GBA 6210 – Data Mining for Business Analytics

Dr. Shuo Zeng

Case Study

Customer Retention – Telco Customer Churn Dataset

By

Norberto Limon, Sepideh Rahmati, Christian Gonzalez

Executive Summary (1 page)

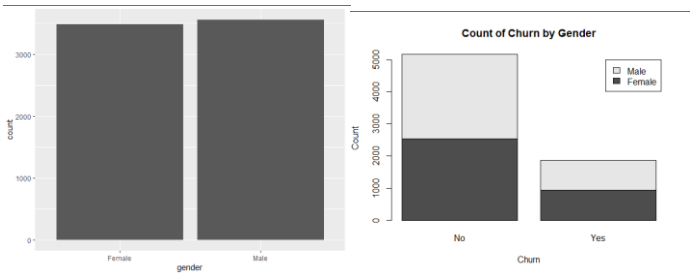
Telco, a telecommunications company, wants to create a model to predict the churn rate, termination of existing contracts, to avoid losing customers and in return increase its revenue. Telco wants to have a proactive approach therefore the exploration of data is necessary to see what variables are causing churn. Data must be explored by visualization and summary statistics to better understand the problem. Data must also be cleaned and positioned by setting dummy variables to be inputs for the categorical data models. Logistic Regression Model and Naïve Bayes were the two models chosen to solve the problem.

Task 1

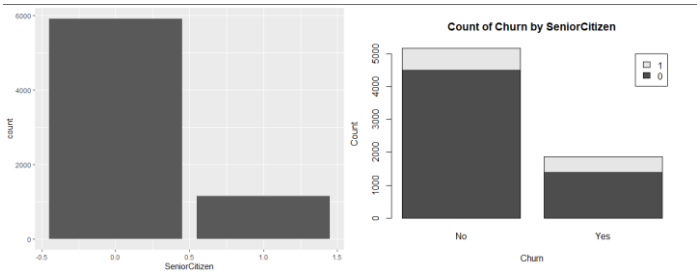
Data exploration (required): use proper summary statistics and visualization tools to understand the distribution of individual variables and relationship between two or more variables.

Developed visualizations of all variables individually as well as visualizations of the relationship between all variables with the dependent variable "Churn." Used the ggplot() function with geom_bar() to develop a bar chart visualization for all categorical variables. Used hist() function to develop a histogram visualization for all continuous variables. Used barplot() function to develop a stacked bar chart for visualizing the relationship between two categorical variables and used geom_boxplot() to develop a visualization between a continuous and categorical variable. Finally, we used corPlot() to find the correlation between the numerical variables. The following are the visualizations for the variables and their relationship.

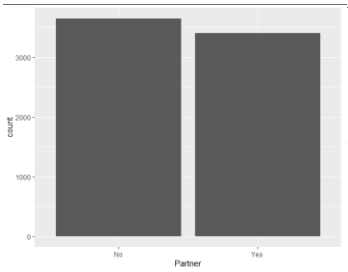
Gender:

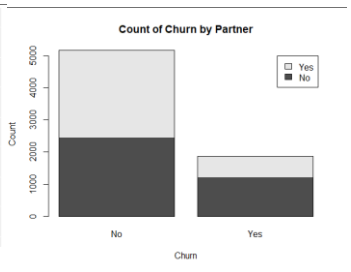


Senior Citizen:

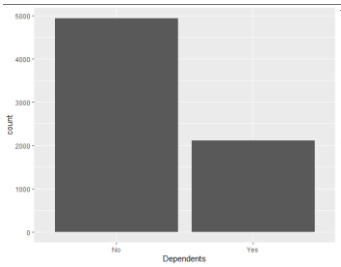


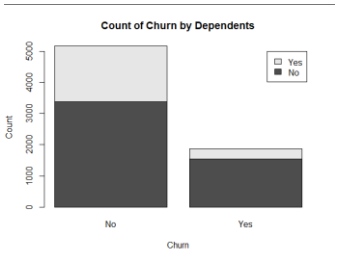
Partner:



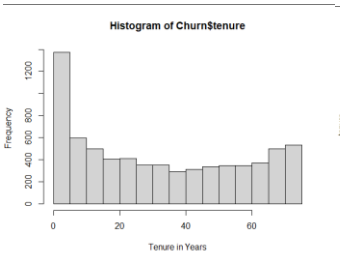


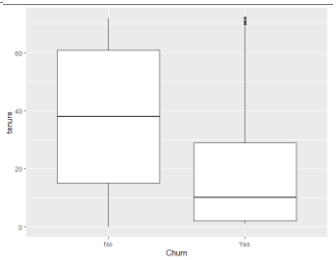
Dependents:



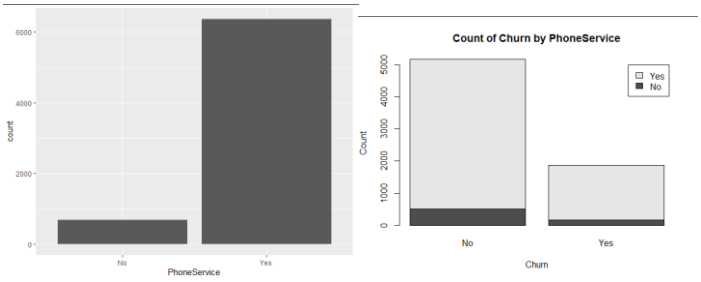


Tenure:

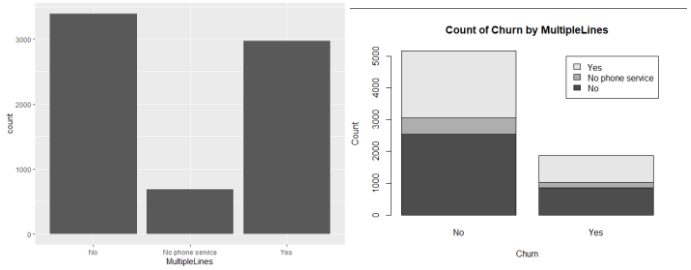




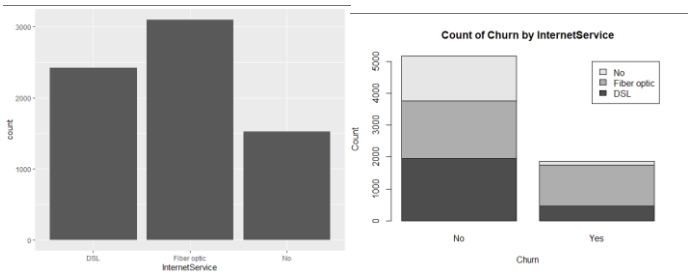
Phone Service:



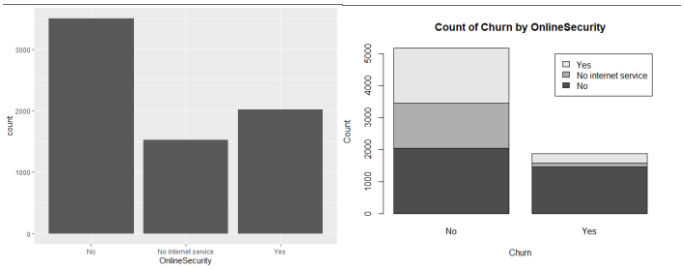
Multiple Lines:



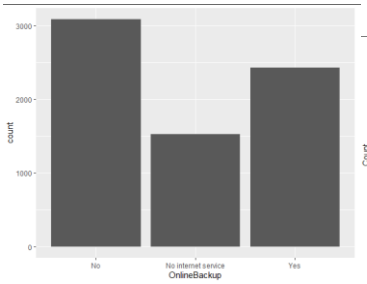
Internet Service:

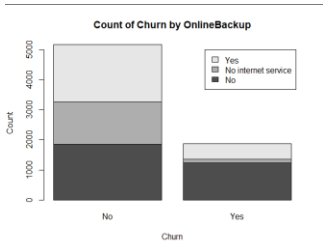


Online Security:

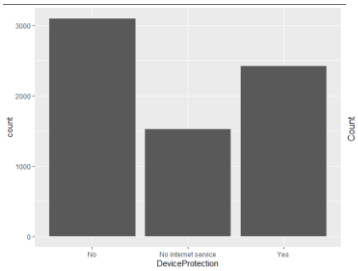


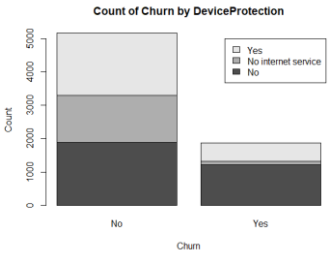
Online backup:



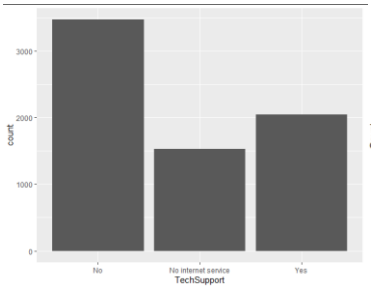


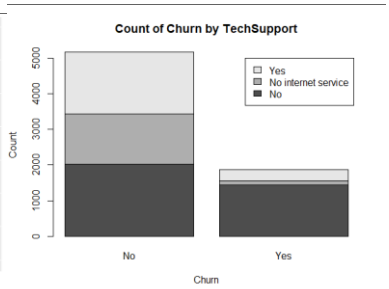
Device Protection:



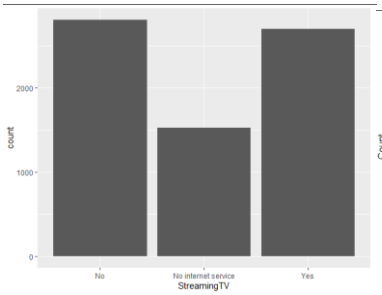


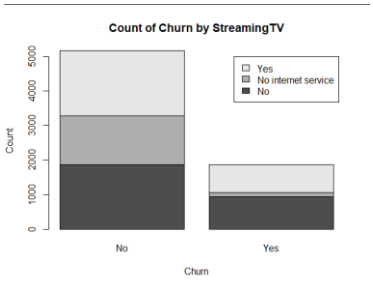
Tech Support:



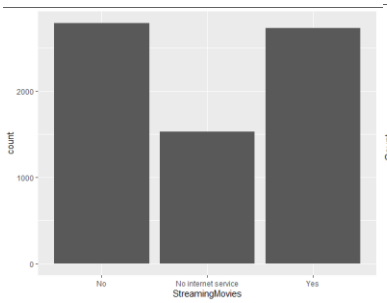


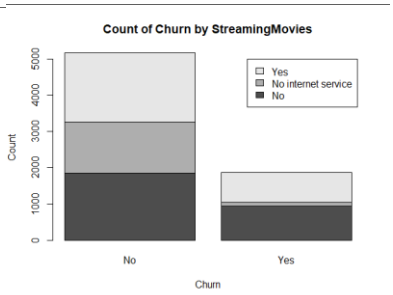
Streaming TV:



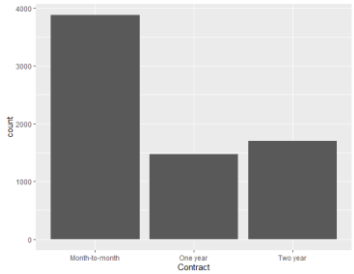


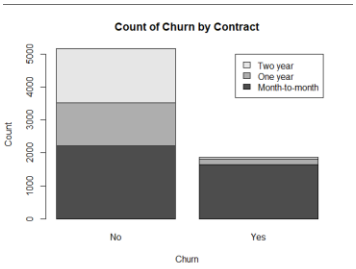
Streaming Movies:



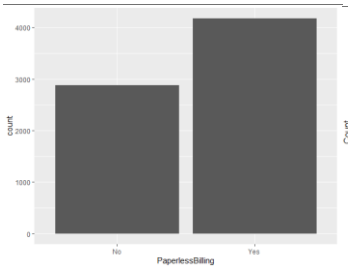


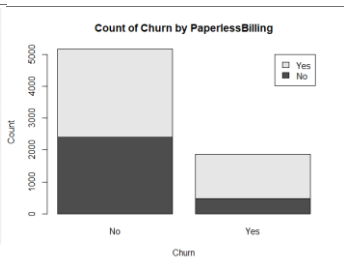
Contract:



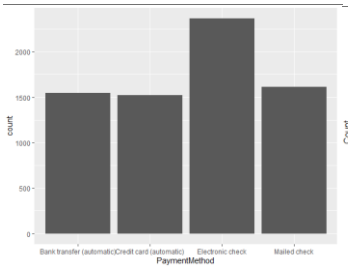


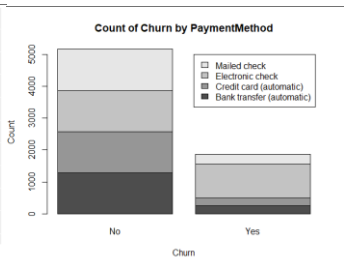
Paperless Billing:



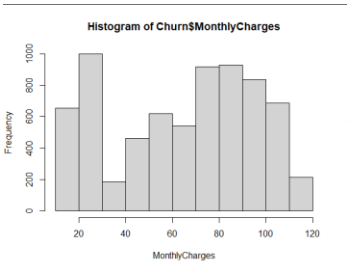


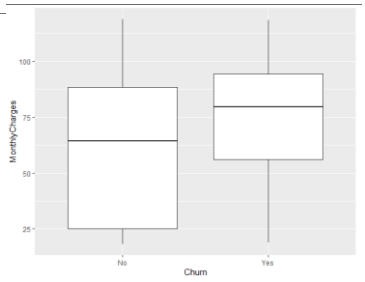
Payment Method:



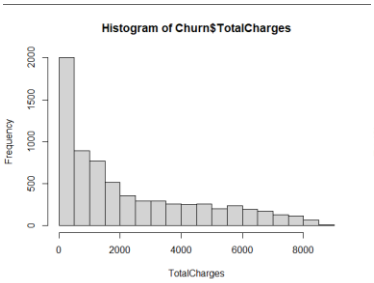


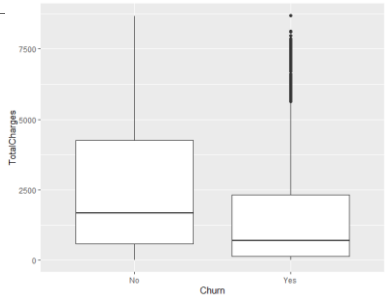
Monthly Charges:



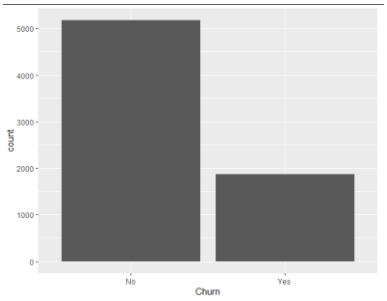


Total Charges:

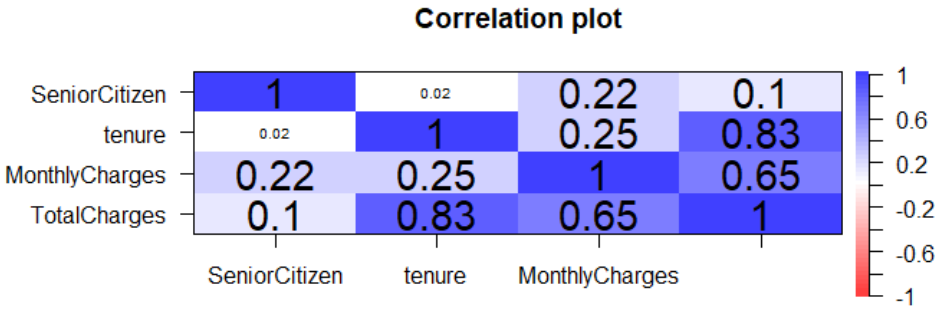




Churn:



Correlation Matrix:


Correlation plot
SeniorCitizen
tenure
MonthlyCharges
TotalCharges
1
0.02
0.22
0.1
0.02
1
0.25
0.83
0.22
0.25
1
0.65
0.1
0.83
0.65
1
SeniorCitizen
tenure
MonthlyCharges
1
0.6
0.2
-0.2
-0.6
-1

Task 2

Data preprocessing (optional): detect and remove outliers, compute new variables from existing variables if necessary.

To remove outliers, we first determined the variables that had numerical values. We used ± 3 standard deviations from the mean to determine the lower and upper bounds to eliminate outliers. We found no outliers in the dataset.We discretized the following variables: Tenure, MonthlyCharges, and TotalCharges. We broke tenure down into 3 variables low, medium, and large. We broke monthly charges and total charges into 4 intervals low, medium, large, and very high. We then trimmed out the original variables out of the dataset.

We then created dummy variables for InternetService, OnlineSecurity, OnlineBackup, DevicProtection, TechSupport, StreamingTV, StreamingMovies, Contract, PaperlessBilling, PaymentMethod and Churn. We created three dummy variables for InternetService: no_internet, fiber_internet, and dsl_internet with each having 1s and 0s as their values. We split OnlineSecurity into two dummy variables no_online_security and yes_online_security. OnlineBackup was split into two dummy variables no_online_backup and yes_online_backup. Device protection into two dummy variables no_online_protection and yes_online_protection. TechSupport into two dummy variables no_tech_support and yes_tech_support. StreamingTV into two dummy variables no_tv_streaming and yes_tv_streaming. StreamingMovies into two dummy variables no_movie_streaming and yes_movies_streaming. PaperlessBilling into two dummy variables no_paper_billing and yes_paper_billing. All these variables have 1s and 0s as their values in order to be interpretable by our classifier models. 1s indicates yes to each variable and 0s indicates no. Contract was split into three dummy variables month_to_month_contract, one_year_contract, and two_year_contract. PaymentMethod was split into four dummy variables payment_echeck, payment_check, payment_bank_transfer, and payment_credit_card.

After all the variables were broken down into categories that could be identified with discrete values, we used the subset function to specify the variables that we were going to use while omitting the unused variables that they were derived from. This brought our initial number of columns from 21 to , and then down to 46 (+1 for churn column). In the next step we used

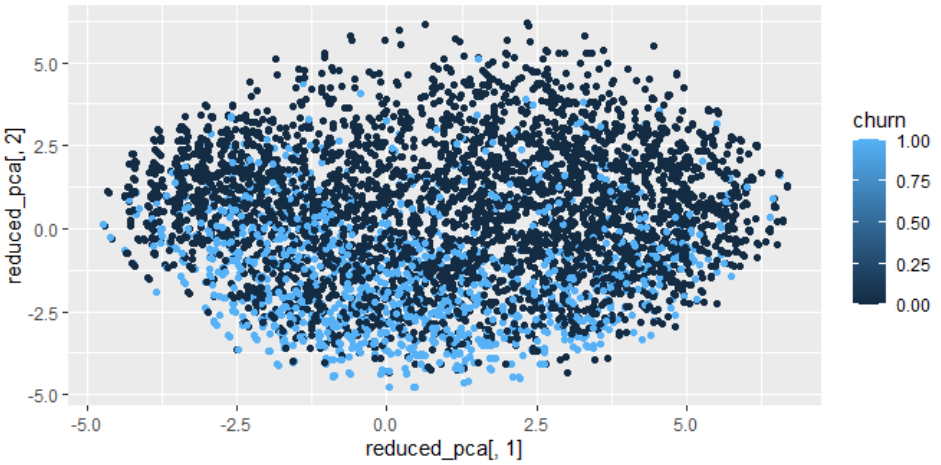
dimensionality reduction to further bring that number down to only 27 Principal Components (plus the independent y_churn variable for a total of 28).

Task 3

*Data and dimension reduction **(optional)**: choose a subset of variables for model building (dimension reduction), partition the records into homogeneous groups (using unsupervised learning techniques such as cluster analysis) and build separate models (data reduction).*

In this section we used Principle Component Analysis to restructure the data in a way that highlighted the most prevalent components that contributed to the outcome of our independent variable, which was churn. After inspecting the discretized data set of numeric 1s and 0s we were able to standardize the data using the scale function. We then performed the Principle Component Anlysis (PCA) by applying the prcomp function to the now standardized data. By viewing a summary of the pca data we found that the first 27 components already accounted for about 99% of the churn rate. Since the PCs were created in order of its variance contribution to the churn rate we set the cutoff point at 27 and reduced the pca data into just he first 27 components. In the end this meant we were able to reduce the data from 46 discretized binomial columns to 27 columns. This could have been reduced further depending on the level of comprehensiveness that is desired. However while it may or may not have been the most efficient choice, after 27 it was very clear that the relative contribution of the remaining PCs was negligible (at less than 1%), so it was easy to settle on that as a notable cutoff point.

After reducing our dimensions with PCA, we decided to use an index matrix to split the data into training and testing dataframes as a 90-10 split. This was done simply by applying the index matrix as the row discriminator and its inverse for the selection of the subset of test data from 'FINAL.DATA'. We also used the trainControl method in this step to apply 10-fold cross validation to the models discussed in Task 5.


5.0
2.5
0.0
-2.5
-5.0
reduced_pca[, 2]
-5.0
-2.5
0.0
2.5
5.0
reduced_pca[, 1]
churn
1.00
0.75
0.50
0.25
0.00

Task 4

Partition the data (required): use 10-fold cross validation to evaluate model performance. Please refer to the description of k-fold cross validation in my slides on data mining process (Chapter 2), and implement 10-fold cross validation using "for" loops.

To partition the data into 10-folds, instead of using a for loop we used the function trainControl() with method set to cv and number set to 10. Then when we trained / built our models we incorporated that setting data into our Logistic Regression and Naïve Bayes models.

Task 5

Model building (required): build at least two classification models using only the algorithms we have introduced in class, at least one of the built models must be interpretable (in terms of model structure and parameters, which essentially reflects the factors that contributed to customer's decision to churn).

For this project we chose to use Logistic Regression Model and a Naïve Bayes Classifier to extrapolate from the data, as both models were particularly well-suited for binary classification. To train / estimate our models we appended y_churn from the discretized dataset to the reduced PCA dataset. We then used the reduced PCA data and trained it to the y_churn column, using the train function, with the method and family attributes set to 'glm' & 'binomial' for the Logistic Regression Model and 'nb' & 'gaussian' respectively for the Gaussian Naïve Bayes model. We also applied the trControl settings we saved in the previous step using the trainControl function in order to apply 10-fold cross-validation to our models.

Task 6

Model evaluation (required): evaluate performance of each model built. Specifically, record and pool all training outcomes and validation outcomes from all folds separately, and evaluate model performance using the following metrics and tools: accuracy, sensitivity, specificity, precision, FDR, FOR, ROC chart, AUC of ROC curve, and Lift chart. Compare the performance between training and validation and discuss whether there is any overfitting issue or not.

For this section we compared the output of the logistic and Bayesian models. Accuracy for the logistic regression model was found to be 0.7963 and kappa was calculated as 0.4476. Conversely the Bayesian model registered a 0.7348 retention rate and a 0.2651 rate lost to customer churn. On the test data for logistic regression the model was found to have an accuracy of 0.8023 and regarding the matter of sensitivity it was calculated to be 0.5916 and the specificity was found to be 0.8809 whereas the Bayesian classifier was determined to have an accuracy of 0.7696, a sensitivity rate of 0.5916 and a specificity rate of 0.8359 using the test data. All together this seemed to indicate that the Logistic Regression Model was optimal choice in this scenario.

Task 7

Model deployment (required): choose one of the interpretable algorithms from step 5 and build a model using the whole dataset. Interpret the model from both technical and managerial perspectives.

Finally after comparing the two models with our training data, we selected our Logistic Regression model for the final deployment. Accuracy of the model was 0.7923 and kappa score was 0.4403

Appendix

Description

A telecommunication company (known as Telco company) provides subscription-based telecommunication service, which is its major revenue source. In order to grow their revenue generating customer base, it is important for a Telco company to attract new customers as well as avoid termination of existing contracts, which is known as churn. Customer turnover, or churn rate, is the percentage of a company's customer base lost during a given period of time, usually on monthly or annual basis. A high churn rate may hurt revenue and profit badly. Many different reasons may trigger customers to terminate their contracts, such as better price offers and/or more interesting packages from competitors, bad service experiences, or change of customers' personal situations.

In order to reduce churn rate, many Telco companies adopt a reactive approach: if a customer called with a request to cancel his or her contract, then the customer service representative would try to convince the customer to extend the contract, most often by offering free services or discounts on existing services. However, it would be more effective to estimate the probability that a given customer would churn in the near future, identify the factors that contributed most to that customer's decision, and then actively reach out to the customer to enhance his or her service experience and divert churn without giving up costly discounts. The goal of this project is to build a classification model using R to predict customer churn (probability and classification) for a Telco company.

Dataset

Dataset used in this project comes from IBM sample datasets from Kaggle (Telco-Customer-Churn.csv). It contains 7043 rows and 21 columns. Each row represents a customer, and each column contains one of customer's attributes, described below.

Customers demographic information

customerID — Customer ID

gender — Whether the customer is a Male or a Female

SeniorCitizen — Whether the customer is a senior citizen or not (1, 0)

Partner — Whether the customer has a partner or not (Yes, No)

Dependents — Whether the customer has dependents or not (Yes, No)

Customer account information

tenure — Number of months the customer has stayed with the company

Contract — The contract term of the customer (Month-to-month, One year, Two year)

PaperlessBilling — Whether the customer has paperless billing (Yes, No)

PaymentMethod — The customer's payment method (Electronic check, Mailed check, Bank transfer (automatic), Credit card (automatic))

MonthlyCharges — The amount charged to the customer monthly

TotalCharges — The total amount charged to the customer

Customer services booked

PhoneService — Whether the customer has a phone service (Yes, No)

MultipleLines — Whether the customer has multiple lines (Yes, No, No phone service)

InternetService — Customer's internet service provider (DSL, Fiber optic, No)

OnlineSecurity — Whether the customer has online security (Yes, No, No internet service)

OnlineBackup — Whether the customer has online backup (Yes, No, No internet service)

DeviceProtection — Whether the customer has device protection (Yes, No, No internet service)

TechSupport — Whether the customer has tech support (Yes, No, No internet service)

StreamingTV — Whether the customer has streaming TV (Yes, No, No internet service)

StreamingMovies — Whether the customer has streaming movies (Yes, No, No internet service)

Classification labels

Churn — Whether the customer churned or not (Yes or No)