­

**Performance Assessment for D208: Predictive Analytics  
Task 1: Multiple Linear Regression**

Christopher Fischer - 011933891

College of Information Technology, Western Governors University

June 12, 2024

Performance Assessment for D206: Data Cleaning – Task 1

This document contains the tasks and outputs required for the “NBM3 TASK 1: LINEAR REGRESSION MODELING” assessment. All work is original to the author unless otherwise indicated by a citation.

# A - Research Question

Over the past year, the company’s costs for delivering reliable communications services have increased faster than revenue. Among the management team, an intuitive hypothesis is forming that a small segment of high-usage customers disproportionately contributes to the unexpected rise in operating costs.

With this motivation, the data analytics team has been given the budget and scope to perform a study to determine if it is possible to predict which customers are most likely to exhibit a high usage pattern. Armed with a useful predictive model, management can identify methods for adjusting pricing and packaging to extract more revenue from those customers, thereby offsetting our increased costs.

In summary, this effort aims to build a model that can identify how much bandwidth a given customer is likely to consume annually so that they can be directed to a pricing package that best matches their usage.

# B – Method Justification

Data analysts first turn to a modeling technique called multiple linear regression to predict values. With this method, an analyst can define a target numeric variable and a set of predictor variables that may explain the target variable. The regression algorithm then analyzes the data, developing a formula that estimates the relationship between the predictors and the target using a straight line. (Larose & Larose, 2019)

Linear regression is a tried-and-true workhorse for prediction studies. However, there are several critical assumptions that, if unmet, may render the study results unusable. These assumptions include:

1. **Linear relationship between the predictors and the target** – This means that the target variable should generally change value at a steady, proportional rate as the predictors change value.
2. **Independent observations** – This means that each data record used in the model is independent of all the other records.
3. **Independent predictors** – This means that the model's prediction variables are generally independent. In data science speak, this is called the absence of *multicollinearity*.
4. **Model variances are evenly distributed** – Also called *homoscedasticity*, this means that the differences between the actual target values and the values predicted by the model are evenly and randomly distributed around an average (mean) of zero. (Bobbitt, 2021)

Since this study aims to predict a target numeric value (annual bandwidth consumption) from a set of predictor variables, multiple linear regression is the ideal statistical method to employ.

The data science team will develop the code for this study using Python. In its nearly 35 years, Python has become one of the most widely used general-purpose programming languages. Python is backed by a massive community of developers and a vast collection of libraries that extend the core capabilities of the language. (Datacamp, 2022) Another reason for selecting Python is that, while R is an excellent choice for interactive studies, Python is better suited for deployment on production servers as part of a data pipeline. (WGU Information Technology, n.d.)

# C – Data Preparation

The following sections outline the team’s plan to clean and prepare the input data for use in predictive modeling.

## C1 – Data Cleaning

Although the source data provided to the team for this study is labeled “clean,” it behooves us to ensure the quality of the data before beginning the predictive modeling activity. Poor data will negatively impact the model's accuracy, making it unusable for decision-making.

The cleaning steps which will be performed in this study include:

* Identify and treat missing data
* Identify and treat outlying values
* Identify and treat any invalid categorical values

### Cleaning code and results

# Check missing data

msno.matrix(df, fontsize = 12, labels=True)

plt.title('Missing Data Matrix')

plt.show()

A black and white striped pattern

Description automatically generated with medium confidence

No variables exhibit missing data.

# Check survey question validity

print(df[survey\_variables][~df[survey\_variables].isin(survey\_answers)].count())

A black screen with white text

Description automatically generated

Survey response variables all contain valid values.

#Check values in yes/no variables

print(df[yes\_no\_variables][~df[yes\_no\_variables].isin(['Yes','No'])].count())

A screen shot of a computer

Description automatically generated

Boolean variables all contain valid values.

#Detect potential outliers

df\_z = (df[numerical\_variables] - df[numerical\_variables].mean())/df[numerical\_variables].std(ddof=0)

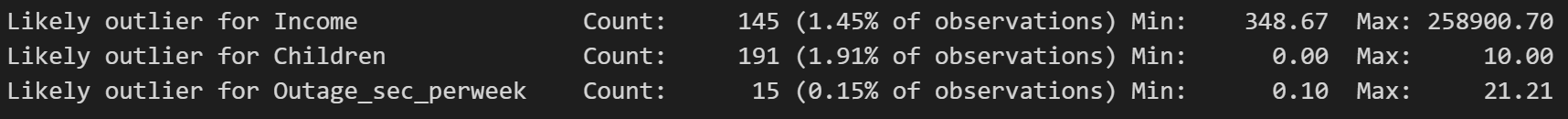
outlier\_cols = df\_z.loc[: , (df\_z > 3.0).any()].columns

for col in outlier\_cols :

    cnt = len(df\_z[df\_z[col]>3])

    min, max = df[col].min(), df[col].max()

    print('Likely outlier for {0:<20}\t Count: {1:7d} ({2:5.2%} of observations)\tMin: {3:>9.2f}\tMax: {4:>9.2f}'.format(col,cnt,cnt/10000,min,max))



While a small percentage of values exceed the norms for the variables *Income*, *Children*, and *Outage\_sec\_perweek*, none is definitively invalid (e.g., a negative number of seconds or 100 children). Therefore, all values will be retained in the study.

## C2 – Summary Statistics for Target and Predictor Variables

This section provides a brief exploration of the predictor variables to be included in the study, as well as the target variable. Please note that while the table includes all the variables that will be included in the initial model, the final model will likely include fewer variables once tuning is complete.

| Variable | Statistics | Comment |
| --- | --- | --- |
| Target | | |
| Bandwidth\_GB\_Year | count 10000.000000  mean 3392.341550  std 2185.294852  min 155.506715  25% 1236.470827  50% 3279.536903  75% 5586.141370  max 7158.981530 | With a range from 155 GB – 7159 GB and an average (mean) of 3392 GB, the hypothesis put forth by management seems to be supported. |
| Predictors - Numeric | | |
| Age | count 10000.000000  mean 53.078400  std 20.698882  min 18.000000  25% 35.000000  50% 53.000000  75% 71.000000  max 89.000000 | Our customers skew older than expected. Hypothesis: younger customers use more bandwidth. |
| Income | count 10000.000000  mean 39806.926771  std 28199.916702  min 348.670000  25% 19224.717500  50% 33170.605000  75% 53246.170000  max 258900.700000 | Our customers skew toward lower income levels. Hypothesis: higher-income customers use more bandwidth. |
| Children | count 10000.0000  mean 2.0877  std 2.1472  min 0.0000  25% 0.0000  50% 1.0000  75% 3.0000  max 10.0000 | Our customers tend to have small to average-sized families. Hypothesis: having more children predicts more bandwidth consumed. |
| Tenure | count 10000.000000  mean 34.526188  std 26.443063  min 1.000259  25% 7.917694  50% 35.430507  75% 61.479795  max 71.999280 | Our customers' average tenure is close to three years. |
| Outage\_sec\_perweek | count 10000.000000  mean 10.001848  std 2.976019  min 0.099747  25% 8.018214  50% 10.018560  75% 11.969485  max 21.207230 | Our service is very reliable, with an average outage of ten seconds per week. |
| MonthlyCharge | count 10000.000000  mean 172.624816  std 42.943094  min 79.978860  25% 139.979239  50% 167.484700  75% 200.734725  max 290.160419 | Our average monthly bill is relatively high. Hypothesis: customers who pay more use more bandwidth. |
| Predictors – Categorical | | |
| Marital | Divorced 2092  Widowed 2027  Separated 2014  Never Married 1956  Married 1911  Mode: Divorced | Note: The “Never Married” was changed to “NeverMarried” to prevent an error in a later data transformation step. |
| Gender | Female 5025  Male 4744  Nonbinary 231  Mode: Female | Slightly more females than males. |
| Techie | No 8321  Yes 1679  Mode: No | Hypothesis: self-declared “techies” use more bandwidth. |
| Port\_modem | No 5166  Yes 4834  Mode: No | About half of our customers have a portable modem device. Hypothesis: having a portable modem enables higher usage. |
| Tablet | No 7009  Yes 2991  Mode: No | About 30% of our customers report having a tablet device. Hypothesis: having more devices predicts higher bandwidth consumption |
| Phone | Yes 9067  No 933  Mode: Yes | (same) |
| Multiple | No 5392  Yes 4608  Mode: No | (same) |
| OnlineSecurity | No 6424  Yes 3576  Mode: No | Hypothesis: the more services customers subscribe to, the higher their bandwidth consumption. |
| OnlineBackup | No 5494  Yes 4506  Mode: No | (same) |
| DeviceProtection | No 5614  Yes 4386  Mode: No | (same) |
| TechSupport | No 6250  Yes 3750  Mode: No | (same) |
| StreamingTV | No 5071  Yes 4929  Mode: No | Hypothesis: streaming is likely the biggest consumer of bandwidth |
| StreamingMovies | No 5110  Yes 4890  Mode: No | (same) |
| Item1 | 3 3448  4 3358  2 1393  5 1359  1 224  6 199  7 19  Mode: 3 | The survey questions have to do with customer satisfaction. Hypothesis: survey results do not predict higher bandwidth usage. |
| Item2 | 3 3415  4 3412  5 1368  2 1360  1 217  6 215  7 13  Mode: 3 | (same) |
| Item3 | 3 3435  4 3410  2 1424  5 1313  6 203  1 202  7 12  8 1  Mode: 3 | (same) |
| Item4 | 4 3452  3 3430  2 1350  5 1335  1 221  6 203  7 9  Mode: 4 | (same) |
| Item5 | 3 3462  4 3417  2 1378  5 1321  1 206  6 204  7 12  Mode: 3 | (same) |
| Item6 | 3 3445  4 3333  2 1427  5 1382  6 210  1 190  7 12  8 1  Mode: 3 | (same) |
| Item7 | 4 3456  3 3446  5 1335  2 1309  6 224  1 219  7 11  Mode: 4 | (same) |
| Item8 | 3 3461  4 3400  2 1378  5 1335  1 206  6 205  7 14  8 1  Mode: 3 | (same) |

## C3 – Univariate and Bivariate Visualizations

The following graphs visualize the univariate and bivariate statistics for each study variable.

### Target Variable

A graph of a number of blue bars

Description automatically generated

### Predictor Numeric Variables

A group of blue bars

Description automatically generated

### Predictor Categorical Variables

|  |  |  |
| --- | --- | --- |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

### Numeric Predictors versus Target

|  |  |  |
| --- | --- | --- |
|  |  |  |
|  |  |  |

Only Tenure shows a strong correlation to the target variable of these predictors. That is a discouraging start.

### Categorical Predictors versus Target

|  |  |  |
| --- | --- | --- |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

By looking at the mean lines and the shape of these box plots, the following predictors show some correlation to the target variable: *Gender, Techie, OnlineBackup, DeviceProtection, StreamingTV,* and several of the survey responses (*Item1 – Item8)*. Nevertheless, further assessment will occur during the modeling phase of this study.

## C4 – Data Transformation

The approach taken in this study aims to create an initial model that includes as many of the candidate predictor variables as possible. Subsequently, the model will be reduced to eliminate variables that do not contribute materially to the model’s accuracy.

This requires the team to transform certain variables to permit their inclusion in the model. The transformations planned for this study include:

* **Re-express *Gender* and *Marital* categorical variables using one-hot encoding**  
  As multiple linear regression works only on variables with numeric values, categorical variables with textual values must be re-expressed into a numeric form. One-hot encoding involves creating a new variable for each category's values. For example, the variable *Gender* has three values: Male, Female, and Non-binary. Using one-hot encoding, new variables will be created for two values, *Gender\_Male* and *Gender\_Nonbinary*, each containing a zero or one to indicate which gender applies to the observation. Notice that only two new variables will be created, not three. This is known as the *k-1 rule,* which stipulates that one-hot encoding must always create one less new variable than there are category values. This is done to prevent multi-collinearity. (Mahto, 2019)
* **Re-express any Boolean (yes/no) variables as numeric**This is a more straightforward implementation of the previous transformation. To be used in the regression model, these yes/no values must be converted to one and zero, respectively.
* **Re-express the survey answer variables to an ascending scale**  
  One oddity of our data is that one is the best value, and eight is the worst value for the eight survey variables. This scale will be inverted to eliminate any confusion caused by this apparent reversal, making one the lowest and eight the highest.
* **Variables exhibiting high multicollinearity will be removed from consideration before producing the initial model**  
  Multicollinearity occurs when two or more predictor variables are strongly correlated. If such variables were included in a linear regression model, it would be impossible for the model to determine which predictors were responsible for a change in the target variable. To detect problematic variables, the *Variance Inflation Factor* (VIF) will be calculated for each variable. The variable with the highest VIF over a threshold of 5.0 will be removed from consideration. This process will be repeated until variables with a VIF > 5.0 remain in the set to be modeled. (Frost, 2017)

### Transformation code and results

# Reexpress yes/no columns as numbers [In-Text Citation:(Eiler, 2017)]

yesno\_dict = {'No': 0, 'Yes': 1}

for col in yes\_no\_variables:

    df[col] = df[col].map(yesno\_dict)

df[yes\_no\_variables].info()

A screen shot of a computer

Description automatically generated

# Reexpress survey responses so that 1 is worst and eight is best

survey\_dict = {1: 8, 2: 7, 3: 6, 4: 5, 5: 4, 6: 3, 7: 2, 8: 1}

for col in survey\_variables:

    df[col] = df[col].map(survey\_dict)

# One-hot encoding

for feature in ['Gender','Marital'] :

    dummies = pd.get\_dummies(df[feature], drop\_first=True, prefix=feature, dtype=np.int64)

    print(dummies.info())

    X\_full.remove(feature)

    for newcol in dummies.columns :

        X\_full.append(newcol)

    df = pd.concat([df, dummies], axis=1).drop(feature, axis=1)

A screenshot of a computer screen

Description automatically generated

# Check for feature colliniarity and drop high VIFs [In-Text Citation: (Prashant, 2016)]

from statsmodels.stats.outliers\_influence import variance\_inflation\_factor

def calculate\_vif(X, thresh=5.0):

    X = X.assign(const=1)

    variables = list(range(X.shape[1]))

    dropped = True

    while dropped:

        dropped = False

        vif = [variance\_inflation\_factor(X.iloc[:, variables].values, ix)

               for ix in range(X.iloc[:, variables].shape[1])]

        vif = vif[:-1]  # always preserve the constant

        maxvif = np.max(vif)

        maxloc = vif.index(maxvif)

        if maxvif > thresh:

            print(f'dropping {X.iloc[:, variables].columns[maxloc]:30} at index: {str(maxloc)} with VIF: {maxvif:.5f}')

            del variables[maxloc]

            dropped = True

    return X.iloc[:, variables[:-1]], X.columns[variables[:-1]]

tmp\_df, X\_full = calculate\_vif(df[X\_full],5.0)

# Merge post VIF reduction df with the target variable from the df

df = pd.concat([tmp\_df, df[y]], axis=1)



## C5 – Prepared Data File

This file contains the cleaned and prepared data used in subsequent modeling activities.



# D – Initial and Reduced Model

The initial model built for this study included the following predictor variables:

* Age
* Income
* Children
* Tenure
* Outage\_sec\_perweek
* Techie
* Port\_modem
* Tablet
* Phone
* Multiple
* OnlineSecurity
* OnlineBackup
* DeviceProtection
* TechSupport
* StreamingTV
* StreamingMovies
* Item1
* Item2
* Item3
* Item4
* Item5
* Item6
* Item7
* Item8
* Gender\_Male
* Gender\_Nonbinary
* Marital\_Married
* Marital\_NeverMarried
* Marital\_Separated
* Marital\_Widowed

Here is the summary of the initial model:

OLS Regression Results

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

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

Model: OLS Adj. R-squared: 0.992

Method: Least Squares F-statistic: 3.991e+04

Date: Wed, 12 Jun 2024 Prob (F-statistic): 0.00

Time: 15:25:23 Log-Likelihood: -67100.

No. Observations: 10000 AIC: 1.343e+05

Df Residuals: 9969 BIC: 1.345e+05

Df Model: 30

Covariance Type: nonrobust

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

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

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

Intercept 332.6146 29.982 11.094 0.000 273.843 391.386

Age -3.3226 0.096 -34.520 0.000 -3.511 -3.134

Income 0.0001 7.06e-05 1.920 0.055 -2.83e-06 0.000

Children 30.5602 0.927 32.953 0.000 28.742 32.378

Tenure 82.0249 0.075 1088.831 0.000 81.877 82.173

Outage\_sec\_perweek -0.8643 0.670 -1.291 0.197 -2.177 0.448

Techie 2.8616 5.328 0.537 0.591 -7.582 13.305

Port\_modem -2.2737 3.982 -0.571 0.568 -10.079 5.532

Tablet 0.1809 4.353 0.042 0.967 -8.351 8.713

Phone -2.5243 6.853 -0.368 0.713 -15.957 10.908

Multiple 76.6077 3.996 19.172 0.000 68.775 84.440

OnlineSecurity 76.0937 4.154 18.317 0.000 67.951 84.237

OnlineBackup 94.0603 4.002 23.502 0.000 86.215 101.906

DeviceProtection 81.2153 4.019 20.209 0.000 73.338 89.093

TechSupport 14.3095 4.114 3.478 0.001 6.246 22.373

StreamingTV 229.3275 3.982 57.586 0.000 221.521 237.134

StreamingMovies 210.3298 3.983 52.805 0.000 202.522 218.138

Item1 6.6838 2.851 2.344 0.019 1.095 12.273

Item2 -4.0386 2.672 -1.511 0.131 -9.277 1.200

Item3 0.7451 2.450 0.304 0.761 -4.057 5.547

Item4 -0.8003 2.193 -0.365 0.715 -5.099 3.498

Item5 -3.9641 2.276 -1.742 0.082 -8.426 0.498

Item6 -0.8716 2.343 -0.372 0.710 -5.464 3.721

Item7 -0.6959 2.216 -0.314 0.754 -5.040 3.648

Item8 -4.3899 2.109 -2.081 0.037 -8.525 -0.255

Gender\_Male 68.5948 4.035 16.999 0.000 60.685 76.505

Gender\_Nonbinary -23.6752 13.397 -1.767 0.077 -49.936 2.586

Marital\_Married -4.8897 6.301 -0.776 0.438 -17.240 7.461

Marital\_NeverMarried -8.1664 6.261 -1.304 0.192 -20.440 4.107

Marital\_Separated -10.1278 6.213 -1.630 0.103 -22.307 2.052

Marital\_Widowed -12.5434 6.210 -2.020 0.043 -24.716 -0.371

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

Omnibus: 59185.340 Durbin-Watson: 1.976

Prob(Omnibus): 0.000 Jarque-Bera (JB): 1611.204

Skew: 0.626 Prob(JB): 0.00

Kurtosis: 1.484 Cond. No. 7.39e+05

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

Notes:

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

[2] The condition number is large, 7.39e+05. This might indicate that there are

strong multicollinearity or other numerical problems.

The initial model exhibits substantial accuracy, as measured by R2 and Adjusted R2. However, note 2 (highlighted in yellow) raises concerns about the model's reliability. Furthermore, there are several variables with high p-values. It would be best to remove these variables to improve computational efficiency and reduce the chance of model overfitting. (Tripathi, 2019)

To improve the initial model, backward feature elimination will be performed to remove the least significant features from the initial model. Backward elimination, one of the wrapper methods of feature selection, is an iterative process where the feature with the highest p-value above a given alpha threshold is removed, and the model is re-fitted. These steps are repeated until no further improvement in the model can be made. (Middleton, 2023) This feature selection method was selected because it is quite easy to explain to stakeholders.

For this study, the alpha threshold selected was 0.05. This resulted in 19 features being eliminated. After backward elimination was performed, a reduced model was built that included the following predictor variables:

* Children
* Tenure
* Multiple
* OnlineSecurity
* OnlineBackup
* DeviceProtection
* TechSupport
* StreamingTV
* StreamingMovies
* Gender\_Male

Here is the summary of the reduced model:

OLS Regression Results

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

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

Model: OLS Adj. R-squared: 0.992

Method: Least Squares F-statistic: 1.088e+05

Date: Wed, 12 Jun 2024 Prob (F-statistic): 0.00

Time: 16:49:55 Log-Likelihood: -67113.

No. Observations: 10000 AIC: 1.343e+05

Df Residuals: 9988 BIC: 1.343e+05

Df Model: 11

Covariance Type: nonrobust

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

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

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

Intercept 277.8850 8.093 34.336 0.000 262.021 293.749

Age -3.3120 0.096 -34.424 0.000 -3.501 -3.123

Children 30.5308 0.927 32.925 0.000 28.713 32.349

Tenure 82.0274 0.075 1089.274 0.000 81.880 82.175

Multiple 76.1885 3.993 19.082 0.000 68.362 84.015

OnlineSecurity 75.9920 4.153 18.300 0.000 67.852 84.132

OnlineBackup 93.9543 4.000 23.489 0.000 86.113 101.795

DeviceProtection 81.4044 4.013 20.284 0.000 73.538 89.271

TechSupport 14.1885 4.111 3.451 0.001 6.130 22.247

StreamingTV 229.1057 3.980 57.562 0.000 221.304 236.908

StreamingMovies 210.4190 3.982 52.845 0.000 202.614 218.224

Gender\_Male 69.4716 3.987 17.426 0.000 61.657 77.286

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

Omnibus: 58332.768 Durbin-Watson: 1.977

Prob(Omnibus): 0.000 Jarque-Bera (JB): 1620.950

Skew: 0.627 Prob(JB): 0.00

Kurtosis: 1.478 Cond. No. 288.

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

Notes:

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

The reduced model also exhibits strong accuracy, as measured by R2 and Adjusted R2. Additionally, the warning observed on the initial model’s summary has been resolved.

# E – Model Evaluation

It is important to assess the changes to the model’s performance after feature selection has been performed. For this study, the results are as follows:

╒═════════╤═════════╤══════════╤═════════════╤═══════════╤══════════════╕

│ **Model │ r2 │ r2-adj │ mse │ rse │ resid\_mean** │

╞═════════╪═════════╪══════════╪═════════════╪═══════════╪══════════════╡

│ Full │ 0.99174 │ 0.99172 │ 39549.12873 │ 198.86963 │ 0.00000 │

│ Reduced │ 0.99172 │ 0.99171 │ 39577.14192 │ 198.94005 │ 0.00000 │

╘═════════╧═════════╧══════════╧═════════════╧═══════════╧══════════════╛

By all measures, the reduced model is almost identical in performance to the initial model despite containing 19 fewer predictor variables.

To ensure that the key assumption of *homoscedasticity* is met (see [B – Method Justification](#_B_–_Method) for more information about the assumptions of multiple linear regression), the model’s residuals were plotted against its fitted values, including a regression line.

A graph with blue lines and a red line

Description automatically generated

We can see that the residuals are well distributed around a mean value of 0.0. One unusual observation is that the positive residuals tend to be greater than the negative residuals, yet the mean is still zero. The results of a deeper examination of the residuals can explain this. Approximately 65% of the residuals have a value less than zero, while just 35% have a value greater than zero. This accounts for the mean being 0.0 despite the difference in degree.

A black background with white text

Description automatically generated

Here is a copy of the actual code used to perform the study.



# F – Summary and Implications

## F1 – Analysis and Discussion

The multiple linear regression of the reduced model can be expressed in the following equation:

(Middleton, Getting Started with D208 - Part 2, 2023)

This regression equation can be understood more simply by saying that, keeping all things constant:

* A one-unit increase in *Age* is associated with a 3.3 GB increase in annual bandwidth consumption
* A one-unit increase in *Children* is associated with a 30.5 GB increase in annual bandwidth consumption
* A one-unit increase in *Tenure* is associated with an 82.0 GB increase in annual bandwidth consumption
* A customer subscribing to *Multiple* phone lines is associated with a 76.2 GB increase in annual bandwidth consumption
* A customer subscribing to *OnlineSecurity* is associated with a 76.0 GB increase in annual bandwidth consumption
* A customer subscribing to *OnlineBackup* is associated with a 94.0 GB increase in annual bandwidth consumption
* A customer subscribing to *DeviceProtection* is associated with a 81.4 GB increase in annual bandwidth consumption
* A customer subscribing to *TechSupport* is associated with a 14.2 GB increase in annual bandwidth consumption
* A customer subscribing to *StreamingTV* is associated with a 229.1 GB increase in annual bandwidth consumption
* A customer subscribing to *StreamingMovies* is associated with a 210.4 GB increase in annual bandwidth consumption
* A customer who self-identifies as *Male* is associated with a 69.5 GB increase in annual bandwidth consumption

(Middleton, Getting Started with D208 - Part 2, 2023)

The results documented in [E – Model Evaluation](#_E_–_Model) clearly demonstrate the statistical significance of the reduced model. The model yields extremely high R2 and Adjusted R2 values, very low p-value, and low residual standard error. These are all helpful since a model with poor performance is useless in making practical applications.

Looking at the predictor variables that remained in the reduced model after feature selection, signs can be seen that indicate the practical significance of the model. Seven of the 11 predictors are categorical variables indicating whether a customer subscribes to one of our additional service offerings. As each service offering consumes bandwidth as it is used, it is entirely sensible that customers who subscribe to more services use more bandwidth each year. It is also no surprise that video streaming services for television and movies are the two services associated with the most significant bandwidth consumption increases.

The model also supports several hypotheses documented in [C2 – Summary Statistics for Target and Predictor Variables](#_C2_–_Summary). Namely:

* Older customers are associated with somewhat lower bandwidth consumption
* Larger families are associated with somewhat higher bandwidth consumption
* Subscribing to multiple services is associated with higher bandwidth consumption
* Identifying as male is associated with somewhat higher bandwidth consumption
* Subscribing to streaming video services is associated with much higher bandwidth consumption

Nevertheless, no model is perfect, and no modeling technique is perfect. There are implications to every decision made in a study. In this study, there are several important implications:

* **Outliers were retained**.The suspected outlier values of three numeric variables (Income, Children, Outage\_sec\_perweek) were retained. While only one of the three remained in the reduced model, keeping outliers impacted the leverage of the variables. If outliers had been removed, it is possible that the outcome of feature selection would have differed.
* **Survey responses were not transformed**. The survey response variable values were inverted, so one was low, and eight was high. A different approach that could have influenced their contribution to the model would be to have reduced their values to Low, Medium, and High and then perform one-hot encoding on the resulting variables.
* **VIF removal was performed before the initial model.** The team used VIF to filter out variables with high multicollinearity before the initial model was created. An alternative approach would have been to do so as part of the feature selection process. Doing so may have resulted in a different subset of variables retained in the reduced model.
* **Backward Feature Elimination was chosen for feature selection**. While wrapper methods are more computationally intensive than other feature selection methods, the impact of this limitation on our study is quite small because there are relatively few observations in the population. As an alternative, several feature selection methods could have been employed to compare the results and identify which method yielded the best model. (Oleszak, 2023)

## F2 – Recommendations

The research question posed for this survey indicates that management is concerned that some customers are using more bandwidth than others, costing the company more by doing so. The linear regression model produced to address this concern can help the marketing team identify services that should be considered for price adjustment. Specifically, attention should be paid to our Streaming TV and Movies offerings. Subscribing to either of these services is associated with much higher annual bandwidth consumption.

There is room for further study to help identify additional price adjustment opportunities. For example, customers subscribing to both streaming services could be fed into a linear regression. Similarly, a count of subscribed services could be added as a predictor.

Lastly, additional studies should be used to compare customer reactions to several proposed pricing strategies. This could take the form of A/B testing or a logistic regression in which the likelihood of customer churn when faced with a price increase is evaluated.

# G – Recorded Code Review

A recording of the code review presentation was uploaded with this submission. For quick reference, that video may be found here: [Panopto Recording](https://wgu.hosted.panopto.com/Panopto/Pages/Viewer.aspx?id=69112358-cd10-4461-9649-b18e00e57bfd)

# H – Third-Party Code References

Astinin, S. (2022, 11 25). *tabulate 0.9.0*. Retrieved from Python Package Index: https://pypi.org/project/tabulate/

Eiler, J. (2017, 01 16). *Remap values in pandas column with a dict, preserve NaNs*. Retrieved from Stackoverflow: https://stackoverflow.com/questions/20250771/remap-values-in-pandas-column-with-a-dict-preserve-nans

Prashant. (2016, 12 28). *How to systematically remove collinear variables (pandas columns) in Python?* Retrieved from Cross Validated: https://stats.stackexchange.com/questions/155028/how-to-systematically-remove-collinear-variables-pandas-columns-in-python

Ray, S. (2020, 10 21). *Show all colums of a pandas-dataframe in ".describe()"*. Retrieved from Stackoverflow: https://stackoverflow.com/questions/64455605/show-all-colums-of-a-pandas-dataframe-in-describe

Vijayakumar, A. (2019, 02 11). *Stepwise Regression*. Retrieved from Github: https://github.com/AakkashVijayakumar/stepwise-regression

# I – Referenced Works

Bobbitt, Z. (2021, 11 16). *The Five Assumptions of Multiple Linear Regression*. Retrieved from Statology: https://www.statology.org/multiple-linear-regression-assumptions/

Datacamp. (2022). *Python vs R for Data Science: Which Should You Learn?* Retrieved 03 22, 2024, from Datacamp Blog: https://www.datacamp.com/blog/python-vs-r-for-data-science-whats-the-difference

Frost, J. (2017, 09 22). *Multicollinearity in Regression Analysis: Problems, Detection, and Solutions*. Retrieved from Statistics by Jim: https://statisticsbyjim.com/regression/multicollinearity-in-regression-analysis/#comments

Larose, C. D., & Larose, D. T. (2019). *Data Science Using Python and R.* Hoboken, NJ, USA: Wiley. Retrieved from https://search.ebscohost.com/login.aspx?direct=true&db=nlebk&AN=2091371&site=eds-live&scope=site

Mahto, K. K. (2019, 07 08). *One-Hot-Encoding, Multicollinearity and the Dummy Variable Trap*. Retrieved from Towards Data Science: https://towardsdatascience.com/one-hot-encoding-multicollinearity-and-the-dummy-variable-trap-b5840be3c41a

Middleton, K. D. (2023, 07 13). *Getting Started with D208 - Part 1*. Retrieved from Data Science Team - Student Facing Resources: https://westerngovernorsuniversity.sharepoint.com/sites/DataScienceTeam/Shared%20Documents/Forms/AllItems.aspx?csf=1&web=1&e=9ccodm&cid=656002fd%2Da3e2%2D4286%2Da361%2D8300bc813a87&FolderCTID=0x01200022092E63FD85A64A8ABFB4F5AEA4839A&id=%2Fsites%2FDataScie

Middleton, K. D. (2023, 07 13). *Getting Started with D208 - Part 2*. Retrieved from Data Science Team - Student Facing Resources: https://westerngovernorsuniversity.sharepoint.com/sites/DataScienceTeam/Shared%20Documents/Forms/AllItems.aspx?csf=1&web=1&e=9ccodm&cid=656002fd%2Da3e2%2D4286%2Da361%2D8300bc813a87&FolderCTID=0x01200022092E63FD85A64A8ABFB4F5AEA4839A&id=%2Fsites%2FDataScie

Oleszak, M. (2023, 08 01). *Feature Selection Methods and How to Choose Them*. Retrieved from Neptune.ai: https://neptune.ai/blog/feature-selection-methods

Tripathi, A. (2019, 06 10). *Feature Selection Techniques in Regression Model*. Retrieved from Towards Data Science: https://towardsdatascience.com/feature-selection-techniques-in-regression-model-26878fe0e24e

WGU Information Technology. (n.d.). *R or Python*. Retrieved 03 22, 2024, from https://www.wgu.edu/online-it-degrees/programming-languages/r-or-python.html