**Part I: Research Question**

A. Describe **one** question or decision that could be addressed using the data set you chose. The summarized question or decision must be relevant to a realistic organizational need or situation.

What factors contribute most to positive customer retention? In other words, what factors keep customers from leaving?

B. Describe *all* variables in the data set (regardless of the research question) and indicate the data type for *each* variable. Use examples from the data set to support your claims.

These are the descriptions of each of the variables as described by the file that came with the data.

* **CaseOrder**: A placeholder variable to preserve the original order of the raw data file
* **Customer\_id**: Unique customer ID
* **Interaction**: Unique IDs related to customer transactions, technical support, and sign-ups
* **City**: Customer city of residence as listed on the billing statement
* **State**: Customer state of residence as listed on the billing statement
* **County**: Customer county of residence as listed on the billing statement
* **Zip**: Customer zip code of residence as listed on the billing statement
* **Lat, Lng**: GPS coordinates of customer residence as listed on the billing statement
* **Population**: Population within a mile radius of customer, based on census data
* **Area**: Area type (rural, urban, suburban), based on census data
* **TimeZone**: Time zone of customer residence based on customer’s sign-up information
* **Job**: Job of the customer (or invoiced person) as reported in sign-up information
* **Children**: Number of children in customer’s household as reported in sign-up information
* **Age**: Age of customer as reported in sign-up information
* **Education**: Highest degree earned by customer as reported in sign-up information
* **Employment**: Employment status of customer as reported in sign-up information
* **Income**: Annual income of customer as reported at time of sign-up
* **Marital**: Marital status of customer as reported in sign-up information
* **Gender**: Customer self-identification as male, female, or nonbinary
* **Churn**: Whether the customer discontinued service within the last month (yes, no)
* **Outage\_sec\_perweek**: Average number of seconds per week of system outages in the customer’s neighborhood
* **Email**: Number of emails sent to the customer in the last year (marketing or correspondence)
* **Contacts**: Number of times customer contacted technical support
* **Yearly\_equip\_failure**: The number of times customer’s equipment failed and had to be reset/replaced in the past year
* **Techie**: Whether the customer considers themselves technically inclined (based on customer questionnaire when they signed up for services) (yes, no)
* **Contract**: The contract term of the customer (month-to-month, one year, two year)
* **Port\_modem**: Whether the customer has a portable modem (yes, no)
* **Tablet**: Whether the customer owns a tablet such as iPad, Surface, etc. (yes, no)
* **InternetService**: Customer’s internet service provider (DSL, fiber optic, None)
* **Phone**: Whether the customer has a phone service (yes, no)
* **Multiple**: Whether the customer has multiple lines (yes, no)
* **OnlineSecurity**: Whether the customer has an online security add-on (yes, no)
* **OnlineBackup**: Whether the customer has an online backup add-on (yes, no)
* **DeviceProtection**: Whether the customer has device protection add-on (yes, no)
* **TechSupport**: Whether the customer has a technical support add-on (yes, no)
* **StreamingTV**: Whether the customer has streaming TV (yes, no)
* **StreamingMovies**: Whether the customer has streaming movies (yes, no)
* **PaperlessBilling**: Whether the customer has paperless billing (yes, no)
* **PaymentMethod**: The customer’s payment method (electronic check, mailed check, bank (automatic bank transfer), credit card (automatic))
* **Tenure**: Number of months the customer has stayed with the provider
* **MonthlyCharge**: The amount charged to the customer monthly. This value reflects an average per customer.
* **Bandwidth\_GB\_Year**: The average amount of data used, in GB, in a year by the customer

There are also 8 survey questions that asked customers to rate the importance of different features with 8 being the least important and 1 being the most important.

* **Item1**: Quick response
* **Item2**: Quick fixes
* **Item3**: Quick replacements
* **Item4**: Reliability
* **Item5**: Enough\_Options
* **Item6**: Respectfulness
* **Item7**: Politeness
* **Item8**: Evidence of active listening

This portion below I took directly from my notebook after using the info.() method on the dataset. It shows that there are 10,000 rows and 52 columns in the dataset. It also shows how some of the columns are missing values.

RangeIndex: 10000 entries, 0 to 9999

Data columns (total 52 columns):

# Column Non-Null Count Dtype

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

0 Unnamed: 0 10000 non-null int64

1 CaseOrder 10000 non-null int64 (categorical)

2 Customer\_id 10000 non-null object (categorical)

3 Interaction 10000 non-null object (categorical)

4 City 10000 non-null object (categorical)

5 State 10000 non-null object (categorical)

6 County 10000 non-null object (categorical)

7 Zip 10000 non-null int64 (categorical)

8 Lat 10000 non-null float64 (categorical)

9 Lng 10000 non-null float64 (categorical)

10 Population 10000 non-null int64 (numeric)

11 Area 10000 non-null object (categorical)

12 Timezone 10000 non-null object (categorical)

13 Job 10000 non-null object (categorical)

14 Children 7505 non-null float64 (numeric)

15 Age 7525 non-null float64 (numeric)

16 Education 10000 non-null object (categorical)

17 Employment 10000 non-null object (categorical)

18 Income 7510 non-null float64 (numeric)

19 Marital 10000 non-null object (categorical)

20 Gender 10000 non-null object (categorical)

21 Churn 10000 non-null object (categorical)

22 Outage\_sec\_perweek 10000 non-null float64 (numeric)

23 Email 10000 non-null int64 (categorical)

24 Contacts 10000 non-null int64 (numeric)

25 Yearly\_equip\_failure 10000 non-null int64 (numeric)

26 Techie 7523 non-null object (categorical)

27 Contract 10000 non-null object (categorical)

28 Port\_modem 10000 non-null object (categorical)

29 Tablet 10000 non-null object (categorical)

30 InternetService 7871 non-null object (categorical)

31 Phone 8974 non-null object (categorical)

32 Multiple 10000 non-null object (categorical)

33 OnlineSecurity 10000 non-null object (categorical)

34 OnlineBackup 10000 non-null object (categorical)

35 DeviceProtection 10000 non-null object (categorical)

36 TechSupport 9009 non-null object (categorical)

37 StreamingTV 10000 non-null object (categorical)

38 StreamingMovies 10000 non-null object (categorical)

39 PaperlessBilling 10000 non-null object (categorical)

40 PaymentMethod 10000 non-null object (categorical)

41 Tenure 9069 non-null float64 (numeric)

42 MonthlyCharge 10000 non-null float64 (numeric)

43 Bandwidth\_GB\_Year 8979 non-null float64 (numeric)

44 item1 10000 non-null int64 (categorical)

45 item2 10000 non-null int64 (categorical)

46 item3 10000 non-null int64 (categorical)

47 item4 10000 non-null int64 (categorical)

48 item5 10000 non-null int64 (categorical)

49 item6 10000 non-null int64 (categorical)

50 item7 10000 non-null int64 (categorical)

51 item8 10000 non-null int64 (categorical)

dtypes: float64(9), int64(15), object(28)

memory usage: 4.0+ MB

**Part II: Data-Cleaning Plan**

C. Explain the plan for cleaning the data by doing the following:

1. Propose a plan that includes the relevant techniques and specific steps needed to assess the quality of the data in the data set.

After reading in the csv file to my notebook, my data cleaning workflow will be as follows:

1. View the “head” of the data and use the info function to get a more detailed idea of what I am working with.
2. Make sure that the dictionary and actual table match up properly, renaming or dropping columns as/if needed.
3. Make sure that there is no duplicate information/columns.
4. Find the columns with Nan values and determine how to go about handling them, then handle them.
5. Look at all of your columns and ensure that their data types are uniform and make sense for what is included in the column.
6. Find outliers, if any, and handle them.

2. Justify your approach for assessing the quality of the data, including the following:

• characteristics of the data being assessed

• the approach used to assess the quality of the data

From what I have learned, doing the steps above are considered best practices when cleaning data and ensuring data quality. Since that is the case it is good to at a minimum do those things. For example, there are null values in the dataset that could very well skew the results if not handled accordingly. Also it seems that besides the Caseorder column there is another unnamed column that has duplicated the same information. That will need to be removed as well.

3. Justify your selected programming language and any libraries and packages that will support the data-cleaning process.

I am using Python over R because I have found myself to be more comfortable with it and I like its interface more than R’s. I also like how the packages have specified roles with functions that can just be called on. Speaking of packages, I will mainly be using the pandas package. I can use some of its functions like the .describe() and .head() functions to learn more about my data and then manipulate it from there.

4. Provide the annotated code you will use to assess the quality of the data in an executable script file.

# First import pandas and read in your dataset as a csv file

import pandas as pd

churn\_df = pd.read\_csv('churn\_raw\_data.csv', index\_col='CaseOrder', na\_values = 'Nan')

# Use the info method to determine the datatypes of your variables

print(churn\_df.info())

# Lets import the rest of the libraries and functions that we will need

import numpy as np

import matplotlib.pyplot as plt

import seaborn as sns

# Lets also set it so that we are able to see all the columns and rows in the tables.

# To set max columns:

pd.set\_option('display.max\_columns', None)

# Let's take a look at the head of the dataframe to get a feel for it

churn\_df.head(10)

**Part III: Data Cleaning**

D. Summarize the data-cleaning process by doing the following:

1. Describe the findings for the data quality issues found from the implementation of the data-cleaning plan from part C.

Upon completing the plan from part C, I noticed that:

- There was an unnamed column that was a duplicate of the CaseOrder column

- The columns titled ‘Item1’ - ‘item8’ were ambiguously named

- There were missing values in the 'Children', 'Age', 'Income', 'Techie', 'Internet Service', 'Phone', 'TechSupport', 'Tenure', and 'Bandwidth\_GB\_Year' columns

2. Justify your methods for mitigating the data quality issues in the data set.

* I dropped the duplicate column because it was not adding anything to the dataset since it was just a copy of the CaseOrder column.
* I renamed the survey columns so that they would be easier to understand if you were just looking at the columns without the dictionary explaining what they were.
* I decided to use fillna() instead of dropna() due to the fact that the missingness percentages were not less than 5%. I read that any more that 5% means that it is missing too much data for it to be considered insignificant.

3. Summarize the outcome from the implementation of *each* data-cleaning step.

*Shown in code attached*

4. Provide the annotated code you will use to mitigate the data quality issues—including anomalies—in the data set in an executable script file.

#Immediately I can see that the unamed column is a duplicate of the CaseOrder column.

#Let's drop that one:

churn\_df.drop(columns='Unnamed: 0')

#I also noticed that there were columns labeled item1 - item8, pretty ambiguous.

#After looking at the dictionary for this dataset, I figured it was useful to change those names to something more recognizable.

#First we create a dictionary matching the columns to their proper names:

customer\_survey = {'item1':'Quick\_Responses', 'item2':'Quick\_Solutions', 'item3':'Quick\_Replacements', 'item4':'Reliability', 'item5':'Enough\_Options', 'item6':'Respectfulness', 'item7':'Ploliteness', 'item8':'Active\_Listening'}

#Now we can rename the columns using this dictionary that we just made.

churn\_df = churn\_df.rename(columns=customer\_survey)

#Lets take another look at the dataframe now.

churn\_df.head(10)

#Next we will check for any more duplicate values. This is one of the best practice steps we should always do.

churn\_df.duplicated()

#The results show that there are no duplicated values.

#Now lets look and see about handling these null values. We do this series of steps to get a percentage of missing values.

# 1) Get the number of missing values using the .isnull and .sum() methods

missing\_values = churn\_df.isnull().sum()

# 2) Get the total number of values

total\_values = churn\_df.count() + missing\_values

# 3) Calculate the percentage of missing values

percentage\_missing = (missing\_values/total\_values) \* 100

# 4) Print and see

print(percentage\_missing)

#From this we can see that the columns missing data are:

#'Children', 'Age', 'Income', 'Techie', 'Internet Service',

#'Phone', 'TechSupport', 'Tenure', 'Bandwidth\_GB\_Year'

#These percentages are too big to just drop the columns so we will do some imputation to handle them.

#I will begin with the numerical columns:

#'Children'

churn\_df['Children'].mean() #Mean = 1.8225 so 2

churn\_df['Children'] = churn\_df['Children'].fillna(2)

#'Age'

churn\_df['Age'].mean() #Mean = 53.27574750830565 so 53

churn\_df['Age'] = churn\_df['Age'].fillna(53)

#'Income'

churn\_df['Income'].mean() #Mean = 39936.76222636485 so $39,937

churn\_df['Income'] = churn\_df['Income'].fillna(39,937)

#'Tenure'

churn\_df['Tenure'].mean() #Mean = 34.49885764604521 so 34.498858 since the filled in values end at the 6th decimal

churn\_df['Tenure'] = churn\_df['Tenure'].fillna(34.498858)

#'Bandwidth\_GB\_Year'

churn\_df['Bandwidth\_GB\_Year'].mean() #Mean = 3398.842752015135 so 3398.842752

churn\_df['Bandwidth\_GB\_Year'] = churn\_df['Bandwidth\_GB\_Year'].fillna(3398.842752)

#Now with the numerical columns out of the way, lets look at these categorical columns.

#'Techie', 'InternetService', 'Phone', 'TechSupport'

#Now with the numerical columns out of the way, lets look at these categorical columns.

#'Techie', 'InternetService', 'Phone', 'TechSupport'

Techie\_mode = churn\_df['Techie'].mode()[0]

churn\_df['Techie'] = churn\_df['Techie'].fillna(Techie\_mode)

InternetService\_mode = churn\_df['InternetService'].mode()[0]

churn\_df['InternetService'] = churn\_df['InternetService'].fillna(InternetService\_mode)

Phone\_mode = churn\_df['Phone'].mode()[0]

churn\_df['Phone'] = churn\_df['Phone'].fillna(Phone\_mode)

TechSupport\_mode = churn\_df['TechSupport'].mode()[0]

churn\_df['TechSupport'] = churn\_df['TechSupport'].fillna(TechSupport\_mode)

5. Provide a copy of the cleaned data set as a CSV file.

#Lets save the newly cleaned dataset now.

churn\_df.to\_csv("cleaned\_churn.csv")

6. Summarize the limitations of the data-cleaning process.

Data cleaning can be limiting in the fact that as you alter, reduce, or completely drop data, you will inevitably lose some potentially valuable data in the process. It is challenging to clean your data while maintaining 100 percent of the quality.

7. Discuss how the limitations summarized in part D6 could affect the analysis of the question or decision from part A.

The choice to use the mean for the numerical data and the mode for the categorical data could end up causing the data to be a little more skewed than it would have been otherwise. However, if we left them blank that could happen to a more drastic effect.

E. Apply principal component analysis (PCA) to identify the significant features of the data set by doing the following:

1. Identify the total number of principal components and provide the output of the principal components loading matrix.

There are 9 principal components. The output of the loadings matrix is included in the attached code

2. Justify the reduced number of the principal components and include a screenshot of a scree plot.

The most vital PCs are PC1, PC2 and PC3. These three have a eigenvalue greater than 1 and based on the Kaiser rule, the ones that are greater than or equal to 1 are the ones that carry the most weight.

3. Describe how the organization would benefit from the use of PCA.

The organization (or all organizations really) would benefit by knowing what specific factors carry the most weight and which ones do not.

**Part IV. Supporting Documents**

F. Provide a Panopto video recording that includes the presenter and a vocalized demonstration of the functionality of the code used for the analysis of the programming environment.

*Note: For instructions on how to access and use Panopto, use the "Panopto How-To Videos" web link provided below. To access Panopto's website, navigate to the web link titled "Panopto Access," and then choose to log in using the “WGU” option. If prompted, log in using your WGU student portal credentials, and then it will forward you to Panopto’s website.*

*To submit your recording, upload it to the Panopto drop box titled “Data Cleaning NUM3 | D206 (Student Creators) [assignments].” Once the recording has been uploaded and processed in Panopto's system, retrieve the URL of the recording from Panopto and copy and paste it into the Links option. Upload the remaining task requirements using the Attachments option.*

G. Acknowledge web sources, using in-text citations and references, for segments of third-party code used to support the application. Be sure the web sources are reliable.

*Panopto*, wgu.hosted.panopto.com/Panopto/Pages/Viewer.aspx?id=a371ff38-6c4f-403a-8018-b07d00ee8ba3&query=d206+pca. Accessed 14 Feb. 2024.

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

I. Demonstrate professional communication in the content and presentation of your submission.