**Jason Pena**

**WGU University**

# D208 PERFORMANCE ASSESSMENT NBM2 TASK 1

# MULTIPLE REGRESSION FOR PREDICTIVE MODELING

**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 multiple regression.

How much Gigabytes of data will a customer use on a yearly basis? Can the measurement be predicted accurately from a list of explanatory variables that are provided?

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 goal or objective of this analysis is to be able to predict in advance how much data a customer will use. This will provide supporting evidence for decisions to increase cloud computing resources for increased bandwidth and to expand or decrease data limits for customers in media streaming.

**Part II: Method Justification**

B.  Describe multiple regression methods by doing the following:

1.  Summarize the assumptions of a multiple regression model.

* Linear Relationship

“Linear regression needs the relationship between the independent and dependent variables to be linear.”(Kalbande, 2020) Visualization tools such as scatterplots can show whether there is a linear relationship.

Chart, scatter chart

Description automatically generated

* Multivariate Normality

“The linear regression analysis requires all variables to be multivariate normal. Means data should be normally distributed.” (Kalbande, 2020)

* No Multicollinearity

“Multiple regression assumes that the independent variables are not highly correlated with each other.” (Kalbande, 2020)

* Homoscedasticity

This assumption states that the variance of error terms is similar across the values of the independent variables. A plot of standardized residuals versus predicted values can show whether points are equally distributed across all values of the independent variables.

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

Python will be used to support the various phases of the analysis. Python will allow the user to implement coding solutions, manipulating the data, and creating visual representations for the performance assessment.

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

Multiple linear regression (MLR), also known simply as multiple regression, is a statistical technique that uses several explanatory variables to predict the outcome of a response variable. The goal of multiple linear regression is to model the linear relationship between the explanatory (independent) variables and response(dependent variables. Multiple regression is an extension of linear (OLS) regression because it involves more than one explanatory variable.(Investopedia, 2021)

Multiple regression is an appropriate technique to analyze the research question because the target variable, Bandwidth\_GB\_Year, is a continuous variable which allows us to create a prediction of how much data is used. There are also several explanatory variables that will add to our understanding when trying to predict how much data a customer will use in a given year. When adding or removing independent variables from the regression analysis, this will indicate whether or not they have a positive or negative relationship to the target variable and how it might affect weighted decisions.

C. Summarize the data preparation process for multiple regression analysis by doing the following:

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

My data preparation goals and data manipulations will include:

1. Import the dataset into Python.
2. Evaluate the data structure to gain a better understanding of the variables and data types.
3. Provide a name to identify my dataset. The naming convention I chose for my dataset is: Churn\_df
4. Data manipulations made to the data set will be named: df
5. Check for any misleading variable names and rename them.
6. Check for any missing data that could skew the model.
7. Missing data will be inputted with measures of central tendency.
8. Create visualizations to identify any outliers that could affect the model.
9. Summaries of univariate and bivariate statistics to search for any flags
10. Discuss the summary statistics, including the target variable and all predictor variables that you will need to gather from the dataset to answer the research question

Once the data is imported into Python, the output provides that the dataset consists of 50 original columns and 10,000 records. For purposes of this analysis, columns such as User ID, & demographic variables such as (Caseorder, Customer\_id, Interaction, UID, City, State, County, Zip, Lat, Lng, Population, Area, TimeZone, Job, Marial, and PaymentMethod will be removed from the dataframe. Binary categorical variables were encoded to 1 / 0 as well as the ordinal variables being converted into numeric. After analyzing summary statistics, it appears that the dataset has been sufficiently cleaned leaving no null, NAs, or missing data points.

This results in having 34 remaining numerical independent predictor variables including the target variable. The dependent variable “Bandwidth\_GB\_Year” which is the most relevant to the decision-making process, will be the continuous target variable used. The predictor variables will consist of continuous variables, binary categorical variables that will be converted into numerical data types and discrete ordinal variables.

The continuous predictor variables: “Children, Age, Income, Outage\_sec\_perweek, Email, Contacts, Yearly\_equip\_failure, Tenure, MonthlyCharge, Gender, Contract, InternetService.

The categorical predictor variables: “Churn, Techie, Port\_modem, Tablet, Phone, Multiple, OnlineSecurity, OnlineBackup, DeviceProtection, TechSupport, StreamingTV, StreamingMovies.

The discrete ordinal variables (These variables are the survey questions provided using a rating system of 1 to 8 (1 = most important, 8 = least important):

The average customer is 53 years old with a standard deviation of 20.69, has two children (with a standard deviation of 2), an income of $39,806.93 (with a standard deviation of 28,199.92), experienced 10 second outages every so often, had less than 1 yearly equipment failure, has been with the company for 34.5 years and has a monthly charge of $172.53 and uses 3392.34 gigabytes of data per year.

1. Explain the steps used to prepare the data for the analysis, including the annotated code.

Annotated Code with explanation of each step:

# Standard Data Science Imports

import numpy as np

import pandas as pd

from pandas import Series, DataFrame

Here I am importing NumPy as pd and pandas as pd. I am also importing Series, and DataFrame

“NumPy is the fundamental package for scientific computing in Python. It is a Python library that provides a multidimensional array object, various derived objects (such as masked arrays and matrices), and an assortment of routines for fast operations on arrays, including mathematical, logical, shape manipulation, sorting, selecting, I/O, discrete Fourier transforms, basic linear algebra, basic statistical operations, random simulation and much more.” (Numpy.org, 2021)

“Similar to Numpy, pandas deals primarily with data in 1-D and 2-D arrays; however, pandas handles them differently.“ (Educative.io, 2021)

“In pandas, 1-D arrays are referred to a series. A **series** is created through the pd.Series constructor, which has a lot of optional arguments. The most common argument is data, which specifies the elements of the series.” (Educative.io, 2021)

“A DataFrame is simply a 2-D array. It can be created through the pd.DataFrame constructor, which takes in essentially the same arguments as pd.Series. However, while a series could be constructed from a scalar (representing a single value Series), a DataFrame cannot.” (Educative.io, 2021)

# Visualization libraries

import seaborn as sns

import matplotlib.pyplot as plt

%matplotlib inline

Next I will be importing the Visualization libraries.

Seaborn is a library for making statistical graphics in Python. It builds on top of matplotlib and integrates closely with pandas data structures.

Seaborn helps you explore and understand your data. Its plotting functions operate on dataframes and arrays containing whole datasets and internally perform the necessary semantic mapping and statistical aggregation to produce informative plots. Its dataset-oriented, declarative API lets you focus on what the different elements of your plots mean, rather than on the details of how to draw them. (Seaborn.pydata.org, 2021)

# Statistics packages

import pylab

from statsmodels.formula.api import ols

import statistics

from scipy import stats

After, are the Statistics packages

PyLab is a procedural interface to the Matplotlib object-oriented plotting library. Matplotlib is the whole package; matplotlib.pyplot is a module in Matplotlib; and PyLab is a module that gets installed alongside Matplotlib. PyLab is a convenience module that bulk imports matplotlib.pyplot (for plotting) and NumPy (for Mathematics and working with arrays) in a single name space. (tutorialspoints.com, 2021)

# Scikit-learn

import sklearn

from sklearn import preprocessing

from sklearn.preprocessing import LabelEncoder

from sklearn.linear\_model import LinearRegression

from sklearn import metrics

Then I will now import the Scikit-learn

Scikit-learn (Sklearn) is the most useful and robust library for machine learning in Python. It provides a selection of efficient tools for machine learning and statistical modeling including classification, regression, clustering, and dimensionality reduction via a consistence interface in Python. This library, which is largely written in Python, is built upon NumPy, SciPy and Matplotlib. (tutorialspoints.com, 2021)

# Load data set into Pandas Dataframe

churn\_df = pd.read\_csv(r'C:\Users\Hydraconix\Desktop\churn\_clean.csv')

Here I am importing the dataset using Pandas. Once I have imported the data, I will check the description of dataframe, structure (columns & rows) & data types.

# Checking for Null Values

churn\_df.isna().sum()

Output:

CaseOrder 0

Customer\_id 0

Interaction 0

UID 0

City 0

State 0

County 0

Zip 0

Lat 0

Lng 0

Population 0

Area 0

TimeZone 0

Job 0

Children 0

Age 0

Income 0

Marital 0

Gender 0

Churn 0

Outage\_sec\_perweek 0

Email 0

Contacts 0

Yearly\_equip\_failure 0

Techie 0

Contract 0

Port\_modem 0

Tablet 0

InternetService 0

Phone 0

Multiple 0

OnlineSecurity 0

OnlineBackup 0

DeviceProtection 0

TechSupport 0

StreamingTV 0

StreamingMovies 0

PaperlessBilling 0

PaymentMethod 0

Tenure 0

MonthlyCharge 0

Bandwidth\_GB\_Year 0

Item1 0

Item2 0

Item3 0

Item4 0

Item5 0

Item6 0

Item7 0

Item8 0

dtype: int64

Here I am checking for any missing values. If any were found, they would be replaced with dummy variables using one of the central tendencies. (Mean, Median, Mode)

# Rename Last 8 Survey Columns for better description of variables

churn\_df.rename(columns = {'Item1' : 'TimelyResponse',

'Item2' : 'Fixes' ,

'Item3' : 'Replacements' ,

'Item4' : 'Reliability' ,

'Item5' : 'Options' ,

'Item6' : 'Respectfulness' ,

'Item7' : 'Courteous' ,

'Item8' : 'Listening'},

inplace=True)

In this step I am renaming columns/variables of survey to easily recognizable features (ex: “Item1” will be renamed “TimelyResponse”).

# Remove less meaningful demographic variables from statistics description

churn\_df = churn.drop(columns=['Caseorder' ,

'Customer\_id' ,

'Interaction' ,

'UID' ,

'City' ,

'State' ,

'County' ,

'Zip' ,

'Lat' ,

'Lng' ,

'Population' ,

'Area' ,

'TimeZone' ,

'Job' ,

'Marital' ,

'PaymentMethod'])

Drop less meaningful identifiers (ex: “Customer ID”) & demographic columns (ex: zip code) from dataframe.

# Converting binary categorical variables to numeric variables

churn\_df['DummyChurn'] = [1 if v == 'Yes' else 0 for v in churn\_df['Churn']]

churn\_df['DummyTechie'] = [1 if v == 'Yes' else 0 for v in churn\_df['Techie']]

churn\_df['DummyPort\_modem'] = [1 if v == 'Yes' else 0 for v in churn\_df['Port\_modem']]

churn\_df['DummyTablet'] = [1 if v == 'Yes' else 0 for v in churn\_df['Tablet']]

churn\_df['DummyPhone'] = [1 if v == 'Yes' else 0 for v in churn\_df['Phone']]

churn\_df['DummyMultiple'] = [1 if v == 'Yes' else 0 for v in churn\_df['Multiple']]

churn\_df['DummyOnlineSecurity'] = [1 if v == 'Yes' else 0 for v in churn\_df['OnlineSecurity']]

churn\_df['DummyOnlineBackup'] = [1 if v == 'Yes' else 0 for v in churn\_df['OnlineBackup']]

churn\_df['DummyDeviceProtection'] = [1 if v == 'Yes' else 0 for v in churn\_df['DeviceProtection']]

churn\_df['DummyTechSupport'] = [1 if v == 'Yes' else 0 for v in churn\_df['TechSupport']]

churn\_df['DummyStreamingTV'] = [1 if v == 'Yes' else 0 for v in churn\_df['StreamingTV']]

churn\_df['DummyStreamingMovies'] = [1 if v == 'Yes' else 0 for v in churn\_df['StreamingMovies']]

churn\_df['DummyPaperlessBilling'] = [1 if v == 'Yes' else 0 for v in churn\_df['PaperlessBilling']]

Created dummy variables to encode categorical, yes/no data points into 1/0 numerical values.

# Converting ordinal categorical data into numeric variables

churn\_df['DummyInternetService'] = churn\_df.InternetService.map({'None' : 0, 'DSL' : 1, 'Fiber Optic' : 2})

churn\_df['DummyContract'] = churn\_df.InternetService.map({'Month-to-month' : 0, 'One year' : 1, 'Two Year' : 2})

churn\_df['DummyGender'] = churn\_df.Gender.map({'Nonbinary' : 0, 'Male' : 1, 'Female' : 2})

Created dummy variables to encode ordinal categorical data, ordinal data points into 0,1,2 numerical values.

# Drop original categorical features from dataframe

churn\_df = churn\_df.drop(columns=['Gender' ,

'Churn' ,

'Techie' ,

'Contract' ,

'Port\_modem' ,

'Tablet' ,

'InternetService' ,

'Phone' ,

'Multiple' ,

'OnlineSecurity' ,

'OnlineBackup',

'DeviceProtection' ,

'TechSupport' ,

'StreamingTV',

'StreamingMovies',

'PaperlessBilling'])

Removing original categorical features from dataframe

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

Univariate Statistics

#Create histograms of continuous variables

churn\_df[['Children',

'Age' ,

'Income' ,

'Outage\_sec\_perweek' ,

'Email' ,

'Contacts' ,

'Yearly\_equip\_failure' ,

'Tenure' ,

'MonthlyCharge' ,

'Bandwidth\_GB\_Year']].hist()

plt.savefig('churn\_pyplot.jpg')

plt.tight\_layout()

Diagram, schematic

Description automatically generated

The visualizations of central tendency have revealed normal distribution for: Outage\_sec\_perweek, Email, and MonthlyCharge. When analyzing the visualization for Bandwidth\_GB\_Year and Tenure, the histogram displays a bimodal distribution.

# Create Seaborn Boxplots for continuous variables

sns.boxplot('Tenure' , data = churn\_df)

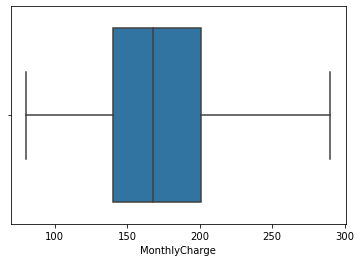
plt.show()

Chart

Description automatically generated

sns.boxplot('MonthlyCharge' , data = churn\_df)

plt.show()



sns.boxplot('Bandwidth\_GB\_Year' , data = churn\_df)

plt.show()

Chart, histogram

Description automatically generated

It appears that anomalies have been removed from the dataset present "churn\_clean.csv" as there are no remaining outliers.

Bivariate Statistics

# Run Scatterplots to show direct or inverse relationships between the target & independent variables

sns.scatterplot(x=churn\_df['Children'], y=churn\_df['Bandwidth\_GB\_Year'],

color='red')

plt.show()

Chart

Description automatically generated

sns.scatterplot(x=churn\_df['Age'], y=churn\_df['Bandwidth\_GB\_Year'],

color='red')

plt.show()

Chart

Description automatically generated

sns.scatterplot(x=churn\_df['Income'], y=churn\_df['Bandwidth\_GB\_Year'],

color='red')

plt.show()

Chart, scatter chart

Description automatically generated

sns.scatterplot(x=churn\_df['Outage\_sec\_perweek'], y=churn\_df['Bandwidth\_GB\_Year'],

color='red')

plt.show()

Chart, scatter chart

Description automatically generated

sns.scatterplot(x=churn\_df['Email'], y=churn\_df['Bandwidth\_GB\_Year'],

color='red')

plt.show()

Chart, scatter chart

Description automatically generated

sns.scatterplot(x=churn\_df['Contacts'], y=churn\_df['Bandwidth\_GB\_Year'],

color='red')

plt.show()

Chart

Description automatically generated

sns.scatterplot(x=churn\_df['Yearly\_equip\_failure'], y=churn\_df['Bandwidth\_GB\_Year'],

color='red')

plt.show()

Chart

Description automatically generated

sns.scatterplot(x=churn\_df['Tenure'], y=churn\_df['Bandwidth\_GB\_Year'],

color='red')

plt.show()

Chart, scatter chart

Description automatically generated

sns.scatterplot(x=churn\_df['MonthlyCharge'], y=churn\_df['Bandwidth\_GB\_Year'],

color='red')

plt.show()

Chart, scatter chart

Description automatically generated

sns.scatterplot(x=churn\_df['TimelyResponse'], y=churn\_df['Bandwidth\_GB\_Year'],

color='red')

plt.show()

Chart, histogram

Description automatically generated

sns.scatterplot(x=churn\_df['Fixes'], y=churn\_df['Bandwidth\_GB\_Year'],

color='red')

plt.show()

Chart, histogram

Description automatically generated

sns.scatterplot(x=churn\_df['DummyTechie'], y=churn\_df['Bandwidth\_GB\_Year'],

color='red')

plt.show()

Chart

Description automatically generated

sns.scatterplot(x=churn\_df['DummyGender'], y=churn\_df['Bandwidth\_GB\_Year'],

color='red')

plt.show()

Chart

Description automatically generated

sns.scatterplot(x=churn\_df['DummyContract'], y=churn\_df['Bandwidth\_GB\_Year'],

color='red')

plt.show()

Chart, shape

Description automatically generated

sns.scatterplot(x=churn\_df['DummyInternetService'], y=churn\_df['Bandwidth\_GB\_Year'],

color='red')

plt.show()

Chart, shape

Description automatically generated

After analyzing the scatterplots between the target variable and the independent variables, ‘Tenure’ signals a direct linear relationship in the scatterplot.

1. Provide a copy of the prepared data set

churn\_df.to\_csv('churn\_prepared.csv')

df = churn\_df.columns

print(df)

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

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

# Model including all dummy variables

lm\_bandwidth = ols("Bandwidth\_GB\_Year ~ Children + Age + Income + Outage\_sec\_perweek + Email +Contacts +Yearly\_equip\_failure +Tenure + MonthlyCharge + TimelyResponse + Fixes + Replacements + Reliability + Options + Respectfulness + Courteous + Listening + DummyChurn + DummyTechie + DummyPort\_modem + DummyTablet + DummyPhone + DummyMultiple + DummyOnlineSecurity + DummyOnlineBackup + DummyDeviceProtection + DummyTechSupport + DummyStreamingTV + DummyStreamingMovies + DummyPaperlessBilling + DummyInternetService + DummyContract + DummyGender", data=churn\_df).fit()

print(lm\_bandwidth.params)

print(lm\_bandwidth.summary())

Intercept 500.921984

Children 30.422558

Age -3.315139

Income 0.000101

Outage\_sec\_perweek -0.795091

Email -0.028446

Contacts 1.985727

Yearly\_equip\_failure 1.327325

Tenure 82.773076

MonthlyCharge -1.433662

TimelyResponse -5.386324

Fixes 4.945364

Replacements -2.034113

Reliability 0.572896

Options 3.261947

Respectfulness 0.588805

Courteous 0.644903

Listening 4.512656

DummyChurn 92.865973

DummyTechie -2.582109

DummyPort\_modem -2.935664

DummyTablet -0.084313

DummyPhone -2.386146

DummyMultiple 114.059281

DummyOnlineSecurity 83.020282

DummyOnlineBackup 120.250807

DummyDeviceProtection 97.520444

DummyTechSupport 28.109653

DummyStreamingTV 269.736465

DummyStreamingMovies 262.543483

DummyPaperlessBilling -5.941383

DummyInternetService -29.316313

DummyContract 13.562562

DummyGender -48.183704

dtype: float64

OLS Regression Results

===============================================================

Dep. Variable: Bandwidth\_GB\_Year R-squared: 0.992

Model: OLS Adj. R-squared: 0.992

Method: Least Squares F-statistic: 3.852e+04

Date: Mon, 22 Nov 2021 Prob (F-statistic): 0.00

Time: 20:18:13 Log-Likelihood: -66803.

No. Observations: 10000 AIC: 1.337e+05

Df Residuals: 9966 BIC: 1.339e+05

Df Model: 33

Covariance Type: nonrobust

===============================================================

coef std err t P>|t| [0.025 0.975]

-----------------------------------------------------------------------------------------

Intercept 500.9220 27.489 18.223 0.000 447.038 554.806

Children 30.4226 0.901 33.770 0.000 28.657 32.188

Age -3.3151 0.093 -35.477 0.000 -3.498 -3.132

Income 0.0001 6.86e-05 1.474 0.140 -3.33e-05 0.000

Outage\_sec\_perweek -0.7951 0.650 -1.223 0.221 -2.069 0.479

Email -0.0284 0.639 -0.045 0.965 -1.281 1.224

Contacts 1.9857 1.956 1.015 0.310 -1.849 5.821

Yearly\_equip\_failure 1.3273 3.040 0.437 0.662 -4.633 7.287

Tenure 82.7731 0.087 946.944 0.000 82.602 82.944

MonthlyCharge -1.4337 0.217 -6.596 0.000 -1.860 -1.008

TimelyResponse -5.3863 2.769 -1.945 0.052 -10.815 0.042

Fixes 4.9454 2.595 1.906 0.057 -0.142 10.033

Replacements -2.0341 2.380 -0.855 0.393 -6.699 2.631

Reliability 0.5729 2.128 0.269 0.788 -3.598 4.744

Options 3.2619 2.210 1.476 0.140 -1.069 7.593

Respectfulness 0.5888 2.275 0.259 0.796 -3.870 5.047

Courteous 0.6449 2.152 0.300 0.764 -3.574 4.864

Listening 4.5127 2.047 2.204 0.028 0.499 8.526

DummyChurn 92.8660 5.986 15.515 0.000 81.133 104.599

DummyTechie -2.5821 5.189 -0.498 0.619 -12.754 7.590

DummyPort\_modem -2.9357 3.867 -0.759 0.448 -10.516 4.644

DummyTablet -0.0843 4.226 -0.020 0.984 -8.369 8.200

DummyPhone -2.3861 6.655 -0.359 0.720 -15.431 10.658

DummyMultiple 114.0593 8.039 14.189 0.000 98.302 129.817

DummyOnlineSecurity 83.0203 4.078 20.357 0.000 75.026 91.014

DummyOnlineBackup 120.2508 6.238 19.278 0.000 108.024 132.478

DummyDeviceProtection 97.5204 4.750 20.531 0.000 88.210 106.831

DummyTechSupport 28.1097 4.824 5.827 0.000 18.654 37.566

DummyStreamingTV 269.7365 9.915 27.205 0.000 250.301 289.172

DummyStreamingMovies 262.5435 11.964 21.944 0.000 239.091 285.996

DummyPaperlessBilling -5.9414 3.930 -1.512 0.131 -13.645 1.762

DummyInternetService -29.3163 4.429 -6.620 0.000 -37.997 -20.636

DummyContract 13.5626 2.440 5.559 0.000 8.780 18.345

DummyGender -48.1837 3.558 -13.542 0.000 -55.158 -41.209

===============================================================

Omnibus: 126416.010 Durbin-Watson: 1.977

Prob(Omnibus): 0.000 Jarque-Bera (JB): 1054.279

Skew: 0.438 Prob(JB): 1.16e-229

Kurtosis: 1.672 Cond. No. 7.46e+05

===============================================================

Notes:

[1] Standard Errors assume that the covariance matrix of the errors is correctly specified.

[2] The condition number is large, 7.46e+05. This might indicate that there are

strong multicollinearity or other numerical problems.

**Initial Multiple Linear Regression Model**

**With 33 independent variables (18 continuous & 15 categorical): y = 500.92 + 30.42 \* Children - 3.31 \* Age + 0.00 \* Income - 0.80 \* Outage\_sec\_perweek - 0.28 \* Email + 1.99 \* Contacts + 1.32 \* Yearly\_equip\_failure + 82.77 \* Tenure - 1.43 \* MonthlyCharge - 5.39 \* TimelyResponse + 4.95 \* Fixes - 2.04 \* Replacements + 0.57 \* Reliability + 3.26 \* Options + 0.59 \* Respectfulness +0.65 \* Courteous + 4.51 \* Listening + 92.86 \* DummyChurn - 2.58 \* DummyTechie - 2.94 \* DummyPort\_modem - 0.08 \* DummyTablet - 2.37 \* DummyPhone + 114.06 \* DummyMultiple + 83.02 \* DummyOnlineSecurity + 120.25 \* DummyOnlineBackup + 97.52 DummyDeviceProtection + 28.11 \* DummyTechSupport + 268.73 \* DummyStreamingTV + 262.54 \* DummyStreamingMovies - 5.94 \* DummyPaperlessBilling - 28.32 \* DummyInternetService + 13.56 \* DummyContract - 48.18 \*DummyGender**

**Based on an R2 value = 0.989. So, 99% of the variation is explained by this model. The condition number is large which might suggest strong multicollinearity. It appears that we do not need all of these variables to explain the variance. So, let’s run a heatmap for bivariate analysis & a principal component analysis in order to reduce variables.**

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.

# Create dataframe for heatmap bivariate analysis of correlation

churn\_bivariate = churn\_df[['Bandwidth\_GB\_Year',

'Children',

'Age',

'Income',

'Outage\_sec\_perweek',

'Yearly\_equip\_failure',

'DummyTechie',

'DummyContract',

'DummyPort\_modem',

'DummyTablet',

'DummyInternetService',

'DummyPhone',

'DummyMultiple',

'DummyOnlineSecurity',

'DummyOnlineBackup',

'DummyDeviceProtection',

'DummyTechSupport',

'DummyStreamingTV',

'DummyPaperlessBilling',

'DummyGender' ,

'DummyStreamingMovies' ,

'Email',

'Contacts',

'Tenure',

'MonthlyCharge',

'TimelyResponse',

'Fixes',

'Replacements',

'Reliability',

'Options',

'Respectfulness',

'Courteous',

'Listening']]

# Run Seaborn heatmap

sns.heatmap(churn\_bivariate.corr(), annot=False)

plt.show()

Chart

Description automatically generated

I will remove some of the variables that do appear to have less significance.

churn\_bivariate = churn\_df[['Bandwidth\_GB\_Year',

'Children',

'Tenure',

'TimelyResponse',

'Fixes',

'Replacements',

'Respectfulness',

'Courteous',

'Listening']]

sns.heatmap(churn\_bivariate.corr(), annot=True)

plt.show()

Chart

Description automatically generated with medium confidence

It appears that Tenure is the predictor for most of the variance. There is clearly a direct linear relationship between customer tenure and the amount of data being used. I will now run a multiple linear regression model on those variables with 0.5 or above. I will add children because the P value came out 0.00 which showed some significance.

The reduced regression equation will include the continuous variable for tenure and the variables children, fixes, and replacements.

3. Reduced Multiple Regression Model

# Run reduced OLS multiple regression

lm\_bandwidth = ols("Bandwidth\_GB\_Year ~ Children + Tenure + Fixes + Replacements", data=churn\_df).fit()

print(lm\_bandwidth.params)

print(lm\_bandwidth.summary())

Intercept 506.769517

Children 31.176273

Tenure 81.951763

Fixes 1.072830

Replacements -3.658451

dtype: float64

OLS Regression Results

==============================================================================

Dep. Variable: Bandwidth\_GB\_Year R-squared: 0.984

Model: OLS Adj. R-squared: 0.984

Method: Least Squares F-statistic: 1.537e+05

Date: Mon, 22 Nov 2021 Prob (F-statistic): 0.00

Time: 21:50:02 Log-Likelihood: -70407.

No. Observations: 10000 AIC: 1.408e+05

Df Residuals: 9995 BIC: 1.409e+05

Df Model: 4

Covariance Type: nonrobust

================================================================================

coef std err t P>|t| [0.025 0.975]

--------------------------------------------------------------------------------

Intercept 506.7695 11.949 42.413 0.000 483.348 530.191

Children 31.1763 1.288 24.211 0.000 28.652 33.700

Tenure 81.9518 0.105 783.845 0.000 81.747 82.157

Fixes 1.0728 3.129 0.343 0.732 -5.061 7.206

Replacements -3.6585 3.149 -1.162 0.245 -9.831 2.514

==============================================================================

Omnibus: 380.733 Durbin-Watson: 1.978

Prob(Omnibus): 0.000 Jarque-Bera (JB): 295.369

Skew: 0.334 Prob(JB): 7.27e-65

Kurtosis: 2.488 Cond. No. 191.

==============================================================================

Notes:

[1] Standard Errors assume that the covariance matrix of the errors is correctly specified.

After removing all the other predictor variables, the model still explains 98% of the variance.

Reduced Multiple Linear Regression Model

With 4 independent variables: y = 506.77 + 31.18 \* Children + 91.95 \* Tenure + 1.07 \* Fixes – 3.66 \* Replacements

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

1. Explain your data analysis process by comparing the initial and recued multiple regression models, including the following elements:

* The logic of the variable selection technique
* The model evaluation metric
* A residual plot

churn\_df = pd.read\_csv('churn\_prepared.csv')

churn\_df['intercept'] = 1

residuals = churn\_df['Bandwidth\_GB\_Year'] = lm\_bandwidth\_reduced.predict(churn\_df[['Children', 'Tenure', 'Fixes','Replacements','intercept']])

sns.scatterplot(x=churn\_df['Tenure'],y=residuals,color='red')

plt.show()

Chart

Description automatically generated

F1. Results

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

* The final multiple regression equation with independent variables:

y = 506.77 + 31.18 \* Children + 91.95 \* Tenure + 1.07 \* Fixes – 3.66 \* Replacements

* Interpretation of coefficients of the statistically significant variables of the model

The coefficients suggest that for every unit of:

Children – Bandwidth\_GB\_Year will increase 31.19 units

Tenure – Bandwidth\_GB\_Year will increase 91.95 units

Fixes – Bandwidth\_GB\_Year will increase by 1.07 units

Replacements – Bandwidth\_GB\_Year will decrease by 3.66 units

* The statistical and practical significance of the model

P-values for Children & Tenure are statistically significant at 0.00, while p-values for Fixes and replacements were -5.06 and – 9.83.

The limitations of this analysis are that the data set is smaller than anticipated & perhaps more historical data is needed to provide a clearer understanding.

F2. Recommend a course of action based on your results.

With such a direct linear relationship between Bandwidth\_GB\_Year and tenure, it makes the most logical sense to suggest that the company find incentives within customer satisfaction to retain the customers gained as the longer they stay with the company, the more bandwidth they tend to use. This would include to ensure fixes are prompt for the customer and need for fewer replacements.

Panopto Video Link:

<https://wgu.hosted.panopto.com/Panopto/Pages/Viewer.aspx?id=df2aa705-09bd-46f6-8517-adeb012e57ee>

REFERENCES

Frankenfield, J. (2021, May 19). *Churn rate*. Investopedia. Retrieved October 24, 2021, from <https://www.investopedia.com/terms/c/churnrate.asp>

Hayes, A. (2021, September 7). *Multiple linear regression (MLR) definition*. Investopedia. Retrieved October 25, 2021, from <https://www.investopedia.com/terms/m/mlr.asp>

A. K., By, -, Aniruddha Kalbandehttps://www.fireblazeaischool.in/blogsOn the mission to train 1, Kalbande, A., & On the mission to train 1. (2021, July 17). *Assumptions of linear regression algorithm - blogs: Fireblaze AI School*. Blogs | Fireblaze AI School. Retrieved October 24, 2021, from <https://www.fireblazeaischool.in/blogs/assumptions-of-linear-regression/>

*Assumptions of multiple linear regression*. Statistics Solutions. (2021, August 11). Retrieved October 25, 2021, from <https://www.statisticssolutions.com/free-resources/directory-of-statistical-analyses/assumptions-of-multiple-linear-regression/>

*What is numpy?¶*. What is NumPy? - NumPy v1.21 Manual. (n.d.). Retrieved November 20, 2021, from <https://numpy.org/doc/stable/user/whatisnumpy.html>

*What is Pandas in python? Educative.io. (n.d.). Retrieved November 20, 2021, from* [*https://www.educative.io/edpresso/what-is-pandas-in-python*](https://www.educative.io/edpresso/what-is-pandas-in-python)

*An introduction to seaborn¶*. An introduction to seaborn - seaborn 0.11.2 documentation. (n.d.). Retrieved November 20, 2021, from <https://seaborn.pydata.org/introduction.html>

Matplotlib - PyLab module. (n.d.). Retrieved November 20, 2021, from <https://www.tutorialspoint.com/matplotlib/matplotlib_pylab_module.htm>

Scikit learn - introduction. (n.d.). Retrieved November 20, 2021, from <https://www.tutorialspoint.com/scikit_learn/scikit_learn_introduction.html>

Urdan, T. C. (2010). Statistics in Plain English. Routledge.