Telco Customer Churn Prediction

Koushik Modekurti, Rithwik Maramraju, Tejasree Bussa, Vrushali Menthe

San Jose State University, California

Data 245 - Machine Learning

Shih Yu Chang

05/16/2023

Table of contents

Abstract

Churn prediction, a crucial task in customer relationship management, holds immense significance for telecommunication companies when leveraging machine learning models. With the abundance of data available from IBM Cognos, including customer demographics, billing details, network usage patterns, and customer service interactions, telecommunication companies can employ advanced analytics techniques to accurately forecast customer churn. By training supervised machine learning models using historical customer data, such as customer tenure, call duration, plan features, payment history, and customer complaints, telecommunication companies can identify patterns and indicators that signal potential churn behavior.n evaluating churn prediction models for telecommunication companies, several metrics are commonly used to assess their performance, including accuracy, precision, recall and F1-score. These metrics give important information about the efficiency and dependability of churn prediction algorithms. Churn prediction has a significant effect on the operations of telecom firms. By proactively identifying consumers who are at danger of leaving and enabling focused interventions, such as tailored offers, loyalty programs, or committed customer care, it aids in customer retention. Telecommunications firms may protect important client connections and revenue streams by lowering turnover rates. Additionally, by maximizing marketing initiatives and resource allocation and resulting in more effective budget usage, churn prediction provides cost savings. Increased customer happiness, loyalty, and the opportunity for upselling or cross-selling possibilities are further ways that churn prediction helps revenue rise. In conclusion, churn prediction gives telecommunications firms the capacity to make data-driven decisions, improve customer retention, boost profitability, and acquire a competitive edge in a fast-moving market.

1. Introduction

Predicting customer churn for telecommunications companies is essential for retaining consumers. Large telecommunications companies are actively working on developing predictive models to identify customers who are more likely to transfer to a different service provider because the cost of recruiting new customers is higher than retaining existing ones. The goal of this project is to build a model that can accurately forecast the likelihood of customer churn by examining three essential types of data: demographic data, account information, and service usage data. Utilizing these traits, the goal is to develop a data-driven solution that effectively lowers churn rates, increasing customer satisfaction and boosting the company's income.

The major goal of this project is to create five machine learning models that, using data on customer behavior and service consumption patterns, can identify which clients are likely to leave in the future. Models like random forest, support vector machine (SVM), k-nearest neighbors (KNN), logistic regression, and gradient boosting are being used in this project. These five different models create a comprehensive prediction system that makes use of each algorithm's distinct advantages. By using the derived models, the telecommunications business will be able to proactively identify clients who are at risk of leaving and take the necessary steps to keep them. This proactive approach might considerably lower the rate of client churn, resulting in enhanced customer happiness and greater income for the business.

1.1 Problem Statement

The key issue is figuring out how many customers are actively using the services versus how many are leaving. The project also intends to find gender-based patterns in churning customer behavior as well as preferences or trends in the services that churning consumers use.

Identifying the service categories that are most profitable as well as the features and services that help the organization make the most money is another key step. Exploratory data analysis, data preparation, feature engineering, and the application of machine learning models to forecast customer attrition will all be part of this project. By answering these scoping queries, the project can offer insightful information on customer behavior and assist the business in creating plans to increase client retention and increase revenues.

1.2. Project background

The project's main objective is to analyze a dataset from IBM Cognos Analytics that consists of 21 independent variables that represent the traits of clients of a hypothetical telecoms company. The dataset contains a response variable named "Churn," which denotes if a client discontinued their business relationship with the firm within the previous month. The "No" and "Yes" categories in the Churn column represent customers who choose to stay with the business, respectively. The telecommunications company can take proactive steps to keep consumers and lower churn rates by having important information about these aspects in hand. The end result of this project will be insights that may be used to inform data-driven decision-making and successful client retention efforts.

1.3. Literature survey

A paper focuses on the problem of customer turnover and how it affects businesses' profits, especially in the telecom sector. According to the authors, a lot of businesses are attempting to figure out how to forecast prospective customer churn so that they may take the appropriate steps to lessen it. The study proposes a churn prediction model that builds a novel approach to feature engineering and selection using machine learning methods on a big data platform. The authors obtained an AUC value of 93.3% when evaluating the model using the Area Under Curve (AUC) standard measure. This study's usage of the consumer social network in the prediction model by extracting characteristics from social network analysis (SNA) is one of its major achievements. (Ahmad et al., 2019).

Another study uses machine learning methods to create a churn prediction model for the telecom sector. The authors did this by using a dataset that included client data and information about their behavior over a predetermined time. Following pre-processing, the dataset was examined using a number of machine learning algorithms, including Decision Trees, Random Forest, and XGBoost. These algorithms were used by the authors to categorize the clients who are most likely to discontinue their membership. With an accuracy of 96.7%, the results demonstrate that the XGBoost algorithm performs better than competing methods. This study showed how machine learning algorithms can accurately forecast customer turnover in the telecom sector, which can assist businesses in lowering churn rates and raising profitability.(Kavitha et al., 2020).

In the next paper, the significance of customer churn prediction in the telecom sector is covered in the article. The writers emphasize the change from client acquisition to customer retention with a focus on the mobile telecommunications sector. Predicting which clients are most likely to defect to a competitor becomes vital in this situation. The authors suggest a model for churn prediction that makes use of the well-known machine learning method logistic regression. On the available dataset, the study compares the algorithm's effectiveness. The findings show that the suggested approach is capable of accurately forecasting customer churn, which can help telecommunications businesses retain consumers and increase profitability. (Rani, 2021).

Lastly, in order to increase the precision of customer churn prediction, the study examines the problem of customer turnover in the telecommunications sector as well as a number of machine learning models and data transformation techniques. The authors optimize prediction models utilizing feature selection and the grid search approach to choose the optimum hyperparameters, using publically available TCI datasets. They show the value of using feature selection and data transformation techniques for training an optimized CCP model, which increased prediction performance by up to 17% and 26.2%, respectively, in terms of AUC and F-measure. Insights into the effectiveness of several machine learning models for customer churn prediction in the TCI setting are provided in the paper, which adds to the body of literature. Various machine learning models, data transformation methods, and performance evaluation measures are some of the techniques and technology employed. (Sana et al., 2022).

2. CRISP-DM

CRISP DM is a popular framework used for machine learning projects. It delivers a disciplined and iterative strategy to guide the project from the very beginning to the end. It also describes the normal stages of a project, the activities associated with every stage, and the connection between these tasks.

Business Understanding *:* The goal is to anticipate churn customer attrition using telco defining features, Establish the project's needs, success criteria, and limits and also understand why customer churn is an issue for the company and how precise forecasting may help the company. Collaborate with stakeholders in order to obtain their project requirements and expectations. The business understanding phase lays the groundwork for the project, ensuring that organization objectives, project needs, and stakeholder expectations are all in sync. It enables a comprehensive grasp of the telecom industry background and the importance of anticipating customer turnover. By creating a comprehensive grasp of the business issues, you may concentrate on designing a model for predicting churn that suits the organization's unique objectives and goals. In the business understanding phase we will be accessing the problem through literature survey and this is the stage where we will be determining the models that are used to implement in order to solve the problem statement.

Data Understanding *:* The dataset here is collected from IBM Cognos Analytics which is the primary source of the project. We reviewed this dataset to evaluate its size, format, structure, quality, completeness, and general characteristics. We also identified the total number of columns and rows in the dataset and understood the datatype of each and every column. The assessment of the data's quality and completeness is a key stage in the data comprehension process. It includes finding missing numbers, outliers, and inconsistent data inputs that may need to be addressed. Exploratory Data Analysis (EDA) is used to acquire a better knowledge of the dataset and to understand the relationships between variables. EDA entails inspecting the dataset with various visualizations and statistical tools in order to discover insights, locate trends, and identify outliers. EDA can give a thorough knowledge of the dataset, establish relationships between variables, identify trends or patterns connected to customer turnover, and spot any outliers or anomalies. These insights may help with feature selection, model development, and decision-making when it comes to anticipating and managing customer turnover.

Data Preparation : The data preparation step focuses on converting raw data into a clean, organized format that can be analyzed and modeled. Data preparation is crucial to ensure data quality before modeling. It involves exploratory data analysis, data preprocessing, and data transformation. Exploratory data analysis includes analyzing statistics, detecting outliers, and identifying patterns. Data preprocessing focuses on cleaning data, handling missing values, and transforming it for modeling, using techniques like min-max normalization, label encoding, and one-hot encoding. The dataset is then split into training, validation, and testing sets to assess the model's performance. The diagram below illustrates the data flow from raw data collection to preparation, highlighting the key steps involved.

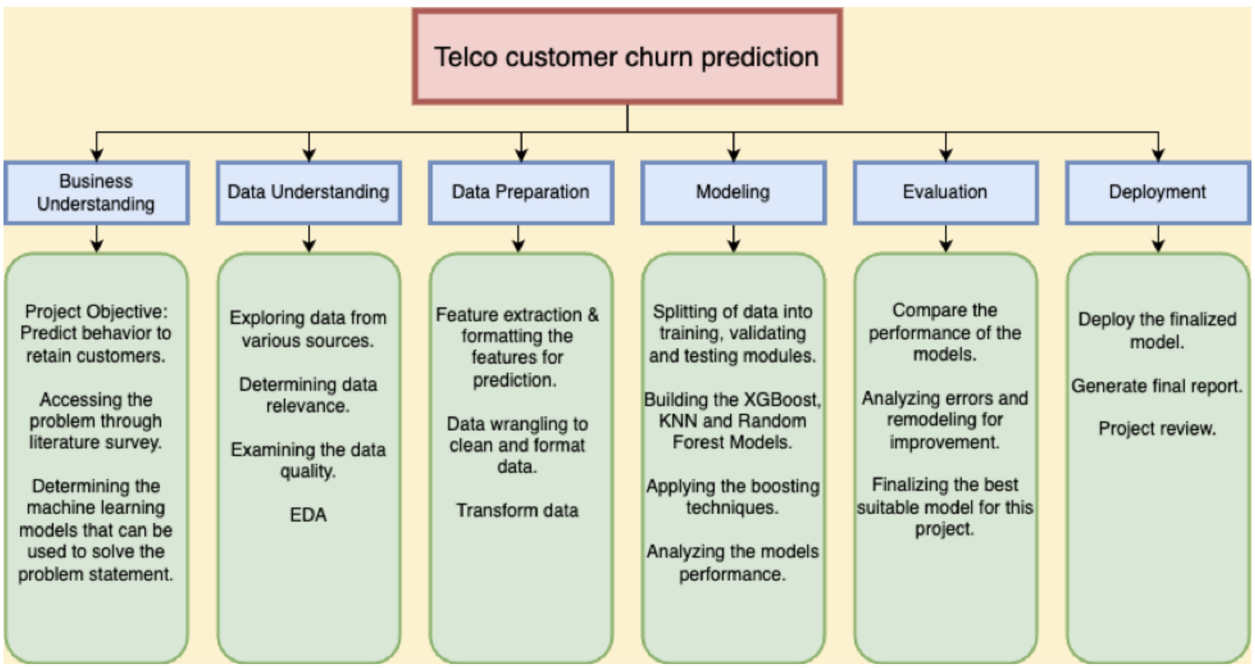
Modeling : The modeling step includes developing and testing prediction models based on prepared and preprocessed data. The machine learning models used in this are Random Forest, Gradient Boosting, Logistic Regression, SVM, and KNN. After this data will be split into a Training, Testing , Validation in 70:20:10 ratio. Train each chosen model on the training set, with telecom characteristics as input and the churn variable as the goal. Model is evaluated on performance based on the validation set using appropriate evaluation measures like accuracy, F1 score, precision, and recall. Hyperparameters of the selected models are implemented to improve their performance. This can include strategies such as grid search or random search, which entail experimenting with various combinations of hyperparameters to discover the ideal configuration.

Evaluation : The CRISP-DM model's Evaluation phase strives to guarantee that data quality criteria are fulfilled, thereby laying the groundwork for later analysis and modeling phases. This step entails studying the data and evaluating the performance of the models. We can identify any data issues, rectify class imbalances, and pick the best models for forecasting customer churn or any other specialized purpose by extensively reviewing the models. Various parts of the data and models are examined during the review process. This involves inspecting the dataset for discrepancies, missing numbers, and outliers. Since they significantly affect the models' reliability and accuracy, these issues need to be recognized and solved. Additionally, when one class in the dataset has a disproportionately high number of occurrences compared to another, this might affect the performance of the model. Techniques like undersampling, oversampling, or using weighted approaches may be utilized to lessen the effects of class imbalances. In order to Balance the data we are using an undersampling technique. Proper assessment measures are utilized to evaluate the performance of the models. Addressing class imbalance in a dataset is crucial to ensure optimal model performance and accuracy. Two common techniques used to tackle class imbalance are undersampling and oversampling.

***Deployment* :** The CRISP-DM methodology's deployment phase is a crucial stage when the chosen predictive model is implemented and integrated into a production environment. In this stage, the model is operationalized and put into use so that stakeholders may benefit from the forecasts and insights it offers.The organization's infrastructure and current systems need to be connected with the predictive model. To guarantee that the integration is smooth and that the model performs as intended in the production environment, extensive testing is done. It is essential to regularly check the model's functionality and performance once it has been put into use. The model receives routine maintenance to resolve any problems or flaws that may appear, ensuring that it continues to function correctly. A record of the deployment process must be kept, including information on the model version distributed, the data utilized, and any changes made during deployment.The model's usage guidelines, explanations of its results, and any dependencies or prerequisites are all included in the documentation. Organizations may make sure that the chosen predictive model is successfully incorporated into their operations by adhering to these procedures throughout the deployment phase. The model's predictions and insights can help stakeholders, empowering them to take action and make educated decisions based on the model's outputs.

Figure 1

CRISP-DM Methodology for Telco Customer Churn Prediction



3. System Architecture

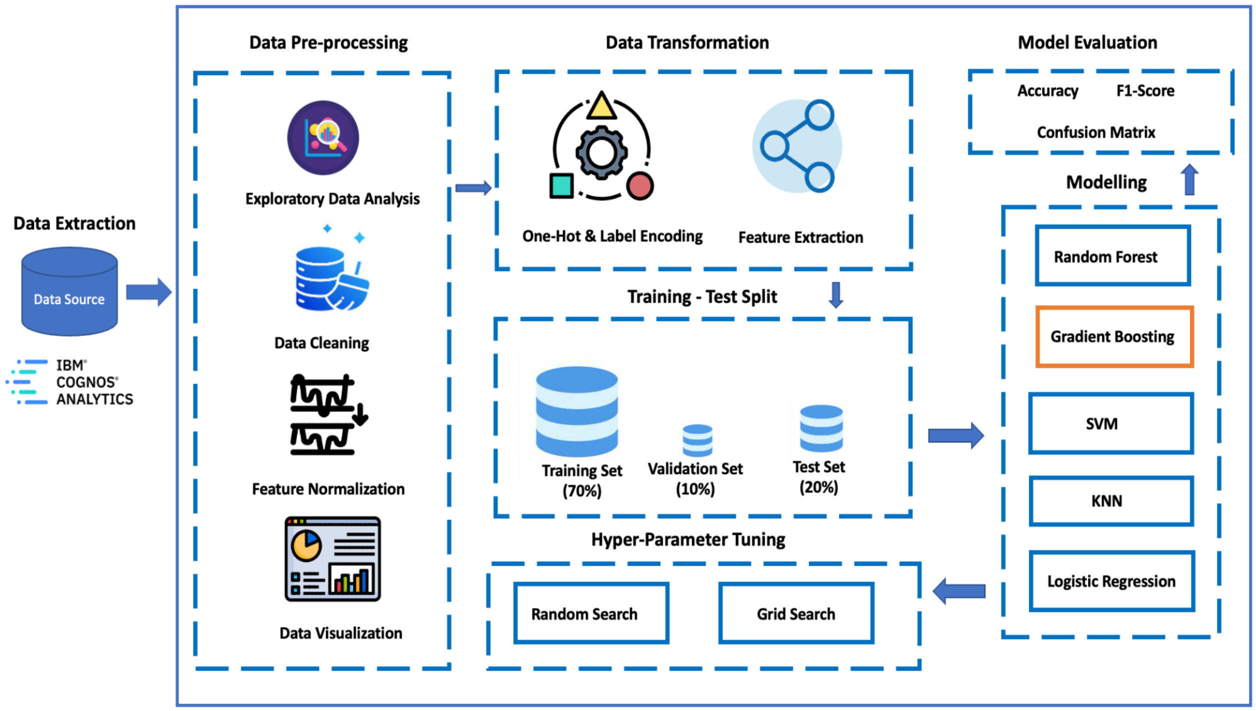
The data for this churn prediction has been taken from IBM Cognos Analytics, which is an analytics platform developed by IBM. The data is initially pre-processed, In which at first an exploratory data analysis is performed on the data to identify the patterns, potential relationships or anomalies present in the data and also gain initial insights from the data. Analysis is performed on different columns such as identifying churn distribution with respect to gender, churn distribution with respect to different services and customer payment method distribution to gain the insights from the data. Visualizations are employed to display the insights and trends found in the data. After this the raw data is checked for null values, The numerical columns such as total tenure, total charges had null values which are dropped and some records in the total charges column had null values but those records had a certain tenure associated with it, Hence those null values were replaced with mean total charges. Also in this step the customer ID column has been removed, In order to keep the data free from irrelevant information.Then normalization is performed on the columns. Min-Max normalization is applied on the numerical columns to keep them aligned to a fixed range. Such scaling of the numerical columns helps in maintaining a consistent scale, which in turn makes it easier to analyze and interpret the results.

Next, Certain transformation techniques are applied on the dataset. One-Hot encoding and label encoding are performed on the categorical columns present in the data in order to convert them into numerical column equivalents so that the machine learning model can effectively utilize them for making classifications and predictions. The dataset has multiple categorical columns such as Internet Service,OnlineSecurity on which one-hot encoding is applied and label encoding is applied on the gender column as gender inherent ordinal relationship. Also feature extraction is performed on the dataset in order to derive new columns from the data. Three new columns such as Monthly to Total charges ratio, Tenure times monthly charges, Has InternetService are created, These columns help in better analyzing the relationship between the numerical columns and also provide valuable insights and improve the performance of machine learning models. After this the dataset is checked for imbalance i.e if the classification present in the data is skewed or not. It is found that the dataset is significantly skewed, It is found that almost two third of the records contain no churn whereas the rest one third predict the customer churn. Having or using such an imbalance dataset results in the building of a biased machine learning model. Hence Undersampling technique is performed to balance the dataset.

After this, The balanced dataset is splitted into training, validation and testing datasets in a 70:10:20 ratio. In order to accurately predict the customer five different machine learning algorithms are used which are Random Forest, Gradient Boosting, Support Vector Machine,K-Nearest Neighbor and Logistic Regression. The train data is then fed to the model and these models are evaluated on the basis of different evaluation metrics such as accuracy,f1 score and confusion matrix. The results of these models are compared using the metrics also hyperparameter tuning is performed on these models. The best parameter grid is found using random search and grid search. These best parameters are then fed to the model to get the best accuracy of the model. The model with best accuracy is selected and then deployed into a production environment to start making real-world predictions using API's. The deployed model is also frequently monitored in order to ensure that the model delivers reliable predictions. Figure 1 shows the complete project work flow.

Figure 2

Architecture of Churn Prediction Model



4. Data Preparation

In the third phase of the CRISP-DM model, data preparation plays a crucial role in ensuring the quality of data before it can be used for modeling. This phase involves several key steps, including Exploratory Data Analysis, data preprocessing, and data transformation. Exploratory Data Analysis involves exploring and understanding the dataset by analyzing its statistical properties and identifying any potential outliers or patterns. Data preprocessing involves cleaning the data, handling missing values, and transforming the data into a format that can be used for modeling. This step includes techniques such as min-max normalization, label encoding, and one-hot encoding. Finally, the dataset is split into training, validation, and testing sets to evaluate the performance of the model.

4.1. Data Exploration

The data used in this project was obtained from IBM Cognos Analytics and comprises 21 variables that characterize the attributes of customers from a telecommunications company. The variables include customer demographics, account information, and usage patterns. A detailed view of the variables along with their data types is shown in the figure below. It is important to note that these variables were carefully chosen to study customer churn in the telecommunications industry.

Figure 3

Columns with datatype

```
customerID           object
gender               object
SeniorCitizen         int64
Partner              object
Dependents           object
tenure                int64
PhoneService         object
MultipleLines        object
InternetService      object
OnlineSecurity       object
OnlineBackup         object
DeviceProtection     object
TechSupport          object
StreamingTV          object
StreamingMovies      object
Contract             object
PaperlessBilling     object
PaymentMethod        object
MonthlyCharges      float64
TotalCharges         object
Churn                object
```

Once the data types of all columns in the dataset were verified, we proceeded to check the statistical properties of the numeric features. The purpose of this step is to understand the distribution and range of values for these features. The statistics that we calculated include measures of central tendency (such as mean, median, and mode) as well as measures of dispersion (such as standard deviation and range). These statistics can provide insights into the data and help in identifying any potential outliers or anomalies. The results of this analysis are presented in Figure below.

Figure 4

Statistical numerical analysis

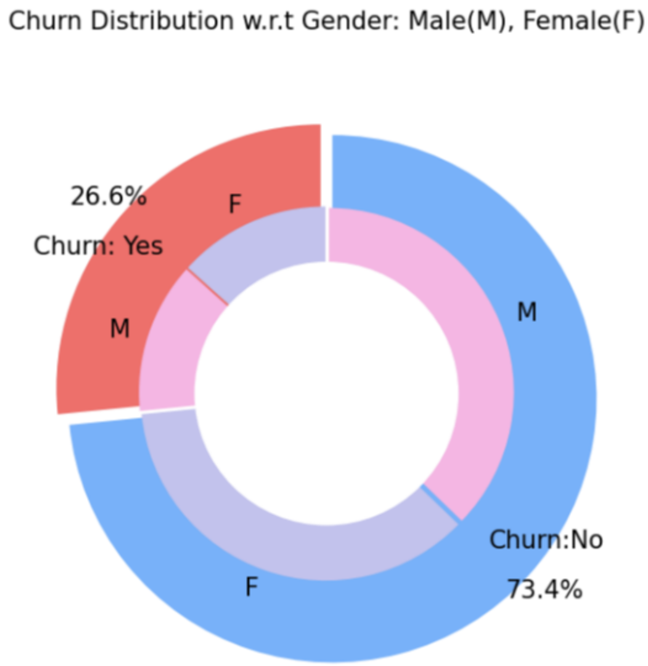
	tenure	MonthlyCharges	TotalCharges
count	7032.000000	7032.000000	7032.000000
mean	32.421786	64.798208	2283.300441
std	24.545260	30.085974	2266.771362
min	1.000000	18.250000	18.800000
25%	9.000000	35.587500	401.450000
50%	29.000000	70.350000	1397.475000
75%	55.000000	89.862500	3794.737500
max	72.000000	118.750000	8684.800000

Data Visualization

Regarding the visualization aspect, presented below are a few graphical representations of the dataset in order to facilitate better comprehension. These visualizations aim to provide insights into the data and allow for easier interpretation of the results.

Figure 5

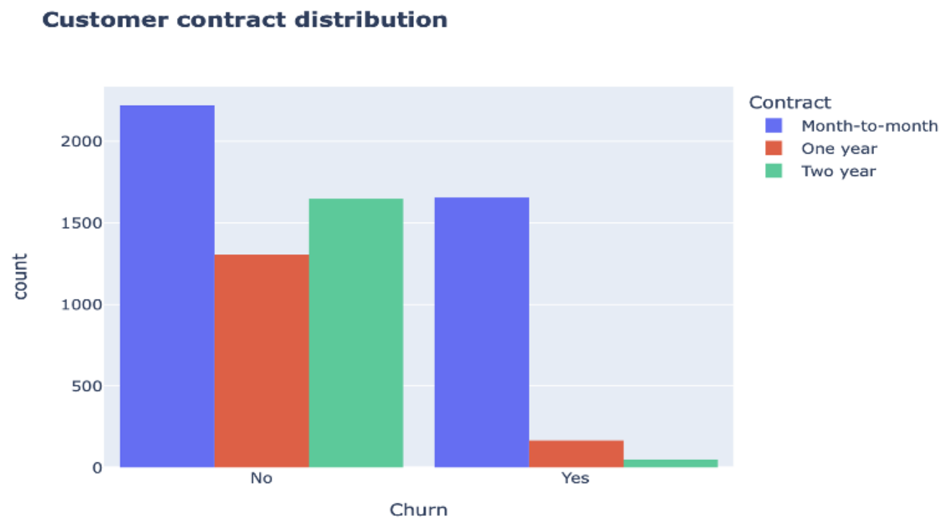
Churn Distribution wrt Gender



The pie chart above represents the distribution of churn rate based on gender. The blue portion represents customers who have churned, and the red portion represents those who have retained. It can be observed that the churn rate is almost the same for both genders, indicating that gender does not have a significant effect on customer churn.

Figure 6

Customer contract distribution



The above bar plot shows the distribution of customer contracts. Customers who opt for longer subscriptions, such as one-year or two-year subscriptions, have a much lower churn rate compared to those who opt for monthly subscriptions.

Figure 7

Churn Distribution wrt Internet services and gender

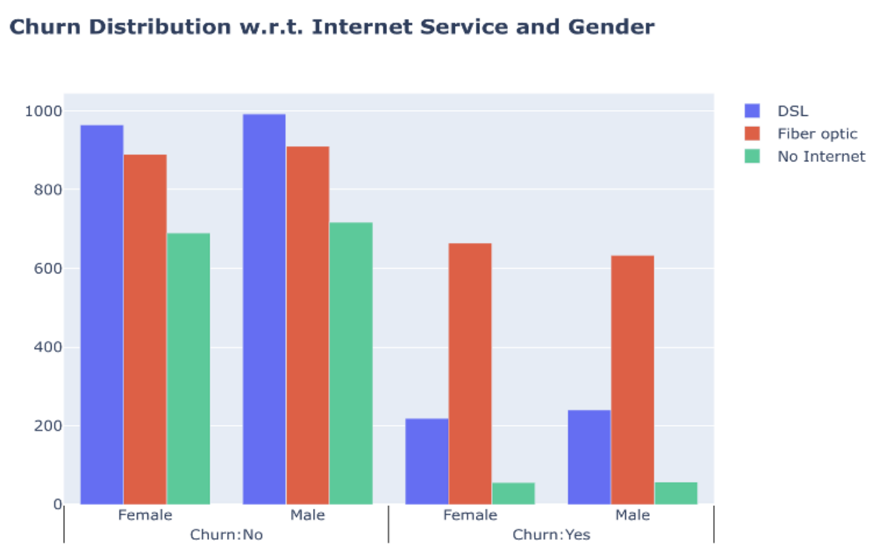
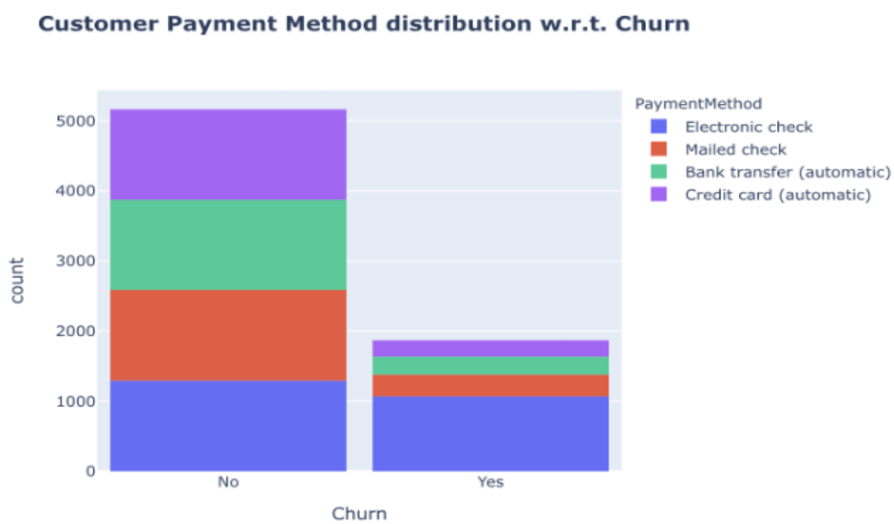


Figure above depicts the churn rate distribution based on Internet services and gender. The data suggests that gender has no significant impact on customer churn. However, customers who opt for fiber optic services have a higher churn rate, and those who do not have any internet service have a lower churn rate.

Figure 8

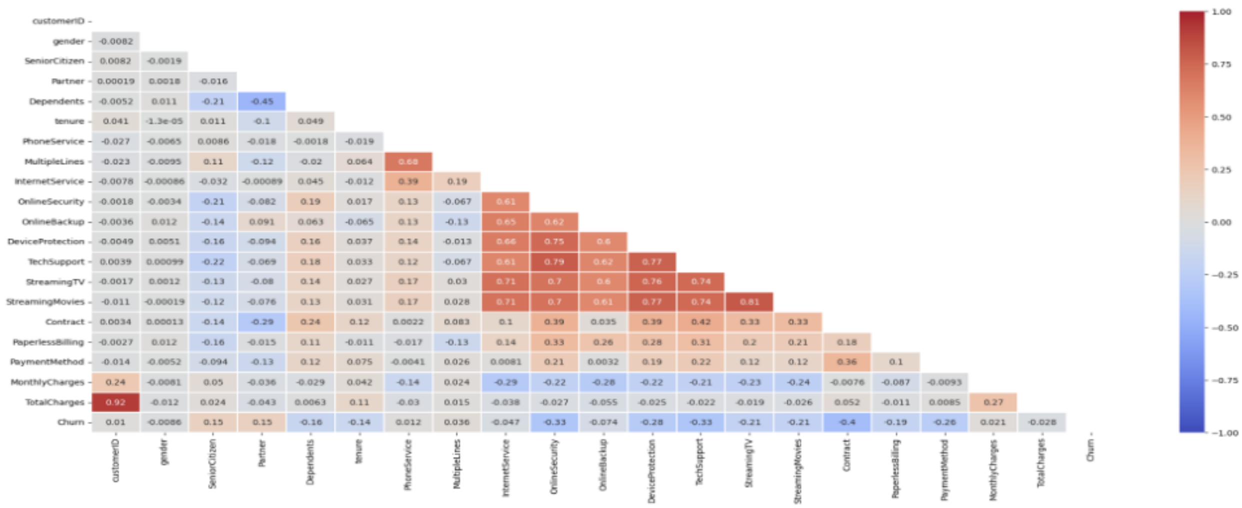
Customer payment method distribution wrt churn



The above bar plot shows the distribution of customer payment methods based on churn rate. The data indicates that customers who opted for electronic check payment methods have a higher churn rate. In contrast, other payment methods have almost similar churn rate ratios.

Figure 9

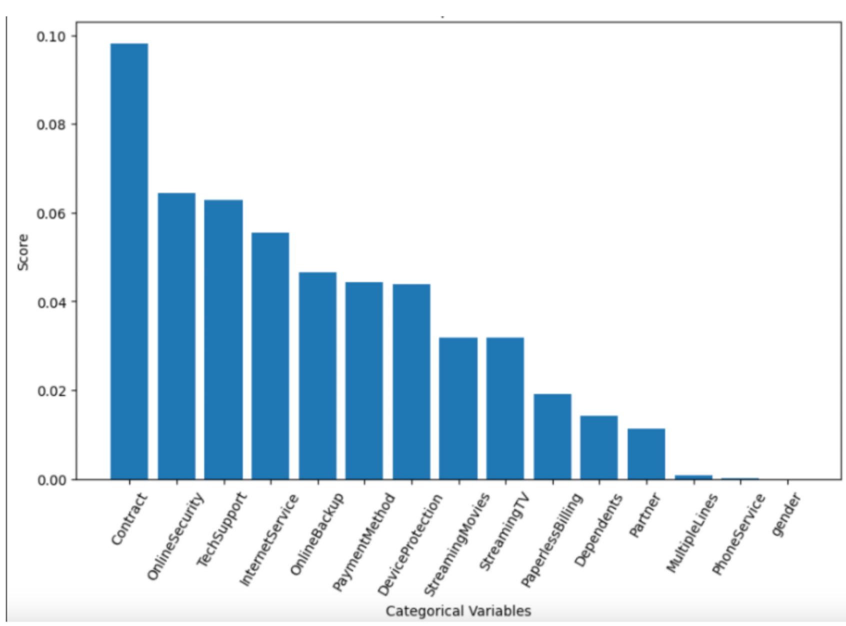
Correlation Matrix



The correlation matrix above shows the correlation between different columns in the dataset. Columns ranging from Internet services to Streaming TV have a high correlation, while gender and customer ID have the least correlation.

Figure 10

Feature importance score



Moreover, feature selection was performed to identify the most and least important features in the dataset. The analysis showed that gender had no significant impact, while the Contract column had a very high impact on customer churn.

4.2. Data Preprocessing

The figure below shows the Tenure column. The first line of code filters the dataframe to select all the rows where the tenure value is equal to 0. The output of this line is the index values of the selected rows, which are displayed as an Int64Index. In this case, the index values are [488, 753, 936, 1082, 1340, 3331, 3826, 4380, 5218, 6670, 6754]. The second line of code drops all the rows that have a tenure value of 0 from the dataframe using the drop method. The output of this line shows the index values of the rows that were dropped from the dataframe, which is an empty Int64Index. This indicates that all the rows with a tenure value of 0 have been successfully dropped from the dataframe.

Figure 11

Indexing

```
[28] df[df['tenure'] == 0].index

     Int64Index([488, 753, 936, 1082, 1340, 3331, 3826, 4380, 5218, 6670, 6754], dtype='int64')

     df.drop(labels=df[df['tenure'] == 0].index, axis=0, inplace=True)
     df[df['tenure'] == 0].index

     Int64Index([], dtype='int64')
```

Upon inspection of the data, it was discovered that there were some missing values (NaN) in the TotalCharges column, while all other columns had zero null values. To address this issue, the dropna method was applied, which removed the 11 NaN values from the TotalCharges column. As a result, the dataset no longer contained any null or NaN values. Figure () illustrates the presence of NaN values in the dataset, while figure below depicts the dataset with no null values.

Figure 12

Handling NaN values

OnlineSecurity	...	DeviceProtection	TechSupport	StreamingTV	StreamingMovies	Contract	PaperlessBilling	PaymentMethod	MonthlyCharges	TotalCharges	Churn
Yes	...	Yes	Yes	Yes	No	Two year	Yes	Bank transfer (automatic)	52.55	NaN	No
No internet service	...	No internet service	No internet service	No internet service	No internet service	Two year	No	Mailed check	20.25	NaN	No
Yes	...	Yes	No	Yes	Yes	Two year	No	Mailed check	80.85	NaN	No
No internet service	...	No internet service	No internet service	No internet service	No internet service	Two year	No	Mailed check	25.75	NaN	No
Yes	...	Yes	Yes	Yes	No	Two year	No	Credit card (automatic)	56.05	NaN	No
No internet service	...	No internet service	No internet service	No internet service	No internet service	Two year	No	Mailed check	19.85	NaN	No
No internet service	...	No internet service	No internet service	No internet service	No internet service	Two year	No	Mailed check	25.35	NaN	No
No internet service	...	No internet service	No internet service	No internet service	No internet service	Two year	No	Mailed check	20.00	NaN	No
No internet service	...	No internet service	No internet service	No internet service	No internet service	One year	Yes	Mailed check	19.70	NaN	No

Figure 13

Checking for NAN values post handling

```
df.isnull().sum()
```

```
customerID          0
gender              0
SeniorCitizen       0
Partner             0
Dependents          0
tenure              0
PhoneService        0
MultipleLines       0
InternetService     0
OnlineSecurity      0
OnlineBackup        0
DeviceProtection    0
TechSupport         0
StreamingTV         0
StreamingMovies     0
Contract            0
PaperlessBilling    0
PaymentMethod       0
MonthlyCharges      0
TotalCharges        0
Churn               0
dtype: int64
```

Min-max normalization rescales the values of a numeric feature to a fixed range between 0 and 1, where the minimum value in the feature becomes 0 and the maximum value becomes 1. This is done to avoid the influence of large values that may exist in some features, which can cause a bias in the machine learning model towards those features. In this project, the min-max normalization is applied to the tenure and MonthlyCharges columns of the Telco Customer Churn dataset to rescale the values between 0 and 1. This is done using the MinMaxScaler function from the Scikit-learn library in Python. Figure below depicts the outcome.

Figure 14

Numerical column values after Min-Max Normalization

	tenure	MonthlyCharges	TotalCharges
0	0.000000	0.115423	0.001275
1	0.464789	0.385075	0.215867
2	0.014085	0.354229	0.010310
3	0.619718	0.239303	0.210241
4	0.014085	0.521891	0.015330
...	...	...	...
7038	0.323944	0.662189	0.227521
7039	1.000000	0.845274	0.847461
7040	0.140845	0.112935	0.037809
7041	0.042254	0.558706	0.033210
7042	0.915493	0.869652	0.787641

7032 rows × 3 columns

Label encoding and one-hot encoding are two popular methods for encoding categorical variables in machine learning. Label encoding refers to the process of converting categorical values into numeric values. In label encoding, each category is assigned a unique integer value. For example, in the telco customer churn prediction project, the categorical feature "gender" has been encoded as 0 for "female" and 1 for "male". Figure below illustrates Label encoding for Gender feature.

On the other hand, one-hot encoding creates a binary variable for each category in a categorical feature. Each category is represented by a binary variable (0 or 1), and only one of these variables can have the value of 1 at any given time. In this project, the categorical feature "contract" has been converted into three features: "contract_month-to-month", "contract_one-year", and "contract_two-year". Then One-hot encoding is used in all these features. One-hot encoding is useful when the categorical variable has no inherent order or hierarchy, and when the number of categories is relatively small. Figure 16 illustrates One-Hot encoding technique which was used in the project to encode all the categorical features, such as "gender", "Partner", "Dependents", "PhoneService", "MultipleLines", "InternetService", "OnlineSecurity", "OnlineBackup", "DeviceProtection", "TechSupport", "StreamingTV", "StreamingMovies", "Contract", "PaperlessBilling", and "PaymentMethod". The encoded features were then used as inputs to the machine learning models to predict customer churn.

Figure 15

Label Encoding on Gender column

	gender	gender_encoded
0	Female	0
1	Male	1
2	Male	1
3	Male	1
4	Female	0
...	...	...
7038	Male	1
7039	Female	0
7040	Female	0
7041	Male	1
7042	Male	1

7032 rows × 2 columns

Figure 16

One-hot encoding on Categorical columns

StreamingMovies_No	StreamingMovies_No internet service	StreamingMovies_Yes	Contract_Month-to-month	Contract_One year	Contract_Two year	PaymentMethod_Bank transfer (automatic)	PaymentMethod_Credit card (automatic)
1	0	0	1	0	0	0	0
1	0	0	0	1	0	0	0
1	0	0	1	0	0	0	0
1	0	0	0	1	0	1	0
1	0	0	1	0	0	0	0
...	...	...	...	...	...	...	...
0	0	1	0	1	0	0	0
0	0	1	0	1	0	0	1
1	0	0	1	0	0	0	0
1	0	0	1	0	0	0	0
0	0	1	0	0	1	1	0

4.3. Data Transformation

Additionally, three new features have been derived based on the existing features. The first new feature is "hasInternetServices" which is derived from the two columns "InternetService" and "OnlineSecurity". The values in the "InternetService" column are either "DSL", "Fiber optic", or "No". The values in the "OnlineSecurity" column are either "Yes", "No", or "No internet service". If "InternetService" is "No", then "hasInternetServices" is set to "0". Otherwise, if "OnlineSecurity" is "No internet service" or "No", then "hasInternetServices" is set to "1", otherwise, it is set to "0".

The second new feature is "MonthlytoTotalCharges" which is calculated by dividing the "MonthlyCharges" column by the "TotalCharges" column. The third new feature is "TenureTimesMonthlyCharges" which is calculated by multiplying the "tenure" column by the "MonthlyCharges" column. These new features are useful in predicting customer churn. "hasInternetServices" can indicate if a customer has multiple services with the company, and

"MonthlytoTotalCharges" and "TenureTimesMonthlyCharges" can give an idea of the customer's financial commitment to the company. Figure shows the derived new feature.

Figure 17

Deriving new features

	HasInternetService	MonthlyToTotalChargesRatio	TenureTimesMonthlyCharges
0	1	1.000000	29.85
1	1	0.030140	1936.30
2	1	0.497920	107.70
3	1	0.022980	1903.50
4	1	0.466205	141.40
...	...	...	...
7038	1	0.042602	2035.20
7039	1	0.014016	7430.40
7040	1	0.085438	325.60
7041	1	0.242661	297.60
7042	1	0.015436	6972.90

7032 rows × 3 columns

The data is then checked for Imbalance. An imbalanced dataset has one class or category with significantly more occurrences than the other(s), which causes an unbalanced distribution of the classes. The dataset is checked and found to be imbalanced. The target column churn is significantly skewed in this dataset. The dataset shows that approximately 66% of the customers would likely continue with the company and only 33% would churn out from the company. Thus showing a class imbalance, The dataset shows that class ‘No’ significantly dominates the class ‘Yes’ Building models using such an imbalanced data would significantly impact the accuracy and would lead to erroneous predictions.

The issue of data imbalance can be resolved using undersampling techniques. undersampling involves reducing the number of samples in the majority class to achieve a more balanced distribution between the classes. The number of samples in the class ‘No’ are reduced with respect to the number of samples in class ’Yes’. Hence data balance is obtained. This balanced data is then splitted in a 70:10:20 ratio for training, validation and testing datasets. Validation dataset acts as an independent dataset separate from the training dataset which can be used to run and fine tune the model. The hyperparameter tuning will be initially performed on the validation data and checked for accuracy and also provides an estimation about the performance of the model. Utilizing a validation dataset is crucial to preventing overfitting. Overfitting occurs when a machine learning model learns to perform exceptionally well on the training data but fails to generalize to new data. The validation dataset makes it simpler to detect and mitigate overfitting.

5. Model selection

5.1. Models implemented

Extreme Gradient Boosting (XGBOOST)

Extreme gradient boosting is a popular ensemble learning algorithm that is known for high accuracy frequently employed for jobs involving predictive modeling, where the objective is to correctly forecast a target variable from a set of input features. One of XGBoost's primary advantages is its capacity to offer feature significance scores, which show the extent to which each feature adds to the model's ability to make predictions accurately. This data may be used to select the elements to focus on when building churn reduction initiatives,

Furthermore, XGBoost includes regularization options like L1 and L2 regularization, which helps in the prevention of overfitting and the improvement of generalization. Overfitting happens when a model is very complicated and overly well matches the training data, resulting in poor performance on fresh, unexplored data. Regularization prevents this by including a penalty term in the loss function of the model, which favors models with fewer variables that are less prone to overfit.

Figure 18

Classification table of Gradient Boosting

Accuracy: 0.786096256684492
f1_score: 0.7837837837837837

	precision	recall	f1-score	support
No	0.79	0.79	0.79	379
Yes	0.78	0.79	0.78	369
accuracy			0.79	748
macro avg	0.79	0.79	0.79	748
weighted avg	0.79	0.79	0.79	748

Support Vector Machines (SVM)

Support vector machine is a strong machine learning technique that may be utilized for both classification and regression applications. SVM is very effective when working with large datasets with many features, making it suited for feature spaces that are high-dimensional. SVM can be useful in context-dependent telecom datasets, which frequently contain a large number of factors that might possibly impact customer attrition. It seeks the hyperplane with the greatest margin between distinct groups or classes in the data. The margin is the distance that exists between the decision border (hyperplane) and the closest data points in each class. SVM enables improved generalization and can deliver strong predictions on unseen data by increasing this margin.

The application of a kernel function allows SVM to perform effectively in high-dimensional spaces. The kernel function enables SVM to implicitly transfer the input data into a higher-dimensional feature space, making it simpler to divide the classes or groups using a hyperplane. This is referred to as the "kernel trick." SVM's capacity to handle feature spaces with high dimensions and discover optimum boundaries for decisions with maximum margins makes it a good choice for telecom datasets with numerous variables that might affect customer turnover.

Figure 19

Classification table of SVM

```
Accuracy: 0.6216577540106952
f1_score: 0.5856515373352855
```

	precision	recall	f1-score	support
No	0.61	0.70	0.65	379
Yes	0.64	0.54	0.59	369
accuracy			0.62	748
macro avg	0.62	0.62	0.62	748
weighted avg	0.62	0.62	0.62	748

K-Nearest Neighbors

An approach for non-parametric machine learning known as K-Nearest Neighbors (KNN), KNN works well with tiny datasets, making it an excellent solution for data-limited issues. When the size of a dataset is minimal, it might be difficult to develop complicated models that generalize effectively. However, there is no such thing as a training period in KNN. During the prediction phase, it instead depends on the stored instances. KNN is endowed with this property, which enables it to make accurate predictions with fewer training instances. Because of its simplicity and versatility, KNN is especially effective in situations when getting a big labeled dataset is challenging or expensive to obtain. It is utilized for both regression and classification applications. It is a basic and straightforward algorithm that predicts based on its closeness to surrounding data points. The training data in KNN is made up of labeled examples with their accompanying class or regression values. When given a new, unlabeled point, KNN assigns a class or forecasts a value using the majority vote or an average of each value of its k nearest neighbors. KNN is often utilized in a wide range of applications, including systems for recommendation, recognition of images, mining texts, and anomaly detection. It is a valuable approach in a variety of machine learning applications due to its simplicity, versatility, and ability to catch local patterns.

KNN can perform well with small datasets, making it a suitable choice for problems with limited data. Here the Dataset is imbalanced and KNN can handle imbalanced data by using appropriate distance metrics or by applying oversampling or undersampling techniques. Here we have used an undersampling technique to Balance the data.

Figure 20

Classification table of KNN

```
 Accuracy: 0.6163101604278075
 f1_score: 0.6198675496688741
              precision    recall  f1-score   support

          No       0.63      0.60      0.61       379
         Yes       0.61      0.63      0.62       369

    accuracy                           0.62       748
   macro avg       0.62      0.62      0.62       748
weighted avg       0.62      0.62      0.62       748

Confusion Matrix - Gradient Boosting:
[[227 152]
 [135 234]]
```

Random Forest

An ensemble learning approach called Random Forest combines many decision trees using a random sample of the data and the characteristics. This method reduces overfitting, improves generalization, and makes the model more resistant to outliers. Because the technique can generate feature relevance ratings, it is useful for finding key variables in the dataset. Because of its accuracy, versatility, and robustness, Random Forest is frequently employed in a variety of disciplines.

Figure 21

Classification table of Random Foresting

```
Accuracy: 0.7606951871657754
f1_score: 0.7503486750348675
              precision    recall  f1-score   support

          No       0.75      0.79      0.77       379
         Yes       0.77      0.73      0.75       369

    accuracy                           0.76       748
   macro avg       0.76      0.76      0.76       748
weighted avg       0.76      0.76      0.76       748
```

Logistic Regression

For binary classification issues, logistic regression is a popular statistical modeling tool. where the purpose is to anticipate one of two probable outcomes or classes. In customer churn prediction, for example, the two classifications may be "churn" and "no churn." Logistic Regression simulates the link between input characteristics and the likelihood of the observation belonging to a specific class. For each observation, the probability of the positive class (such as churn) is estimated. Rather than explicitly predicting class labels, Logistic Regression produces probabilities ranging from 0 to 1. This can be viewed as the probability of a positive class observation. Setting a suitable threshold, such as 0.5, allows one to classify observations as positive or negative based on their projected probability. Logistic Regression is frequently used as a baseline model for binary classification tasks.

Figure 22

Classification table of Logistic Regression

```
Accuracy: 0.7687165775401069
f1_score: 0.7690253671562084
              precision    recall  f1-score   support

          No       0.78      0.76      0.77       379
         Yes       0.76      0.78      0.77       369

    accuracy                           0.77       748
   macro avg       0.77      0.77      0.77       748
weighted avg       0.77      0.77      0.77       748
```

5.2. Training model and results

Accuracy

Accuracy as another evaluation metrics to assess the performance of your models in predicting customer churn. Accuracy is a commonly used metric in classification tasks and provides a measure of how well the model predicts the correct class labels. Accuracy counts the proportion of correctly identified cases over all instances. It is a straightforward and understandable metric, however for unbalanced datasets, it may be deceptive.

It is crucial to highlight that a model's accuracy may not necessarily offer a whole view of model performance, particularly when dealing with unbalanced datasets. Accuracy can be deceptive in cases when the classes are unbalanced, such as in the prediction of customer churn, where the proportion of churned customers is frequently substantially lower than that of non-churned customers. The model may attain high accuracy by predicting the majority of the class (non-churn) the vast majority of the time. To measure the models' success in forecasting customer churn, accuracy is utilized in conjunction with other assessment criteria. It offers an initial insight of how well the models are working, but it should be read in conjunction with other metrics to provide a complete picture of the models' ability to predict.

Figure 23

Accuracy formula

$$\text{Accuracy} = \frac{\text{True Positives} + \text{True Negatives}}{\text{All Samples}}$$

Precision

A performance indicator called precision is employed to assess how well a model predicts the future. It evaluates the model's capacity to accurately pinpoint instances of positivity while reducing false positive predictions. It is especially helpful for anticipating customer turnover or other scenarios when the cost or impact of false positive predictions is large. A true positive in the context of anticipating customer churn would mean accurately identifying a client who had left, whereas a false positive would mean wrongly classifying a customer who had remained. Minimizing the number of false positives is the main goal of precision. The model is reliable in forecasting customer churn when it identifies a customer as churned if it has a high accuracy value, which means it has a low probability of false positives. Precision is a statistic that measures how well a model predicts the future. The ratio of true positives to the total of true positives and false positives is used to compute it. A high precision suggests a low number of false positives, which increases the model's accuracy in properly recognizing positive events. Accuracy must be considered in addition to other factors in order to provide a full evaluation of the model's performance.

Figure 24

Precision formula

$$Precision = \frac{True\ Positive(TP)}{True\ Positive(TP) + False\ Positive(FP)}$$

Recall

Recall is referred to as sensitivity or true positive rate, is a performance indicator used to assess a model's capacity to accurately detect all occurrences of positivity. It assesses how well the model captures the real positives while reducing the false negatives. Recall is especially vital in circumstances where it is essential to identify all successful occurrences, such as when forecasting customer attrition. Recall is focused on reducing the rate of false negatives. A high recall value means there are fewer instances of lost chances for intervention or retention because the model can successfully catch the majority of the real consumers who have churned. Recall is a statistic that measures how well a model can properly identify every positive case. A high recall means the model can successfully capture the majority of the real positive cases, lowering the possibility of missing chances for retention or intervention. Recall should be taken into account together with other metrics to provide a complete assessment of the model's performance.

Figure 25

Recall formula

$$Recall = \frac{True\ Positive(TP)}{True\ Positive(TP) + False\ Negative(FN)}$$

F1 score

The F1 score is a widely used metric to evaluate the performance of models in predicting customer attrition. It is particularly useful in binary classification tasks, such as forecasting churn, where the goal is to classify instances into two categories: churn or no churn. The F1 score takes into account both precision and recall, providing a balanced measure of a model's accuracy. The F1 score is calculated based on both precision and recall. Precision represents the ratio of true positives to all predicted positives, while recall represents the ratio of true positives to all actual positives. The F1 score is a metric that combines these two measures by computing their harmonic mean. This results in a balanced evaluation metric that takes into account both false positives and false negatives. The F1 score helps to establish a compromise between accuracy and recall, weighting both criteria equally. A higher F1 score suggests a better trade-off between accuracy and recall, indicating the capacity of a model to properly identify positive occurrences (churned clients) while avoiding false negatives and false positives. Analyze the models' success in forecasting customer turnover by utilizing the F1 score as one of the assessment measures, taking into consideration both recall and accuracy and accounting for the unbalanced nature of the dataset. This statistic aids you in comparing and choosing the top-performing model for deployment by demonstrating how well the models perform in precisely identifying consumers who have churned.

Figure 26

F1 score formula

$$F1\ score = 2 * \frac{Precision\ * Recall}{Precision + Recall}$$

Confusion Matrix

The confusion matrix as another evaluation tool is utilized to evaluate the performance of the machine learning models used in predicting customer churn. And the confusion matrix is categorized into four quadrants namely, True positives (TP), true negatives (TN), false positives (FP), and false negatives (FN) . Each quadrant reflects a distinct forecast outcome. The confusion matrix assists in understanding the model's predictions ,strengths and flaws. For example, a significant amount of false positives suggests the model is wrongly classifying many consumers as churned while they actually remain. A significant number of false negatives, on the other side, indicates that the model is overlooking prospective churned consumers. Confusion matrix has been a great tool for assessing the performance of models as well as gaining a deep grasp of customer churn forecasts. It enables one to accurately evaluate a model's accuracy and make Knowledgeable decisions based on the precise sorts of mistakes it creates.

Figure 27

Confusion Matrix table

	Predicted	
	Churn	Not Churn
Actual Churn	TP	FN
Actual Not Churn	FP	TN

Table 1

Baseline models

Model	**Accuracy**	**Precision**	**Recall**	**F1 Score**
Gradient Boosting	0.786	0.785	0.79	0.783
Random Forest	0.75	0.76	0.75	0.78
K-Nearest Neighbor	0.6163	0.62	0.615	0.6163
Support Vector Machine	0.62	0.63	0.62	0.58
Logistic Regression	0.76	0.77	0.77	0.76

5.3 Hyper Parameter Tuning

Hyperparameter tuning refers to the process of identifying and selecting the best values for the hyperparameters of a machine learning model. Hyperparameters are the values defined by the user and are not learned from the data. Hyperparameters have a significant influence over the model’s performance. Tuning of Hyperparameters is crucial as by selecting appropriate parameters values can greatly enhance the performance of the model and model’s generalization capabilities. There are several methods for tuning the hyperparameter models. For this project we have chosen Grid Search and Random Search.

Grid search is a method for hyperparameter optimization that is used to systematically find the ideal set of hyperparameter values for a machine learning model. Each hyperparameter must first have a set of values or a range specified before the model's performance is assessed for all possible combinations of these values.Grid search thoroughly explores all potential pairings of hyperparameter values, ensuring that the ideal configuration will be discovered within the given search space. However, it can be computationally expensive, especially if the search space is large or the model is computationally intensive.

Random Search is a hyperparameter optimization technique that involves randomly sampling hyperparameter values from a specified search space.In comparison to time-consuming methods like grid search, random search provides a more effective way to explore the hyperparameter space. It enables a more thorough examination of hyperparameter combinations and frequently identifies effective configurations without considering all potential pairings. When the search space is huge, the significance of several hyperparameters is ambiguous, or there aren't enough computational resources, random search is very helpful.

From the models we have developed, Gradient Boosting model is the one giving the highest accuracy hence both grid search and random search are performed on this model. The hyperparameter values considered for both grid and random search are n_estimators, learning_rate,max_depth,min_samples_split,min_samples_leaf,max_features. The n_estimators indicate the number of individual decision trees to be considered. The learning rate controls the contribution of each tree in the ensemble. It determines how fast or slow the model learns from the mistakes of the previous trees. Max_depth parameter determines the maximum depth of each decision tree in the ensemble. It restricts the number of nodes or levels in the tree. min_samples_split parameter sets the minimum number of samples required to split an internal node in a decision tree. Min_samples_leaf parameter defines the minimum number of samples required to be at a leaf node in a decision tree. Max_features parameter determines the number of features to consider when looking for the best split at each node. Table shows the parameter grid used for grid and random search along with the values found for gradient boosting model

Table 2

Hyperparameter tuning results

Method	Parameter Grid	Best Parameters	Accuracy	F1 Score
Random Search	'n_estimators': sp_randint(100, 1000), ' Learning_rate': [0.001, 0.01, 0.1] 'max_depth':sp_randint(3, 10), 'min_samples_split': sp_randint(2, 20), 'min_samples_leaf': sp_randint(1, 10), 'max_features': ['sqrt', 'log2', None]	N_estimators': 609 'Learning_rate': 0.01 'max_depth': 3 'Min_samples_split': 13 'min_samples_leaf': 1 'max_features': 'sqrt'	0.763	0.7665

Grid Search	n_estimators': sp_randint(100, 1000), ' learning_rate': [0.001, 0.01, 0.1] 'max_depth': sp_randint(3, 10), 'min_samples_split': sp_randint(2, 20), 'min_samples_leaf': sp_randint(1, 10), 'max_features': ['sqrt', 'log2', None]	N_estimators': 609 'Learning_rate': 0.01 'max_depth': 3 'Min_samples_split': 13 'min_samples_leaf': 1 'max_features': 'sqrt'	0.758	0.7621

6. Deployment

There are numerous critical processes involved in deploying a churn prediction machine learning (ML) model utilizing the CRISP-DM technique. Organizations must specify their goals and needs for churn prediction at the business understanding phase. The model may be deployed in a production environment for real-time predictions after it has been trained and evaluated. In order to receive input data for churn prediction, an interface or API must first be established as part of the deployment process. Web frameworks like Flask or FastAPI may be used to create this interface, which should conform to the anticipated input format for the model which would be accomplished in future. The API enables the model to acquire fresh customer data and generate in-the-moment predictions about which customers are most likely to leave. It is critical to guarantee that the interface is scalable and capable of effectively responding to a large number of incoming requests by hosting the application on AWS or GCP cloud environments.

Following the deployment of the churn prediction model, ongoing monitoring and upkeep are crucial. Monitoring entails keeping track of important indicators to evaluate the model's effectiveness and churn prediction accuracy. Measures of the model's efficiency include precision, recall, and the total accuracy of predictions. It can be essential to retrain or update the model if its performance declines over time or if it falls short of the acceptable accuracy levels. To enhance performance, this can entail adding fresh data, modifying hyperparameters, or investigating various techniques.

Another key stage in the deployment of a churn prediction model is integration with business processes. It is necessary to incorporate the churn estimates into the appropriate business workflows and decision-making processes. Stakeholders like customer retention teams or marketing departments should be given the results. Personalized offers, customer engagement programs, and targeted retention tactics may all be implemented using actionable information and suggestions based on attrition projections. The efficient use of the churn projections to influence business results and customer happiness is ensured by this integration.

Finally, it is crucial to assess the results of the implemented churn prediction model. It includes tracking customer satisfaction levels, revenue indicators, and client retention rates over time. Organizations may evaluate the success of the tactics deployed based on the predictions by comparing these metrics between before and after applying the churn prediction model. The model or technique may be improved upon iteratively to constantly maximize results after identifying potential areas for improvement.

7. Source Code Link

The source code is publicly accessible and is at the following link:
https://drive.google.com/file/d/1Ko07hos3zDVqwUqL7krSrIy8o4_Ngy_s/view?usp=share_link

Conclusion

In this project we aimed to use a variety of telco parameters in this project to forecast customer attrition in the telecom sector. We used the CRISP-DM technique, which enabled us to address the issue methodically and create a powerful churn prediction model. The effectiveness of the model was assessed using a variety of measures, and the top model was chosen based on accuracy, F1 score, and overall predictive ability. The implemented model offers the telecom sector a useful tool for identifying clients at threat of leaving, allowing proactive retention efforts and wise decision-making. It is important to remember that the success of the deployed model depends on ongoing maintenance, periodic updates utilizing fresh data, and regular monitoring to guarantee its sustained efficacy in foretelling client turnover. This study shows the value of data analysis, preprocessing, modeling, and assessment in creating a reliable customer churn prediction system. With the knowledge gathered from this study, telecom firms may create focused strategies and take proactive steps to reduce customer churn, improve customer retention, and promote company growth.

Future Scope

The project's long-term objective is to successfully introduce the customer churn prediction model into a telecom company's production environment. By reaching this objective, the initiative hopes to provide these businesses the ability to actively recognize and keep consumers who are at risk of leaving, thereby enhancing customer happiness and financial results. For telecom firms, using the customer churn prediction model in a production setting provides a number of advantages. Additionally, the implementation of the churn prediction model may increase customer satisfaction. Telecom firms may improve the customer experience by taking proactive steps to resolve possible problems or concerns that could cause churn. As a result, there may be an improvement in total client retention, customer satisfaction levels, and customer loyalty. Overall, telecom businesses' approach to customer retention and happiness might be completely changed if the customer churn prediction model is successfully implemented. These firms may proactively handle customer churn, improve customer happiness, and achieve better financial results by utilizing sophisticated analytics and predictive modeling methodologies.

References

"Customer churn prediction in telecommunications using decision trees"

Authors: Syed Amjad Ali, Nadeem Javaid, Waleed Bin Shahid, et al.

Published in: 2017 14th International Bhurban Conference on Applied Sciences and Technology (IBCAST) https://ieeexplore.ieee.org/abstract/document/7880149

"A novel approach for customer churn prediction in telecommunication industry"

Authors: Rabia Noor Enam, Aniqa Humayun, Muhammad Asif, et al.

Published in: 2017 4th International Conference on Control, Decision and Information Technologies (CoDIT) https://ieeexplore.ieee.org/abstract/document/8102640

"Customer churn prediction in telecommunications industry using machine learning techniques"

Authors: Sumeet Kumar, Anu Chaudhary, Aditya Khamparia, et al.

Published in: 2018 5th International Conference on Signal Processing and Integrated Networks (SPIN) https://ieeexplore.ieee.org/abstract/document/8474232

"Customer churn prediction in telecom using big data analytics: A case study of a telecommunication company in India"

Authors: Akashdeep Bhardwaj, Rajkumar Jain

Published in: 2016 International Conference on Computational Techniques in Information and Communication Technologies (ICCTICT)

https://ieeexplore.ieee.org/abstract/document/7519326

"Customer churn prediction in telecommunications using ensemble methods and principal component analysis"

Authors: Jingjing Gao, Min Zhang, Chen Gong

Published in: 2016 8th International Conference on Information Technology in Medicine and Education (ITME) https://ieeexplore.ieee.org/abstract/document/7815179

Sana, J. K., Abedin, M. Z., Rahman, M. S., & Rahman, M. M. (2022). A novel customer churn prediction model for the telecommunication industry using data transformation methods and feature selection. *PLOS ONE*, *17*(12), e0278095. https://doi.org/10.1371/journal.pone.0278095