Tucker Atwood
WGU MSDA
D206 Performance Assessment
2/29/24

## A. Research Question

Using data from the "churn" database, I will consider the research question, "Which factors in churn analysis are affected by education level?"

## B. Dataset Variables

The variables to be used in considering this question are as follows.

| Variable: | Data type: | Description: | Example: |
|---|---|---|---|
| CaseOrder | Integer | Original order of data. | 1, 2, 3, etc. |
| Customer_id | Character | ID for each customer. | X00931 |
| Interaction | Character | ID for each customer interaction. | dce8258e-a729-4597-93b7-e678c62e3ab4 |
| City | Character | Customer's city of residence. | Rimrock |
| State | Character | Customer's state of residence (2-letter state code). | AZ |
| County | Character | Customer's county of residence. | Yavapai |
| Zip | Integer | Zip code for customer's residence. | 86335 |
| Lat | Numeric | Latitude coordinate for customer's residence. | 34.69424 |
| Lng | Numeric | Longitude coordinate for customer's residence. | -111.68285 |
| Population | Integer | Census population of customer's area of residence. | 5364 |
| Area | Character | Census area description. | Urban |
| Timezone | Character | Time zone of customer's residence. | America/Phoenix |
| Job | Character | Customer's job title. | Multimedia specialist |
| Children | Integer | Number of children in customer's household. | 2 |
| Age | Integer | Customer's age in years. | 70 |
| Education | Character | Customer's highest level of education attained. | Associate's Degree |
| Employment | Character | Customer's employment status. | Part Time |

| | | | |
|---|---|---|---|
| Income | Numeric | Customer's annual income in US dollars. | 60476.86 |
| Marital | Character | Customer's marital status. | Separated |
| Gender | Character | Customer's self-identified gender. | Male |
| Churn | Character | Whether or not the customer stopped using the service in the last month. | Yes/No |
| Outage_sec_perweek | Numeric | Average seconds of system outages per week in customer's surrounding area. | 9.349593 |
| Email | Integer | Emails sent to customer in the past year. | 9 |
| Contacts | Integer | Number of times customer requested support. | 2 |
| Yearly_equip_failure | Integer | Number of times customer's equipment failed and/or had to be replaced in the past year. | 1 |
| Techie | Character | Whether or not customer considers themself capable of solving technical problems. | Yes/No |
| Contract | Character | Contract length for customer's service plan. | Month-to-month |
| Port_modem | Character | Whether or not customer has a portable modem. | Yes/No |
| Tablet | Character | Whether or not customer has a tablet. | Yes/No |
| InternetService | Character | Customer's internet service type. | Fiber optic |
| Phone | Character | Whether or not customer has a phone service. | Yes/No |
| Multiple | Character | Whether or not customer has more than one phone line. | Yes/No |
| OnlineSecurity | Character | Whether or not customer has an online security program. | Yes/No |
| OnlineBackup | Character | Whether or not customer has an online backup program. | Yes/No |
| DeviceProtection | Character | Whether or not customer has signed up for device protection. | Yes/No |
| TechSupport | Character | Whether or not customer has a technical support program. | Yes/No |

| | | | |
|---|---|---|---|
| StreamingTV | Character | Whether or not customer has signed up for TV streaming. | Yes/No |
| StreamingMovies | Character | Whether or not customer has signed up for movie streaming. | Yes/No |
| PaperlessBilling | Character | Whether or not customer has signed up for paperless billing. | Yes/No |
| PaymentMethod | Character | Customer's payment method. | Electronic check |
| Tenure | Numeric | Number of months customer has been with the service. | 34.496428 |
| MonthlyCharge | Numeric | Average amount customer has been charged per month. | 170.52421 |
| Bandwidth_GB_Year | Numeric | Average amount of data customer has used per year in gigabytes. | 717.6824 |
| Item1 | Integer | Customer's rating of the importance of timely responses. | 1-8<br>1: most important<br>8: least important |
| Item2 | Integer | Customer's rating of the importance of timely fixes. | 1-8<br>1: most important<br>8: least important |
| Item3 | Integer | Customer's rating of the importance of timely replacements. | 1-8<br>1: most important<br>8: least important |
| Item4 | Integer | Customer's rating of the importance of reliability. | 1-8<br>1: most important<br>8: least important |
| Item5 | Integer | Customer's rating of the importance of having multiple options available. | 1-8<br>1: most important<br>8: least important |
| Item6 | Integer | Customer's rating of the importance of respectful responses. | 1-8<br>1: most important<br>8: least important |
| Item7 | Integer | Customer's rating of the importance of courteous exchanges. | 1-8<br>1: most important<br>8: least important |
| Item8 | Integer | Customer's rating of the importance of active listening. | 1-8<br>1: most important<br>8: least important |

## C1. Detecting Data Quality Issues

The following methods were used to detect the presence and scope of dirty data (defined as data with inherent quality issues that must be addressed) within the churn dataset.

- **Duplicates:**
  - Full duplicates were checked for by using the "duplicated" function.
  - Potential partial duplicates were checked for according to the Customer_id factor using the "filter" function and the "count" function in the dplyr package.
  - Other potential partial duplicates were checked using the "filter" and "duplicated" functions along six factors: State, City, Job, Gender, Employment, and Education.
- **Missing values:**
  - The total and proportion of "NA" values in the entire dataset and each individual variable were detected using the "n_miss," "prop_miss," and "miss_var_summary" functions from the naniar package.
  - These missing values were visualized according to each variable using the "vis_miss" function from the visdat package, with the missing values clustered to see potential patterns in missingness.
  - Population values of 0 were detected using the "filter" and "count" functions.
- **Re-expression of numeric variables:**
  - The data type of Zip was assessed using the "str" function, and values with less than 5 digits were found using the "count" and "filter" functions, grouped by State.
- **Outliers:**
  - Outliers were detected for all quantitative variables: Latitude, Longitude, Population, Children, Age, Income, Outage_sec_perweek, Email, Contacts, Tenure, Yearly_equip_failure, MonthlyCharge, Bandwidth_GB_Year, and Items 1-8.
  - The Latitude and Longitude data were visualized using the "geom_point" (scatterplot) function from the ggplot2 package.
  - For items 1-8, values below 1 or above 8 were detected using the "filter" function.
  - The spread of all other variables was visualized using the "hist" (histogram) and "boxplot" functions.
  - If the data appeared to follow a Normal distribution, outliers were detected by normalizing the data and filtering for z-scores less than -3 or greater than 3. This was done using the "filter," "mutate," and "scale" functions.
  - If the data did not appear to follow a Normal distribution, outliers were detected using the Interquartile Range (IQR) method: by multiplying the IQR by 1.5, subtracting this value from Q1 (first quartile) and adding it to Q3 (third quartile), and finding values below or above these boundaries. This was done using the "filter," "quantile," and "IQR" functions.
  - The total number of outliers for each variable was found using the "count" function and summary statistics of the outliers were found using the "summary" function.

- **Re-expression of categorical variables:**
    - Categorical variables to be used in calculations such as regressions to answer the research question require alteration: Churn and Education.
    - The "str" function was used to confirm the data type for each factor.
    - For each categorical variable, the "unique" function provided a list of all distinct values.

## C2: Data Quality Detection Reasoning

The above methods of dirty data detection have been chosen for the following reasons.

- **Duplicates:**
    - These must be detected and treated so that customers do not represent multiple cases within the data.
    - Full duplicates (customer rows that contain the same response for every variable) were detected first because their presence is clearest and easiest to determine.
    - Since customers may be incorrectly entered into the system twice with slightly different information in one or more columns, partial duplicate detection was also required.
    - Matches in the Customer_id variable were found first, since each customer had a unique identifier which should be used exactly once.
    - Matches in the factors State, City, Job, Gender, Employment, and Education were then found, as these were determined to be highly explanatory of each customer and any matches along all six factors may be either a customer entered into the data twice with two different Customer_ids, or a coincidence.
- **Missing values:**
    - These must be detected and treated so that all customer information is as complete and accurate as possible to perform analysis.
    - The total NA values for the dataset provided an overall understanding of how much inherent missing data was present.
    - Expressing this value as a proportion then put this value into an easily understood context.
    - Further breaking down the NA values by variable allowed a clearer look into the specific areas where data was missing.
    - Additionally, since each customer can be assumed to be a member of their listed population, all population values should be at least 1; therefore, any population values of 0 were flagged as missing.
- **Re-expression of numeric variables:**
    - This was necessary for the Zip factor because, although a zip code is presented as a numeric entry, it represents a location and is better suited to be used as a categorical variable.

    - In particular, problems arose with zip codes that began with one or more "0" digits, which were erroneously dropped when expressed as integers. Finding zip codes of less than 5 digits and grouping them by State provided insight into whether they were found exclusively in states with zip codes starting with at least one "0."
- **Outliers:**
    - These must be detected and analyzed to ensure there have not been data entry errors or mismatched units that would significantly affect analysis.
    - For the Latitude and Longitude data, a scatterplot was determined to be the best visualization of potential outliers, as these factors are inherently linked and refer to physical locations easiest to view together.
    - For items 1-8, values are presented on a scale from 1 to 8, so any values outside that range have been treated as outliers.
    - For all other factors, histograms provided a first look at the shape of the data, including whether or not a Normal distribution shape was apparent (which would inform the following outlier detection method) and whether there are values far away from all others.
    - Boxplots were then used to further explore the shape and spread of the data, and were particularly useful by clearly identifying outliers using the IQR method.
    - The z-score method was used to identify outliers only for data appearing to follow a Normal distribution shape, because this is a requirement for z-score applications.
    - For other data shapes and spreads, the IQR method was used to identify outliers, as it is the standard method for non-Normal data.
- **Re-expression of categorical variables:**
    - This was necessary for the Churn and Education factors, which are particularly relevant to the research question and would be used for calculations such as regressions in answering the question. Confirming that these factors had character data types identified that they must be re-expressed later.
    - Examining lists of unique values for each categorical variable was used to find potential redundant and/or overlapping identifiers that could later be consolidated into fewer, clearer values.

## C3. Programming Language

This research question was considered using the programming language R in the RStudio environment. This language was chosen because it is ideal for initial exploration and examination of data, as well as conducting quick and efficient statistical modeling. Within R, the following packages were utilized: plyr, dplyr, naniar, visdat, stringr, ggplot2, simputation, and factoextra. These packages have been chosen to provide several functions that will make the data cleaning process more accessible and effective.

## C4. Detection Code

The following code written in the language R executes the detection of dirty data as described. An executable version of this code can be found in the attached file: Atwood_D206_PA_Code.R.

```
churn <- read.csv("C:/Users/atwoo/OneDrive/Desktop/206Directory/churn_raw_data.csv")
# imports dataset

# DETECTING DUPLICATES:

sum(duplicated(churn)) # checks for full duplicates (0)

library(plyr) # using plyr package
library(dplyr) # using dplyr package
churn %>%
  count(Customer_id) %>%
  filter(n > 1) # checks for partial duplicates with matching Customer_id (0)

churn %>%
  filter(duplicated(churn[,c('State','City','Job','Gender','Employment','Education')]) |
         duplicated(churn[,c('State','City','Job','Gender','Employment','Education')], fromLast =
TRUE))
# partial duplicates based on specified columns
# 1 match, but other factors make this highly unlikely to be a duplicate

# DETECTING MISSING VALUES:

library(naniar) # using naniar package
n_miss(churn) # total missing values (13,906)
prop_miss(churn) # proportion of missing values (approx. 0.027)
miss_var_summary(churn) # gives total and proportion of missing values for each variable
# data is missing from 8 factors: Children, Age, Income, Techie, Phone, TechSupport, Tenure,
Bandwidth_GB_Year

library(visdat) # using visdat package
vis_miss(churn, cluster = TRUE) # visualize missing data; no clear patterns between missingness

churn %>%
  filter(Population == 0) %>%
  count() # finds cases with population listed as 0 (97)
```

```
# DETECTING INCORRECT APPLICATION OF ZIP CODE:

str(churn$Zip) # checks data type of Zip

library(stringr) # using stringr package
churn %>%
  filter(str_length(Zip) != 5) %>%
  count(State)
# finds zip codes less than 5 digits, by state (verifies this only takes place in states with zip codes
starting with 0)

# DETECTING OUTLIERS:

library(ggplot2) # using ggplot2 package
churn %>%
  ggplot(aes(x=Lat, y=Lng)) + geom_point()
# scatterplot of Latitude and Longitude values; outliers present on each end of Latitude, low end
of Longitude

churn %>%
  filter(Lng < -125) %>%
  count()
# 112 low Longitude values

churn %>%
  filter(Lat < 24) %>%
  count()
# 75 low Latitude values

churn %>%
  filter(Lat > 50) %>%
  count()
# 77 high Latitude values

item_outliers <- churn %>%
  filter(item1 < 1 | item1 > 8
         | item2 < 1 | item2 > 8
         | item3 < 1 | item3 > 8
         | item4 < 1 | item4 > 8
```

```
        | item5 < 1 | item5 > 8
        | item6 < 1 | item6 > 8
        | item7 < 1 | item7 > 8
        | item8 < 1 | item8 > 8)
# items 1-8 listed as scale from 1 to 8; finds values outside range
count(item_outliers) # confirms no values outside range

hist(churn$Population) # visualization of Population data; skewed right
boxplot(churn$Population) # many outliers present
pop_outliers <- churn %>%
  filter(Population < quantile(churn$Population, 0.25) - IQR(churn$Population) * 1.5
       | Population > (quantile(churn$Population, 0.75) + IQR(churn$Population) * 1.5))
# find outliers using IQR method
count(pop_outliers) # 937 outliers
summary(pop_outliers$Population) # outlier range is 31,816-111,850

hist(churn$Children) # visualization of Children data; skewed right
boxplot(churn$Children) # 3 outlier values appear
chi_outliers <- churn %>%
  filter((Children < quantile(churn$Children, 0.25, na.rm = TRUE) - IQR(churn$Children, na.rm
= TRUE) * 1.5)
       | (Children > quantile(churn$Children, 0.75, na.rm = TRUE) + IQR(churn$Children, na.rm
= TRUE) * 1.5))
# find outliers using IQR method
count(chi_outliers) # 302 outliers
summary(chi_outliers$Children) # outlier range is 8-10

hist(churn$Age) # visualization of Age data; relatively uniform
boxplot(churn$Age) # no outliers present
age_outliers <- churn %>%
  filter((Age < quantile(churn$Age, 0.25, na.rm = TRUE) - IQR(churn$Age, na.rm = TRUE) *
1.5)
       | (Age > quantile(churn$Age, 0.75, na.rm = TRUE) + IQR(churn$Age, na.rm = TRUE) *
1.5))
# find outliers using IQR method
count(age_outliers) # confirms zero Age outliers

hist(churn$Income) # visualization of Income data; skewed right
boxplot(churn$Income) # many outliers present
inc_outliers <- churn %>%
```

```
  filter((Income < quantile(churn$Income, 0.25, na.rm = TRUE) - IQR(churn$Income, na.rm =
TRUE) * 1.5)
        | (Income > quantile(churn$Income, 0.75, na.rm = TRUE) + IQR(churn$Income, na.rm =
TRUE) * 1.5))
# find outliers using IQR method
count(inc_outliers) # 249 outliers
summary(inc_outliers$Income) # outlier range is 104,868-258,901

hist(churn$Outage_sec_perweek) # visualization of Outage_sec_perweek data; skewed right
boxplot(churn$Outage_sec_perweek) # many outliers present
outage_outliers <- churn %>%
  filter((Outage_sec_perweek < quantile(churn$Outage_sec_perweek, 0.25) -
IQR(churn$Outage_sec_perweek) * 1.5)
        | (Outage_sec_perweek > quantile(churn$Outage_sec_perweek, 0.75) +
IQR(churn$Outage_sec_perweek) * 1.5))
# find outliers using IQR method
count(outage_outliers) # 539 outliers
summary(outage_outliers$Outage_sec_perweek) # outlier range is (-1.349)-47.049

hist(churn$Email) # visualization of Email data; normal distribution
boxplot(churn$Email) # 6 outlier values appear
email_outliers <- churn %>%
  mutate(email_z = scale(churn$Email)) %>%
  filter(email_z > 3 | email_z < -3)  # find outliers using z-score method
count(email_outliers) # 12 outliers
summary(email_outliers$Email) # outlier range is 1-23

hist(churn$Contacts) # visualization of Contacts data; skewed right
boxplot(churn$Contacts) # 2 outlier values appear
con_outliers <- churn %>%
  filter((Contacts < quantile(churn$Contacts, 0.25) - IQR(churn$Contacts) * 1.5)
        | (Contacts > quantile(churn$Contacts, 0.75) + IQR(churn$Contacts) * 1.5))
# find outliers using IQR method
count(con_outliers) # 8 outliers
summary(con_outliers$Contacts) # outlier range is 6-7

hist(churn$Yearly_equip_failure) # visualization of Yearly_equip_failure data; skewed right
boxplot(churn$Yearly_equip_failure) # 3 outlier values appear
yef_outliers <- churn %>%
```

```
  filter((Yearly_equip_failure < quantile(churn$Yearly_equip_failure, 0.25) -
IQR(churn$Yearly_equip_failure) * 1.5)
        | (Yearly_equip_failure > quantile(churn$Yearly_equip_failure, 0.75) +
IQR(churn$Yearly_equip_failure) * 1.5))
# find outliers using IQR method
count(yef_outliers) # 94 outliers
summary(yef_outliers$Yearly_equip_failure) # outlier range is 3-6

hist(churn$Tenure) # visualization of Tenure data; bimodal
boxplot(churn$Tenure) # no outliers present
ten_outliers <- churn %>%
  filter((Tenure < quantile(churn$Tenure, 0.25, na.rm = TRUE) - IQR(churn$Tenure, na.rm =
TRUE) * 1.5)
        | (Tenure > quantile(churn$Tenure, 0.75, na.rm = TRUE) + IQR(churn$Tenure, na.rm =
TRUE) * 1.5))
# find outliers using IQR method
count(ten_outliers) # confirms zero Tenure outliers

hist(churn$MonthlyCharge) # visualization of MonthlyCharge data; normal distribution
boxplot(churn$MonthlyCharge) # 5 outlier values appear
mon_outliers <- churn %>%
  mutate(mon_z = scale(churn$MonthlyCharge)) %>%
  filter(mon_z > 3 | mon_z < -3)  # find outliers using z-score method
count(mon_outliers) # 3 outliers
summary(mon_outliers$MonthlyCharge) # outlier range is 306.3-315.9

hist(churn$Bandwidth_GB_Year) # visualization of Bandwidth_GB_Year data; bimodal
boxplot(churn$Bandwidth_GB_Year) # no outliers present
bgy_outliers <- churn %>%
  filter((Bandwidth_GB_Year < quantile(churn$Bandwidth_GB_Year, 0.25, na.rm = TRUE) -
IQR(churn$Bandwidth_GB_Year, na.rm = TRUE) * 1.5)
        | (Bandwidth_GB_Year > quantile(churn$Bandwidth_GB_Year, 0.75, na.rm = TRUE) +
IQR(churn$Bandwidth_GB_Year, na.rm = TRUE) * 1.5))
# find outliers using IQR method
count(bgy_outliers) # confirms zero Bandwidth_GB_Year outliers

# DETECTING CATEGORICAL VARIABLE DATA QUALITY ISSUES:

str(churn) # checks data type for all variables
```

```
unique(churn$Area) # 3 distinct answers
unique(churn$Timezone) # 25 answers, some redundant
# will be reduced to reflect only necessary amount of distinct time zones
unique(churn$Education) # 12 distinct answers
# will be re-expressed as numeric to allow regression calculations
unique(churn$Employment) # 5 distinct answers
unique(churn$Marital) # 5 distinct answers
unique(churn$Gender) # 3 distinct answers
unique(churn$Churn) # all values either "Yes" or "No"
# will be re-expressed as numeric to allow regression calculations
unique(churn$Techie) # all values either "Yes," "No," or "NA"
unique(churn$Contract) # 3 distinct answers
unique(churn$Port_modem) # all values either "Yes" or "No"
unique(churn$Tablet) # all values either "Yes" or "No"
unique(churn$InternetService) # 3 distinct answers
unique(churn$Phone) # all values either "Yes" or "No"
unique(churn$Multiple) # all values either "Yes" or "No"
unique(churn$OnlineSecurity) # all values either "Yes" or "No"
unique(churn$OnlineBackup) # all values either "Yes" or "No"
unique(churn$DeviceProtection) # all values either "Yes" or "No"
unique(churn$TechSupport) # all values either "Yes," "No," or "NA"
unique(churn$StreamingTV) # all values either "Yes" or "No"
unique(churn$StreamingMovies) # all values either "Yes" or "No"
unique(churn$PaperlessBilling) # all values either "Yes" or "No"
unique(churn$PaymentMethod) # 4 distinct answers
```

**D1. Data Quality Issue Detection Findings**

The following data quality issues were found using the data cleaning process described.

- **Duplicates:**
  - Zero full duplicates were found.
  - Zero partial duplicates of Customer_id values were found.
  - One match was found along the specified six factors: Customer_ids 1137 and 1189 each had the same values in State, City, Job, Gender, Employment, and Education.
- **Missing values:**
  - In total, 13,906 “NA” values were found, which represented about 2.7% of the data.
  - Data was found to be missing in 8 factors (total in parentheses): Children (2495), Age (2475), Income (2490), Techie (2477), Phone (1026), TechSupport (991), Tenure (931), and Bandwidth_GB_Year (1021).
  - In addition, 97 values of 0 for the Population factor were found.

- **Re-expression of numeric variables:**
    - Zip codes of less than 5 digits were found in 8 states: Maine, New Hampshire, Vermont, Massachusetts, Connecticut, New Jersey, Rhode Island, and Puerto Rico. In checking zip code information on https://www.unitedstateszipcodes.org/, it was confirmed the first seven of these states primarily have zip codes starting with "0" and Puerto Rico has zip codes starting with "00."
- **Outliers:**
    - In the scatterplot of Latitude and Longitude data, clear groups of data were found on each end of Latitude and the low end of Longitude. Filtering by the apparent cutoffs for these groups revealed 75 low Latitude values, 77 high Latitude values, and 112 low Longitude values.
    - For items 1-8, for which values below 1 or above 8 were to be considered outliers, zero such values were found.
    - For the Population factor, 937 outliers were found, ranging from 31,816 to 111,850.
    - For the Children factor, 302 outliers were found, ranging from 8 to 10.
    - For the Age factor, zero outliers were found.
    - For the Income factor, 249 outliers were found, ranging from 104,868 to 258,901.
    - For the Outage_sec_perweek factor, 539 outliers were found, ranging from -1.349 to 47.049.
    - For the Email factor, 12 outliers were found, ranging from 1 to 23.
    - For the Contacts factor, 8 outliers were found, ranging from 6 to 7.
    - For the Yearly_equip_failure factor, 94 outliers were found, ranging from 3 to 6.
    - For the Tenure factor, zero outliers were found.
    - For the MonthlyCharge factor, 3 outliers were found, ranging from 306.3 to 315.9.
    - For the Bandwidth_GB_Year factor, zero outliers were found.
- **Re-expression of categorical variables:**
    - The data types of Churn and Education were confirmed to be "character."
    - In evaluating the unique lists of values for each categorical variable, data quality issues were found in the Timezone factor. In total, 25 values existed, but further exploration revealed these values represented only 7 unique time zones.

## D2. Treatment of Data Quality Issues

The following methods were used to treat the referenced dirty data, with a new dataset named churn_cln created to hold the cleaned data.

- **Duplicates:**
    - An analysis of other factors in the potential partial duplicate pair (customers 1137 and 1189) showed this match was likely not a duplicate. Most notably, customer 1137 was listed with 0 children and a tenure of approximately 14.6 months, while customer 1189 was listed with 2 children and a tenure of approximately 13.7

months. If these cases represented the same person, it is highly unlikely that the number of children would decrease from 2 to 0 after 0.9 months with the service. Therefore, it was determined that zero full or partial duplicates were present in the dataset and no treatment was necessary.

- **Missing values:**
  - Missing data in the Children factor was replaced with "0" using the "mutate" function, because a customer who does not provide a response when asked how many children they have is likely implying they have zero children.
  - Similarly, missing data in the Techie, Phone, and TechSupport factors was replaced with "No" using the "mutate" function, because a lack of response on these Yes/No questions implies an answer of No.
  - For the remaining variables with missing values, the "gg_miss_span" function from the ggplot2 package was used to detect potential patterns in the missingness, with spans of 1000 for each. No clear patterns were found.
  - Missing values in Age were treated by grouping customers by Employment, calculating the mean Age value for each group, and replacing the NA values with the resulting value. This was done because Employment was determined to have the most relevant real-world connection to Age; for example, the average age for customers whose employment is "Student" is theoretically expected to be lower than customers whose employment is "Retired." This process was completed using the "group" and "mutate" functions.
  - The same process was used to replace NA values in Income, except with the median used instead of the mean due to the skew of the data. Groups were created using the Job factor, since customers with the same jobs are likely to have similar income levels.
  - The process was also used to replace Population values of 0, also using the median due to skew in the data, with groups created using the State factor, since customers from the same state are likely to have similar surrounding population totals.
  - Using the "impute_lm" function from the simputation package, the NA values in the Tenure variable were imputed using a linear model, with the updated Age variable as the predictor. This was chosen because older customers were surmised to have been with the service for longer time periods.
  - This same process was used to impute the NA values in Bandwidth_GB_Year, using MonthlyCharge as the predictor. This was chosen because there is an assumed connection between internet data used and the monthly charge to use the data.
  - Finally, the "n_miss" function was used on the new churn_cln dataset to confirm that zero missing values remain.
- **Re-expression of numeric variables:**
  - Using the "mutate" and "as.character" functions, all zip codes were converted to characters.

- This allowed the “str_length” (from the stringr package) and “paste0” functions to be used to add leading “0” digits to the necessary zip codes.
- The “filter,” “str_length,” and “count” functions were then used to confirm all zip codes in the cleaned dataset were 5 digits.
- This was necessary for the Zip factor because, although a zip code is presented as a numeric entry, it represents a location and is better suited to be used as a categorical variable.

- **Outliers:**
  - For Latitude and Longitude data, the functions “filter” and “count” were used to determine which states were represented in the values designated as outliers. All low longitude values were found to be from Alaska (77 values) or Hawaii (35 values). All low latitude values were found to be from Hawaii (35 values) or Puerto Rico (40 values). All high latitude values were found to be from Alaska (77 values). These values were determined to be reasonable based on the geographic location of these regions. All outliers were retained.
  - For the following factors, the “summary” function was used to find the ranges and maximum values.
  - For Population data, the outlier range (31,816-111,850) and maximum value (111,850) were determined to be reasonable and acceptable. All outliers were retained.
  - For Children data, the outlier range (8-10) and maximum value (10) were determined to be reasonable and acceptable. All outliers were retained.
  - For Income data, the outlier range (104,868-258,901) and maximum value (258,901) were determined to be reasonable and acceptable. All outliers were retained.
  - For Outage_sec_perweek data, the minimum value was negative, which does not make sense in the context of the factor. Using the “mutate” function, all negative values were replaced with 0. The maximum value (47.049) was determined to be reasonable and acceptable. All other non-negative outliers were retained. The “summary” function was used to confirm the new minimum value for this factor was 0.
  - For Email data, the outlier range (1-23) and maximum value (23) were determined to be reasonable and acceptable. All outliers were retained.
  - For Contacts data, the outlier range (6-7) and maximum value (7) were determined to be reasonable and acceptable. All outliers were retained.
  - For Yearly_equip_failure data, the outlier range (3-6) and maximum value (6) were determined to be reasonable and acceptable. All outliers were retained.
  - For MonthlyCharge data, the outlier range (306.3-315.9) and maximum value (315.9) were determined to be reasonable and acceptable. All outliers were retained.
  - All other relevant factors not listed here were found to have zero outliers.

- **Re-expression of categorical variables:**
    - The Churn factor was converted to numeric by replacing all "No" responses with "0" and all "Yes" responses with "1." This was performed using the "as.numeric" and "revalue" (from the plyr package) functions.
    - With the same functions, the Education factor was converted to numeric by creating the following ordinal ranking of education level:
        - 1: "No Schooling Completed"
        - 2: "Nursery School to 8th Grade"
        - 3: "9th Grade to 12th Grade, No Diploma"
        - 4: "GED or Alternative Credential"
        - 5: "Regular High School Diploma"
        - 6: "Some College, Less than 1 Year"
        - 7: "Some College, 1 or More Years, No Degree"
        - 8: "Associate's Degree"
        - 9: "Bachelor's Degree"
        - 10: "Master's Degree"
        - 11: "Professional School Degree"
        - 12: "Doctorate Degree"
    - These conversions were conducted so that the Churn and Education factors could later be used for calculations such as regressions to address the research question.
    - Using https://www.timeanddate.com/time/map/, all 25 unique listed values in the Timezone factor were found to belong to one of the following 7 time zones and replaced as such: Alaska, Atlantic, Central, Eastern, Hawaii, Mountain, and Pacific. This was performed using the "revalue" function.
    - This conversion was conducted because it simplified the data and allowed better comparisons between cases with the same time zones.
    - The "unique" function was used to confirm the new values of the re-expressed categorical variables.

## D3. Summary of Treatment

In summary, the following processes of detection and treatment of dirty data were conducted.

- **Duplicates:**
    - A check for full duplicates was run, with none found.
    - A check for partial duplicates was run on the Customer_id factor, with none found.
    - A check for partial duplicates was run to find matches on the factors State, City, Job, Gender, Employment, and Education. One potential match was found, but an examination of other factors (in particular, Tenure and Children) showed this was highly unlikely to be a duplicate value.
    - It was determined that there were no full or partial duplicates to be treated.

- **Missing values:**
    - All "NA" values were found, including their total, proportion, and breakdown by variable. The factors containing NA values were Children, Age, Income, Techie, Phone, TechSupport, Tenure, and Bandwidth_GB_Year.
    - Missing values in Children were replaced with "0."
    - Missing values in Techie, Phone, and TechSupport were replaced with "No."
    - Missing values in Age were replaced with the mean Age of all customers with the same Employment value.
    - Missing values in Income were replaced with the mean Income of all customers with the same Job value.
    - Missing values in Tenure were imputed using a linear model, with the updated Age factor as the predictive variable.
    - Missing values in Bandwidth_GB_Year were imputed using a linear model, with the MonthlyCharge factor as the predictive variable.
    - Values of "0" in the Population factor were determined to be missing and were replaced with the mean Population of all customers with the same State value.
- **Re-expression of numeric data:**
    - The data type of Zip was found to be integer.
    - All Zip values less than 5 digits were found and grouped by State.
    - All Zip values were converted to character data type.
    - For 4-digit Zip values, a leading "0" was added.
    - For 3-digit Zip values, a leading "00" was added.
- **Outliers:**
    - Outliers were visualized using a scatterplot for Latitude/Longitude. Histograms and boxplots were used for most other quantitative variables: Population, Children, Age, Income, Outage_sec_perweek, Email, Contacts, Yearly_equip_failure, Tenure, MonthlyCharge, and Bandwidth_GB_Year.
    - Outliers were defined for Items 1-8 as any value outside the provided scale of responses: lower than 1 or higher than 8.
    - Outliers were detected using the z-score method for variables that appeared to follow a Normal distribution, and using the IQR method for all others.
    - Outliers in the Outage_sec_perweek factor were determined to be quality issues, since the minimum value showed there were negative values within this factor. This did not make sense in context, so all negative values were replaced with "0."
    - All other outliers found were examined in the context of the factors and determined to be reasonable and acceptable.
- **Re-expression of categorical data:**
    - The Churn and Education factors were relevant to the research question and required conversion to numeric variables in order for operational calculations such as regressions to be run on them.

- Churn was converted into a numeric variable, with “0” representing “No” and “1” representing “Yes.”
- Education was converted into a numeric variable using a common-sense ordinal ranking of the given education levels.
- The Timezone factor contained 25 separate values which represented 7 distinct time zones. The values were replaced with their respective time zones and relisted as one of the following: Alaska, Atlantic, Central, Eastern, Hawaii, Mountain, and Pacific.

**D4. Treatment Code**

The following code written in the language R executes the treatment of dirty data as described. An executable version of this code can be found in the attached file: Atwood_D206_PA_Code.R.

```
# TREATING MISSING VALUES:

churn_cln <- churn %>%
  mutate(Children = ifelse(is.na(Children), 0, Children)) %>%
  mutate(Techie = ifelse(is.na(Techie), "No", Techie)) %>%
  mutate(Phone = ifelse(is.na(Phone), "No", Phone)) %>%
  mutate(TechSupport = ifelse(is.na(TechSupport), "No", TechSupport))
# replaces missing values in Children with 0 and missing values in Techie, Phone, and
TechSupport with “No”

gg_miss_span(churn_cln, Age, 1000)
gg_miss_span(churn_cln, Income, 1000)
gg_miss_span(churn_cln, Tenure, 1000)
gg_miss_span(churn_cln, Bandwidth_GB_Year, 1000)
# checks for clear patterns of missingness in remaining quantitative variables (none)

churn_cln <- churn_cln %>%
  group_by(Employment) %>%
  mutate(Age = ifelse(is.na(Age), as.integer(mean(Age, na.rm = TRUE)), Age)) %>%
  ungroup()
# replaces NA Age values with mean age from that Employment type

churn_cln <- churn_cln %>%
  group_by(Job) %>%
  mutate(Income = ifelse(is.na(Income), as.integer(median(Income, na.rm = TRUE)), Income))
%>%
  ungroup()
```

```
# replaces NA Income values with median income from that Job

churn_cln <- churn_cln %>%
  mutate(Population = replace(Population, Population == 0, NA)) %>%
  group_by(State) %>%
  mutate(Population = ifelse(is.na(Population), as.integer(median(Population, na.rm = TRUE)),
Population)) %>%
  ungroup()
# replaces Population values of 0 with median Population from that state

library(simputation) # using simputation package

churn_cln <- churn_cln %>%
  impute_lm(Tenure ~ Age)
# replaces NA Tenure values with linear model using Age factor

churn_cln <- churn_cln %>%
  impute_lm(Bandwidth_GB_Year ~ MonthlyCharge)
# replaces NA Bandwidth_GB_Year values with linear model using MonthlyCharge factor

n_miss(churn_cln) # confirm 0 missing values

# TREATING INCORRECT APPLICATION OF ZIP CODE:

churn_cln <- churn_cln %>%
  mutate(Zip = as.character(Zip)) %>%
  mutate(Zip = ifelse(str_length(Zip) == 3, paste0("00",Zip), Zip)) %>%
  mutate(Zip = ifelse(str_length(Zip) == 4, paste0("0",Zip), Zip))
# converts zip code to character, adds leading 00 to zip codes with 3 digits and 0 to zip codes
with 4 digits

churn_cln %>%
  filter(str_length(Zip) != 5) %>%
  count() # confirms all zip codes are now 5 digits

# TREATING OUTLIERS:

churn_cln %>%
  filter(Lng < -125) %>%
  count(State)
```

```
# confirms all low Longitude values are from Alaska (77 values) and Hawaii (35 values)
# Longitude outliers will be retained

churn_cln %>%
  filter(Lat < 24) %>%
  count(State)
# confirms all low Latitude values are from Puerto Rico (40 values) and Hawaii (35 values)

churn_cln %>%
  filter(Lat > 50) %>%
  count(State)
# confirms all high Latitude values are from Alaska (77 values)
# Latitude outliers will be retained

summary(pop_outliers$Population)
# outlier range (31,816-111,850) and maximum value (111,850) are reasonable and acceptable
# Population outliers will be retained

summary(chi_outliers$Children)
# outlier range (8-10) and maximum value (10) are reasonable and acceptable
# Children outliers will be retained

summary(inc_outliers$Income)
# outlier range (104,868-258,901) and maximum value (258,901) are reasonable and acceptable
# Income outliers will be retained

summary(outage_outliers$Outage_sec_perweek)
# minimum value shows negative values, maximum value (47.049) is reasonable and acceptable
# Negative Outage_sec_perweek values will be imputed and other outliers will be retained

churn_cln <- churn_cln %>%
  mutate(Outage_sec_perweek = ifelse(Outage_sec_perweek < 0, Outage_sec_perweek == 0,
Outage_sec_perweek))
# replaces negative Outage_sec_perweek values with 0

summary(churn_cln$Outage_sec_perweek)
# confirms minimum value is now 0

summary(email_outliers$Email)
# outlier range (1-23) and maximum value (23) are reasonable and acceptable
```

```
# Email outliers will be retained

summary(con_outliers$Contacts)
# outlier range (6-7) and maximum value (7) are reasonable and acceptable
# Contacts outliers will be retained

summary(yef_outliers$Yearly_equip_failure)
# outlier range (3-6) and maximum value (6) are reasonable and acceptable
# Yearly_equip_failure outliers will be retained

summary(mon_outliers$MonthlyCharge)
# outlier range (306.3-315.9) and maximum value (315.9) are reasonable and acceptable
# MonthlyCharge outliers will be retained

# RE-EXPRESSING CATEGORICAL VARIABLES:

churn_cln$Timezone <- revalue(churn_cln$Timezone, replace = c("America/Sitka" = "Alaska",
"America/Detroit" = "Eastern", "America/Los_Angeles" = "Pacific", "America/Chicago" =
"Central", "America/New_York" = "Eastern", "America/Puerto_Rico" = "Atlantic",
"America/Denver" = "Mountain", "America/Menominee" = "Central", "America/Phoenix" =
"Mountain", "America/Indiana/Indianapolis" = "Eastern", "America/Boise" = "Mountain",
"America/Kentucky/Louisville" = "Eastern", "Pacific/Honolulu" = "Hawaii",
"America/Indiana/Petersburg" = "Eastern", "America/Nome" = "Alaska", "America/Anchorage"
= "Alaska", "America/Indiana/Knox" = "Central", "America/Juneau" = "Alaska",
"America/Toronto" = "Eastern", "America/Indiana/Winamac" = "Eastern",
"America/Indiana/Vincennes" = "Eastern", "America/North_Dakota/New_Salem" = "Central",
"America/Indiana/Tell_City" = "Central", "America/Indiana/Marengo" = "Eastern",
"America/Ojinaga" = "Central"))
# consolidates redundant time zones into 7 distinct time zones

churn_cln$Education <- as.numeric(revalue(churn_cln$Education, replace = c("No Schooling
Completed" = 1, "Nursery School to 8th Grade" = 2, "9th Grade to 12th Grade, No Diploma" =
3, "GED or Alternative Credential" = 4, "Regular High School Diploma" = 5, "Some College,
Less than 1 Year" = 6, "Some College, 1 or More Years, No Degree" = 7, "Associate's Degree" =
8, "Bachelor's Degree" = 9, "Master's Degree" = 10, "Professional School Degree" = 11,
"Doctorate Degree" = 12)))
# converts Education to numeric

churn_cln$Churn <- as.numeric(revalue(churn_cln$Churn, replace = c("No" = 0, "Yes" = 1)))
# converts Churn to numeric
```

```
unique(churn_cln$Timezone) # confirms new time zones
unique(churn_cln$Education) # confirms new education values
unique(churn_cln$Churn) # confirms new churn values
```

## D5. Cleaned Dataset

The resulting cleaned dataset has been written into a CSV file and attached with the following name: churn_cln.csv. If opened with Excel, ensure to not convert the removal of leading zeros.

## D6. Disadvantages of Methods Used

Potential disadvantages of the detection and treatment methods used include the following.

- **Duplicates**
  - Though partial duplicates were checked according to Customer_id, as well as a combination of six factors deemed most explanatory of each customer, it is still possible for partial duplicates to exist in this dataset without being found with these methods. A more thorough check for partial duplicates may include several other combinations of key factors, though the sheer amount of possible combinations would make this a difficult task.
- **Missing values**
  - It was assumed that missing values in Children, Techie, Phone, and TechSupport implied a negative answer from the customer ("0" or "No"). However, these values may have been missing simply due to data entry errors, in which case replacing these values with "0" or "No" could lead to inaccurate conclusions.
  - Missing values in Age and Income, and values of 0 in Population, were replaced using the mean values of these factors when customers were grouped by other factors: Employment, Job, and State, respectively. These factors were chosen for common-sense reasons, as there are theoretical connections between the factors with missing values and their grouping factors. However, a deeper analysis of correlations between Age, Income, and Population and all other variables in this dataset may reveal stronger connections elsewhere.
  - Likewise, imputations of missing values in Tenure and Bandwidth_GB_Year were made using linear models with Age and MonthlyCharge, respectively, as predictive variables. This choice, too, could prove to be inefficient with a deeper analysis of correlations between Tenure and Bandwidth_GB_Year and all other variables.
- **Re-expression of numeric variables**
  - Though zip codes are typically used only to refer to geographic locations and are not ideal for performing calculations, there may be some circumstances where it is beneficial to leave them as integers.

- **Outliers**
    - Many outliers were determined to be reasonable and acceptable, but further analysis could reveal them to not be so. For example, a population of 111,850 is not unlikely, but if it was listed as the population of a small town, this may represent a data quality issue. The same potential issue could be argued for any of the factors for which outliers were found.
    - The only outliers that required treatment were those in Outage_sec_perweek, which contained negative values. While negative values were replaced with "0," their presence may point to a larger problem with this factor.
- **Re-expression of categorical variables**
    - While the Churn and Education variables were re-expressed as numeric to make further calculations possible, this conversion may not be necessary if, for example, those calculations only require grouping by response. Other categorical variables within the dataset could have been re-expressed the same way but were not deemed necessary due to the research question. If a conversion was in fact required, these factors should have been converted alongside Churn and Education.
    - Reducing the number of time zones from 25 to 7 may help consolidate information, but the regions may have been more specific than necessary for a reason.

## D7. Cleaned Dataset Challenges

The listed disadvantages to the methods used could result in the following limitations or challenges a data analyst may face when using this cleaned dataset.

- **Duplicates**
    - Since partial duplicates may still exist in the dataset, if a customer has been added to the data after, for example, moving to a new location or changing jobs, this match would not be found since there would be no match in the relevant factors.
- **Missing values**
    - If missing values in Children, Techie, Phone, and TechSupport were data entry errors, replacing them with "0" or "No" would change the data and any findings from the data; any resulting correlations found between these factors relating to the research question may be inaccurate.
    - If missing values in Age, Income, Population, Tenure, and Bandwidth_GB_Year should have been replaced with respect to different variables than they were, connections between them and their grouping variables may have been erroneously created and misinterpreted in further data analysis relating to the research question.
- **Re-expression of numeric variables**
    - Though using zip codes as integers is unlikely to be relevant to the research question, potential analysis of zip codes as integers is possible and the conversion to character may pose a problem. Locations that are near each other generally have

closer zip codes than locations that are further apart, with more consistency on the leading digits (for example, almost all Maine zip codes begin with "04"), so any analysis of zip code location may require them to be of the integer data type.

- **Outliers**
  - If outliers that were determined reasonable and acceptable were later found to be data entry errors or misplaced values, this could affect further analysis concerning the research question. For example, if the population of 111,850 was paired with the wrong city, correlations between city and churn analysis would be inaccurate.
  - The measurement or calculation of outage seconds per week may need alteration if it produces negative values in the first place. If alterations are required, this may indicate that all values in this factor are inaccurate, which would affect the analysis of the research question by providing incorrect data.
- **Re-expression of categorical variables**
  - If the wrong factors were re-expressed as numeric variables (or were not, but should have been), analysis of the research question may have to include further re-expression which should have been done during the data cleaning phase.
  - If more specific time zones were required for analysis of the research question, reducing the number of time zones would be detrimental to the process. For example, if one region does not observe Daylight Savings Time (DST), it may need to be listed separately from a region that technically exists in the same time zone but does observe DST.

## E1. Principal Component Analysis

A Principal Component Analysis (PCA) of the cleaned dataset, churn_cln, was conducted, with the following information:

- **Variables used for PCA:**
  - Latitude, Longitude, Income, Outage_sec_perweek, Tenure, MonthlyCharge, and Bandwidth_GB_Year.
  - These were chosen because they are continuous quantitative variables.
- **PCA execution:**
  - A PCA was conducted using the "prcomp" function, with the data being centered and scaled.
- **PCA loading matrix:**
  - The following is a screenshot of the loadings matrix for the PCA:

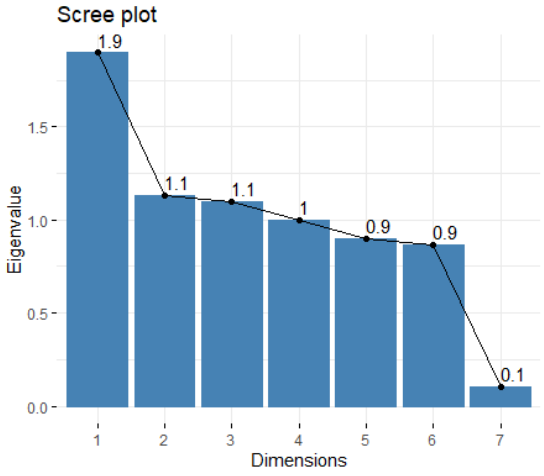
| | PC1 | PC2 | PC3 | PC4 | PC5 | PC6 | PC7 |
|---|---|---|---|---|---|---|---|
| Lat | -0.022596055 | -0.218750560 | 0.66919272 | -0.0005827425 | 0.709747398 | -0.00876467 | -0.0004390605 |
| Lng | 0.008464480 | 0.142638264 | -0.69188353 | -0.0952783384 | 0.695381398 | -0.09080800 | -0.0007224004 |
| Income | 0.005039417 | -0.029025344 | 0.05990002 | -0.9948058755 | -0.065585749 | 0.04003746 | -0.0009502473 |
| Outage_sec_perweek | 0.022705828 | -0.690483603 | -0.13871688 | -0.0104810081 | -0.089999195 | -0.70375041 | -0.0003551124 |
| Tenure | 0.704802721 | 0.055875482 | 0.03302215 | 0.0010848139 | 0.008467318 | -0.04004625 | 0.7052403978 |
| MonthlyCharge | 0.050574270 | -0.671590708 | -0.22199044 | 0.0338594733 | 0.012657300 | 0.70216907 | 0.0527288380 |
| Bandwidth_GB_Year | 0.706804208 | 0.006024345 | 0.01666413 | 0.0050474205 | 0.008372259 | 0.01281979 | -0.7070034704 |

- **PCA code:**
    - The following code was used to conduct the PCA. An executable version of this code can be found in the attached file: Atwood_D206_PA_Code.R.

```
# PRINCIPAL COMPONENT ANALYSIS:

vars <- churn_cln %>%
  select(Lat, Lng, Income, Outage_sec_perweek, Tenure, MonthlyCharge, Bandwidth_GB_Year)
# variables to be used for PCA: all continuous quantitative variables

vars_pca <- vars %>%
  prcomp(center = TRUE, scale = TRUE)
# conducts PCA, centering and scaling the data

vars_pca$rotation
# creates PCA loading matrix

library(factoextra) # using factoextra package

fviz_eig(vars_pca, choice = "eigenvalue", addlabels = TRUE)
# creates scree plot for PCA
# PC1, PC2, PC3, and PC4 all have eigenvalues greater than or equal to 1, so they will be kept, per the Kaiser Rule
```

## E2. Retained Principal Components

Using the Kaiser Rule to determine which principal components should be retained from the analysis, it was determined that PC1, PC2, PC3, and PC4 will be retained as the most important principal components for analysis of the data, because each has an eigenvalue of at least 1.0. The following scree plot provides a visualization with these values:



## E3. Organizational Benefits to PCA

This Principal Component Analysis could be beneficial to an organization by reducing the dimensionality of the data to its most important components. This could, in turn, help create a more effective data application process, especially if the data will be used in machine learning processes. For example, PC1, which is significantly influenced by the Tenure and Bandwidth_GB_Year variables and is the most important principal component in this analysis, can explain as much variance in the data as 1.9 of its original variables. By consolidating the data into a smaller amount of factors which can explain a similar amount of variance as the original data, the process of finding connections within the data can become quicker and can create more efficient results.

## F. Panopto Recording

Please refer to the link attached with a Panopto video recording. The video includes an explanation, execution, and output results of the referred code.

## G. Third-Party Code References

WGU Courseware was used as a resource to learn the methods, concepts, and functions used to create the codes in this project, including DataCamp course tracks (datacamp.com), Dr. Keiona Middleton's D206 webinar videos, and the book *Data Science Using Python and R* by Chantal D. Larose and Daniel T. Larose, and there are no codes taken directly any other sources.

## H. References

The site https://www.unitedstateszipcodes.org/ was used to find information about zip codes of specific states.
The site https://www.timeanddate.com/time/map/ was used to determine time zones for specific cities and regions.