Matthew Blessing
Student ID: ***000507424***
D208 Predictive Modeling

Performance Assessment Task 1: Predictive Modeling - NBM2

Part I: Research Question

A1.

Using the provided ‘Churn’ data set and a multiple regression model, we will attempt to see if it is possible to somewhat accurately predict the amount of bandwidth a customer will use annually based on customer age, income, gender identity, and tenure.

A2.

The goal of this analysis is to use a multiple regression model to determine the viability of being able to predict customer bandwidth usage. The ability to somewhat accurately predict bandwidth usage by customer demographics will allow the business to more accurately cater services to customers and be able to build the infrastructure necessary to support the required bandwidth.

Part II: Method Justification

B1.

A multiple regression model generally assumes four things:

1. The relationship between the independent and dependent variables must be linear
2. Errors between the observed and predicted values (residuals) should be normally distributed
3. There should be no multicollinearity in the data - the independent variables should not be highly correlated with each other
4. Homoscedasticity - the residuals should be equal across the line of regression (*Assumptions of Multiple Linear Regression*, 2021)

B2.

For this analysis, we will be utilizing Python within Jupyter Notebooks.

Python is an object-oriented programming language that is extremely popular for data science due to it being a powerful, easy to learn language that is extremely expandable with a large library of data science packages, such as NumPy, SciPy, Pandas, and Matplotlib. These libraries easily allow users to implement classification, regression, machine learning and more on chosen data sets (*Advantages of Learning Python for Data Science*, n.d.).

Jupyter Notebook is an open-source web application that allows you to create and share documents that contain live code, equations, visualizations and narrative text (*Project Jupyter*, n.d.).

B3.

Multiple regression should be useful in this analysis because we would like to determine if we are able to predict the value of a dependent variable (bandwidth used by customers per year) based on other independent/predictor variables (age, income, gender identity, and tenure).

Part III: Data Preparation

C1.

Before we can create a multiple regression model, there are some preparations required for the data. To begin, we will want to ensure the data is clean by removing duplicate rows of data using the *drop_duplicates()* function. The data will also be checked for null values using the *isnull().sum()* function. Luckily, the data appears to be clean with no null values or duplicate rows.

Since we are also using a categorical variable, we will need to use Pandas *get_dummies()* function to convert the categorical variables to binary variables (1 = presence of the categorical variable, 0 = absence of the categorical variable) in order to be able to use this variable in our model.

We will also need to explore the variables we want to use for our model using summary statistics to ensure our independent variables have a linear relationship with our dependent variable.

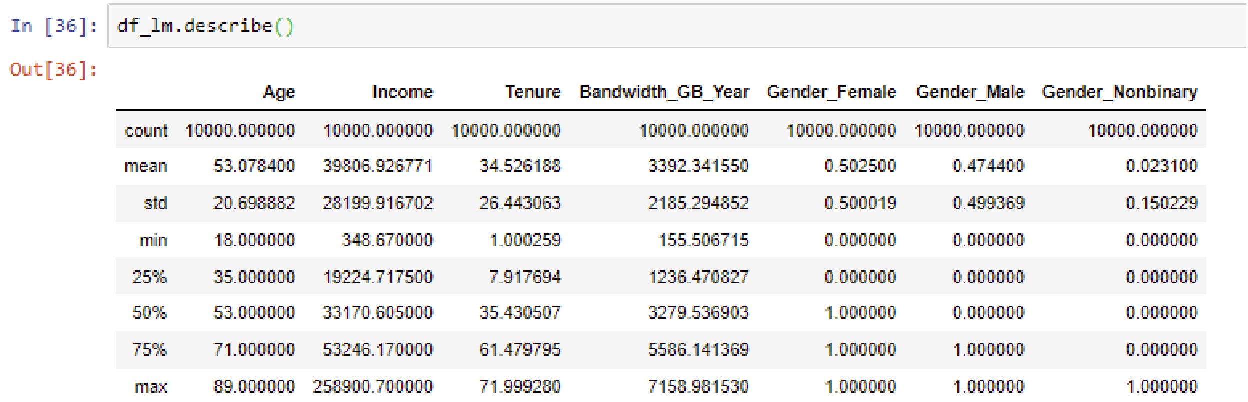
Once we have determined all of the variables we will want to use for the model, that the variables appear to have a linear relationship, and that our independent variables are not highly correlated with each other, we will create a new dataframe containing only the variables we will be using for our model.

C2.

To answer the question of whether we can use age, income, gender identity, and tenure to predict how much bandwidth a customer will use annually, we will need to first look at some summary statistics to understand the distributions and of our independent and dependent variables. We will also want to look at bivariate scatter plots to confirm a linear relationship between age, income, gender identity, and tenure and our dependent variable, ‘Bandwidth_GB_Year’. Lastly, we will want to compare all of our independent variables to look for multicollinearity between them, and if we discover multicollinearity, we will need to reframe our research question and select new variables.

Before we begin preparing our data for our model, we should explore our independent and dependent variables to explore the number of observations, measures of central tendency.

We can get a good look at the measures of central tendency with our selected variables by using pandas *.describe()* function.



```
In [36]: df_lm.describe()
Out[36]:
```

	Age	Income	Tenure	Bandwidth_GB_Year	Gender_Female	Gender_Male	Gender_Nonbinary
count	10000.000000	10000.000000	10000.000000	10000.000000	10000.000000	10000.000000	10000.000000
mean	53.078400	39806.926771	34.526188	3392.341550	0.502500	0.474400	0.023100
std	20.698882	28199.916702	26.443063	2185.294852	0.500019	0.499369	0.150229
min	18.000000	348.670000	1.000259	155.506715	0.000000	0.000000	0.000000
25%	35.000000	19224.717500	7.917694	1236.470827	0.000000	0.000000	0.000000
50%	53.000000	33170.605000	35.430507	3279.536903	1.000000	0.000000	0.000000
75%	71.000000	53246.170000	61.479795	5586.141369	1.000000	1.000000	0.000000
max	89.000000	258900.700000	71.999280	7158.981530	1.000000	1.000000	1.000000

As we can see, our selected variables for our model include five variables - four independent variables - Age, Income, Tenure, Gender (Note that in the above screenshot, 'Gender' has been broken out into dummy variables for input into our model, the process for this is discussed below in section C3, the 'Gender' variable specifics are discussed below), and one dependent variable, 'Bandwidth_GB_Year'. Some notes on our variables:

- All of our selected variables have 10,000 entries in this data set
- 'Age' is a continuous variable with a mean of 53.07 years
- 'Income' is a continuous variable with a mean of $39,808
- 'Tenure' is a discrete variable, with a mean of 34.5 months
- 'Bandwidth_GB_Year' is a continuous variable with a mean of 3,392GB

Since Gender is a nominal variable, we will have to look at the mode as a measure of central tendency. We can look at this using Panda's *.value_counts()* function on the 'Gender' column with the following code:

```
df.Gender.value_counts()

Female        5025
Male          4744
Nonbinary      231
Name: Gender, dtype: int64
```

As we can see, there are 10,000 entries for 'Gender', with 5,025 Female, 4,744 Male, and 231 Nonbinary.

C3.

To prepare our data for the analysis, we begin by importing our data science libraries and tools we will use for the analysis, which include Pandas, NumPy, Matplotlib, StatsModel and Seaborn:

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
%matplotlib inline
import seaborn as sns
import statsmodels.api as sm
sns.set_style('darkgrid')

#sets the jupyter notebook window to take up 90% width of the
browser window
from IPython.core.display import display, HTML
display(HTML("<style>.container { width:90% !important;
}</style>"))
```

We then read in the data into a dataframe via Pandas with the following code:

```
df = pd.read_csv('churn_clean.csv')
```

We then drop any duplicate rows:

```
#drops any duplicate rows
df.drop_duplicates()
```

We also check for any null values:

```
#gives us the count of any null values in the data
df.isnull().sum()
```

We then create a new dataframe containing only our independent and dependent variables:

```
#creates a new dataframe containing only the columns ‘Age’,
‘Income’, ‘Gender’, ‘Tenure’, and ‘Bandwidth_GB_year’ by copying
those columns and values from the existing dataframe
df_lm = df[['Age', ‘Income’, ‘Gender’, 'Tenure',
'Bandwidth_GB_Year']].copy()
```

We will need to create dummy variables for our categorical variables (Gender), in order to use this independent variable in our model:

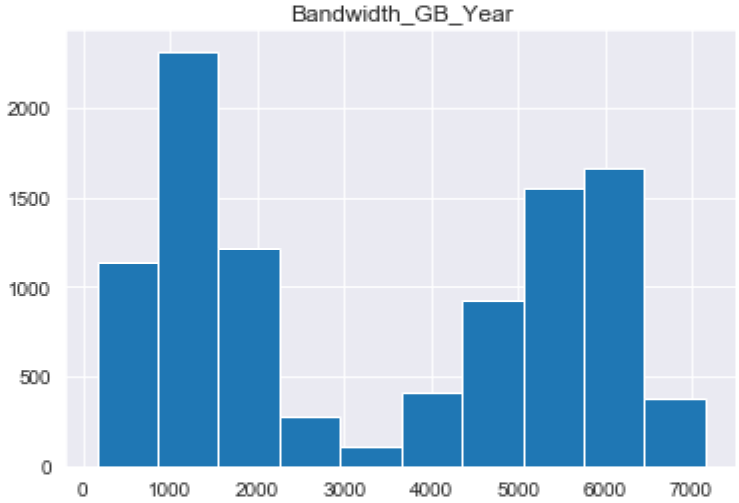
```
#creates dummy variables to convert the categorical variables to
binary values in order to be able to use them in our model
df_lm = pd.get_dummies(data=df_lm, drop_first=False)
df_lm.head()
```

We also export the newly created dataframe to a .csv:

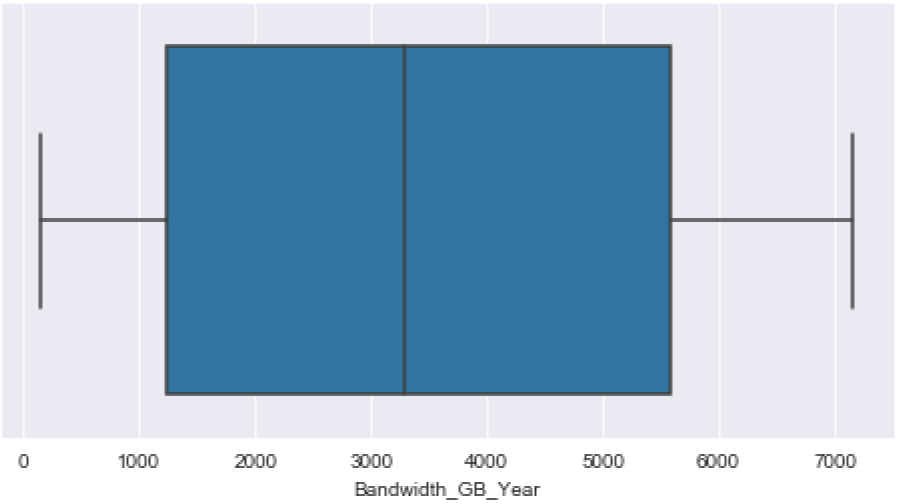
```
#exports the new dataframe to a .csv
df_lm.to_csv('D208_Data_set.csv')
```

C4.

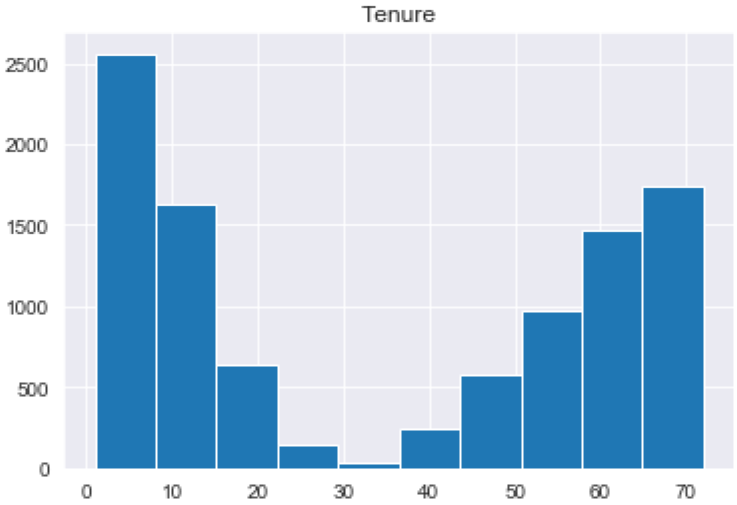
To begin exploring the data, we will first look at the target variable we will be attempting to predict, 'Bandwidth_GB_Year', which is defined in the data dictionary as "the average amount of data used, in GB, in a year by the customer (if the customer is newer than a year, this value is approximated based on initial use or of average usage for a typical customer in their demographic profile)". We will examine the distribution and use a box plot to look at the quartiles.



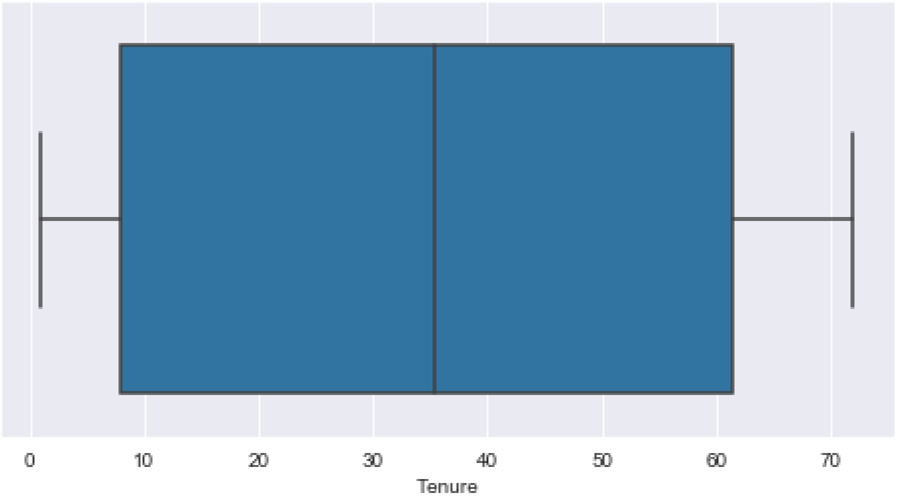
Interestingly enough, it appears as though 'Bandwidth_GB_Year' has a bimodal distribution.

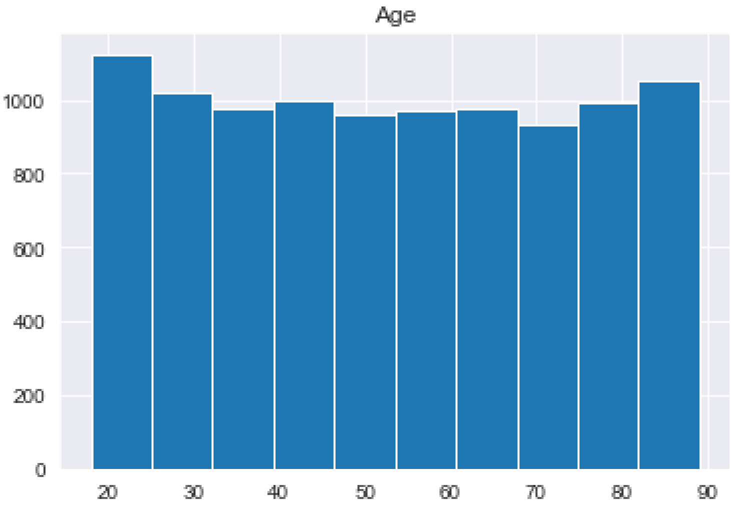


Next, we will look at our chosen independent variables distributions and box plots:

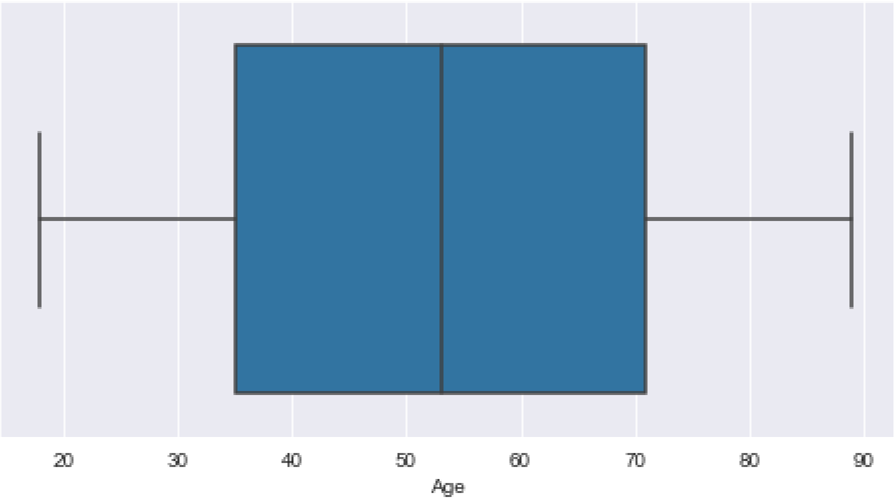


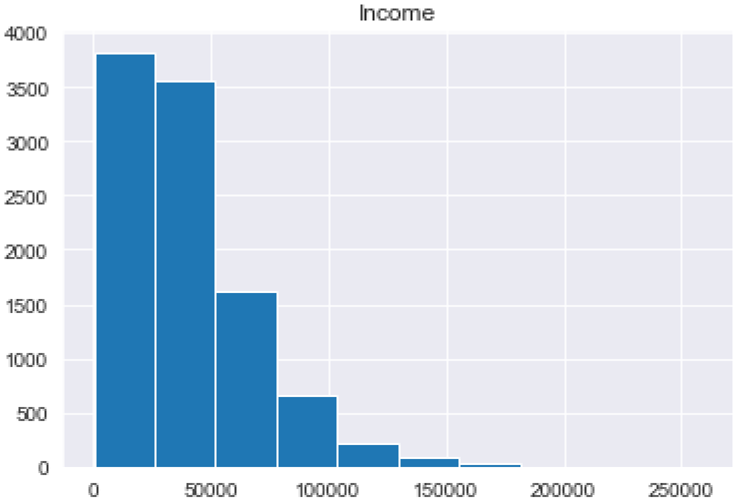
‘Tenure’ also appears to have a bimodal distribution.



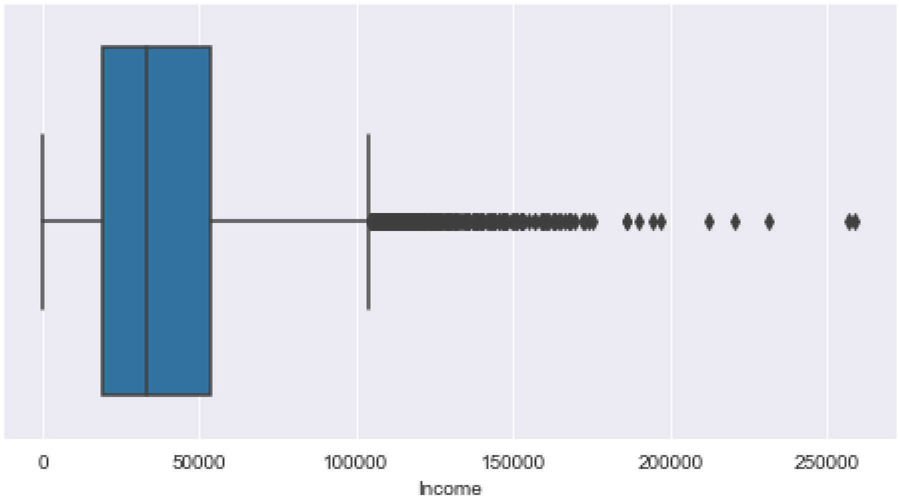


‘Age’ has a relatively uniform distribution.

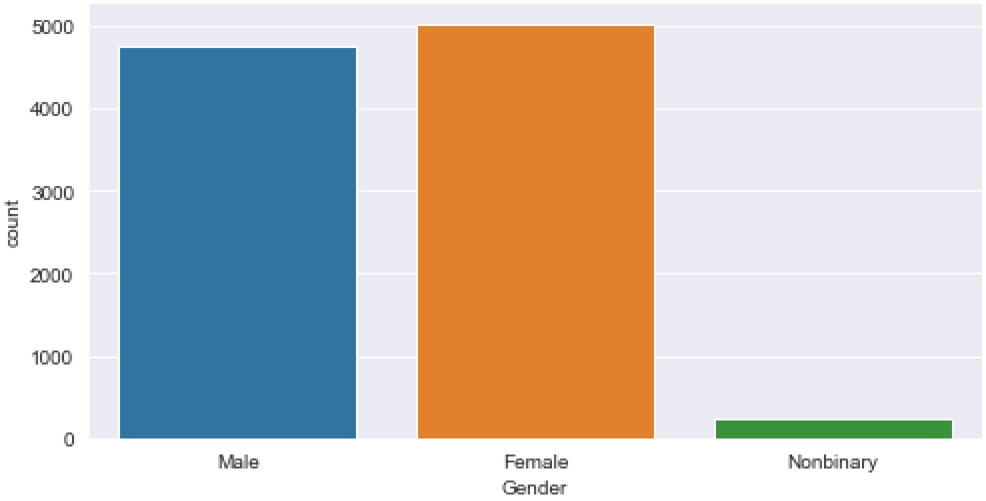




The distribution for ‘Income’ is right-skewed, which is to be expected.

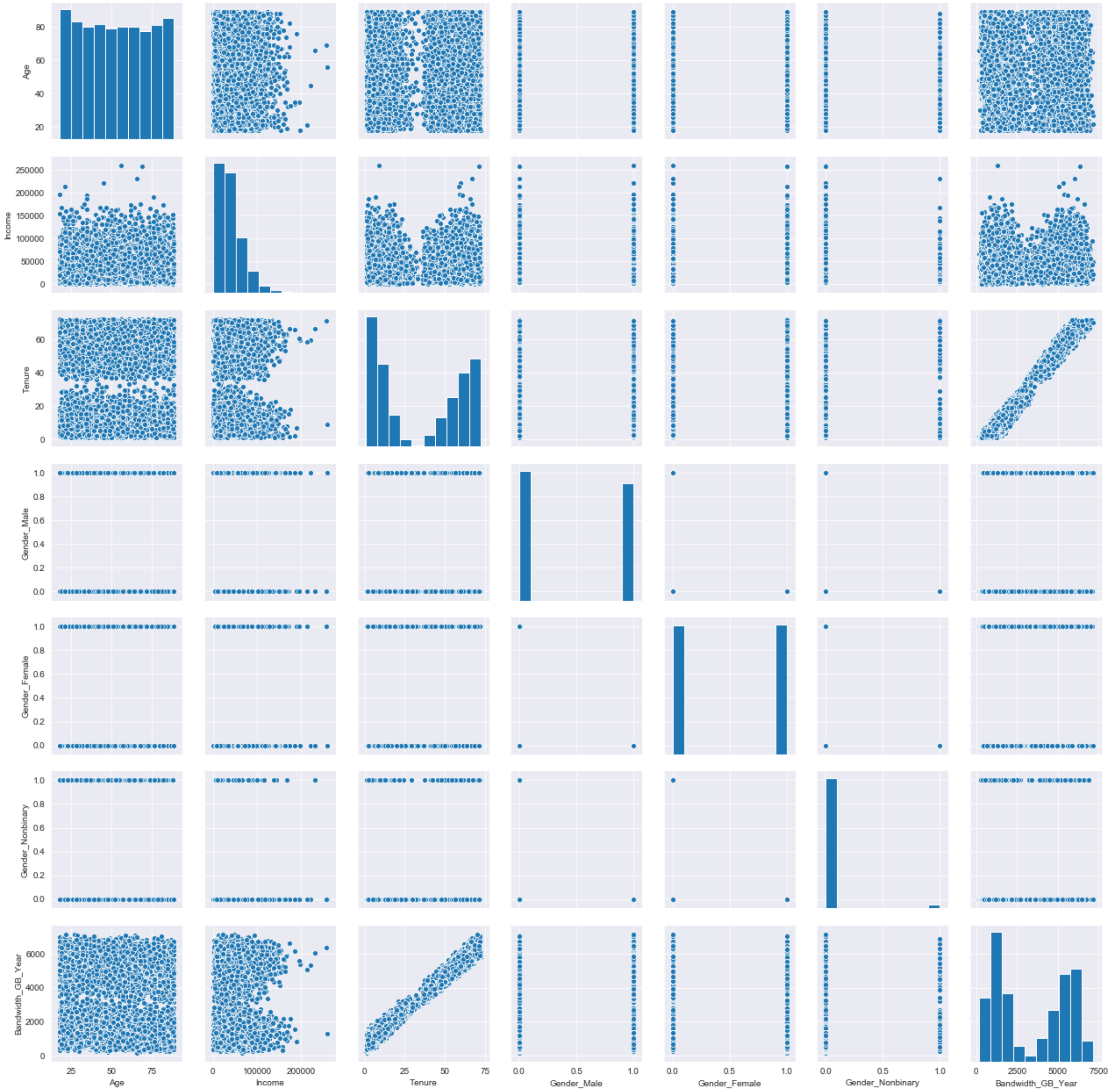


We see several outliers on the high-end of ‘Income’, which is also to be expected.



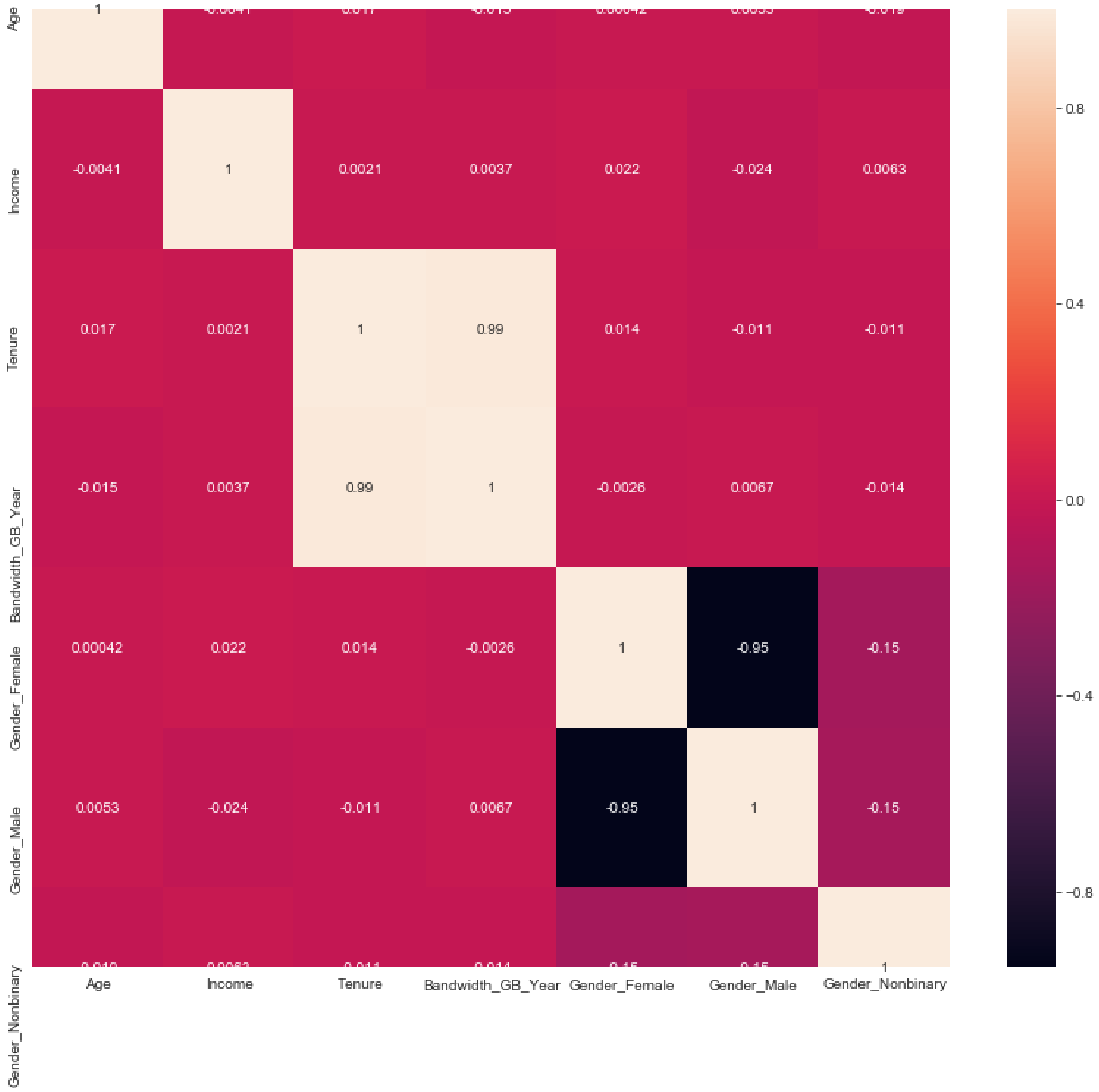
A countplot of ‘Gender’ reveals that Female customers slightly outnumber Male, followed by Nonbinary.

In addition to the above descriptive statistics of our chosen variables, we can use Seaborn's 'pairplot' function to generate a pair plot between our dependent and independent variables to get a quick look at their relationships:



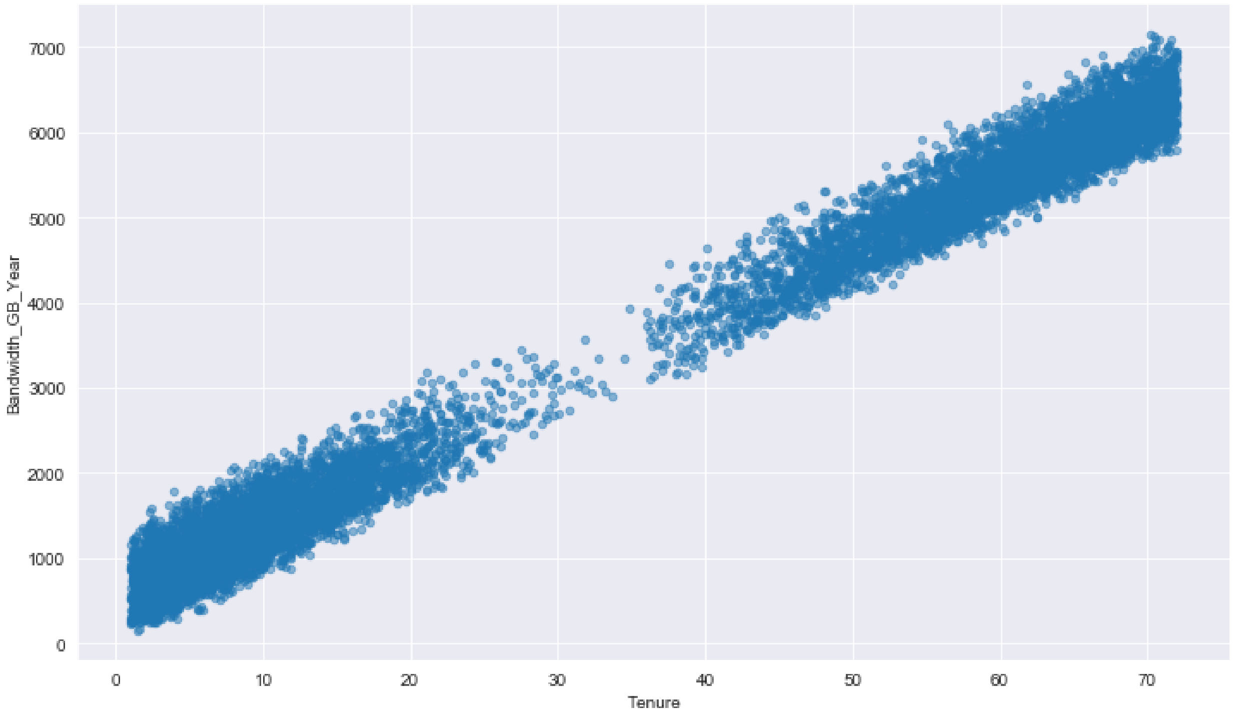
As we can see in this pair plot, there is a very strong linear relationship between 'Tenure' and 'Bandwidth_GB_Year'. Our other independent variables also appear to have a linear relationship, but not nearly as strong as 'Tenure'. Our independent variables also do not appear to be strongly correlated with each other, so we will avoid the problem of multicollinearity.

Correlation heat map that shows an extremely strong correlation (0.99) between ‘Tenure’ and ‘Bandwidth_GB_Year’:

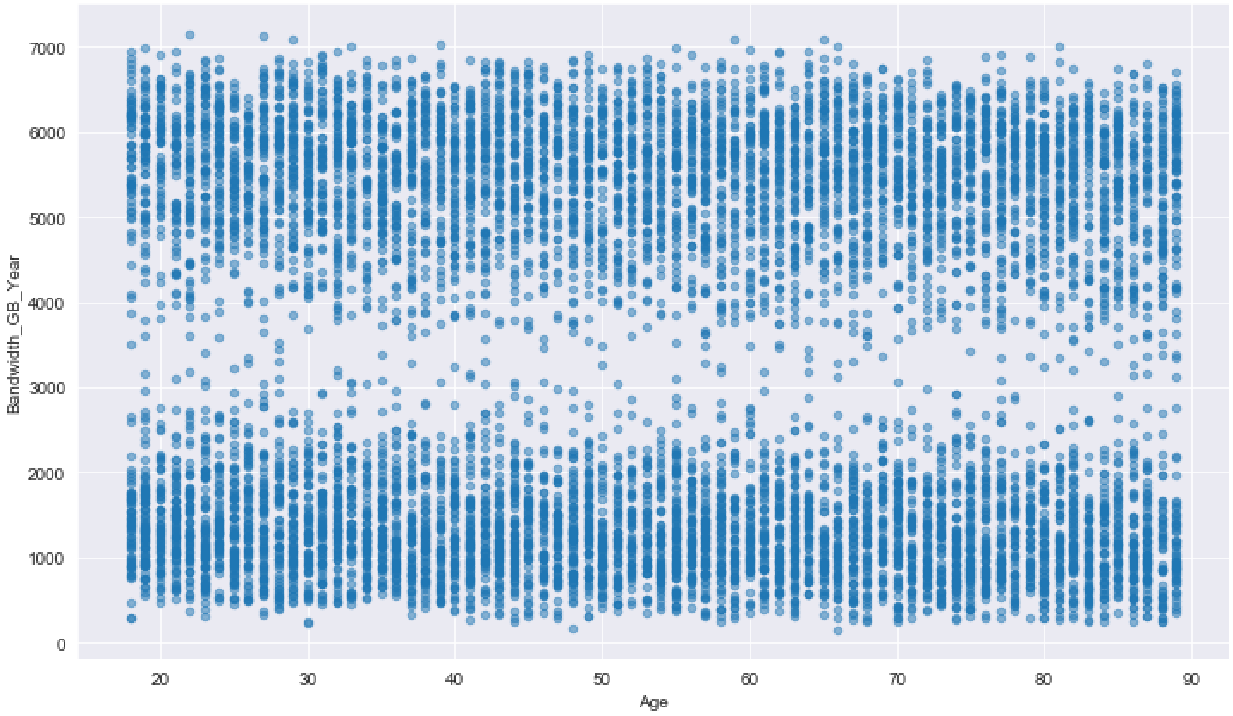


We can also look at individual bivariate plots between our dependent and independent variables:

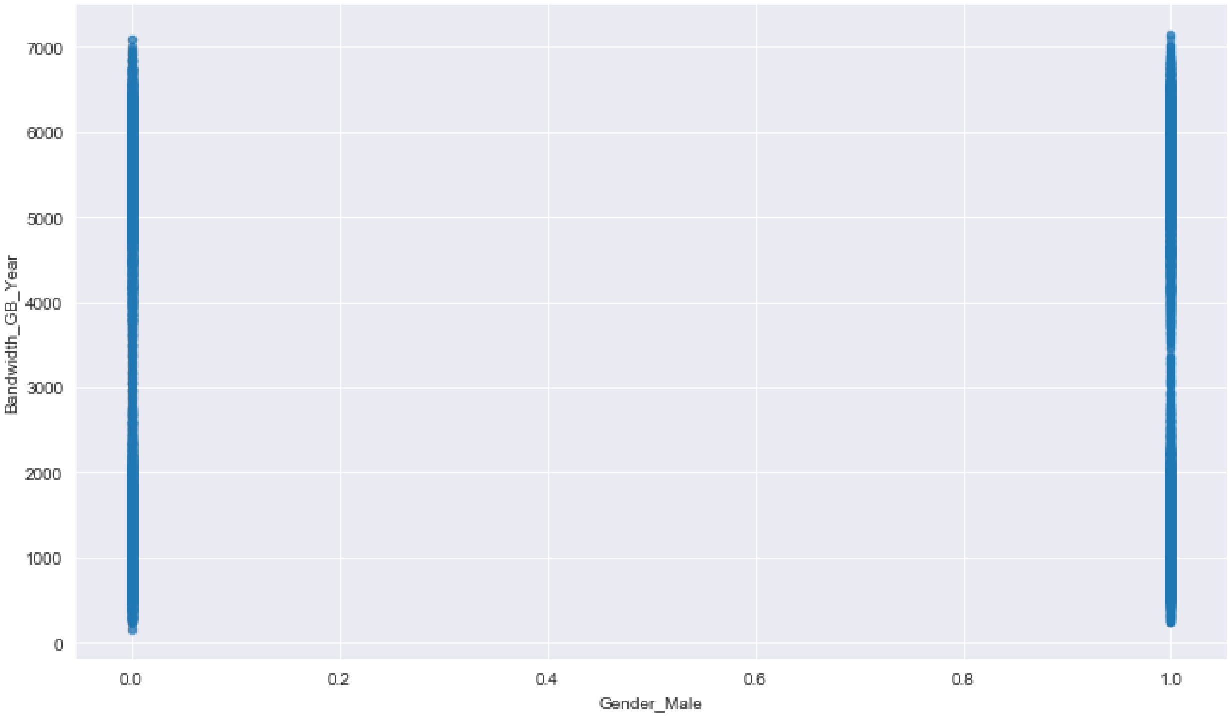
Tenure vs. Bandwidth_GB_Year



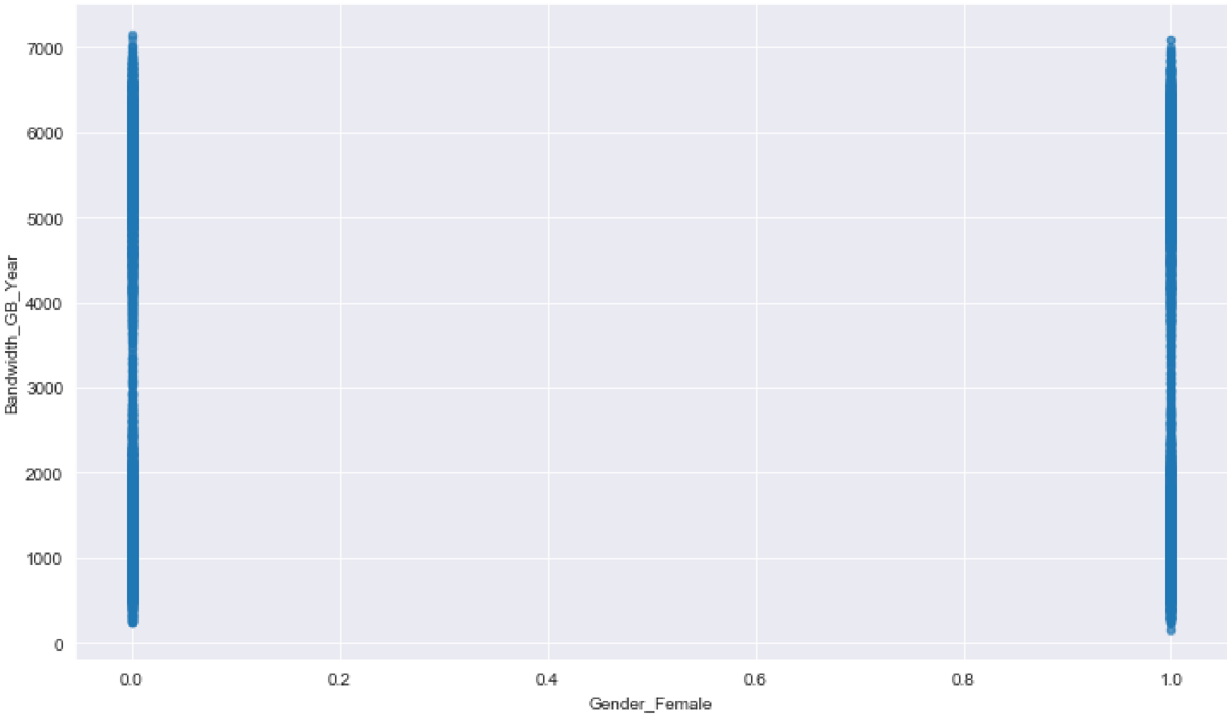
Age vs. Bandwidth_GB_Year



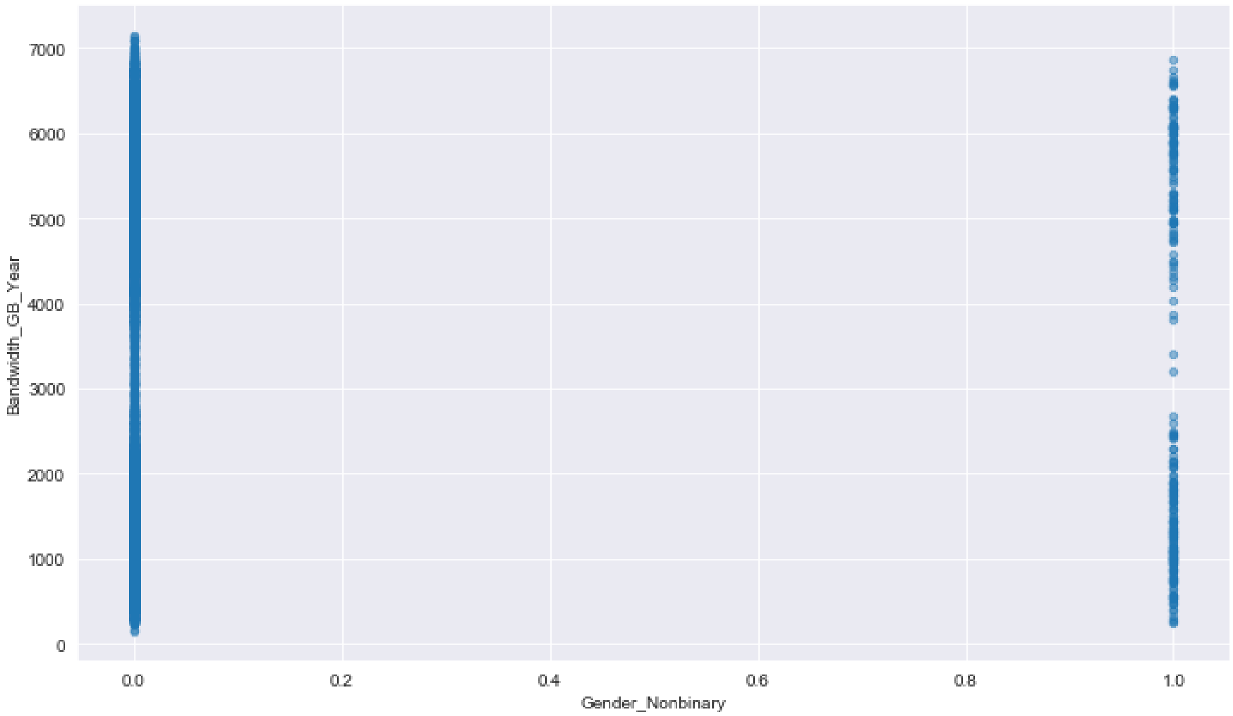
Gender_Male vs Bandwidth_GB_Year



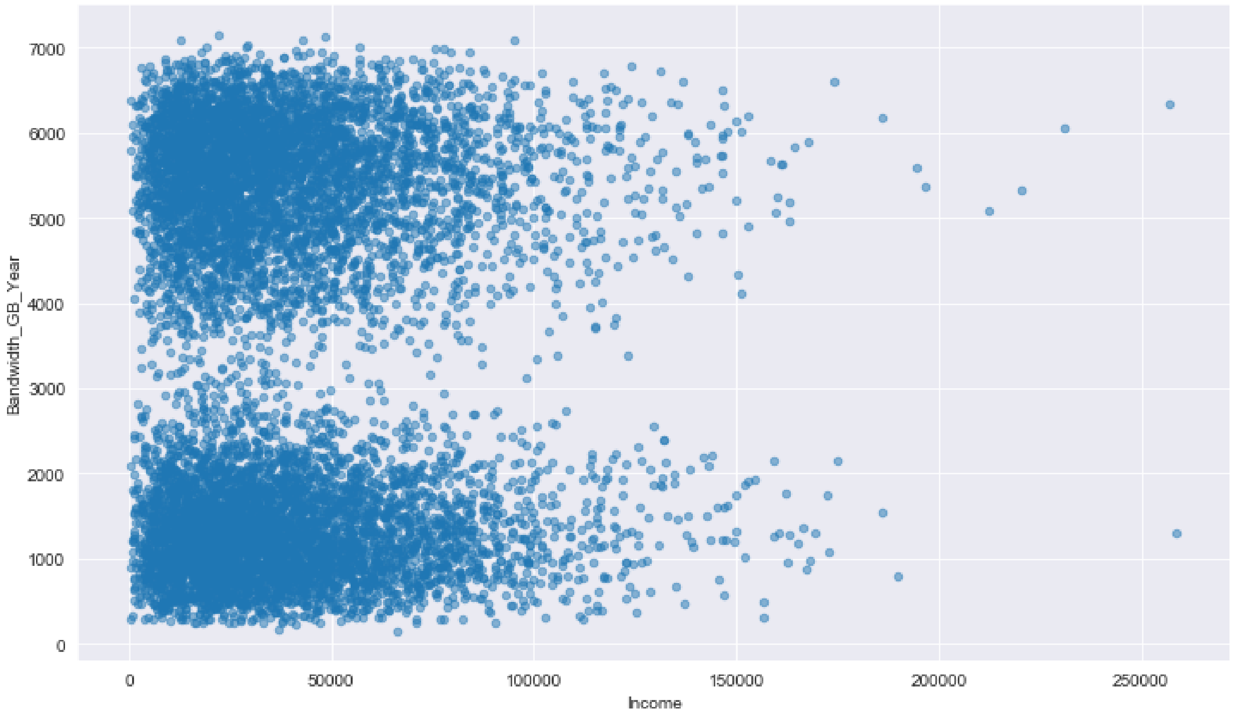
Gender_Female vs. Bandwidth_GB_Year



Gender_Nonbinary vs. Bandwidth_GB_Year



Income vs. Bandwidth_GB_Year



C5.

Please see the attached “D208_Data_set.csv”.

Part IV: Model Comparison and Analysis

D1.

We construct our multiple regression model by defining our independent / predictor variables and our dependent / target variable:

```
X = df_lm[['Age', 'Income',
'Tenure','Gender_Male','Gender_Female','Gender_Nonbinary']]
# X = independent variable(s) / predictor variable
X = sm.add_constant(X) # adds the constant coefficient /
intercept
y = target['Bandwidth_GB_Year']
# y = dependent variable / response / target variable

model = sm.OLS(y, X).fit()
predictions = model.predict(X)

model.summary()
```

The model summary returns the following output:

Dep. Variable:	Bandwidth_GB_ Year	**R-squared:**	0.984			
Model:	OLS	**Adj. R-squared:**	0.984			
Method:	Least Squares	**F-statistic:**	1.26E+05			
Date:	Wed, 14 Jul 2021	**Prob (F-statistic):**	0			
Time:	22:24:41	**Log-Likelihood:**	-70289			
No. Observations:	10000	**AIC:**	1.41E+05			
Df Residuals:	9994	**BIC:**	1.41E+05			
Df Model:	5					
Covariance Type:	nonrobust					
	coef	**std err**	**t**	**P>|t|**	**[0.025**	**0.975]**
const	538.0111	7.998	67.271	0	522.334	553.688
Age	-3.3443	0.132	-25.328	0	-3.603	-3.086
Income	0.0001	9.69E-05	1.498	0.134	-4.48E-05	0
Tenure	81.9977	0.103	793.378	0	81.795	82.2
Gender_Male	235.8771	5.91	39.914	0	224.293	247.461
Gender_Female	158.771	5.899	26.914	0	147.207	170.334
Gender_Nonbinary	143.3631	13.694	10.469	0	116.52	170.206
Omnibus:	419.543	**Durbin-Watson:**	1.969			
Prob(Omnibus):	0	**Jarque-Bera (JB):**	343.499			
Skew:	0.377	**Prob(JB):**	2.57E-75			
Kurtosis:	2.494	**Cond. No.**	8.61E+19			
Warnings:						
[1] Standard Errors assume that the covariance matrix of the errors is correctly specified.						
[2] The smallest eigenvalue is 3.21e-27. This might indicate that there are						
strong multicollinearity problems or that the design matrix is singular.						

D2.

With a *p* value of 0.134, 'Income' appears to be the most statistically insignificant variable in our model. 'Income' has a *p* value that is much greater than the statistically significant value of 0.05. It is safe to assume that 'Income' has little, if any, impact on the 'Bandwidth_GB_Year' variable. We also see that the standard error on the 'Gender_Nonbinary' and 'Income' variables are high, particularly on ‘Income’, indicating the observed values are further from the fitted regression line (Frost, 2017).

We can also note that in our initial model, the R-squared value is 0.984, meaning 98% of the variance can be explained using this model. This model also suggests that due to the large condition number, we could potentially be running into a strong multicollinearity or other issues.

Given this information, we will remove ‘Income’ and ‘Gender_Nonbinary’ from our independent variables and run the model again, with the goal of decreasing the condition number, eliminating a possibility of having multicollinearity issues, and, ideally, increasing the R-squared value.

D3.

We prepare our reduced regression model by creating a new model without the ‘Income’ and ‘Gender_Nonbinary’ independent variables:

```
X_red = df_lm[['Age', 'Tenure','Gender_Male','Gender_Female']]
# X = independent variable(s) / predictor variable
X_red = sm.add_constant(X_red) # adds the constant coefficient  /
intercept
y_red = target['Bandwidth_GB_Year']
# y = dependent variable / response / target variable

model_red = sm.OLS(y_red, X_red).fit()
predictions = model_red.predict(X_red)

model_red.summary()
```

Our reduced model summary gives us the following results:

Dep. Variable:	Bandwidth_GB_Year	**R-squared:**	0.984			
Model:	OLS	**Adj. R-squared:**	0.984			
Method:	Least Squares	**F-statistic:**	1.57E+05			
Date:	Wed, 14 Jul 2021	**Prob (F-statistic):**	0			
Time:	22:25:14	**Log-Likelihood:**	-70290			
No. Observations:	10000	**AIC:**	1.41E+05			
Df Residuals:	9995	**BIC:**	1.41E+05			
Df Model:	4					
Covariance Type:	nonrobust					
	coef	**std err**	**t**	**P>|t|**	**[0.025**	**0.975]**
const	687.3501	19.449	35.341	0	649.226	725.475
Age	-3.3451	0.132	-25.332	0	-3.604	-3.086
Tenure	81.998	0.103	793.333	0	81.795	82.201
Gender_Male	92.2451	18.413	5.01	0	56.151	128.339
Gender_Female	15.3309	18.39	0.834	0.404	-20.717	51.379
Omnibus:	421.009	**Durbin-Watson:**	1.969			
Prob(Omnibus):	0	**Jarque-Bera (JB):**	343.515			
Skew:	0.376	**Prob(JB):**	2.55E-75			
Kurtosis:	2.492	**Cond. No.**	777			
Warnings:						
[1] Standard Errors assume that the covariance matrix of the errors is correctly specified.						

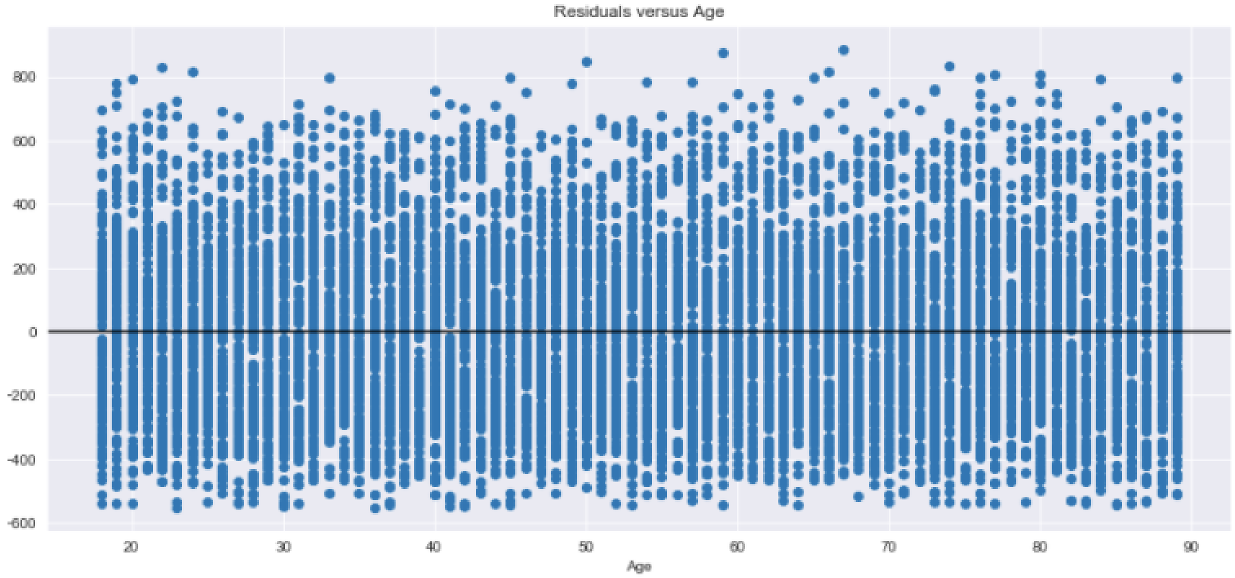
E1.

The reduced regression model looks better than our initial model, with the reduced model featuring a drastically reduced Condition number and has also seen the elimination of the potential for multicollinearity issues. The reduced model's R-squared value remains the same at 98%, however, still suggesting 98% of the variation can still be explained with this model.

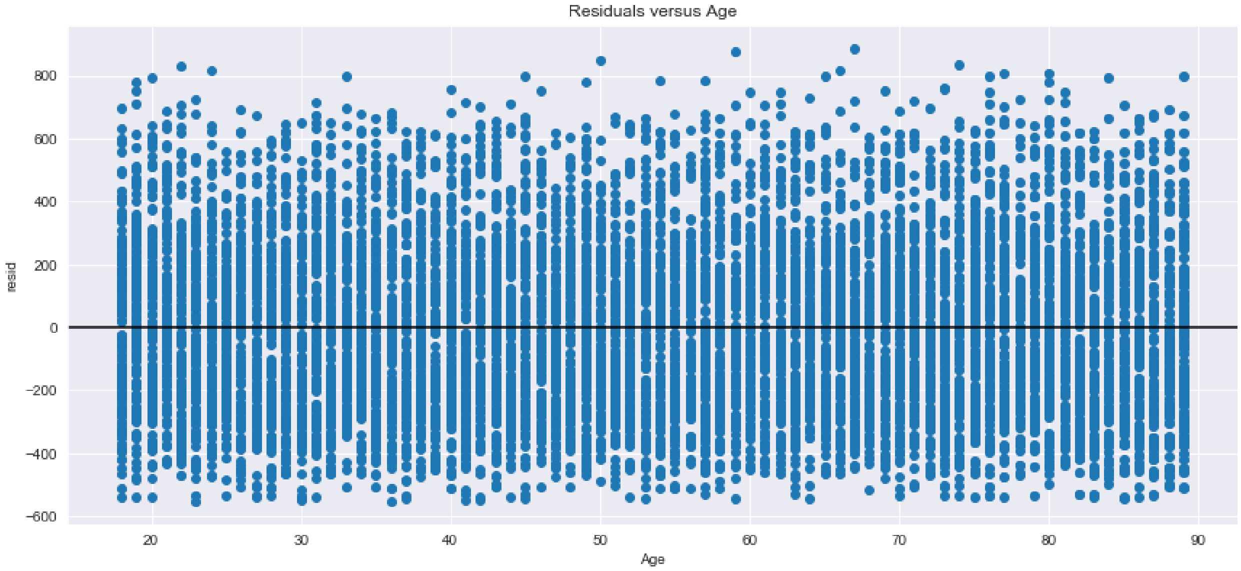
The decrease in the condition number and multicollinearity issue is due to the removal of the 'Income' and 'Gender_Nonbinary' variables, which were responsible for deteriorating the accuracy of the model. However, the R-squared value for the reduced model has not increased. Interestingly enough, in the reduced model, we see the *p* value for 'Gender_Female' increase, suggesting this variable is now also statistically insignificant. However, the removal of this independent variable would not benefit the model in a meaningful way in looking to predict customer bandwidth use based on demographics in this analysis.

We can see slight differences in the plotted residuals of our independent variables between the initial and reduced model as well:

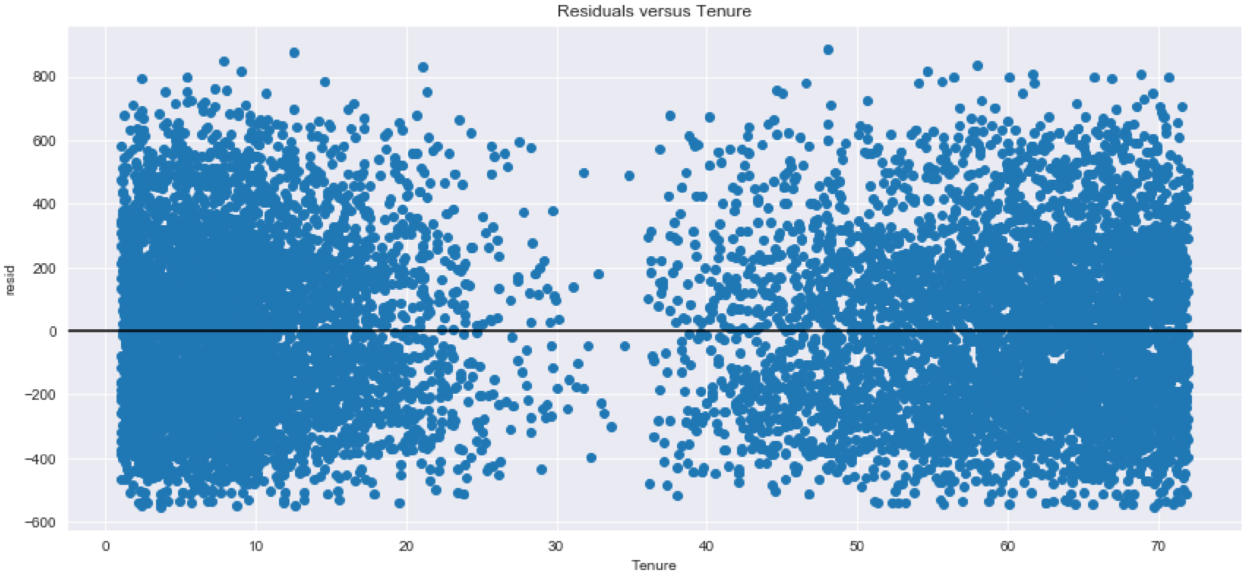
'Age' - Initial Model



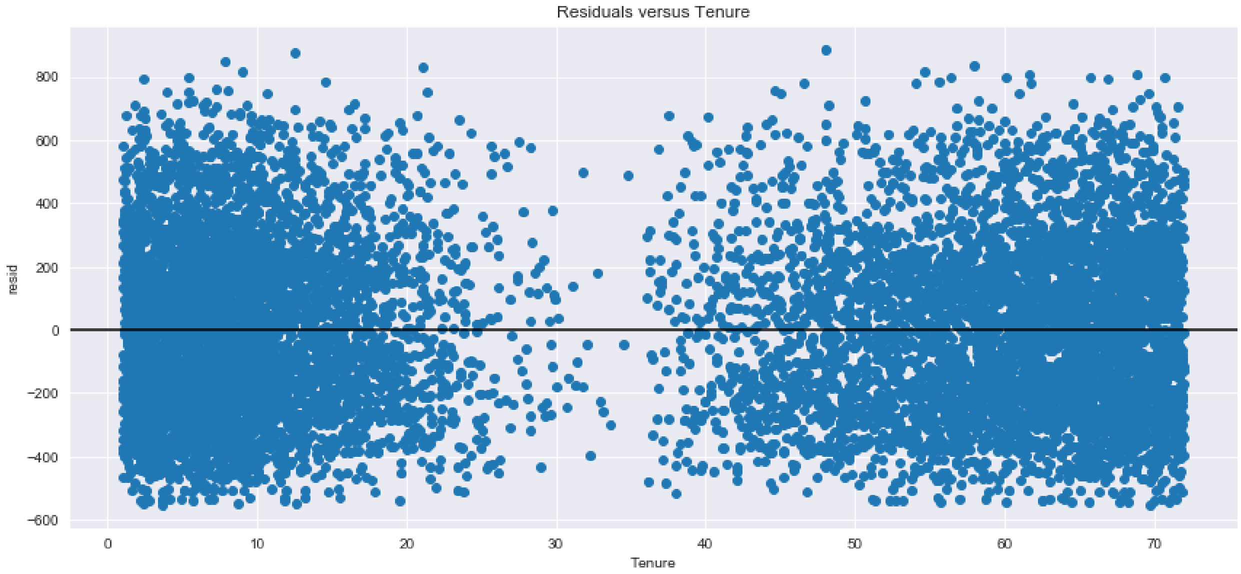
'Age' - Reduced Model



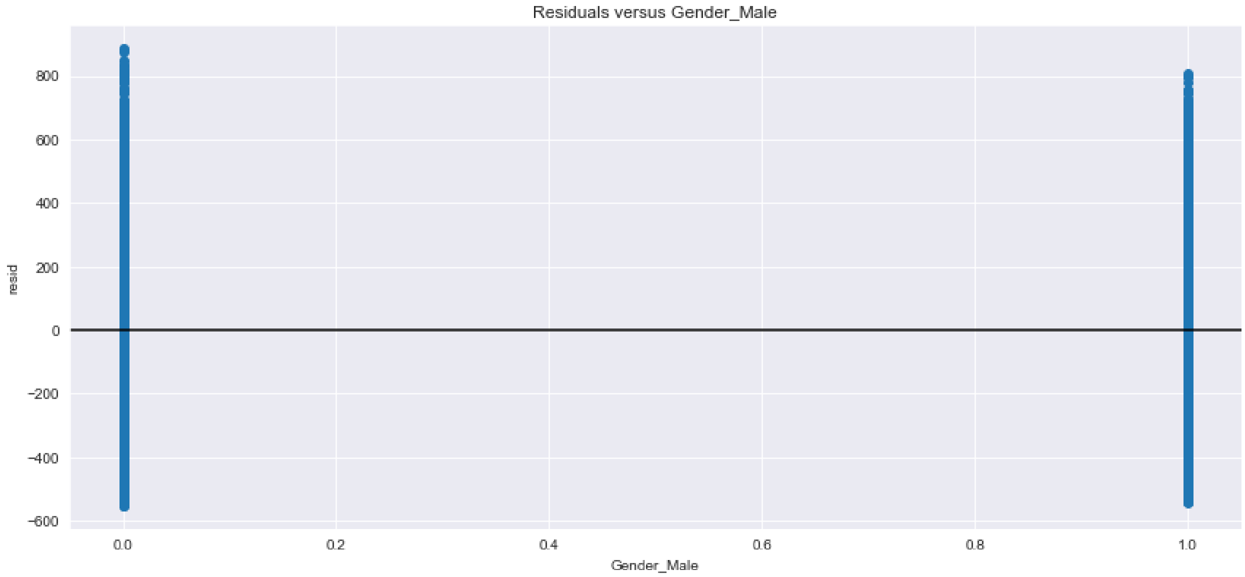
'Tenure' - Initial Model



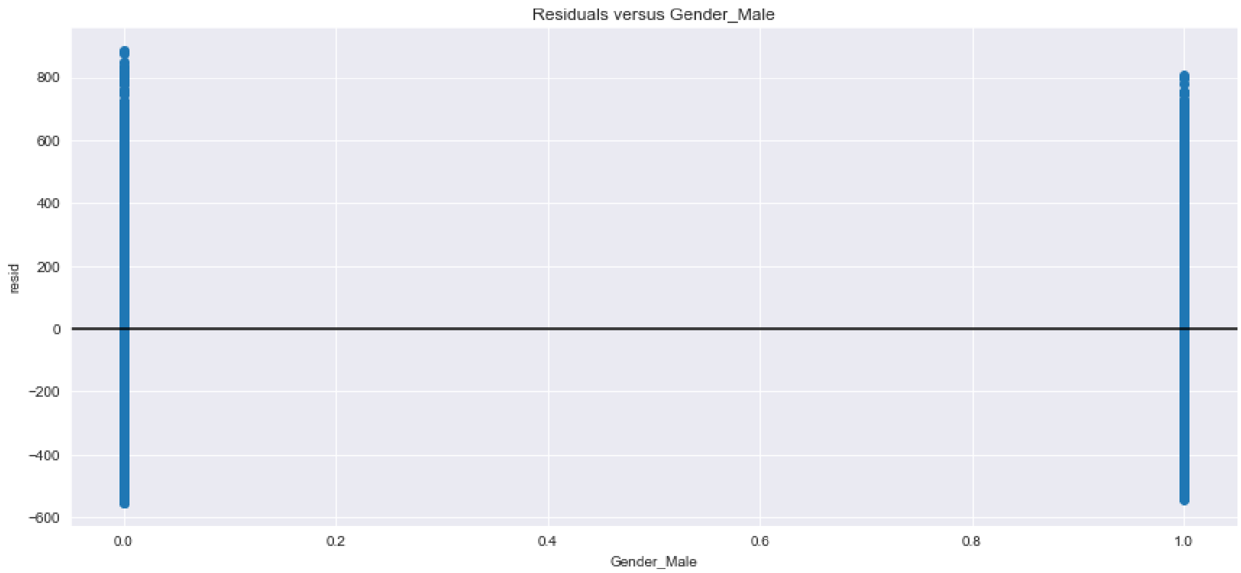
‘Tenure’ - Reduced Model



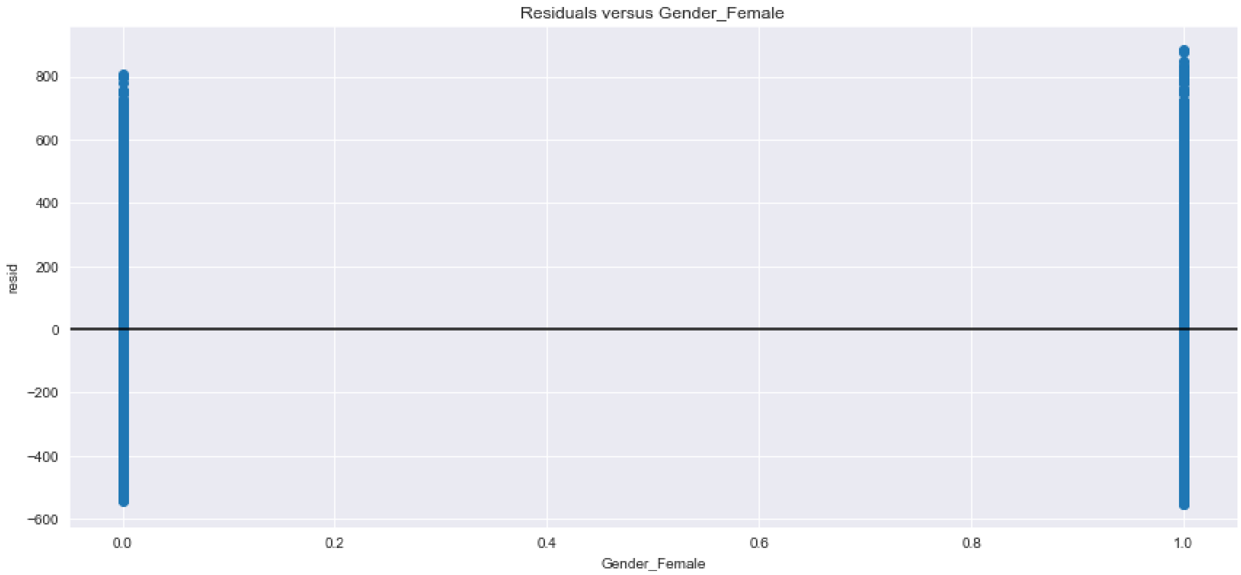
‘Gender_Male’ - Initial Model



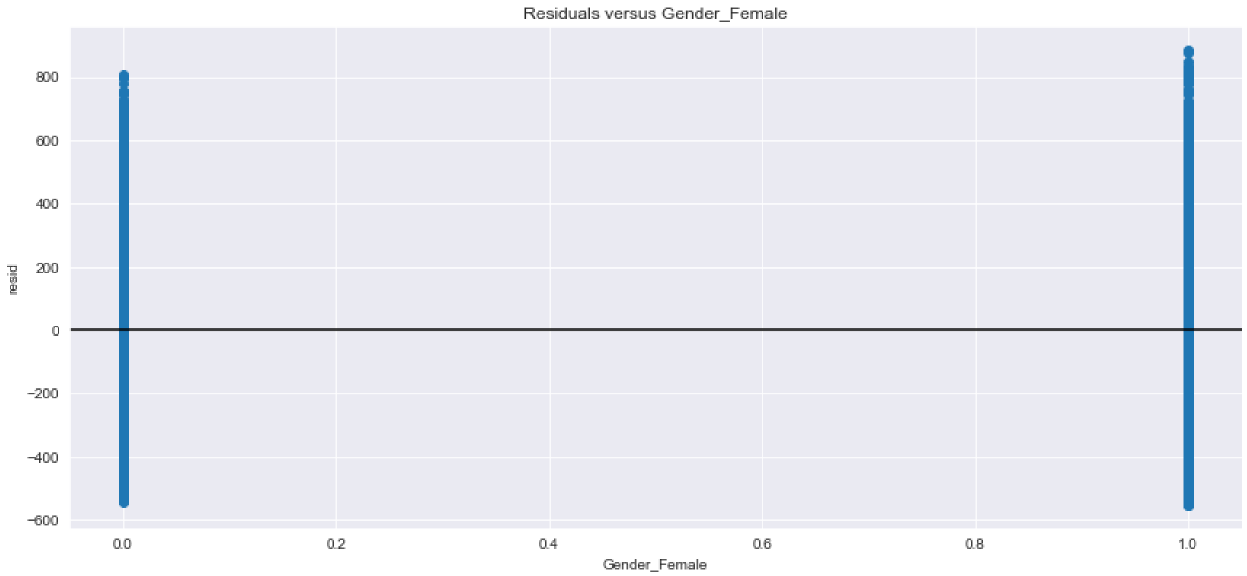
'Gender_Male' - Reduced Model



'Gender_Female' - Initial Model



'Gender_Female' - Reduced Model



E2.

To calculate the mean squared error of the reduced model's residuals, we use the statsmodel 'mse_resid' function:

```
#calculates the model's mean squared error of the residuals

model_red.mse_resid
74650.4499854058
```

The reduced model gives us the following predictions based on our dependent and independent variables:

- As a customer's age increases, their bandwidth usage per year will decline
- As a customer's tenure increases, their bandwidth usage per year will increase
- A male customer is likely to use more bandwidth than a female customer

E3.

Code:

```
# Initial Regression Model

X = df_lm[['Age', 'Income',
'Tenure','Gender_Male','Gender_Female','Gender_Nonbinary']] # X =
independent variable(s) / predictor variable
X = sm.add_constant(X) # adds the constant coefficient /
intercept
y = df_lm['Bandwidth_GB_Year'] # y = dependent variable /
response / target variable

model = sm.OLS(y, X).fit()
predictions = model.predict(X)

model.summary()


# INITIAL REGRESSION MODEL PLOTS

age_fig = plt.figure(figsize=(24,12))
age_fig = sm.graphics.plot_regress_exog(model, 'Age',
fig=age_fig);

income_fig = plt.figure(figsize=(24,12))
#produce regression plots
income_fig = sm.graphics.plot_regress_exog(model, 'Income',
fig=income_fig);


tenure_fig = plt.figure(figsize=(24,12))
#produce regression plots
tenure_fig = sm.graphics.plot_regress_exog(model, 'Tenure',
fig=tenure_fig);

male_fig = plt.figure(figsize=(24,12))
#produce regression plots
male_fig = sm.graphics.plot_regress_exog(model, 'Gender_Male',
fig=male_fig);
```

```
female_fig = plt.figure(figsize=(24,12))
#produce regression plots
female_fig = sm.graphics.plot_regress_exog(model,
'Gender_Female', fig=female_fig);

nb_fig = plt.figure(figsize=(24,12))
#produce regression plots
nb_fig = sm.graphics.plot_regress_exog(model, 'Gender_Nonbinary',
fig=nb_fig);


# Reduced Regression Model

X_red = df_lm[['Age','Tenure','Gender_Male','Gender_Female']] # X
= independent variable(s) / predictor variable
X_red = sm.add_constant(X_red) # adds the constant coefficient /
intercept
y_red = df_lm['Bandwidth_GB_Year'] # y = dependent variable /
response / target variable

model_red = sm.OLS(y_red, X_red).fit()
predictions = model_red.predict(X_red)

model_red.summary()


# REDUCED REGRESSION MODEL PLOTS

age_fig = plt.figure(figsize=(24,12))
age_fig = sm.graphics.plot_regress_exog(model_red, 'Age',
fig=age_fig);

tenure_fig = plt.figure(figsize=(24,12))
#produce regression plots
tenure_fig = sm.graphics.plot_regress_exog(model_red, 'Tenure',
fig=tenure_fig);

male_fig = plt.figure(figsize=(24,12))
#produce regression plots
male_fig = sm.graphics.plot_regress_exog(model_red,
'Gender_Male', fig=male_fig);

female_fig = plt.figure(figsize=(24,12))
#produce regression plots
```

```
female_fig = sm.graphics.plot_regress_exog(model_red,
'Gender_Female', fig=female_fig);
```

Part V: Data Summary and Implications

F1.

Regression Equation

$$\hat{Y}=b_0+b_1X_1+b_2X_2+...+b_pX_p ,$$

Bandwidth_GB_Year = 687.3 + -3.345 Age + 81.99 Tenure + 92.24 Gender_Male + 15.33 Gender_Female

The reduced multiple regression model, while far from perfect, does allow us to predict certain variables with some amount of confidence.

```
                              Coef

Age                           -3.345114
Tenure                        81.998043
Gender_Male                   92.245060
Gender_Female                 15.330932
```

In a regression model, the coefficients of each independent variable shows a positive or negative correlation with the dependent variable. A positive coefficient value tells us that as the independent variable increases, the mean of the dependent variable will tend to increase. A negative coefficient value indicates that as the independent variable decreases, the mean of the dependent variable will tend to decrease (Frost, 2021).

The amount by which the mean of the dependent variable is likely to change is indicated by the coefficient value. For every one unit change in the independent variable, assuming all other variables remain the same, the mean of the dependent variable will change by the coefficient value.

With this information in mind, if we look at our reduced model's coefficient values, we can assume the following:

- For every one unit increase in 'Age', assuming all other variables remain the same, a customer's mean annual bandwidth use will decrease by 3.34GB

- For every one unit increase in 'Tenure', assuming all other variables remain the same, a customer's mean annual bandwidth use will increase by 81.99GB

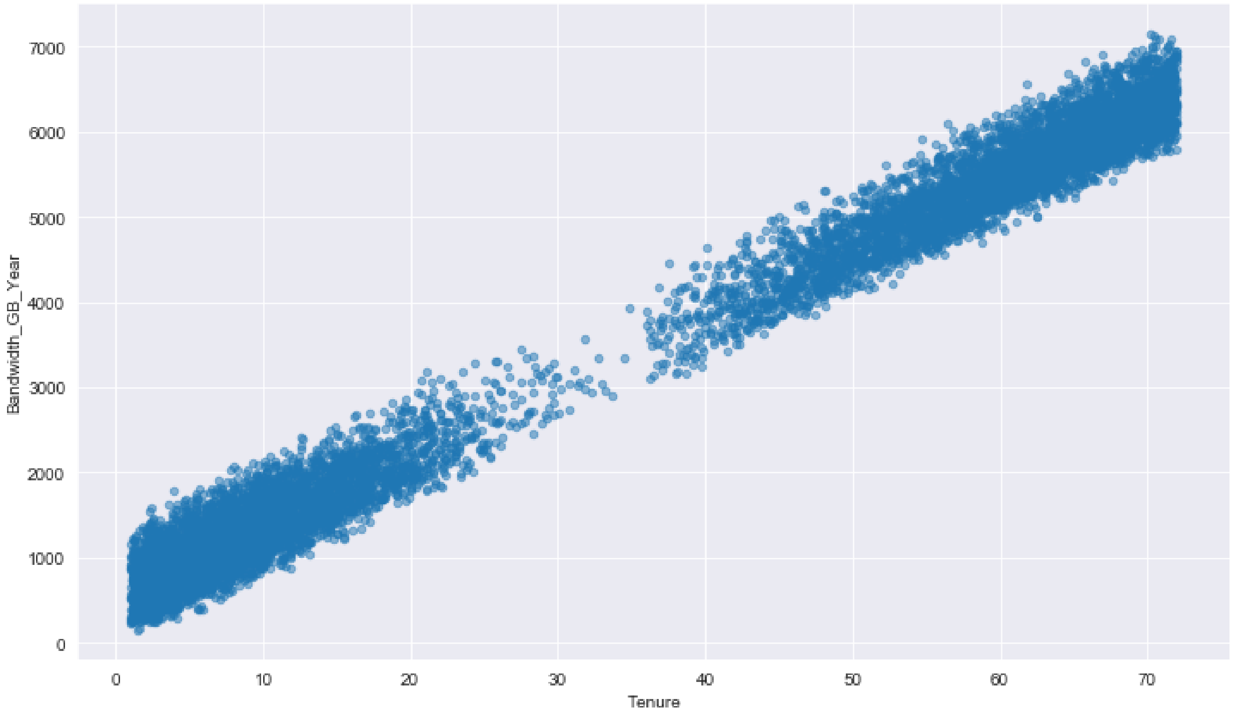
- For every one unit increase in 'Gender_Male' (we will assume this to be one new customer whose gender is identified as 'Male'), assuming all other variables remain the same, the customer's mean annual bandwidth use will increase by 92.24GB

- For every one unit increase in 'Gender_Female' (we will assume this to be one new customer whose gender is identified as 'Female'), assuming all other variables remain the same, the customer's mean annual bandwidth use will increase by 15.33GB

Statistically, the reduced model's R-Squared value is .984, which tells us that ~98% of the variance in the data can be explained using this model. While this R-squared value is high, we should keep in mind that our reduced model's residual error was quite high, with an error of 74650.449, which does not indicate a great fit. The R-Squared value, in addition to the residual error, suggests we can improve the model, but would need to expand it beyond the original research question posed at the start of this analysis. In the original model, 'Gender_Nonbinary' showed a statistically insignificant *p* value and was removed for the reduced model. However, in the reduced model, 'Gender_Female' showed an insignificant *p* value.

If we were purely exploring how to predict how much bandwidth a customer would use annually, and we were not interested in whether gender was a predicting factor, we would reduce the model further by eliminating the 'Gender_Female' variable from the model. As our research question was to see if gender was a determining factor in bandwidth use, it does not make sense to remove all gender variables. In the real world, we would likely go back to our research question and change our analysis approach, as it appears that gender is not a variable we are able to use, within this data set using multiple linear regression, to accurately predict how much bandwidth a customer will use in a year.

Another limitation of this analysis, as well as the data set, is the extremely correlated relationship between 'Tenure' and 'Bandwidth_GB_Year'.



In fact, the correlation between these two variables is so strong, that if ‘Tenure’ is removed from an even further reduced model, our R-squared value drops to 0 and the remainder of our independent variables’ *p* values become statistically insignificant:

Dep. Variable:	Bandwidth_GB_Year	**R-squared:**	0			
Model:	OLS	**Adj. R-squared:**	0			
Method:	Least Squares	**F-statistic:**	1.464			
Date:	Thu, 15 Jul 2021	**Prob (F-statistic):**	0.222			
Time:	8:58:27	**Log-Likelihood:**	-91082			
No. Observations:	10000	**AIC:**	1.82E+05			
Df Residuals:	9996	**BIC:**	1.82E+05			
Df Model:	3					
Covariance Type:	nonrobust					
	coef	**std err**	**t**	**P>|t|**	**[0.025**	**0.975]**
const	3275.775	153.345	21.362	0	2975.189	3576.361
Age	-1.5842	1.056	-1.5	0.134	-3.654	0.486
Gender_Male	216.3046	147.258	1.469	0.142	-72.351	504.96

Gender_Female	195.1058	147.065	1.327	0.185	-93.17	483.382
Omnibus:	43219.577	**Durbin-Watson:**	0.188			
Prob(Omnibus):	0	**Jarque-Bera (JB):**	1219.108			
Skew:	0.068	**Prob(JB):**	1.88E-265			
Kurtosis:	1.295	**Cond. No.**	657			
Warnings:						
[1] Standard Errors assume that the covariance matrix of the errors is correctly specified.						

This could tell us that ‘Tenure’ is almost entirely responsible for the majority of the variance in the model.

With this information, the analysis is not entirely practical given our research question. While we can make some predictions with confidence, such as the correlations between tenure and age with bandwidth use, we cannot confidently make the same predictions regarding customer gender. The model, while seemingly accurate, appears to only be accurate due to the inclusion of the ‘Tenure’ variable and its strong linear relationship with the dependent variable, leading to the high R-squared value.

F2.

With the information we have gathered from our initial model and reduced model, along with the exploration of the data, the best course of action we can recommend is to reframe our research question or approach the current question with a different model. A multiple linear regression model may not be the best approach with this particular data set. If we are committed to using a multiple linear regression model, it would likely benefit us to approach the research question with a less specific set of predictors and reduce the model until we can identify which variables have the largest impact on predicting the outcome of our chosen dependent variable.

Part VI: Demonstration

See attached Panopto video.

Sources

Assumptions of Multiple Linear Regression. (2021, April 29). Statistics Solutions. https://www.statisticssolutions.com/free-resources/directory-of-statistical-analyses/assumptions-of-multiple-linear-regression/

Advantages of Learning Python for Data Science. (n.d.). BSD MAG. Retrieved July 15, 2021, from https://bsdmag.org/advantages-of-learning-python-for-data-science/

Project Jupyter. (n.d.). Project Jupyter. Retrieved July 15, 2021, from https://jupyter.org/

Frost, J. (2017, May 5). *Standard error of the regression*. Statistics By Jim. https://statisticsbyjim.com/glossary/standard-error-regression/

How to Calculate Standardized Residuals in Python. (n.d.). Statology. Retrieved July 15, 2021, from https://www.statology.org/standardized-residuals-python/

Statsmodels - Linear Regression. (n.d.). Statsmodels. Retrieved July 15, 2021, from https://www.statsmodels.org/devel/generated/statsmodels.regression.linear_model.OLS.html

Frost, J. (2021, June 8). *How to Interpret P-values and Coefficients in Regression Analysis*. Statistics By Jim. https://statisticsbyjim.com/regression/interpret-coefficients-p-values-regression/