ARE PREDICTIVE VARIABLES AFFECTING THE PROBABILITY OF CHURN?

PREDICTIVE MODELING – D208

PERFORMANCE ASSESSMENT

TASK 2: LOGISTIC REGRESSION FOR PREDICTIVE MODELING

STUDENT NAME: IBRAHIM SULEIMAN

DATA ANALYTICS

STUDENT NUMBER: 001429984

COLLEGE OF INFORMATION TECHNOLOGY, WESTERN GOVERNORS UNIVERSITY

NAME OF INSTRUCTOR: KEIONA MIDDLETON

SEPTEMBER 11TH , 2022

**Part I: Research Question**

A.  **Describe the purpose of this data analysis by doing the following:**

**1.  Summarize one research question that is relevant to a real-world organizational situation captured in the data set you have selected and that you will answer using logistic regression.**

***The summary includes 1 research question relevant to a realistic organizational situation and can be addressed using the selected data set and logistic regression.***

Are predictive indicators impacting the probability of churn, this would be our research question. To predict the categorical variable "Churn" and to use a logistic regression model. By employing this model, the company can help themselves predict which customers are most likely to abandon them so that they can make attempts to retain them, and they can also get an assessment of their services and customer service. Churning is described as whether the customer was retained by the business or otherwise.

2.  **Define the objectives or goals of the data analysis. Ensure that your objectives or goals are reasonable within the scope of the data dictionary and are represented in the available data**.

***The submission clearly defines the objectives or goals of the data analysis, and the objectives or goals are reasonable for the scope of the scenario and are represented in the available data.***

The goal is to determine whether the services that clients purchase might encourage them to remain. More robust marketing strategies may be created both for the prospective and current customers by analyzing the services that are integral to retention.

**Part II: Method Justification**

**B.  Describe logistic regression methods by doing the following:**

1. **Summarize the assumptions of a logistic regression model.**

***The submission accurately summarizes the assumptions of a logistic regression model.***

The following are a number of the principles underlying logistic regression methods:

The dependent variable can sometimes be ordinal or binary.

• The discoveries in the sample of data are unconnected from one another.

• There should not be substantial multicollinearity, if there are any.

• Log odds as well as independent variables ought to be linear.

• A significant statistical sample (as per usual, at least 5 - 10 instances of the least frequently response for every independent variable).

Binary choice models' factor values should be quantified and scaled. A binary choice model can also incorporate categorical variables as variables. Therefore, in binary choice models, a regression model is constructed based on the dependence of the probability that the resulting dichotomous variable would take the value of 0 or 1 for a given value of the factors.

An exclusive monotonically growing function that can only accept values between 0 and 1 is used to model the probability of a dichotomous dependent variable.

**2.  Describe the benefits of using the tool(s) you have chosen (i.e., Python, R, or both) in support of various analysis phases.**

***The submission describes the benefits of using the tool(s) chosen in support of the various phases of the logistic regression analysis, and the benefits logically align with the goal of the analysis.***

**Python**: is a programming language that is well-used to read, assemble and organize code capable of execution on several interfaces and platforms designated by workload management systems due to its powerful indentation.

**The Benefits of using Python for this data analysis**

· Eco-friendly environment – The user-friendly environment python entails provides a fast and easy way to adapt to the functions and perform code execution efficiently.

· Python is practical for implementing, reading code, detecting missing values and outliers in the data set, and creating visual representations to spot animalities and identify outliers during the data cleaning execution

· Python made it systematic for the 'Churn\_data.csv' file to be imported for the cleaning operations to begin.

In addition, Python allowed the importation of packages into the new environment, such as Pandas, NumPy, Matplotlib, Sklearn, and Seaborn. These packages offer a variety of features, such as creating visualizations of histograms, boxplots, and data tables. Without a doubt, these packages, alongside the programming languages, are user-friendly, ideal, and intuitive in providing data analysts with efficiency and error-free output in an innovative presentation opposing other tools (Michael Galarnyk,2018).

· **Pandas**: pandas is a software library drafted for Python operating systems for data handling alongside inquiry. Specifically, it presents data design alongside procedures for executing arithmetic groups.

**The Benefits of Pandas**

· It implements a quick and dynamic strategy to take care of data

· It is straightforward to treat data omitted values.

**NumPy**: NumPy is a Python library applicable for dealing with arrays. In addition, it contains operations for functioning in the domain of linear algebra alongside matrics.

**Benefits of Numpy**

· NumPy's arrays appear to be less in proportion compared to Python lists

· The quick execution is magnificent because it acts rapidly in computing than python lists

· **Matplotlib**: Matplotlib is a cross-structure, data representation, and graph plotting library for Python alongside its binary extension NumPy

**Benefit of Matplotlib**

· It presents the user with an interface to represent data by applying various sorts of plots to communicate the data effectively

· We can execute multiple sorts of plots (scatterplots, histograms, bar charts, error charts, boxplots, etc.) by executing a scanty line of code in Python

· **Sklearn**: Scikit-learn is a suitable and powerful library for machine learning in Python. It proffers a collection of powerful setups for machine learning and analytical modeling alongside distribution, regression, clustering, and dimensionality minimization over a consistent platform in Python.

**Benefits of Sklearn**

· Scikit-learn entails diversely supervised & unsupervised learning algorithms. Most significantly, its simplicities as well as the cleanest machine learning library

· It appears to be formative and unify distinctively along several Python libraries, including Matplotlib for charts, Numpy for arithmetic calculations, as well as Pandas for DataFrames

· **Seaborn**: Seaborn is a library regulated by Python that helps represent data and creates enormous and further analytical operations.

**Benefits of Seaborn**

· we could systematically represent our data on a plot

· This library is created to help us reflect on our data; without manipulating the inner technicalities.

3**.  Explain why logistic regression is an appropriate technique to analyze the research question summarized in Part I.**

***The submission accurately explains why logistic regression is an appropriate technique to analyze the research question from Part I.***

A logistic function is used in the logistic regression or logistic regression statistical model to estimate the likelihood of occurring an event of particular interest. Occasionally, a binary choice model with a dichotomous dependent variable is employed to describe logistic regression (binary). The dependent variable can only have two values, including  "yes" or "no," and can imply elements like group membership (reliable client or unreliable bank customer), and the action taken (purchasing products).

**Part III: Data Preparation**

**C.  Summarize the data preparation process for logistic regression by doing the following:**

**1.  Describe your data preparation goals and the data manipulations that will be used to achieve the goals.**

***The submission describes the data preparation goals and the data manipulations that will be used to achieve the goals. The goals and manipulations align with each other and with logistic regression analysis and the research question.***

The set goal of data preparation is to be updated with the demand of data analysis to ascertain insight into the dynamic business environment conditions and streamline the business procedures.

The goal is to ensure it supports the data analyst by preparing multiple kinds of data for analytical objectives.

Furthermore, data preparation, including understanding the data accessible for analysis, is the first step in data preparation. Data on customer churn comprises about 40 fields. A significant problem is deciding which fields can be used for logistic regression analysis.

Some regions, such as customer ID, interaction, and UID, together with continuous numeric data fields like tenure and age and categorical data columns like marital status and gender, may or may not be required for analysis (related to customer service interactions).

The customer's Latitude and Longitude, the case order, and other fields are examples that are not crucial for analysis (used as a serial number). We will also look for null values in the data; if any are detected, they must be adequately treated.

The specific manipulations used to achieve data preparation goals involve the following.

Cleaning the data- In the stage of data cleaning, we will perform multiple analytical techniques which go a long way in examining the variables, analytically detecting missing values and outliers alongside cleaning the data sets to be error-free using the below functions:

|  |
| --- |
| df.dropna()  df.fillna(df.mean(), inplace=True),  boxplot=sns.boxplot(x='Varchar' (input () ),data=df)  outlierFilter=df['Varchar' (input () )] (less than) int(input () )  df = df[outlierFilter]  boxplot=sns.boxplot(x='Varchar' (input () ),data=df)  df.duplicated() |

**Explore the data**- This step illustrates central tendencies, correlations, and variations to illuminate organizational resolution alongside conducting parametric hypothesis examination using the below functions:

|  |
| --- |
| df.head()  df.describe()  df.dtypes  df.info()  df.isna()  boxplot=sns.boxplot(x='Varchar' (input () )  data=df)  df.nunique()  df['Varchar' (input () )].hist()  %matplotlib inline  grouped%matplotlib inline  grouped'Varchar' (input () ).plot.bar()  sns.scatterplot(x='Varchar' (input () ), y='Varchar' (input () ), data=df)  plt.show()  print(list(df.columns))  df.head() |

**Wrangle the data**- The goal of this stage is preprocessing, cleaning, modifying, and renaming variables in a standardized manipulative approach. Our logistic regression analysis would assist us in identifying solutions to the everyday challenges with data processing..

|  |
| --- |
| df.fillna(df.mean(), inplace=True),  grouped'Varchar' (input () ) = df.groupby(by='Varchar' (input () ).size()  grouped'Varchar' (input () )  %matplotlib inline  grouped%matplotlib inline  grouped'Varchar' (input () ).plot.bar()  grouped'Varchar' (input () ) = df.groupby(by='Varchar' (input () ).size()  grouped'Varchar' (input () )  %matplotlib inline  grouped'Varchar' (input () ).plot.bar() |

2**.  Discuss the summary statistics, including the target variable and all predictor variables that you will need to gather from the data set to answer the research question**.

|  |
| --- |
| ***The submission accurately discusses the summary statistics and discusses the target variable and all predictor variables that need to be gathered from the data set to answer the selected research question.*** |

The table contains a number of fields that are immediately apparent as not being used in the overall study. There are many of them, including Lat, Lng, UID, Interaction, Customer id, Case Order, Time Zone, Job, Zip, County, State, City, and Area. An overview of the sites and their importance is provided in the zip field.

The zip contains the state, city, neighborhood, county, latitude, longitude, and time zone. If you look at the zip field distribution in the chart below, you will see that customers are evenly dispersed from zip codes 00\*\*\* to 88\*\*\*. This demonstrates how churn has an equal impact on all areas, not just one. The accompanying table, which contains the field name, minimum and maximum values as well as averages and quartiles, demonstrates how to find extreme values in detail.Table

Description automatically generated

Ages range from 18 to 89 in the table, which would seem perfectly realistic. Users make up 75% of this demographic who are younger than 71. $348 to $258,900 in yearly income is attainable.

Whereas the low end appears weird, it might be college-age customers who have been assisted by their families simply because they may not earn much money.  It seems logical that monthly fees range from $79.97 to $290.Chart, bar chart

Description automatically generated

Churn is the target field. The overall quantity of customers who have already canceled their subscription is represented in this category. It appears that about 26.5% of individuals have discontinued that used the service in the previous decade.Chart, bar chart

Description automatically generated

3.  **Explain the steps to prepare the data for the analysis, including the annotated code**.

***The submission explains all the necessary steps to prepare the data for the analysis. The steps include the annotated code and relate to preparing for logistic regression analysis.***

• A short examination of the information to evaluate how it displays using the following code:

df.info, which provides an overview of the dataset inserted into the data frame df's top five rows and bottom five rows.

• The data frame is quickly analyzed statistically via using: Df.describe(), which provides the count, mean, standard deviation, minimum, first, second, third, and maximum values for every set of variables throughout the dataset.

• Consequently, extreme ideologies should be acknowledged as well as treated.

• Browse across each column's content,  The list of column names inside the data frame will indeed be returned using Df.columns.

• Attempt To analyze every type of variable data.

• A collection of any varying data will be produced by the technique df.dtypes.o Df.dtypes will provide a list of all variables and their associated data types with each, including object, float64, and int64.

• A thorough review will allow you to distinguish the text from numeric fields.

• Explore the overall information in every variable.

• Count each variable using df.info() will include the variable name,  the field's data type, and also the proportion of Non-Null values.

• Finding missing values and their placements utilizing Info() is quick and straightforward in a data frame.

• Fulfill The criteria for any incomplete information.

**4.  Generate univariate and bivariate visualizations of the distributions of variables in the cleaned data set. Include the target variable in your bivariate visualizations.**

***The submission accurately generates  univariate and bivariate visualizations of the distributions of variables in the cleaned data set. The bivariate visualizations include the target variable.***

Univariate statistics entails a single dependent variable and can include one or more independent variables. Therefore, histograms and box plots are some of the most commonly used univariate statistics methods to represent the data visually.

.

**Continuous variables –**

|  |
| --- |
| df[' **MonthlyCharge** '].hist()  <AxesSubplot:>  Chart, histogram  Description automatically generated  boxplot=sns.boxplot(x=' **MonthlyCharge**',data=df)  Chart  Description automatically generated  df['Income'].hist()  Chart, histogram  Description automatically generated  boxplot=sns.boxplot(x='Income',data=df)  Chart, box and whisker chart  Description automatically generated  df['Age'].hist()  <AxesSubplot:>  Chart, histogram  Description automatically generated  boxplot=sns.boxplot(x='Age',data=df)  Chart  Description automatically generated  df['Zip'].hist()    df['Tenure'].hist()  <AxesSubplot:> |

**Categorical variables - Internet service and online security**

|  |
| --- |
| %matplotlib inline  groupedInternetService.plot.bar()  <AxesSubplot:xlabel='InternetService'>  Chart, bar chart  Description automatically generated with medium confidence  df['Phone'].hist()    Chart, bar chart  Description automatically generated  %matplotlib inline  groupedOnlineSecurity.plot.bar()  Chart, bar chart  Description automatically generated  df['Churn'].hist()<AxesSubplot:>    Graphical user interface, application  Description automatically generated |

To ascertain a vivid understanding of the variables, we are systematically applying scatterplot and heatmap for bivariate analysis to unveil the relationship between two continuous variables, which would be measured on the ratio scales and intervals scales.

sns.scatterplot(x='Income', y='MonthlyCharge', data=df)

plt.show()

Chart, scatter chart

Description automatically generated

sns.scatterplot(x='Income', y='Age', data=df)

plt.show()

Chart, scatter chart

Description automatically generated

sns.scatterplot(x='Age', y='MonthlyCharge', data=df)

plt.show()

A picture containing graphical user interface

Description automatically generated

sns.scatterplot(x='OnlineSecurity', y='InternetService', data=df)

plt.show()

Graphical user interface, text, application

Description automatically generated

sns.scatterplot(x='Gender', y='Churn', data=df)

plt.show()

A picture containing text, screenshot, monitor

Description automatically generated

sns.scatterplot(x='Phone', y='OnlineBackup', data=df)

plt.show()

Shape

Description automatically generated with low confidence

sns.scatterplot(x='Churn', y='Age', data=df)

plt.show()

Table

Description automatically generated

|  |
| --- |
| # plotting scatter plot of each numerical column vs others  pd.plotting.scatter\_matrix(data[['Tenure','MonthlyCharge','Bandwidth\_GB\_Year','Income','Outage\_sec\_perweek',  'Email','Contacts','Population','Children','Age']], figsize=(20, 20))  plt.show()  plt.savefig('correlation.png')  A picture containing diagram  Description automatically generated |

5**.  Provide a copy of the prepared data set.**

***The submission provides a copy of the fully prepared data set.***

We saved the prepared data set as prepared\_churn\_data.csv.

**Part IV: Model Comparison and Analysis**

**D.  Compare an initial and a reduced logistic regression model by doing the following:**

**1.  Construct an initial logistic regression model from all predictors that were identified in Part C2**

***The submission provides an accurate initial logistic regression model from all predictors identified in Part C2.***

|  |
| --- |
| # importing relevant libraries  import numpy as np  import pandas as pd  import seaborn as sns  import matplotlib.pyplot as plt  from sklearn.preprocessing import StandardScaler  from sklearn.model\_selection import train\_test\_split  from sklearn.linear\_model import LogisticRegression  from sklearn.metrics import accuracy\_score, f1\_score  # reading csv file  data = pd.read\_csv('churn\_clean.csv')  # Checking for missing values  data.isna().any(axis=0).any()  # checking how many unique values are in each column  data.nunique()  # printing columns with more than 100 unique values, these columns will be either numerical or have too many unique values that they should be left out  for col in data.columns:  if data[col].nunique()>100:  print(col)  # dropping columns with higher number of of unique values  to\_drop = ['City','County','Zip','Job','TimeZone', 'Lat','Lng','UID', 'Customer\_id','Interaction','CaseOrder']  data.drop(columns=to\_drop,inplace=True)  data.shape  # plotting scatter plot of each numerical column vs others  pd.plotting.scatter\_matrix(data[['Tenure','MonthlyCharge','Bandwidth\_GB\_Year','Income','Outage\_sec\_perweek',  'Email','Contacts','Population','Children','Age']], figsize=(20, 20))  plt.show()  plt.savefig('correlation.png')  # plotting correlations plot of each numerical column vs others  plt.subplots(figsize=(15,15))  df = data[['Tenure','MonthlyCharge','Bandwidth\_GB\_Year','Income','Outage\_sec\_perweek',  'Email','Contacts','Population','Children','Age']]  sns.heatmap(df.corr(),annot=True,lw=1);  categorical\_columns = ['State','Area','Marital','Gender','Techie','Contract','Port\_modem','Tablet',  'InternetService','Phone', 'Multiple','OnlineSecurity','OnlineBackup',  'DeviceProtection','TechSupport', 'StreamingTV','StreamingMovies',  'PaperlessBilling','PaymentMethod'  ]  numerical\_columns = ['Population','Children','Age','Income','Outage\_sec\_perweek','Email',  'Contacts','Yearly\_equip\_failure','Tenure',  'MonthlyCharge','Bandwidth\_GB\_Year'  ]  dummy\_data\_file\_index = 0  def get\_dummy\_data\_with\_output(dummy\_variable\_columns, data):  global dummy\_data\_file\_index  dummy\_data = pd.get\_dummies(data, prefix=dummy\_variable\_columns, columns=dummy\_variable\_columns, drop\_first= True)  # dummy\_data.to\_csv('dummy\_var\_data'+str(dummy\_data\_file\_index)+'.csv', index=False)  y = dummy\_data['Churn']  dummy\_data.drop(columns=['Churn'], inplace=True)  dummy\_data\_file\_index += 1  return y, dummy\_data  y, dummy\_data = get\_dummy\_data\_with\_output(categorical\_columns, data)  X\_train, X\_test, y\_train, y\_test = train\_test\_split(dummy\_data,y,test\_size=.2, random\_state=0)  # Creating a function to create prediction model based on sklearn library and  # print details like model Summary, Confusion Matrix and Accuracy Score based on predicted values using test set  model = LogisticRegression(max\_iter=10000)  model.fit(X\_train, y\_train)  arr = np.c\_[X\_train.columns.tolist(), model.coef\_.tolist()[0]]  intercept = model.intercept\_  # print('\nPrinting model coefficients and intercept summary for sklearn model:\n',arr, model.intercept\_)  y\_pred = model.predict(X\_test)  print('\nPrinting predicted and actual values from sklearn:\n',np.c\_[y\_pred, y\_test])  print('\nPrinting Accuracy:\n',(accuracy\_score(y\_test, y\_pred)))  print('\nPrinting F1-Score:\n',(f1\_score(y\_test, y\_pred, pos\_label="Yes")))  equation = 'y = '  for ar in arr:  eq = str(round(float(ar[1]),3))+' x '+str(ar[0])  if eq.startswith('-'):  equation = equation + ' ' + eq  else:  equation = equation + ' + ' +eq  print("Equation for model without data preparation:")  print (equation + ' ' + str(intercept))  data\_reduced = data[[ 'Children', 'Age', 'Gender', 'InternetService', 'Multiple', 'OnlineSecurity',  'OnlineBackup', 'DeviceProtection', 'TechSupport', 'StreamingTV',  'StreamingMovies', 'Tenure', 'Bandwidth\_GB\_Year',  'MonthlyCharge', 'Contract', 'Churn' ]]  dummy\_variable\_columns = ['Gender', 'InternetService', 'Multiple','OnlineSecurity','OnlineBackup',  'DeviceProtection','TechSupport', 'StreamingTV','StreamingMovies','Contract'  ]  data\_diff = [i for i in data.columns.tolist() + data\_reduced.columns.tolist() if i not in data.columns.tolist() or i not in data\_reduced.columns.tolist()]  y, dummy\_data\_reduced = get\_dummy\_data\_with\_output(dummy\_variable\_columns, data\_reduced)  X\_train, X\_test, y\_train, y\_test = train\_test\_split(dummy\_data\_reduced,y,test\_size=.3, random\_state=0)  data\_diff  # Creating a function to create prediction model based on sklearn library and  # print details like model Summary, Confusion Matrix and Accuracy Score based on predicted values using test set  model = LogisticRegression(max\_iter=10000)  model.fit(X\_train, y\_train)  arr = np.c\_[X\_train.columns.tolist(), model.coef\_.tolist()[0]]  intercept = model.intercept\_  # print('\nPrinting model coefficients and intercept summary for sklearn model:\n',arr, model.intercept\_)  y\_pred = model.predict(X\_test)  print('\nPrinting predicted and actual values from sklearn:\n',np.c\_[y\_pred, y\_test])  print('\nPrinting Accuracy:\n',(accuracy\_score(y\_test, y\_pred)))  print('\nPrinting F1-Score:\n',(f1\_score(y\_test, y\_pred, pos\_label="Yes")))  equation = 'y = '  for ar in arr:  eq = str(round(float(ar[1]),3))+' x '+str(ar[0])  if eq.startswith('-'):  equation = equation + ' ' + eq  else:  equation = equation + ' + ' +eq  print("Equation for model with reduced data:")  print (equation + ' ' + str(intercept)) |

**2.  Justify a statistically based variable selection procedure and a model evaluation metric to reduce the initial model in a way that aligns with the research question.**

***The submission justifies a statistically based variable selection procedure and a model evaluation metric to reduce the initial model. The justification is in alignment with the research question.***

The majority of the predictors do not align with the research questions. Additional variables will be reduced to achieve the reduced model. I separated the data into train and test sets specifically for this analysis. We could evaluate the model by feeding it with data and looking for trends and quality. Then, We use the testing data as a subset of the larger data set used to assess the performance of the model and its accuracy.

On the basis of the findings from the previous model, I will make an effort to create a reduced model using the same data but excluding the independent variables that were found to have high p values and multicollinearity. Its accuracy is improved by the decreased model.

A picture containing calendar

Description automatically generated

A correlation heatmap might be helpful, but due to the number of features, it may be somewhat challenging to interpret. Creating the below chart would help.Chart, treemap chart

Description automatically generated

This consist of tenure, monthly charge, bandwidth\_GB\_Year, income, outage\_sec\_perweek, email, contacts, population, children, age, and below was the output of running the initial logistic model.

Graphical user interface, text, application, chat or text message

Description automatically generated

From the above, the predicted values shows one (Yes) for yes and three(Yes) for no.

A screenshot of a computer

Description automatically generated with medium confidence

3.  **Provide a reduced logistic regression model.**

Note: The output should include a screenshot of each model.

***The submission provides a reduced logistic regression model, and the model is in alignment with the justification from part D2.***

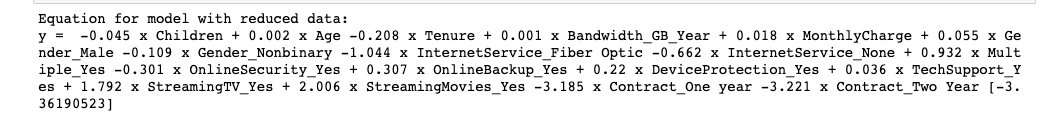
Improved predictive figures have been derived by the Reduced Logistic Regression analysis. The Logistic regression score method produced a 0.806045, a 5.42 percent increase, using ten variables contrary to the original model's thirty and employing the equivalent methodology. The improvements involving reduced false positives and false negatives are depicted in the redesigned confusion matrix for the reduced feature.

Graphical user interface, text, application

Description automatically generated

Graphical user interface, text

Description automatically generated



E.  **Analyze the data set using your reduced logistic regression model by doing the following:**

1**.  Explain your data analysis process by comparing the initial and reduced logistic regression models, including the following elements:**

**•  The model evaluation metric**

***The submission accurately explains the data analysis process by comparing the initial and reduced logistic regression models, including all of the given elements.***

Can we predict turnovers? is the query we're seeking to solve. Withdrawal of customers is churn. Churn is when a consumer no longer wishes to pay for our services due to an issue concerning our service or our product. For any of these services, they would probably seek somewhere. Consequently, engagements with the customer should constitute the attention of the critical metrics. These interactions include phone calls, contracts, Yearly equipment failure, tenure, monthly payments, Survey responses (items 1 through 8), and the services they use, including the Internet, modem, phone, security, backup, tech support, and streaming services.

It seems reasonable that as consumers utilize the company's services more consistently, customers become more vested in them and are less motivated to leave.

**The logic of the variable selection technique**

Using every predictor is a great place to start testing that notion. The final result will not influence the variables or classifiers that have no influence on the final result after evaluating the data. We will be using the summary part of the Logit function in the stats model to analyze that impact.

The summary includes an analysis and a set of statistics for every predictor. The number of predictors is trimmed from 30 to 10 because of this figure. That removes the non-significant predictors. The sophistication of regression is lowered by employing these ten predictors, and the results are enhanced. Reliability, clarity, and recollection were just a small number of the regression's forecasting metrics that have been improved.

**The model evaluation metric**

The model is mainly evaluated on how efficiently it can generate predictions. As indicated previously, these parameters are recall, clarity, and correctness. The capacity to predict true positives and true negatives with correctness. Precision refers to the proportion of anticipated True Positives to any anticipated Positives, including True and False Positives. The recall is the percentage of predicted true positives across all true positives.

Initial model

Graphical user interface, text, application, chat or text message

Description automatically generated

Reduced Model

Graphical user interface, text

Description automatically generated

Compared to the entire predictor method, the lower predictor approach exceeded it throughout the spectrum. More importantly, around 20% of the time, there was a substantial improvement in anticipating customer churn. The implication is that by identifying prospective churn customers, the company can act quickly, which could also convince customers to remain long-term clients.

.

**2.  Provide the output and any calculations of the analysis you performed, including a confusion matrix.**

Note: The output should include the predictions from the refined model you used to perform the analysis.

The submission provides the accurate output and calculations of the analysis performed, including a confusion matrix. The submissions includes all necessary output and calculations.

|  |
| --- |
| *# importing relevant libraries*  ​  **import** numpy **as** np  **import** pandas **as** pd  **import** seaborn **as** sns  **import** matplotlib.pyplot **as** plt  ​  **from** sklearn.preprocessing **import** StandardScaler  **from** sklearn.model\_selection **import** train\_test\_split  **from** sklearn.linear\_model **import** LogisticRegression  **from** sklearn.metrics **import** accuracy\_score, f1\_score  In [2]:  *# reading csv file*  ​  data **=** pd.read\_csv('churn\_clean.csv')  In [3]:  *# Checking for missing values*  ​  data.isna().any(axis**=**0).any()  Out[3]:  False  In [4]:  *# checking how many unique values are in each column*  ​  data.nunique()  Out[4]:  CaseOrder 10000  Customer\_id 10000  Interaction 10000  UID 10000  City 6058  State 52  County 1620  Zip 8583  Lat 8563  Lng 8630  Population 5933  Area 3  TimeZone 25  Job 639  Children 11  Age 72  Income 9993  Marital 5  Gender 3  Churn 2  Outage\_sec\_perweek 9986  Email 23  Contacts 8  Yearly\_equip\_failure 6  Techie 2  Contract 3  Port\_modem 2  Tablet 2  InternetService 3  Phone 2  Multiple 2  OnlineSecurity 2  OnlineBackup 2  DeviceProtection 2  TechSupport 2  StreamingTV 2  StreamingMovies 2  PaperlessBilling 2  PaymentMethod 4  Tenure 9996  MonthlyCharge 750  Bandwidth\_GB\_Year 10000  Item1 7  Item2 7  Item3 8  Item4 7  Item5 7  Item6 8  Item7 7  Item8 8  dtype: int64  In [ ]:  ​  In [5]:  *# printing columns with more than 100 unique values, these columns will be either numerical or have too many unique values that they should be left out*  ​  **for** col **in** data.columns:  **if** data[col].nunique()**>**100:  print(col)  CaseOrder  Customer\_id  Interaction  UID  City  County  Zip  Lat  Lng  Population  Job  Income  Outage\_sec\_perweek  Tenure  MonthlyCharge  Bandwidth\_GB\_Year  In [6]:  *# dropping columns with higher number of of unique values*  ​  to\_drop **=** ['City','County','Zip','Job','TimeZone', 'Lat','Lng','UID', 'Customer\_id','Interaction','CaseOrder']  data.drop(columns**=**to\_drop,inplace**=True**)  data.shape  Out[6]:  (10000, 39)  In [7]:  *# plotting scatter plot of each numerical column vs others*  ​  pd.plotting.scatter\_matrix(data[['Tenure','MonthlyCharge','Bandwidth\_GB\_Year','Income','Outage\_sec\_perweek',  'Email','Contacts','Population','Children','Age']], figsize**=**(20, 20))  plt.show()  plt.savefig('correlation.png')    <Figure size 432x288 with 0 Axes>  In [8]:  *# plotting correlations plot of each numerical column vs others*  ​  plt.subplots(figsize**=**(15,15))  df **=** data[['Tenure','MonthlyCharge','Bandwidth\_GB\_Year','Income','Outage\_sec\_perweek',  'Email','Contacts','Population','Children','Age']]  sns.heatmap(df.corr(),annot**=True**,lw**=**1);    In [9]:  categorical\_columns **=** ['State','Area','Marital','Gender','Techie','Contract','Port\_modem','Tablet',  'InternetService','Phone', 'Multiple','OnlineSecurity','OnlineBackup',  'DeviceProtection','TechSupport', 'StreamingTV','StreamingMovies',  'PaperlessBilling','PaymentMethod'  ]  ​  numerical\_columns **=** ['Population','Children','Age','Income','Outage\_sec\_perweek','Email',  'Contacts','Yearly\_equip\_failure','Tenure',  'MonthlyCharge','Bandwidth\_GB\_Year'  ]  ​  dummy\_data\_file\_index **=** 0  In [10]:  **def** get\_dummy\_data\_with\_output(dummy\_variable\_columns, data):  **global** dummy\_data\_file\_index  dummy\_data **=** pd.get\_dummies(data, prefix**=**dummy\_variable\_columns, columns**=**dummy\_variable\_columns, drop\_first**=** **True**)  *# dummy\_data.to\_csv('dummy\_var\_data'+str(dummy\_data\_file\_index)+'.csv', index=False)*  y **=** dummy\_data['Churn']  dummy\_data.drop(columns**=**['Churn'], inplace**=True**)  dummy\_data\_file\_index **+=** 1  **return** y, dummy\_data  ​  y, dummy\_data **=** get\_dummy\_data\_with\_output(categorical\_columns, data)  In [11]:  X\_train, X\_test, y\_train, y\_test **=** train\_test\_split(dummy\_data,y,test\_size**=**.2, random\_state**=**0)  In [12]:  *# Creating a function to create prediction model based on sklearn library and*  *# print details like model Summary, Confusion Matrix and Accuracy Score based on predicted values using test set*  model **=** LogisticRegression(max\_iter**=**10000)  model.fit(X\_train, y\_train)  arr **=** np.c\_[X\_train.columns.tolist(), model.coef\_.tolist()[0]]  intercept **=** model.intercept\_  *# print('\nPrinting model coefficients and intercept summary for sklearn model:\n',arr, model.intercept\_)*  y\_pred **=** model.predict(X\_test)  print('\nPrinting predicted and actual values from sklearn:\n',np.c\_[y\_pred, y\_test])  print('\nPrinting Accuracy:\n',(accuracy\_score(y\_test, y\_pred)))  print('\nPrinting F1-Score:\n',(f1\_score(y\_test, y\_pred, pos\_label**=**"Yes")))  Printing predicted and actual values from sklearn:  [['No' 'No']  ['No' 'No']  ['Yes' 'Yes']  ...  ['No' 'Yes']  ['No' 'Yes']  ['No' 'No']]  Printing Accuracy:  0.8445  Printing F1-Score:  0.6836215666327569  In [13]:  equation **=** 'y = '  **for** ar **in** arr:  eq **=** str(round(float(ar[1]),3))**+**' x '**+**str(ar[0])  **if** eq.startswith('-'):  equation **=** equation **+** ' ' **+** eq  **else**:  equation **=** equation **+** ' + ' **+**eq  ​  print("Equation for model without data preparation:")  print (equation **+** ' ' **+** str(intercept))  Equation for model without data preparation:  y = -0.0 x Population -0.049 x Children + 0.001 x Age -0.0 x Income -0.082 x Outage\_sec\_perweek -0.092 x Email -0.014 x Contacts -0.01 x Yearly\_equip\_failure -0.306 x Tenure + 0.02 x MonthlyCharge + 0.003 x Bandwidth\_GB\_Year -0.072 x Item1 -0.068 x Item2 -0.067 x Item3 -0.073 x Item4 -0.083 x Item5 -0.065 x Item6 -0.073 x Item7 -0.068 x Item8 -0.0 x State\_AL -0.001 x State\_AR -0.0 x State\_AZ -0.0 x State\_CA -0.001 x State\_CO -0.0 x State\_CT + 0.0 x State\_DC + 0.0 x State\_DE -0.003 x State\_FL + 0.0 x State\_GA -0.0 x State\_HI -0.001 x State\_IA -0.0 x State\_ID -0.002 x State\_IL -0.001 x State\_IN + 0.0 x State\_KS -0.001 x State\_KY + 0.0 x State\_LA -0.001 x State\_MA + 0.0 x State\_MD -0.0 x State\_ME + 0.0 x State\_MI -0.002 x State\_MN + 0.001 x State\_MO + 0.0 x State\_MS + 0.0 x State\_MT -0.002 x State\_NC -0.002 x State\_ND -0.001 x State\_NE -0.0 x State\_NH -0.001 x State\_NJ -0.0 x State\_NM -0.001 x State\_NV -0.006 x State\_NY -0.001 x State\_OH + 0.001 x State\_OK + 0.0 x State\_OR -0.003 x State\_PA -0.0 x State\_PR -0.001 x State\_RI -0.0 x State\_SC -0.0 x State\_SD + 0.002 x State\_TN + 0.001 x State\_TX -0.0 x State\_UT -0.001 x State\_VA + 0.0 x State\_VT + 0.002 x State\_WA -0.0 x State\_WI + 0.003 x State\_WV -0.0 x State\_WY -0.012 x Area\_Suburban -0.004 x Area\_Urban -0.004 x Marital\_Married -0.008 x Marital\_Never Married -0.001 x Marital\_Separated -0.002 x Marital\_Widowed -0.0 x Gender\_Male -0.003 x Gender\_Nonbinary + 0.014 x Techie\_Yes -0.056 x Contract\_One year -0.067 x Contract\_Two Year -0.003 x Port\_modem\_Yes -0.007 x Tablet\_Yes -0.063 x InternetService\_Fiber Optic + 0.001 x InternetService\_None -0.025 x Phone\_Yes + 0.001 x Multiple\_Yes -0.019 x OnlineSecurity\_Yes -0.011 x OnlineBackup\_Yes -0.008 x DeviceProtection\_Yes -0.014 x TechSupport\_Yes + 0.03 x StreamingTV\_Yes + 0.042 x StreamingMovies\_Yes -0.013 x PaperlessBilling\_Yes -0.005 x PaymentMethod\_Credit Card (automatic) + 0.005 x PaymentMethod\_Electronic Check -0.007 x PaymentMethod\_Mailed Check [-0.02205022]  In [14]:  data\_reduced **=** data[[ 'Children', 'Age', 'Gender', 'InternetService', 'Multiple', 'OnlineSecurity',  'OnlineBackup', 'DeviceProtection', 'TechSupport', 'StreamingTV',  'StreamingMovies', 'Tenure', 'Bandwidth\_GB\_Year',  'MonthlyCharge', 'Contract', 'Churn' ]]  ​  dummy\_variable\_columns **=** ['Gender', 'InternetService', 'Multiple','OnlineSecurity','OnlineBackup',  'DeviceProtection','TechSupport', 'StreamingTV','StreamingMovies','Contract'  ]  ​  ​  data\_diff **=** [i **for** i **in** data.columns.tolist() **+** data\_reduced.columns.tolist() **if** i **not** **in** data.columns.tolist() **or** i **not** **in** data\_reduced.columns.tolist()]  ​  y, dummy\_data\_reduced **=** get\_dummy\_data\_with\_output(dummy\_variable\_columns, data\_reduced)  X\_train, X\_test, y\_train, y\_test **=** train\_test\_split(dummy\_data\_reduced,y,test\_size**=**.3, random\_state**=**0)  In [15]:  data\_diff  Out[15]:  ['State',  'Population',  'Area',  'Income',  'Marital',  'Outage\_sec\_perweek',  'Email',  'Contacts',  'Yearly\_equip\_failure',  'Techie',  'Port\_modem',  'Tablet',  'Phone',  'PaperlessBilling',  'PaymentMethod',  'Item1',  'Item2',  'Item3',  'Item4',  'Item5',  'Item6',  'Item7',  'Item8']  In [16]:  *# Creating a function to create prediction model based on sklearn library and*  *# print details like model Summary, Confusion Matrix and Accuracy Score based on predicted values using test set*  model **=** LogisticRegression(max\_iter**=**10000)  model.fit(X\_train, y\_train)  arr **=** np.c\_[X\_train.columns.tolist(), model.coef\_.tolist()[0]]  intercept **=** model.intercept\_  *# print('\nPrinting model coefficients and intercept summary for sklearn model:\n',arr, model.intercept\_)*  y\_pred **=** model.predict(X\_test)  print('\nPrinting predicted and actual values from sklearn:\n',np.c\_[y\_pred, y\_test])  print('\nPrinting Accuracy:\n',(accuracy\_score(y\_test, y\_pred)))  print('\nPrinting F1-Score:\n',(f1\_score(y\_test, y\_pred, pos\_label**=**"Yes")))  Printing predicted and actual values from sklearn:  [['No' 'No']  ['No' 'No']  ['Yes' 'Yes']  ...  ['No' 'No']  ['No' 'No']  ['No' 'No']]  Printing Accuracy:  0.8973333333333333  Printing F1-Score:  0.8060453400503778  In [17]:  equation **=** 'y = '  **for** ar **in** arr:  eq **=** str(round(float(ar[1]),3))**+**' x '**+**str(ar[0])  **if** eq.startswith('-'):  equation **=** equation **+** ' ' **+** eq  **else**:  equation **=** equation **+** ' + ' **+**eq  ​  print("Equation for model with reduced data:")  print (equation **+** ' ' **+** str(intercept))  Equation for model with reduced data:  y = -0.045 x Children + 0.002 x Age -0.208 x Tenure + 0.001 x Bandwidth\_GB\_Year + 0.018 x MonthlyCharge + 0.055 x Gender\_Male -0.109 x Gender\_Nonbinary -1.044 x InternetService\_Fiber Optic -0.662 x InternetService\_None + 0.932 x Multiple\_Yes -0.301 x OnlineSecurity\_Yes + 0.307 x OnlineBackup\_Yes + 0.22 x DeviceProtection\_Yes + 0.036 x TechSupport\_Yes + 1.792 x StreamingTV\_Yes + 2.006 x StreamingMovies\_Yes -3.185 x Contract\_One year -3.221 x Contract\_Two Year [-3.36190523]  In [ ]:  ​ |

3.  **Provide the code used to support the implementation of the logistic regression models.**

***The submission provides the code used to support the implementation of the logistic regression models, and the code is complete and accurate.***

Code See attached file logreg\_churn\_csv.ipynb

**Part V: Data Summary and Implications**

F**.  Summarize your findings and assumptions by doing the following:**

**1.  Discuss the results of your data analysis, including the following elements:**

***The submission accurately discusses the results of the data analysis, and the discussion addresses all of the given elements and is in alignment with the research question and the data analysis.***

**•  A regression equation for the reduced model**

Equation for model with reduced data:

y = -0.045 x Children + 0.002 x Age -0.208 x Tenure + 0.001 x Bandwidth\_GB\_Year + 0.018 x MonthlyCharge + 0.055 x Gender\_Male -0.109 x Gender\_Nonbinary -1.044 x InternetService\_Fiber Optic -0.662 x InternetService\_None + 0.932 x Multiple\_Yes -0.301 x OnlineSecurity\_Yes + 0.307 x OnlineBackup\_Yes + 0.22 x DeviceProtection\_Yes + 0.036 x TechSupport\_Yes + 1.792 x StreamingTV\_Yes + 2.006 x StreamingMovies\_Yes -3.185 x Contract\_One year -3.221 x Contract\_Two Year [-3.36190523]

•  **An interpretation of coefficients of the statistically significant variables of the model**

The exponent of the predictor, either favorable or unfavorable, has been the coefficient. In connection with the current investigation, a substantial coefficient value implies a predictor that encourages churning, while a negative coefficient means a classifier that promotes avoiding churning. Turnover is forecasted strongly by contract, monthly charge, and Internet service. Tenure, children, protection, and phone are all contradictory. However, bandwidth GBY is a churning influence.

The better determinant of customer churn, with a factor of 0.018, is establishing a pay period subscription. The most vital determinant of not churning seems to have a multiplier of -3.361 and therefore is named "Contract Two-year." In addition to the tenure predictor, which has a multiplier of -0.045, the children predictor, which has a multiplier of -0.34, is a good predictor of not churning.

•  **The statistical and Practical significance of the model**

Analytically or logistically, the model has statistical significance. From such a statistical point of view, it is undeniable that there are a variety of reliable indicators that reflect the possibility of a consumer terminating a service.

According to the coefficient associated with every variable, it is possible to organize these from the highest to the lowest classifier. There is a more significant likelihood that the customer will be lost when the predictor variable coefficient increases because the response variable also increases.

The telecom corporation may often accurately determine and keep customers acquired via promotional tools by using coefficients.

•  **The limitations of the data analysis**

It depends upon whether the data is representative of the general population, similar to any other data analysis. There are no duplicates nor derived values in the data, which is pure. Nonetheless, it also only includes more than 90,000 samples from either a higher dimensionality that encompasses a shorter duration time frame.

2.  **Recommend a course of action based on your results.**

***The submission recommends an appropriate course of action based on the results as they relate to the research question.***

The Churn is sure to erupt. However, less than four years ago, anything that enabled the churn rate was more significant than the number of new customers, as demonstrated by the longevity prediction dispersion. So much worse, the entire customer base from twenty-four to 36 months earlier has disappeared. perform analysis regarding promotional campaigns to evaluate performance and take the exact opposite measure.

One expected result of the dataset analysis was that customers would be less inclined to leave more and more features they would spend for. In some ways, the data analysis supports that. It seems that customer stays are impacted by phone, kid, and protection services. Technical issues don't appear to have harmed a particular area or product line, and technical support isn't a minor but essential level.

Technical support inquiries are unrelated to customer churn. The tech support team is performing well. Any suggestions on how to reduce client churn? Create incentives for customers to utilize more of the supplied services. Encourage clients to sign 1- or 2-year contracts by offering incentives.

**Part VI: Demonstration**

G.  Provide a Panopto video recording that includes *all* of the following elements:

•  a demonstration of the functionality of the code used for the analysis

•  an identification of the version of the programming environment

•  a comparison of the **two** logistic regression models you used in your analysis

•  an interpretation of the coefficients

***A Panopto video recording is provided that includes all of the given elements. For the duration of the presentation, the video captures both the presenter and the functioning code in a Panopto video recording.***

<https://wgu.hosted.panopto.com/Panopto/Pages/Viewer.aspx?id=304a5e64-24f3-494f-a8c8-aee700387321>

**H.  List the web sources used to acquire data or segments of third-party code to support the application. Ensure the web sources are reliable.**

***The submission lists all web sources used to acquire data or segments of third-party code, and the web sources are reliable.***

*Grant, P. (2019). Understanding Multiple Regression; The fundamental basis behind this commonly used algorithm.*

Medium***.***[***https://towardsdatascience.com/understanding-multiple-regression-249b16bde83e***](https://towardsdatascience.com/understanding-multiple-regression-249b16bde83e)

Deepanshu, B. (2020). *How to Import Data in Python.*

RSGB Business Consultant Pvt. Ltd. <https://www.listendata.com/2017/02/import-data-in-python.html#Import-CSV-files>

Pierre-Louis B. (2020). *Principle Components Analysis(PCA), Fundamentals, Benefits & Insights for Industry.*

Medium. <https://towardsdatascience.com/principal-components-analysis-pca-fundamentals-benefits-insights-for-industry-2f03ad18c4d7>

John S. (2018). *Data Cleaning with python and Pandas: Detecting Missing Values.*

Medium. <https://towardsdatascience.com/data-cleaning-with-python-and-pandas-detecting-missing-values-3e9c6ebcf78b>

Angelica Lo D. (2021). *How to detect outliers with Python.*

Medium. <https://towardsdatascience.com/how-to-detect-outliers-with-python-pyod-aa7147359e4b>

Michael G. (2018). *Understanding Boxplots.*

Medium. <https://towardsdatascience.com/understanding-boxplots-5e2df7bcbd51>

John S. (2018). *Data Cleaning with python and Pandas: Detecting Missing Values.*

Medium. <https://towardsdatascience.com/data-cleaning-with-python-and-pandas-detecting-missing-values-3e9c6ebcf78b>

<https://scikit-learn.org/stable/modules/generated/sklearn.model_selection.train_test_split.html>

<https://scikit-learn.org/stable/modules/generated/sklearn.metrics.confusion_matrix.html>

<https://pandas.pydata.org/pandas-docs/stable/reference/api/pandas.get_dummies.html>

**I.  Acknowledge sources, using in-text citations and references, for content that is quoted, paraphrased, or summarized.**

***The submission includes in-text citations for sources that are properly quoted, paraphrased, or summarized and a reference list that accurately identifies the author, date, title, and source location as available*.**

**Reference**

Larose, C. D., & Larose, D. T. (2019). Data science using Python and R. ISBN-13: 978-1-119-52684-1.