Telco Customer Churn

Milestone: Project Report

Group 4
Niresh Subramanian
Sachin Dewangan

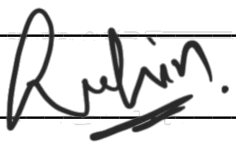
857-693-9624 (Tel of Niresh Subramanian)
332-250-8015 (Tel of Sachin Dewangan)

subramanian.ni@northeastern.edu
dewangan.s@northeastern.edu

Percentage of Effort Contributed by Student 1: 50%

Percentage of Effort Contributed by Student 2: 50%

Signature of Student 1: 

Signature of Student 2: 

Submission Date: 2023-04-21

Table of Content

Problem Setting

Customer churn is a common issue for businesses, particularly in fast-paced and highly competitive industries. It occurs when customers decide to take their business elsewhere, resulting in a significant loss of revenue and profitability. The telecommunications industry is no exception, with thousands of customers canceling their contracts each month. Preventing customer churn is crucial for the success and survival of a company. The loss of thousands of customers can mean the difference between success and failure. By understanding the concept of customer churn and its impact on a business, companies can take the necessary steps to retain their valued customers and stay ahead of the competition.

Problem Definition

The objective is to identify the best model for predicting customer churn in the telecommunications industry by analyzing the telco customer churn dataset, identifying key factors contributing to churn, profiling customers based on their attributes, and developing various classification models. The chosen model would predict which customers are likely to cancel their contracts, enabling the company to take proactive steps to retain them, thus improving revenue and profitability while addressing the root causes of customer churn.

Data Sources

The telco customer churn dataset is sourced from Kaggle, a platform for machine learning and data science competitions. This dataset is from IBM Sample Data Sets.

Data Description

The dataset includes information about customers, including whether they have left within the last month (used to flag customers as "churned" or "not churned"), the services they have signed up for, their account information, and demographic information. It contains 7043 observations and 21 variables, with both numerical and categorical data.

The data dictionary has been listed below for reference:

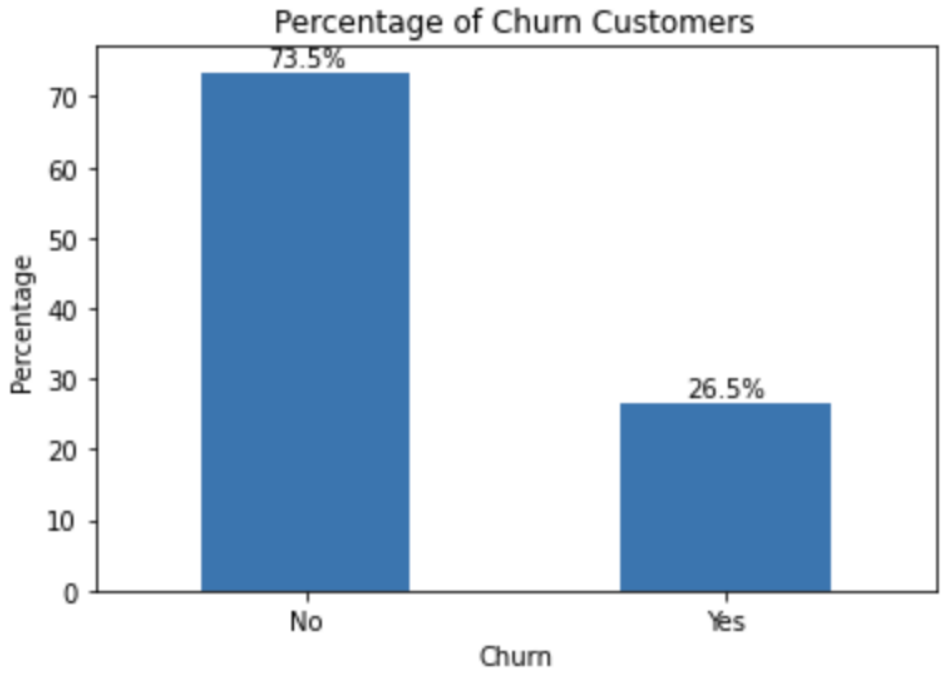
Variables	**Description**
customerID	A unique identifier for each customer.
gender	The gender of the customer: Male or Female.
SeniorCitizen	Whether or not the customer is a senior citizen.
Partner	Whether or not the customer has a partner.
Dependents	Whether or not the customer has dependents.
tenure	The length of time the customer has been a customer in months.
PhoneService	Whether or not the customer has phone service.
MultipleLines	Whether or not the customer has multiple phone lines.
InternetService	The type of internet service the customer has: DSL, Fiber optic, or None.
OnlineSecurity	Whether or not the customer has online security.
OnlineBackup	Whether or not the customer has online backup.
DeviceProtection	Whether or not the customer has device protection.
TechSupport	Whether or not the customer has tech support.
StreamingTV	Whether or not the customer has streaming TV.
StreamingMovies	Whether or not the customer has streaming movies.
Contract	The type of contract the customer has: Month-to-month, One year, or Two years.
PaperlessBilling	Whether or not the customer has paperless billing.
PaymentMethod	The method of payment the customer uses: Electronic check, mailed check, Bank transfer (automatic), or Credit card (automatic).
MonthlyCharges	The amount the customer is charged per month.
TotalCharges	The total amount the customer has been charged.
Churn	Whether or not the customer churned: Yes or No.

Data Collection

Data collection is a crucial step in any data analysis project. In this project, the first step in data collection was to generate a Kaggle API token. This was done to use the Kaggle module to access the Kaggle Telco-Customer-Churn dataset. The credentials were then passed, and the dataset was downloaded. After the dataset was downloaded, it was read as a pandas dataframe, which is a popular data manipulation tool in Python. This step was essential in preparing the data for further analysis.

Data Exploration

Data exploration is a critical step in any data analysis project. In this project, the data exploration process involved several steps to better understand the data and prepare it for further analysis. Firstly, the datatypes and presence of null values were checked in all columns, and it was found that the **"TotalCharges" column had inconsistent datatypes** and needed to be converted to float. Additionally, **blank spaces were identified, replaced with NaN values, and converted to float**. Eleven null values were identified, and it was noted that 26.5% of the records were customers who churned. **As all the rows where "TotalCharges" was NULL (0.15% of records) belonged to non-churn customers, they were dropped**.



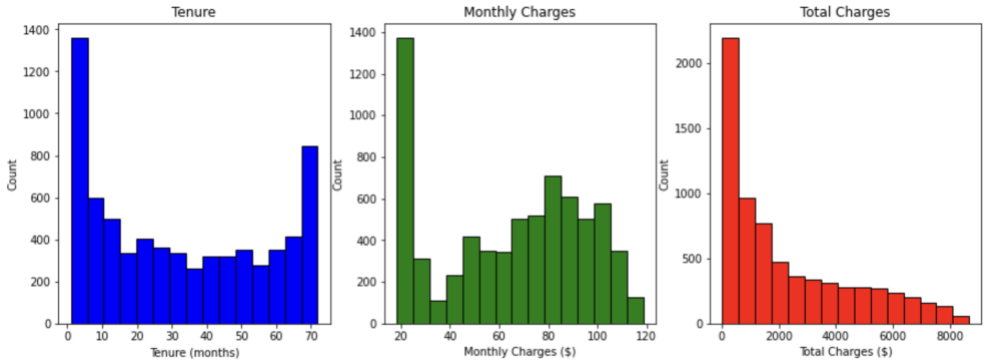
To gain a better understanding of the data, the "describe()" function was used to obtain aggregate statistics. It was observed that **several binary columns had three unique values**, with a couple of values that could be merged into a "No" category. Accordingly, **"No internet service" and "No phone service" were replaced with "No" for several columns**. **Binary categorical columns were then encoded into 1s and 0s**, and changes were made to several columns, including "Partner," "Dependents," and "Churn."

Finally, the level of data was checked, and it was identified that the **customer ID** was not useful and could be **dropped**, as **it was only a row identifier and would not be used as an input to the model**.

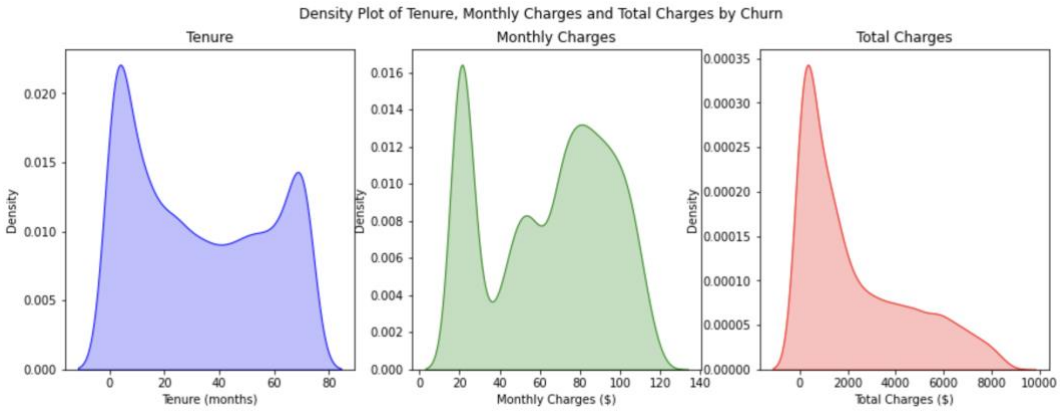
Data Visualization

Data visualization helps in EDA by providing a visual representation of the data that can quickly identify outliers, missing values, and other anomalies. It also helps in identifying trends and patterns between variables that may not be apparent from simple analysis. Visualizations can take many forms and are useful in communicating insights to stakeholders in an easily understandable and actionable way.

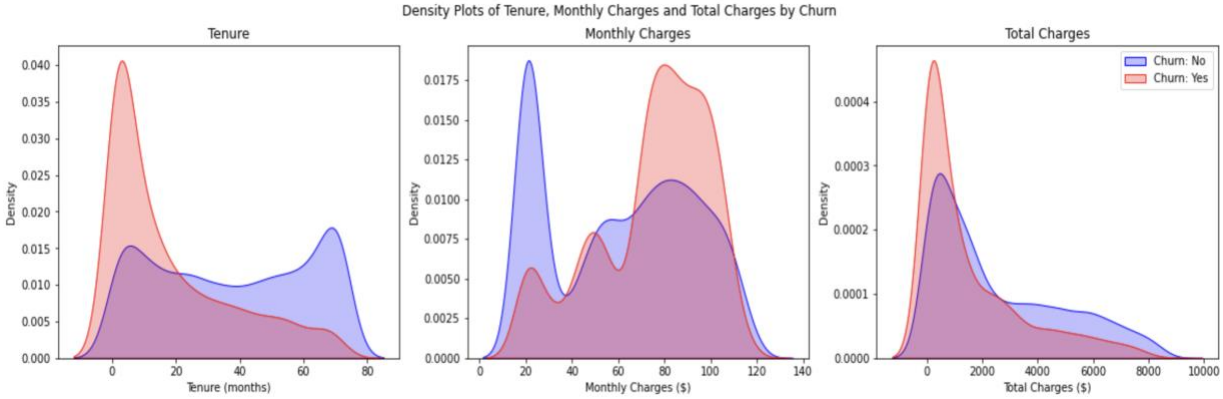
Additionally, during the EDA, we performed various data visualizations to gain a better understanding of the data. Firstly, we used box plots to understand the spread of the data for the three numerical variables: "Tenure", "TotalCharges", and "MonthlyCharges". We observed that **"Tenure" and "MonthlyCharges" have a bimodal shape, while "TotalCharges" is skewed to the right**.



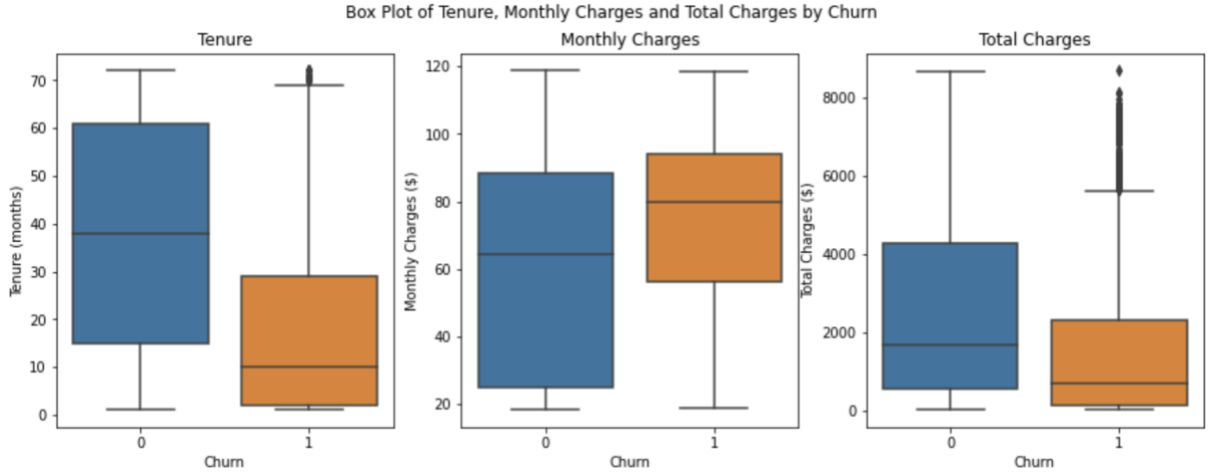
To get a better sense of the number of peaks or modes, we also plotted density curves for these variables, which validated the observations from the box plots.



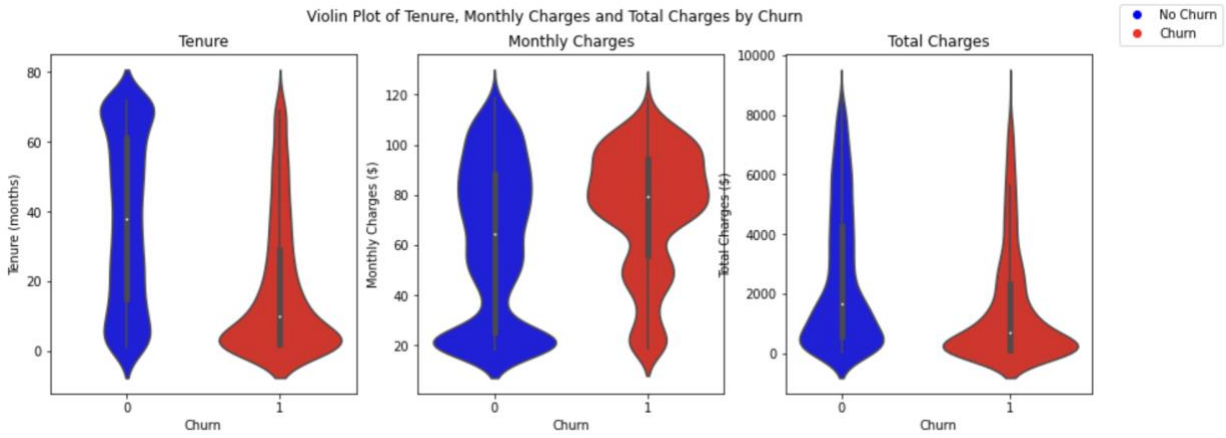
Furthermore, we created plots to understand the distribution of the numerical variables by the different churn groups. By including churn as the legend, we observed that **customers who churn out have a low tenure**, which makes sense as they are likely new customers who unsubscribed. Additionally, **customers who churn out tend to have higher monthly charges compared to those who don't churn out**. Interestingly, the distribution of "TotalCharges" is the same for both customer types. These findings provide valuable insights into the relationship between churn and the different numerical variables, which can inform our modeling and decision-making processes.



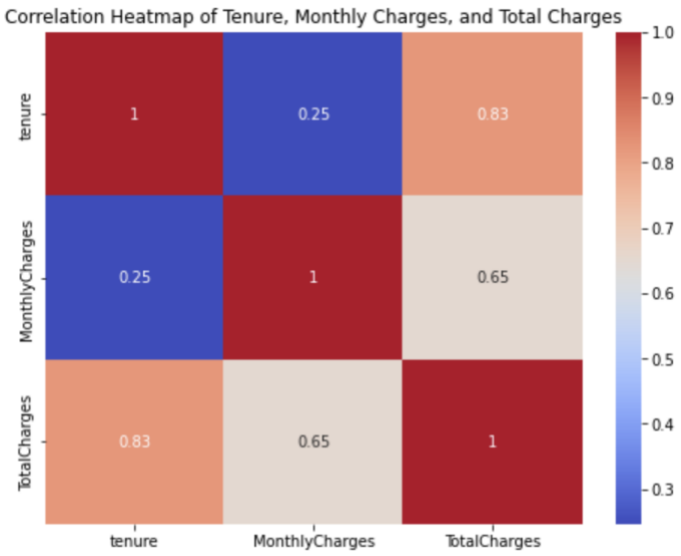
A box plot of the numerical variables was created by dividing customers into two groups based on churn, and the results were observed. The **median tenure for non-churn customers was found to be higher,** at around 40 months, while **most of the churn customers had data points in the initial 2 quartiles**, indicated by the small whiskers, and a few outliers. The **median monthly charges for churn customers were higher**, and there were no outliers. As expected, the median total charges for churn customers were lower, primarily because they were mostly new customers who had unsubscribed. However, the **churn customers had several outliers**, which could be due to customers with less tenure but high rates or those who unsubscribed after spending a considerable amount of time with the telecom provider. This information could be valuable for predicting churn, and it might not be appropriate to exclude these outliers.



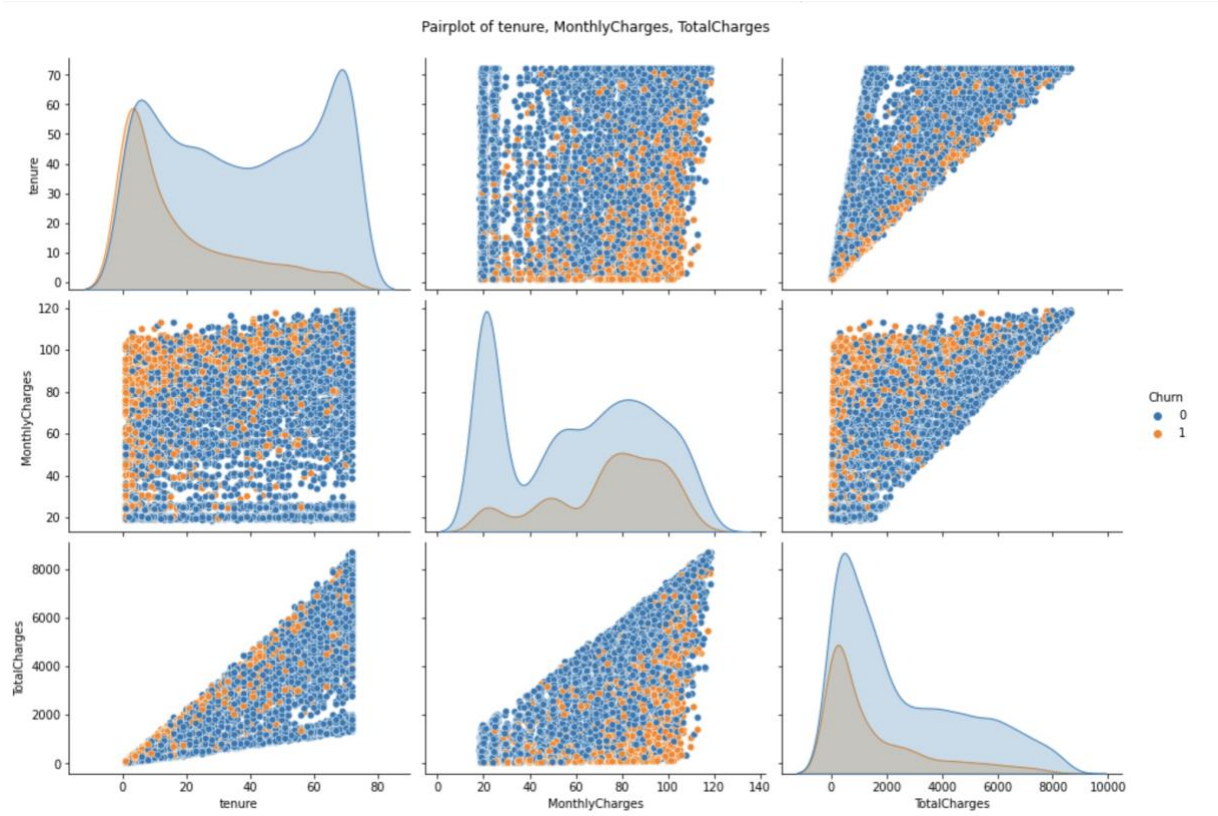
To better understand the spread of data points for both churn and non-churn customers, we created violin plots. The violin plots showed that the **distribution of the data points for non-churn customers is wider and more spread out, while the distribution of data points for churn customers is narrower and more concentrated around the lower quartiles**. This suggests that churn customers tend to be more similar in their characteristics, while non-churn customers have a wider range of characteristics. Overall, the violin plots provided a clearer and more informative view of the data distribution than the box plots alone.



To understand the relationships between the numerical variables, a pairplot matrix and a correlation heatmap were plotted, leading to the following observations. Correlation coefficients ranging from 0.7 to 0.9 indicate a high correlation between variables. From the heatmap, it can be inferred that the variables **tenure and total charges are highly correlated**, which means that as a customer spends more time with the telecom provider, their total charges are likely to increase. On the other hand, the **other variable combinations such as monthly charges and total charges, tenure and monthly charges do not show a strong correlation**. This implies that there is no significant relationship between the monthly charges paid by customers and their total charges or tenure.



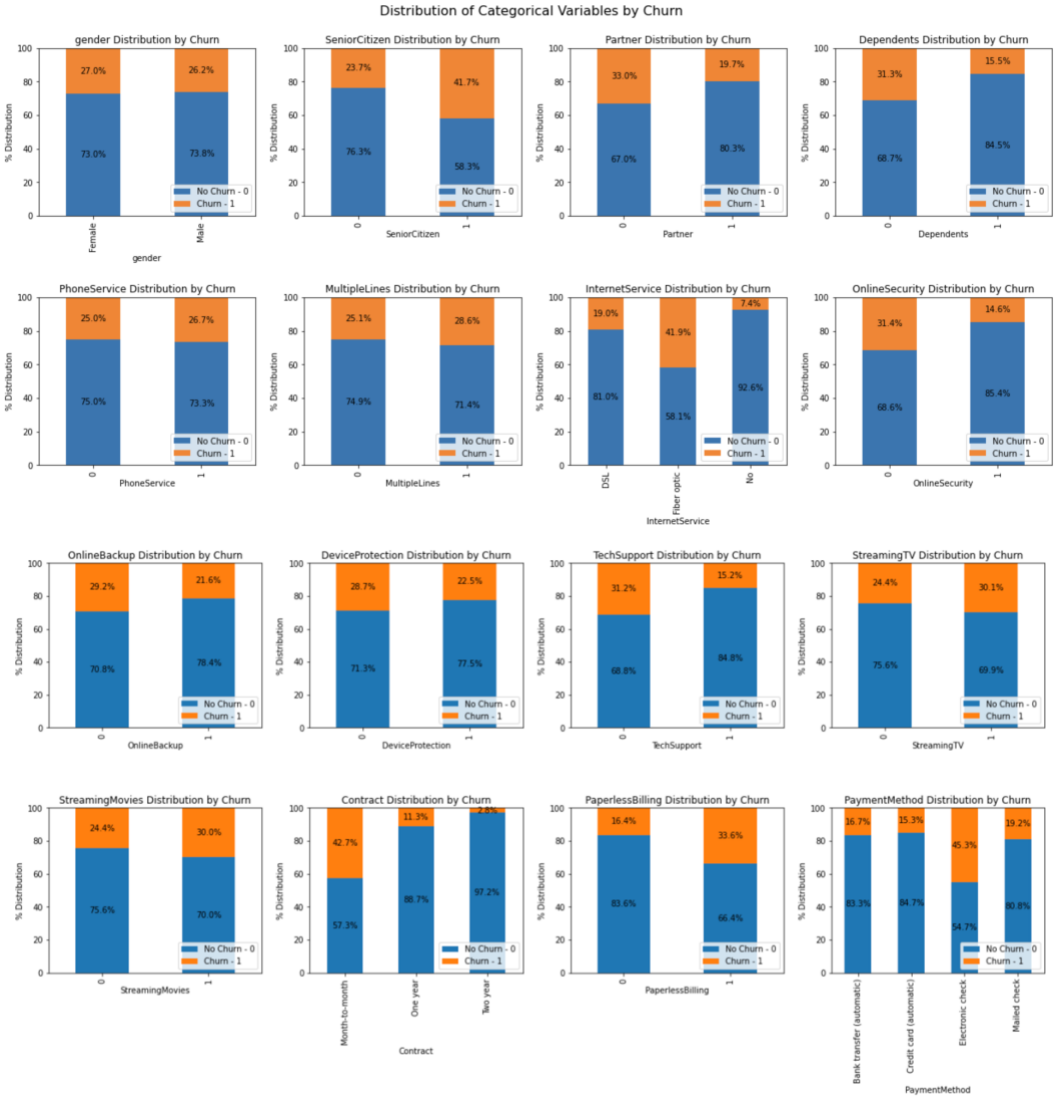
Pairplot helps us better visualize the relationship between continuous variables. We can see tenure and TotalCharges have a positive relationship and **as tenure increases, TotalCharges increases linearly**.



To examine the relationship of each categorical variable with the dependent variable (Churn), a stacked bar charts were plotted. The categorical variables included 'gender', 'SeniorCitizen', 'Partner', 'Dependents', 'PhoneService', 'MultipleLines', 'InternetService', 'OnlineSecurity', 'OnlineBackup', 'DeviceProtection', 'TechSupport', 'StreamingTV', 'StreamingMovies', 'Contract', 'PaperlessBilling', and 'PaymentMethod'. Several insights were observed from this analysis.

- Senior Citizens have a higher churn rate than non-Senior Citizens
- Customers with partners or dependents are less likely to churn
- Customers with Fiber Optic Internet Service are more likely to churn, with a churn rate of 41.9%, followed by DSL and No Internet Service
- Customers without online security have a slightly higher churn rate distribution

- Customers with a month-to-month contract are more likely to churn out. Other yearly subscriptions can be merged into one category
- Customers with paperless billing have a higher churn rate
- Customers who use electronic check as their payment method have a higher churn rate. Other payment categories can be merged into one category
- The following variables have minute differences when it comes to the distribution of churn to no churn. Hence, they probably aren't the best predictors when it comes to predicting churn: gender, phone service, multiple lines, online backup, device protection, streaming TV, and streaming movies



Data Preprocessing

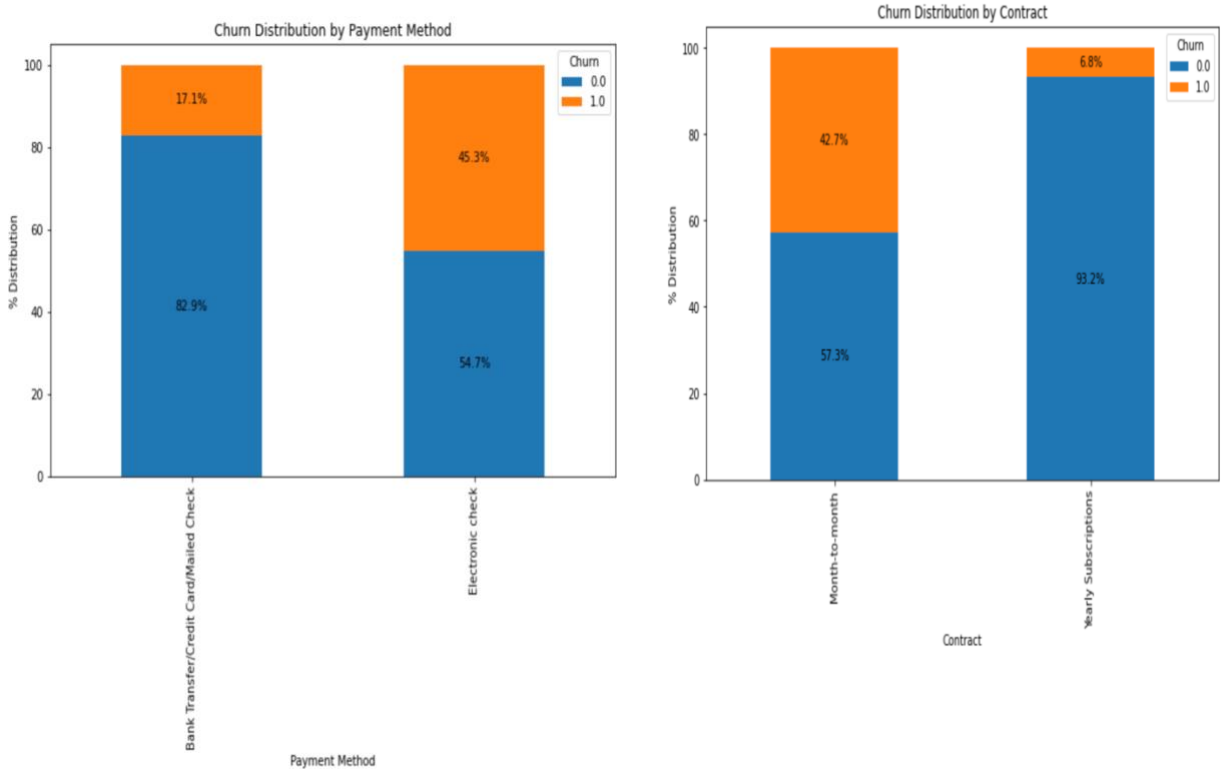
Data preprocessing is the process of cleaning and transforming raw data into a format suitable for analysis. It involves removing irrelevant or duplicate data, filling in missing values, and converting data into a consistent format. Data preprocessing is a crucial step in data analysis as it ensures that the data is accurate, complete, and relevant to the analysis being performed.

As part of data preprocessing, we **performed standardization and principal component analysis (PCA) on the three numerical variables: "tenure", "MonthlyCharges", and "TotalCharges".** Standardization was used to transform the variables to have a mean of 0 and a standard deviation of 1, which helps ensure that all variables are equally important in subsequent analyses. PCA was then applied to identify the principal components that explain most of the variance in the data. We observed that the **first two components capture 98% of the variance**, indicating that these two components can effectively summarize the variability in the original data. By reducing the number of variables in this way, we can simplify subsequent analyses while retaining the most important information from the original data.

After combining the categories of Bank transfer, Credit card, and Mailed check into one category in Payment Method, we aimed to reduce dimensions by creating a stacked bar chart to find the Churn Distribution by Payment Method. From this chart, we observed that customers who chose Electronic check as their payment method had the highest churn rate (45.3%), followed by customers who chose Mailed check, Bank transfer, or Credit card as their payment method. **Combining these categories into one group helps to simplify the analysis of the Payment Method variable while still preserving the information regarding the relationship between Payment Method and Churn**.

As a part of dimensionality reduction, the **categories in the "Contract" column were combined to reduce the number of categories in the dataset**. The categories "One year" and "Two year" were combined into a new category named "Yearly Subscriptions". A stacked bar chart was then created to visualize the churn distribution by contract. The chart revealed that **customers with a**

Month-to-Month contract had the highest likelihood of churning out, followed by those with Yearly Subscriptions. This insight may be useful for the company in designing subscription plans that could help retain customers and reduce churn.



One hot encoding is a technique used to convert categorical data into a format that can be used by machine learning models. It creates binary columns for each unique category, with a value of 1 or 0 to indicate whether the category is present. For a categorical variable with m categories, one hot encoding creates m-1 columns since the presence of all m categories can be inferred from the absence of m-1 categories. In this project, **one hot encoding was used for categorical variables with more than two values** after grouping similar categories within the variable, such as 'PaymentMethod', 'Contract', and 'InternetService', to represent them in a format suitable for machine learning models.

The chi-squared test is a statistical test that is used to determine the association between two categorical variables. The **null hypothesis states that there is no association between the**

variables, while the alternative hypothesis states that there is a significant association between the variables.

The chi-squared statistic measures the difference between the observed frequency and the expected frequency, with higher values indicating a stronger association. The p-value indicates the probability of obtaining the observed result by chance, with **smaller p-values indicating stronger evidence against the null hypothesis**.

In this analysis, we found that there is n**o significant association between gender and churn, and phone service and churn as their p-value was greater than 0.05**, indicating that these predictors may not be strong predictors for churn. A **higher chi-squared score indicates a stronger association** between the variables.

```
Contingency table for SeniorCitizen and Churn: Chi-squared test results for SeniorCitizen and Churn:
Churn           0.0   1.0
SeniorCitizen
0.0            4497  1393
1.0             666   476 chi2 = 158.44, p-value = 2.48e-36, dof = 1
There is a significant association between SeniorCitizen and Churn with p-value 2.48e-36.

-----------------------------------------------------------------------------------------------------------------
Contingency table for Partner and Churn: Chi-squared test results for Partner and Churn:
Churn     0.0   1.0
Partner
0.0      2439  1200
1.0      2724   669 chi2 = 157.50, p-value = 3.97e-36, dof = 1
There is a significant association between Partner and Churn with p-value 3.97e-36.

-----------------------------------------------------------------------------------------------------------------
Contingency table for Dependents and Churn: Chi-squared test results for Dependents and Churn:
Churn        0.0   1.0
Dependents
0.0         3390  1543
1.0         1773   326 chi2 = 186.32, p-value = 2.02e-42, dof = 1
There is a significant association between Dependents and Churn with p-value 2.02e-42.

-----------------------------------------------------------------------------------------------------------------
Contingency table for PhoneService and Churn: Chi-squared test results for PhoneService and Churn:
Churn           0.0   1.0
PhoneService
0.0              510   170
1.0             4653  1699 chi2 = 0.87, p-value = 3.50e-01, dof = 1
There is NO SIGNIFICANT ASSOCIATION between PhoneService and Churn with p-value 3.50e-01.

-----------------------------------------------------------------------------------------------------------------
Contingency table for MultipleLines and Churn: Chi-squared test results for MultipleLines and Churn:
Churn           0.0   1.0
MultipleLines
0.0             3046  1019
1.0             2117   850 chi2 = 11.09, p-value = 8.69e-04, dof = 1
```

In **summary**, we **reduced the dimensionality of three continuous variables to two principal components**, which captured 98% of the data variability. Through a chi-squared test, we found that **PhoneService and gender were not strongly associated with churn and excluded them as predictors**. We also **reduced the dimensionality of the PaymentMethod, Contract, and InternetService variables from 10 to 4 dimensions**. Irrelevant columns like **CustomerID were dropped**, resulting in **17 predictors for the model**.

Data Partitioning

Data partitioning is a crucial step in machine learning, and it involves dividing the dataset into separate sets for training and testing purposes. To avoid overfitting and obtain an accurate estimate of the model's performance on new data, **we split the dataset into training, validation, and test sets (70%, 20%, and 10% respectively)** using scikit-learn. **The training and validation sets will be used to train and evaluate multiple candidate models, respectively.** The model that performs best on the validation set will be chosen as the final candidate model. The **test set will then be used to estimate the final model's performance on new, unseen data.**

Exploration of Candidate Data Mining Models

The telco customer churn dataset has 'Churn' as the response variable which is a binary categorical variable. Most of the chosen predictors, after reduction, are binary categorical variables as well. Additionally, we have a few numerical predictors whose dimensions were reduced using PCA to capture over 98% of the variability in the data. The following are a **few candidate data mining model options available to classify a binary categorical variable using a combination of categorical and numerical predictors**:

Logistic Regression

A statistical model that models the relationship between the response variable and multiple predictors using a logistic function to transform the output into probabilities.

Advantages:

- Simple and interpretable model
- Performs well when the relationship is linear or can be approximated by a linear function
- Produces probabilities as outputs

Disadvantages:

- Assumes a linear relationship between the response variable and predictors
- Sensitive to outliers and multicollinearity
- Cannot handle predictor interactions
- May not perform well when classes are not well separated or have a high degree of overlap

Decision Tree Classifier

A non-parametric model that splits the data based on the values of the predictors using a tree-like structure to make predictions.

Advantages:

- Can handle both numerical and categorical predictors
- Non-parametric and robust to outliers and nonlinear relationships
- Able to capture complex interactions between predictors
- Can be easily visualized and interpreted

Disadvantages:

- Tendency to overfit training data
- Can be unstable and sensitive to small changes in the data
- Can create complex and difficult-to-interpret trees
- May not generalize well to new data

Random Forest

An ensemble learning model that combines multiple decision trees to improve accuracy and robustness by aggregating their predictions.

Advantages:

- Combines the advantages of decision trees with the benefits of ensemble learning
- Non-parametric and robust to outliers and nonlinear relationships
- Able to capture complex interactions between predictors
- Helps reduce overfitting and improves the generalization of the model

Disadvantages:

- Less interpretable than individual decision trees
- Can be computationally expensive and require more resources
- May not work well on small datasets

Support Vector Classification (SVC)

A linear model which finds a hyperplane that maximally separates the data into different classes.

Advantages:

- Effective at handling high-dimensional data and complex decision boundaries
- Tends to perform well with a small number of observations
- Has regularization parameters that help prevent overfitting

Disadvantages:

- Can be computationally expensive with large datasets
- Choosing the appropriate kernel function and tuning parameters can be difficult
- Can be sensitive to outliers in the data
- Can be difficult to interpret and explain the model results

Naive Bayes

A probabilistic model that assumes independence between predictors and can efficiently handle a mix of categorical and numerical predictors.

Advantages:

- Simple and easy to implement with fast training and prediction times
- Can handle many predictors
- Can work well with small datasets
- Performs well when the assumption of independence between predictors holds

Disadvantages:

- Assumes that the predictors are independent, which may not be true in practice
- Can perform poorly when dealing with rare events or rare combinations of predictors
- Cannot handle missing data well
- Can produce overconfident probability estimates for rare events

Gradient Boosting

An ensemble learning model that combines multiple weak learners, such as decision trees, to form a strong classifier.

Advantages:

- Can handle missing data and outliers

- Can detect and model complex nonlinear relationships between predictors and response variables
- Can be tuned to prevent overfitting

Disadvantages:

- Can be computationally expensive and time-consuming to train
- Can be sensitive to hyperparameter choices
- Can be prone to overfitting if not properly tuned
- May require a large amount of training data to achieve good performance

Neural Networks

A class of models that can learn complex non-linear relationships between the response variable and predictors through multiple interconnected nodes.

Advantages:

- Can learn complex nonlinear relationships between predictors and response variables
- Can be highly accurate when properly trained and tuned
- Can be used to classify images, texts, and other unstructured data

Disadvantages:

- Can be computationally expensive and time-consuming to train
- Can be prone to overfitting if not properly tuned or if the model architecture is too complex
- May require a large amount of training data to achieve good performance
- Can be difficult to interpret and explain the results of a trained model

K-Nearest Neighbors (KNN)

A non-parametric model that classifies a new data point based on the class labels of its k-nearest neighbors in the training data.

Advantages:

- Simple and easy to understand and implement
- Can work well with a small number of predictors

- Non-parametric model, meaning that it does not make any assumptions about the underlying distribution of the data

Disadvantages:

- Sensitive to irrelevant features, which can result in poor performance
- Can be computationally expensive, especially with large datasets and/or many predictors
- Performance can be affected by the choice of the value of K
- Needs enough training data to work effectively

Ensemble Learning - Adaptive Boosting using Decision Tree Classifier

Adaptive Boosting using Decision Tree Classifier is an ensemble learning technique that trains a series of weak decision tree classifiers on training data with an emphasis on misclassified examples.

Advantages:

- Can handle both categorical and numerical data
- Can achieve high accuracy with relatively simple models.
- Can handle imbalanced data well, by emphasizing the misclassified examples
- It is a flexible algorithm, which can be easily adapted to other classification tasks.

Disadvantages:

- It is prone to overfitting if the weak classifiers are too complex or if the data is noisy.
- It can be sensitive to outliers, as they may be repeatedly misclassified by the weak classifiers.
- It requires more computational resources and time to train compared to other algorithms, due to the iterative nature of the algorithm.
- The performance can degrade if there is insufficient training data or if the data is biased.

XGBoost

A gradient-boosting library that can handle both categorical and numerical predictors and is known for its scalability and efficiency.

Advantages:

- XGBoost is known for its scalability and efficiency, making it ideal for large datasets
- Designed to handle a mix of categorical and numerical predictors, making it a versatile algorithm
- It can handle missing data, which can be a common issue in real-world datasets
- It has an in-built regularization technique to prevent overfitting, called 'shrinkage'
- It has a built-in cross-validation method to help prevent overfitting and improve model performance

Disadvantages:

- XGBoost can be computationally expensive and slow for very large datasets or complex models
- It requires tuning of hyperparameters to achieve the best performance, which can be time-consuming and require expertise
- It is a black box model, making it difficult to interpret the results and understand how the model is making its predictions
- It may be prone to overfitting if not tuned properly

On implementing the above-mentioned models, we get the following accuracy and F1 score values as presented in the table below:

Model	Accuracy	F1 Score
Gradient Boosting	0.80	0.59
Neural Networks	0.78	0.59
XGBoost	0.79	0.58
Support Vector Classification	0.79	0.57
K-Nearest Neighbors	0.78	0.57
Logistic Regression	0.79	0.56
Random Forest	0.78	0.56
Ensemble Learning	0.80	0.55
Naïve Bayes	0.78	0.55
Decision Tree Classifier	0.73	0.52

The data has been sorted by F1 Score followed by accuracy in descending order. **As flagging a customer as churn is of more importance, it is essential to balance precision and recall**. Hence,

a **higher F1 score is chosen as the preferred evaluation metric** over accuracy. Note that the above models were run using the default model parameters in the sklearn library and hence, can be finetuned to give a higher F1 score.

The **models were tested on an imbalanced dataset** with 74% non-churn and 26% churn customers. **SMOTE oversampling technique was applied to the training dataset to address the class imbalance**. SMOTE creates synthetic samples for the minority class by interpolating new points between existing ones. The table below shows the models and their performance evaluation metrics on the balanced dataset sorted by F1 score and accuracy in descending order:

Model	Accuracy	F1 Score
Ensemble Learning	0.77	0.64
Logistic Regression	0.76	0.64
SVC	0.75	0.63
Gradient Boosting	0.75	0.62
Naive Bayes	0.74	0.62
Neural Networks	0.75	0.61
XGBoost	0.75	0.61
Random Forest	0.76	0.60
K-Nearest Neighbors	0.71	0.57
Decision Tree Classifier	0.70	0.52

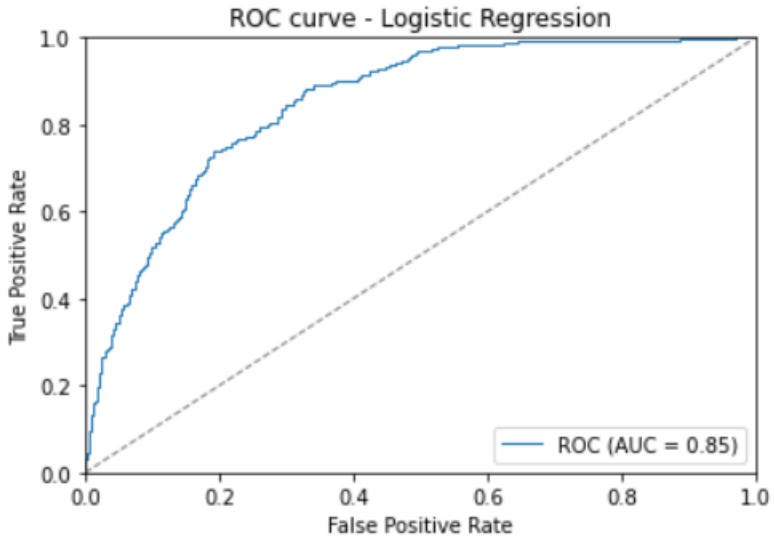
We notice that **models that did not perform that well when trained with the imbalanced dataset, Ensemble learning, and logistic regression, for instance, perform better when trained on a balanced dataset**. By using Dummy Classifier (Assigning the majority class value to all records), we get an accuracy of 0.73 and an F1 score of 0. Hence, our model needs to be better than this.

On examining the top 2-3 models based on the F1 score and accuracy from the above 2 tables, we can select a total of 6 models – Logistic Regression, Ensemble Learning, Gradient Boosting, Neural Networks, and XGBoost, and fine-tune the parameters to obtain maximum F1-score and accuracy and finalize the best model.

Performance Evaluation

Logistic Regression

We used logistic regression to predict customer churn, with hyperparameter tuning focused on the churn class. The **best hyperparameters found were {'solver': 'liblinear', 'penalty': 'l2', 'max_iter': 800, 'fit_intercept': True, 'C': 0.27}**. The best F1-score achieved on the test set was 0.771, indicating moderate ability to identify likely churners. **Sensitivity and specificity were 0.767 and 0.771** respectively, with a **positive predictive value of 0.556**, indicating moderate identification of churners but a high number of false positives. The **overall accuracy on the validation set was 0.770**.

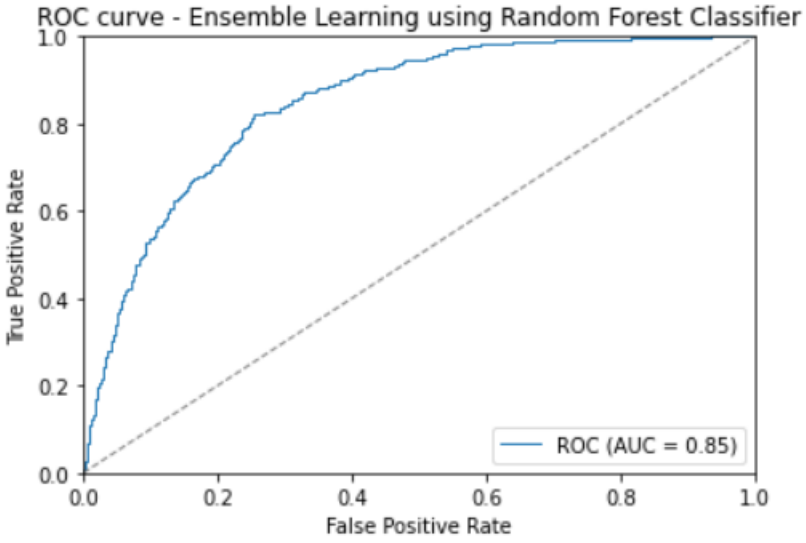


	Predicted False	Predicted True
Actual False	711	211
Actual True	80	264

Random Forest

We used a random forest classifier with hyperparameter tuning to predict customer churn. The random search was used to search over the hyperparameter space, and the **best hyperparameters were found to be {'n_estimators': 460, 'min_samples_split': 4, 'min_samples_leaf': 2, 'max_features': 'log2', 'max_depth': 9}**. The best F1-score achieved on the test set was 0.8102, indicating that the random forest model has a moderate to a strong ability to identify customers who are likely to churn.

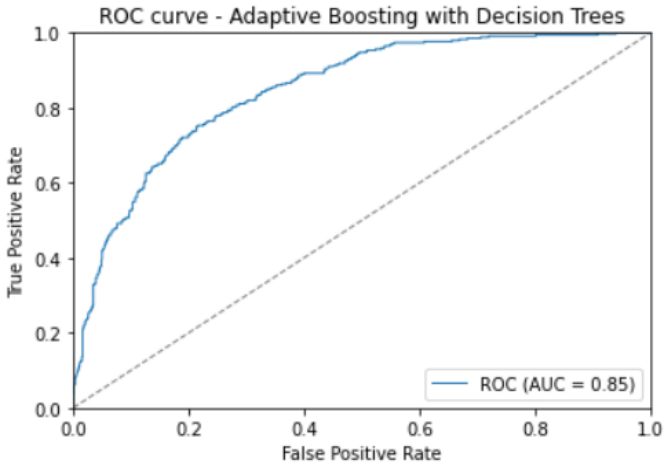
For the churn class, the model achieved a s**ensitivity of 0.7209, specificity of 0.7939, and positive predictive value of 0.5662**. These metrics indicate that the model is able to correctly identify a moderate proportion of customers who are likely to churn, with a relatively high number of false positives. The **overall accuracy of the model was 0.7741** on the validation set.



	Predicted False	Predicted True
Actual False	732	190
Actual True	96	248

Ensemble Learning - Adaptive Boosting using Decision Tree Classifier

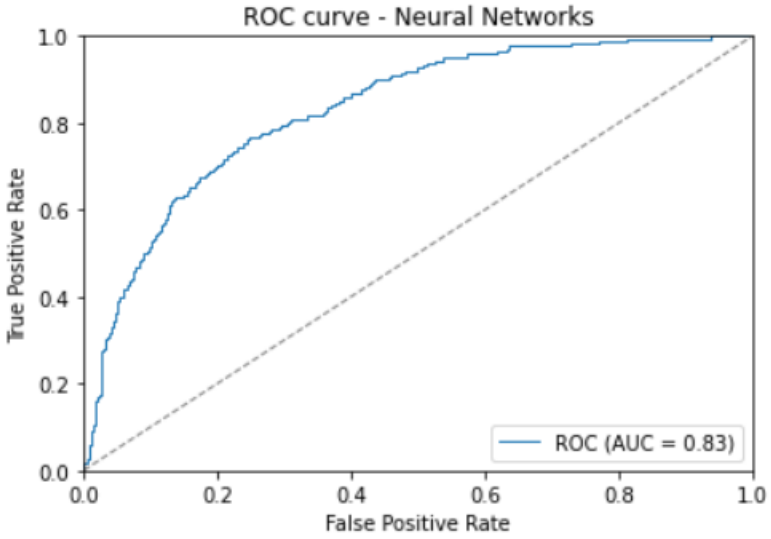
We used adaptive boosting with decision tree classifier to predict customer churn. **Best hyperparameters were {'n_estimators': 330, 'learning_rate': 0.03, 'base_estimator': DecisionTreeClassifier(max_depth=3)}**. Best F1 score on test cross-validation was 0.787. The model achieved a **sensitivity of 0.762, specificity of 0.767**, and **positive predictive value of 0.549** for churn class on the validation set, with an **overall accuracy of 0.765**. The model correctly identified a moderate proportion of churn cases, but also misclassified a significant number of non-churn cases as churn.



	Predicted False	Predicted True
Actual False	707	215
Actual True	82	262

Neural Networks

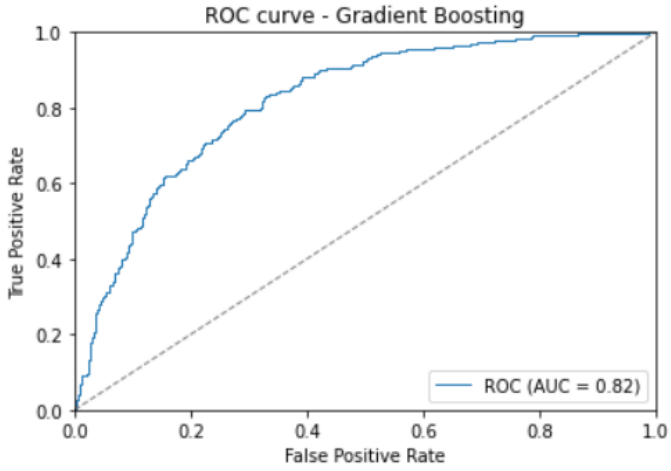
The neural network model achieved a best F1 score of 0.7289 using the hyperparameters **{'activation': 'relu', 'dropout_rate': 0.2, 'hidden_layers': 3, 'optimizer': 'adam', 'units': 64}** during grid search CV. However, when evaluated on the validation set, the model achieved an **accuracy of 0.7520** with a **sensitivity of 0.7683, specificity of 0.7466, PPV of 0.5010**, and an **F1-score of 0.6065**. The model had a higher proportion of false negatives, which indicates that it had difficulty correctly identifying cases of churn.



	Predicted False	Predicted True
Actual False	710	241
Actual True	73	242

Gradient Boosting

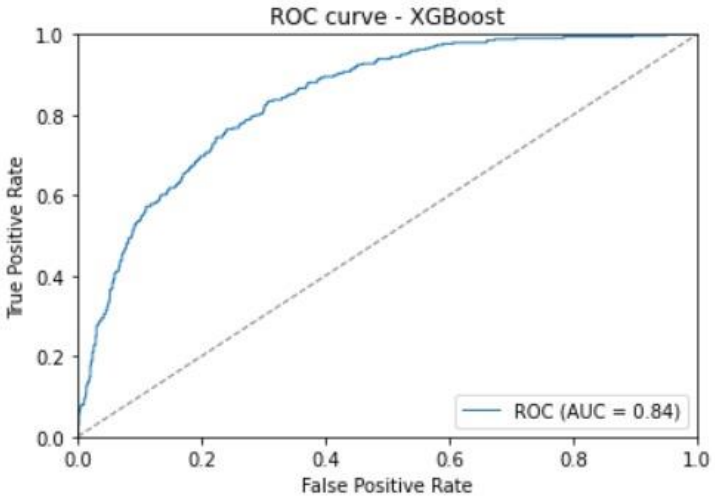
Best hyperparameters for gradient boosting were found using random search cv: **{'subsample': 0.85, 'n_estimators': 170, 'min_samples_split': 4, 'min_samples_leaf': 3, 'max_features': 0.5, 'max_depth': 9, 'learning_rate': 0.21}**. The best F1 score achieved was 0.8213. The model had an **accuracy of 0.7646 and a sensitivity of 0.6570**, indicating the model correctly identified 65.7% of positive cases. The **specificity was 0.8048**, indicating the model correctly identified 80.5% of negative cases. The **precision for positive cases was 0.5567**.



	Predicted False	Predicted True
Actual False	742	180
Actual True	118	226

XGBoost

We used XGBoost with hyperparameter tuning through random search to predict customer churn. The best hyperparameters found were **{'subsample': 0.95, 'n_estimators': 100, 'min_child_weight': 1, 'max_depth': 9, 'learning_rate': 0.13, 'gamma': 0.54, 'colsample_bytree': 0.95}**. The best F1 score achieved on the test set was 0.811, indicating that the model has a moderate ability to identify customers who are likely to churn. For the churn class, the model achieved a **sensitivity of 0.712, specificity of 0.786, and positive predictive value of 0.554**. These metrics indicate that the model is able to correctly identify a moderate proportion of customers who are likely to churn, with a relatively high number of false positives. The **overall accuracy of the model was 0.766** on the validation set. The confusion matrix shows that the model misclassified a significant number of non-churn cases as churn.



	Predicted False	Predicted True
Actual False	725	197
Actual True	99	245

Project Results

Logistic regression, adaptive boosting using decision tree classifier, and random forest after hyperparameter tuning were considered to find the model with the highest AUC. The **focus was on maximizing the accuracy of classifying churn customers, and metrics like F1-score, sensitivity, and precision were used**. Based on these metrics, **logistic regression was determined to be the superior model** due to its higher sensitivity and F1-score, even though random forest had slightly higher precision (the difference in precision was negligible).

Model	Accuracy	F1 Score	Sensitivity	Specificity	Precision	AUC
Logistic Regression	0.7701	0.6447	0.7674	0.7711	0.5558	0.85
Adaptive Boosting using Decision Tree	0.7654	0.6382	0.7616	0.7668	0.5493	0.85
Random Forest	0.7741	0.6343	0.7209	0.7939	0.5662	0.85
XGBoost	0.7662	0.6234	0.7122	0.7863	0.5543	0.84
Neural Network	0.752	0.6065	0.7683	0.7466	0.5010	0.83
Gradient Boosting	0.7646	0.6027	0.657	0.8048	0.5567	0.82

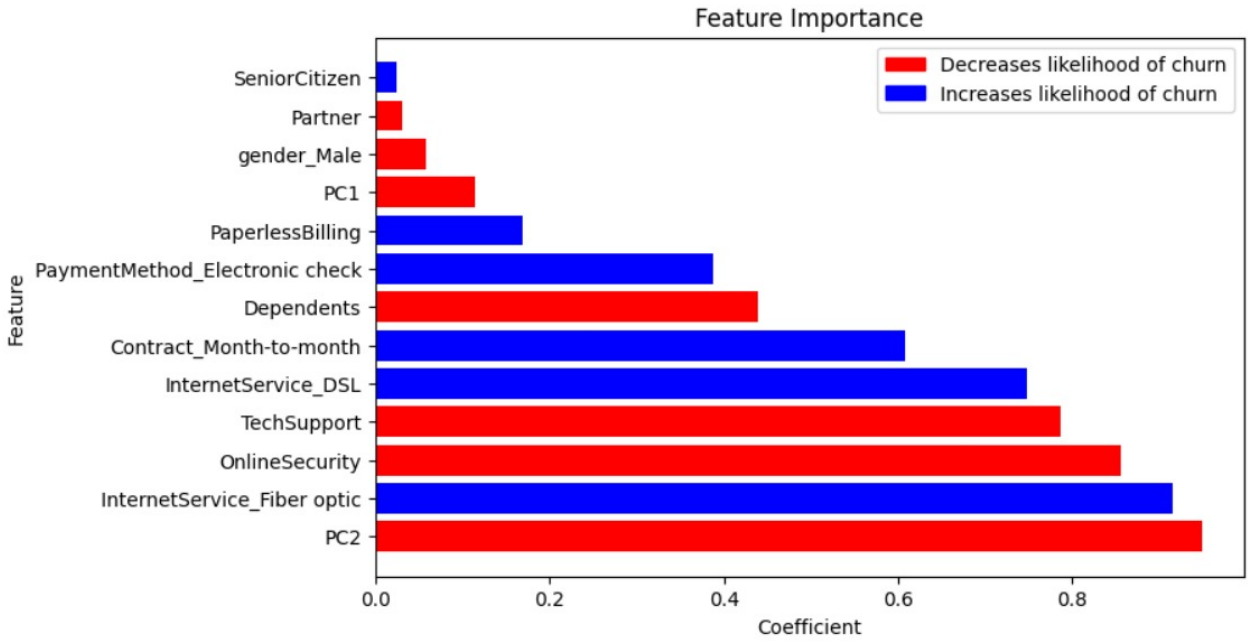
Performance Evaluation of the Best Model on New Data

On exposing the final model to new data (test data), the evaluation metrics and confusion matrix are as follows:

Model	Accuracy	F1 Score	Sensitivity	Specificity	Precision	AUC
Logistic Regression on Test Data	0.71	0.56	0.72	0.71	0.46	0.72

	Predicted False	Predicted True
Actual False	373	151
Actual True	50	130

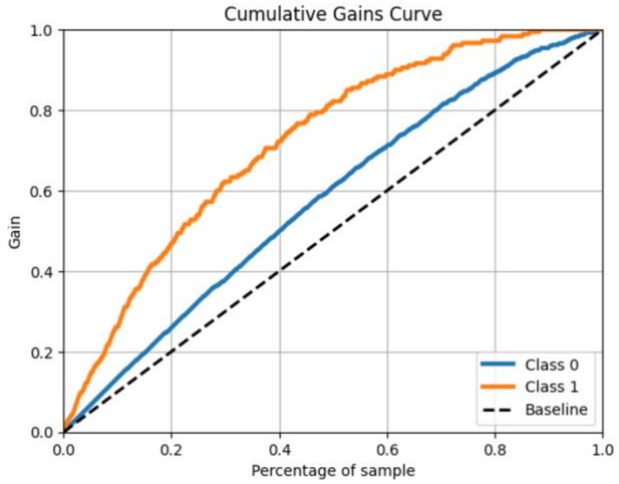
The below graph depicts the feature importance. The numbers in the horizontal bar graph represent the coefficients of the logistic regression model for each feature. These coefficients quantify the effect of each feature on the probability of churn.

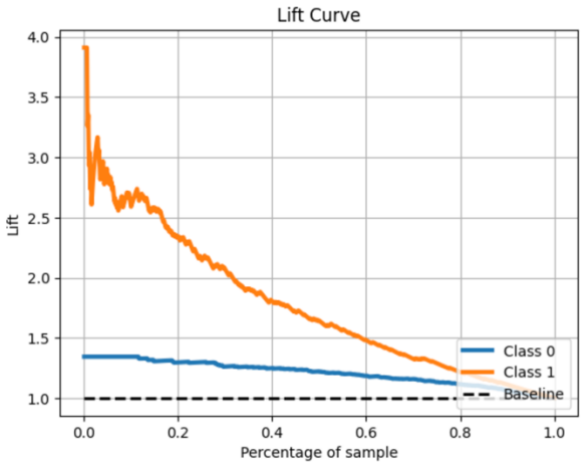


A positive coefficient indicates that an increase in the corresponding feature value increases the likelihood of churn, while a negative coefficient indicates that an increase in the corresponding feature value decreases the likelihood of churn. The magnitude of the coefficient represents the strength of the effect.

For example, in this plot, the features with the largest positive coefficients are “InternetService_Fiber optic" and “InternetService_DSL”. This means that having an Internet Service as Fiber optic and DSL increases the likelihood of churn. Therefore, as the quality of the internet is not satisfactory, the company may need to address this issue to reduce the churn rate. Similarly, having Month-to-Month Contract, Payment Method as Electronic Check, or Paperless Billing increases the likelihood of churn. The feature with the largest negative coefficient is "Online Security" and “Tech Support”. This means that having these services decreases the likelihood of churn, compared to not having them. This makes intuitive sense, as these services provide a sense of security and technical support to customers, which may make them less likely to switch to a different provider. Overall, this indicates that providing reliable online security and tech support services to customers could be an effective strategy to reduce customer churn.

By analyzing the gain and lift chart, we get an understanding of how our model is likely to perform on new data. Class 1 is 'Churn' and Class 0 is 'non-Churn'. We see that the model is a good classifier for the class of interest (Churn customers).





On plotting the decile-wise lift chart, we see that the model has a lift of over 1 for 30% of the data. The model is pretty good at predicting the class of interest for deciles 1 through 3.

Impact of Project Outcomes

In this project, the goal was to develop a classification model to identify customers who are likely to churn for a telecom business, enabling the company to take necessary steps to retain customers and maximize revenue. The logistic regression model outperformed all other classifiers due to its high overall F-1 score, sensitivity, and accuracy, making it the best classification model for identifying and flagging customers who are likely to churn.

By analyzing the most important features and indicators that contribute to churn, the business can develop a predictive model that can estimate the likelihood of churn based on predictor values. This information can be used to take proactive measures to retain at-risk customers and develop targeted marketing and promotional campaigns to address the underlying reasons for churn.

Overall, the impact of this project's outcomes is significant for the telecom business, as it enables them to better understand their customer base and take necessary steps to improve customer retention and maximize revenue. By retaining at-risk customers and reducing customer churn, the business can increase profitability and customer satisfaction.