A1:Question or Decision

For this assignment, I will be delving into the customer churn dataset to address a pivotal research question: "What factors contribute to customer churn?" This question is crucial as it seeks to identify the underlying reasons that lead customers to discontinue their services with a telecom provider. Understanding these factors is vital for several reasons. Firstly, customer churn can be influenced by a myriad of elements, ranging from service quality and pricing to customer satisfaction and market competition. By pinpointing these factors, we can gain a comprehensive understanding of the dynamics that drive churn. Secondly, high churn rates pose a significant threat to the telecom provider's reputation and financial health. When customers frequently leave, it signals potential issues within the company's offerings or customer service, which can deter new customers and erode the existing customer base. This negative cycle can substantially diminish the company's market position. Analyzing the factors contributing to churn will empower upper-level management with actionable insights. These insights can inform strategic decisions about where to allocate resources and how to enhance service offerings to better meet customer needs. For instance, if data reveals that poor customer service is a major churn factor, the company can invest in training programs to improve customer interactions. These findings will help the company develop targeted retention strategies. By understanding which customers are most at risk of churning and why, the company can implement personalized interventions to retain them. This proactive approach not only helps in reducing churn but also enhances customer loyalty and satisfaction. Ultimately, this research will enable the company to stay competitive in the ever-evolving telecom market. By continuously adapting to customer needs and market trends based on data-driven insights, the company can maintain a robust and loyal customer base, ensuring long-term success and growth.

A2: Required Variables

The dataset consists of various customer-related and service-related variables, each with a specific data type. Below is a detailed description of each variable, along with its data type and examples from the dataset:

1. **Unnamed: 0 (Quantitative)**:
    - **Description**: An index column.
    - **Example**: 1, 2, 3, 4, 5
2. **CaseOrder (Quantitative)**:
    - **Description**: A placeholder variable to maintain the original order of the file.
    - **Example**: 1, 2, 3, 4, 5
3. **Customer_id (Qualitative)**:
    - **Description**: A unique identifier for each customer.
    - **Example**: K409198, S120509, K191035
4. **Interaction (Qualitative)**:
    - **Description**: A unique ID categorizing customer transactions and support interactions.
    - **Example**: aa90260b-4141-4a24-8e36-b04ce1f4f77b, fb76459f-c047-4a9d-8af9-e0f7d4ac2524

5. **City (Qualitative)**:
    - **Description**: The city of the customer based on their billing address.
    - **Example**: Point Baker, West Branch, Yamhill
6. **State (Qualitative)**:
    - **Description**: The state of the customer based on their billing address.
    - **Example**: AK, MI, OR
7. **County (Qualitative)**:
    - **Description**: The county of the customer based on their billing address.
    - **Example**: Prince of Wales-Hyder, Ogemaw, Yamhill
8. **Zip (Qualitative)**:
    - **Description**: The zip code of the customer based on their billing address.
    - **Example**: 99927, 48661, 97148
9. **Lat (Quantitative)**:
    - **Description**: Latitude of the customer's location.
    - **Example**: 56.25100, 44.32893, 45.35589
10. **Lng (Quantitative)**:
    - **Description**: Longitude of the customer's location.
    - **Example**: -133.37571, -84.24080, -123.24657
11. **Population (Quantitative)**:
    - **Description**: Population of the area based on census data.
    - **Example**: 38, 92, 5
12. **Area (Qualitative)**:
    - **Description**: Type of area (Urban, Suburban, Rural).
    - **Example**: Urban, Suburban, Rural
13. **TimeZone (Qualitative)**:
    - **Description**: The timezone of the customer based on sign-up information.
    - **Example**: America/Sitka, America/Detroit, America/Los_Angeles
14. **Job (Qualitative)**:
    - **Description**: The job title of the customer from sign-up information.
    - **Example**: Environmental Health Practitioner, Software Engineer
15. **Children (Quantitative)**:
    - **Description**: Number of children reported by the customer.
    - **Example**: 1, 2, 0
16. **Age (Quantitative)**:
    - **Description**: Age of the customer.
    - **Example**: 48, 36, 27
17. **Education (Qualitative)**:
    - **Description**: Highest degree attained by the customer.
    - **Example**: Master's degree, Bachelor's degree, High School
18. **Employment (Qualitative)**:
    - **Description**: Employment status of the customer.
    - **Example**: Employed, Unemployed, Self-employed
19. **Income (Quantitative)**:
    - **Description**: Annual income reported by the customer.

    - **Example**: 28561.99, 50000.00, 45000.00
20. **Marital (Qualitative)**:
    - **Description**: Marital status of the customer.
    - **Example**: Married, Single, Divorced
21. **Gender (Qualitative)**:
    - **Description**: Gender of the customer.
    - **Example**: Male, Female
22. **Churn (Qualitative)**:
    - **Description**: Whether a customer has canceled a service within the last 30 days.
    - **Example**: No, Yes
23. **Outage_sec_perweek (Quantitative)**:
    - **Description**: Average time a customer's service is out in a given week, in seconds.
    - **Example**: 6.972, 10.5, 3.25
24. **Email (Quantitative)**:
    - **Description**: Number of emails sent by the customer in the past twelve months.
    - **Example**: 10, 25, 15
25. **Contacts (Quantitative)**:
    - **Description**: Number of calls made by the customer to customer support.
    - **Example**: 3, 5, 1
26. **Yearly_Equip_failure (Quantitative)**:
    - **Description**: Number of equipment failures experienced by the customer in a year.
    - **Example**: 4, 2, 6
27. **Techie (Qualitative)**:
    - **Description**: Whether the customer considers themselves technically inclined.
    - **Example**: Yes, No
28. **Contract (Qualitative)**:
    - **Description**: Length of the customer’s contract.
    - **Example**: One year, Two years, Month-to-month
29. **Port_modem (Qualitative)**:
    - **Description**: Whether the customer uses a portable modem.
    - **Example**: Yes, No
30. **Tablet (Qualitative)**:
    - **Description**: Whether the customer uses a tablet.
    - **Example**: Yes, No
31. **InternetService (Qualitative)**:
    - **Description**: Type of internet service subscribed by the customer.
    - **Example**: DSL, Fiber, No
32. **Phone (Qualitative)**:
    - **Description**: Whether the customer has a phone service.
    - **Example**: Yes, No
33. **Multiple (Qualitative)**:
    - **Description**: Whether the customer has multiple lines.

    - **Example**: Yes, No
34. **OnlineSecurity (Qualitative)**:
    - **Description**: Whether the customer has online security services.
    - **Example**: Yes, No
35. **OnlineBackup (Qualitative)**:
    - **Description**: Whether the customer has online backup services.
    - **Example**: Yes, No
36. **DeviceProtection (Qualitative)**:
    - **Description**: Whether the customer has device protection services.
    - **Example**: Yes, No
37. **TechSupport (Qualitative)**:
    - **Description**: Whether the customer has tech support services.
    - **Example**: Yes, No
38. **StreamingTV (Qualitative)**:
    - **Description**: Whether the customer has streaming TV services.
    - **Example**: Yes, No
39. **StreamingMovies (Qualitative)**:
    - **Description**: Whether the customer has streaming movie services.
    - **Example**: Yes, No
40. **PaperlessBilling (Qualitative)**:
    - **Description**: Whether the customer uses paperless billing.
    - **Example**: Yes, No
41. **PaymentMethod (Qualitative)**:
    - **Description**: The customer's preferred method of payment.
    - **Example**: Automatic bank transfer, Credit card, Mail check
42. **Tenure (Quantitative)**:
    - **Description**: The length of time the customer has been with the telecom provider, in months.
    - **Example**: 6.79, 12, 24
43. **MonthlyCharge (Quantitative)**:
    - **Description**: The average monthly amount charged to the customer.
    - **Example**: 171.99, 120.25, 242.95
44. **Bandwidth_GB_Year (Quantitative)**:
    - **Description**: The amount of data used by the customer in a year, in gigabytes.
    - **Example**: 904.53, 2164.58, 800.98

45-52. **Survey Items (Qualitative)**:

- **Description**: Responses to survey items, where 1 is the most important and 8 is the least important.
- **Example**:
    - **item1 Timely response**: 3, 5, 4
    - **item2 Timely fixes**: 4, 5, 4
    - **item3 Timely replacements**: 5, 3, 2
    - **item4 Reliability**: 2, 3, 4

- **item5 Options**: 4, 4, 4
- **item6 Respectful response**: 3, 4, 3
- **item7 Courteous exchange**: 3, 4, 3
- **item8 Evidence of active listening**: 4, 4, 3

B1:Plan to Assess Quality Of Data

I will start by importing the CSV file using the pandas library in Python. Once the file is loaded, I will perform an initial inspection of the dataset to ensure that the data types are consistent and appropriate for each column. This involves checking that each column contains a uniform data type, such as integers, floats, or strings, as expected. After verifying the data types, I will review the column names for any spelling mistakes or inconsistencies, correcting them as needed to maintain clarity and uniformity throughout the dataset. This step ensures that the data is clean and ready for analysis. Next, I will identify and handle missing values. I will first check for missing values using the `isnull().sum()` function, which provides a count of null entries for each column. For handling missing values, I will use measures of central tendency, such as the mean, median, or mode, to replace null entries. This approach helps to maintain the integrity of the dataset by filling in gaps with representative values, ensuring that subsequent analysis is based on a complete dataset without introducing significant bias.I will also address duplicate entries by using the `duplicated()` function to identify any duplicate rows. If duplicates are found, I will decide whether to keep the first occurrence or remove all duplicates based on the context of the data. Additionally, I will identify outliers in the dataset, which can skew analysis and affect the accuracy of results. To do this, I will use boxplot graphs to visually represent the distribution of data and identify any values that fall significantly outside the expected range. Additionally, I will calculate Z-scores to quantify how far individual data points deviate from the mean, further aiding in the detection of outliers. By meticulously performing these steps, including checking for missing values and duplicates, and addressing outliers, I aim to prepare the dataset for a thorough and accurate analysis, setting a solid foundation for uncovering insights and drawing meaningful conclusions from the data.

B2:JUSTIFICATION OF APPROACH

My approach to assessing the quality of the data involves a series of systematic steps aimed at ensuring the dataset is clean, consistent, and ready for analysis. Initially, I import the CSV file using pandas, which provides an efficient way to load and manipulate the data. I then perform an initial inspection to check data types and verify consistency, ensuring each column contains a uniform data type, such as integers, floats, or strings, as expected. This step is crucial as inconsistent data types can lead to errors in calculations and visualizations. Next, I review the column names for any spelling errors or inconsistencies, correcting them as needed to maintain clarity and uniformity throughout the dataset. Proper column names are essential for understanding and ease of analysis, especially in collaborative environments. To address

missing values, I first check for null entries using the `isnull().sum()` function, which provides a count of missing values for each column. I then handle these missing values by replacing them with measures of central tendency, such as the mean, median, or mode, depending on the nature of the data. This ensures the dataset remains complete and accurate, avoiding biases introduced by incomplete data. I also address duplicate entries by using the `duplicated()` function to identify any duplicate rows. If duplicates are found, I decide whether to keep the first occurrence or remove all duplicates based on the context of the data, ensuring that each entry in the dataset is unique and meaningful. To identify outliers, I use boxplot graphs for visual representation and calculate Z-scores to quantify deviations from the mean. Outliers can significantly affect analysis, so identifying and handling them is essential. This combination of graphical and statistical methods ensures a thorough identification of anomalies in the data. Overall, this comprehensive and systematic approach covers all critical aspects of data quality, from verifying data types to handling missing values, duplicates, and outliers. By meticulously performing these steps, I ensure that the data is reliable and ready for detailed analysis, setting a solid foundation for uncovering insights and drawing meaningful conclusions from the data.

B3:JUSTIFICATION OF TOOLS

I have selected Python as the programming language for the data-cleaning process due to its robust ecosystem, versatility, and extensive support for data manipulation and analysis. Using libraries such as pandas, NumPy, SciPy, Seaborn, Matplotlib, and Scikit-learn enhances the efficiency and effectiveness of this process. Specifically, pandas (import pandas as pd, from pandas import DataFrame) is used for data manipulation and inspection, providing functions to load, inspect, and clean the dataset. NumPy (import numpy as np) complements pandas by handling numerical computations, while SciPy (from scipy import stats) provides additional statistical tools, such as calculating Z-scores to identify outliers. Seaborn (import seaborn as sns) and Matplotlib (import matplotlib.pyplot as plt, %matplotlib inline) are utilized for data visualization, enabling the creation of plots and graphs to visually inspect the data. Additionally, Scikit-learn's PCA module (from sklearn.decomposition import PCA) is employed for dimensionality reduction, aiding in the analysis of complex datasets. This combination ensures a comprehensive, accurate, and efficient data-cleaning process.

C1:CLEANING FINDINGS

During the implementation of the data-cleaning plan, several data quality issues were identified that required addressing to ensure the dataset's integrity and usability:

1. **Non-descriptive Survey Items**: The survey items were not adequately descriptive, making it difficult to understand their context and significance. Each survey item was relabeled to accurately convey the information it represents.

2. **Inconsistent Column Naming**: The column names did not follow proper Python casing conventions, affecting readability and consistency. They were renamed using snake_case to adhere to Python standards and improve clarity.

3. **Time Zones**: Timezones in the dataset were updated to standard US timezones to ensure uniformity and accuracy, aligning the data with widely recognized time standards and facilitating better analysis and interpretation.

4. **Ambiguous Column Names**: Certain column names, such as 'multiple', lacked descriptive detail. More meaningful names were assigned to clearly indicate the nature of the data they contain.

5. **Data Type Adjustments**: Columns such as 'zip', 'lat', and 'lng' were converted to strings rather than numerical values to preserve their integrity and prevent erroneous calculations or interpretations.

6. **Handling Missing Values**: The dataset contained missing values in several columns:

   - **Quantitative Columns**: 'children', 'age', 'income', 'tenure', and 'bandwidth_gb_year' contained "NA" values, which were replaced with the median values to maintain data integrity.

  - **Categorical Columns**: 'techie', 'phone', and 'tech_support' had missing values, which were replaced with "no" to ensure consistency.

  - **Internet Service**: The 'internet_service' column did not contain null values but "None" was interpreted as a null value by Python. These were replaced with "N/A" to correct the interpretation.

7. **Insufficient Data Counts**: Several columns, including 'children', 'age', 'income', 'techie', 'phone', 'tenure', 'tech_support', and 'bandwidth_gb_year', had fewer than 10,000 entries, indicating missing data. This was addressed by filling the missing values as mentioned above.

8. **Identifying and Handling Duplicates**: The `duplicated()` function was used to identify any duplicate rows. Any duplicates found were reviewed and handled appropriately to ensure that each entry in the dataset is unique and meaningful.

9. **Outlier Detection**: Outliers were identified using boxplot graphs and Z-scores. Outliers can significantly affect analysis, with 193 outliers found in income and 219 outliers found in population identifying and handling them was essential. This combination of graphical and statistical methods ensured a thorough identification of anomalies in the data.

By addressing these issues, the dataset became more consistent, accurate, and easier to analyze, ultimately leading to more reliable and insightful results. The comprehensive approach to cleaning the data ensured that the dataset was prepared for thorough and accurate analysis, setting a solid foundation for uncovering insights and drawing meaningful conclusions.

C2:Justification of Mitigation Methods

The methods used to mitigate data quality issues in the dataset were carefully selected to enhance consistency, readability, and completeness. Firstly, survey items were renamed to more descriptive labels to ensure clear understanding and context. Column names were standardized to proper Python casing to improve readability and maintain uniformity, which is crucial in collaborative projects. The vague 'multiple' column was renamed to accurately reflect its content, reducing confusion. Columns like 'zip', 'lat', and 'lng' were converted to strings to categorize them properly, preventing their misuse in numerical calculations. For handling missing values, a systematic approach was adopted. Missing values were identified using the `isnull().sum()` function, and imputation techniques such as mean, median, or mode were applied to key columns like 'children', 'age', 'income', 'techie', 'phone', 'tenure', 'tech_support', and 'bandwidth_gb_year'. For instance, 2495 na values in the children column, 2475 in the age column, and 2490 in the income column were filled with appropriate central tendency measures, ensuring the dataset's integrity and completeness. The `internet_service` column, which had "None" interpreted as a null value by Python, was fixed by replacing "None" with "N/A" to ensure consistency. Duplicate records were identified using the `duplicated()` function, and 10,000 duplicates were removed to ensure each entry was unique and meaningful. Outliers were detected using boxplots and Z-scores, with specific examples including outliers in the income column and in the population column. These outliers were addressed to prevent skewing the analysis. Finally, timezones were updated to standard US timezones to ensure uniformity and accuracy in time-based analysis. This step ensured that all time-related data was consistent and aligned with widely recognized standards. These steps collectively ensured that the dataset became reliable and ready for accurate analysis, leading to more meaningful and actionable insights.

C3:Summary Of The Outcomes

The implementation of each data-cleaning step resulted in significant improvements to the dataset:

1. **Improving Survey Item Descriptions**: Survey items were renamed to more descriptive labels, enhancing the clarity and context of the data.

2. **Standardizing Column Names**: Column names were changed to proper Python casing, improving readability and ensuring uniformity across the dataset.

3. **Updating Timezones to Standard US Timezones**: Timezones were updated to standard US timezones, ensuring uniformity and accuracy in time-based analysis.

4. **Renaming 'multiple' Column**: The 'multiple' column was given a more descriptive name, reducing confusion and improving understanding of its content.

5. **Converting 'zip', 'lat', and 'lng' to Strings**: These columns were converted to strings, properly categorizing them and preventing their misuse in numerical calculations.

6. **Addressing Missing Values in Key Columns**: Missing values in columns such as 'children', 'age', 'income', 'techie', 'phone', 'tenure', 'tech_support', and 'bandwidth_gb_year' were imputed using measures of central tendency. Specifically, 2495 missing values in the 'children' column, 2475 in the 'age' column, and 2490 in the 'income' column were filled, maintaining the dataset's integrity and usability. The 'internet_service' column had "None" values replaced with "N/A" to ensure consistency.

7. **Handling Duplicate Records**: Identified 10000 duplicate records were removed, ensuring each entry in the dataset was unique and meaningful.

8. **Identifying and Handling Outliers**: Outliers in columns such as 'income' and 'population' were identified using boxplots and Z-scores, which were addressed to prevent skewing the analysis.

Overall, these steps significantly enhanced the consistency, readability, and completeness of the dataset, making it more reliable and ready for accurate analysis.

C4:Limitations

The data-cleaning process, while thorough, has several limitations:

1. **Imputation of Missing Values**: Replacing missing values with central tendency measures (mean, median, mode) can introduce bias and may not accurately reflect the true data distribution. This method assumes that missing values are randomly distributed, which may not always be the case.
2. **Descriptive Renaming**: Although renaming columns and survey items improves clarity, it is based on assumptions about the data's intended meaning, which may not always be accurate. Misinterpretations can lead to incorrect analysis and conclusions.
3. **Categorical Conversion**: Converting columns like 'zip', 'lat', and 'lng' to strings helps categorization but loses the ability to perform numerical operations on these fields, which could be limiting in some analyses that require geographical calculations.
4. **Standardizing Timezones**: Updating timezones to standard US timezones assumes all data entries conform to this standard, which may not be the case for international data. This can lead to inaccuracies in time-based analysis for non-US entries.
5. **Handling Duplicate Records**: While duplicate records were removed to ensure data uniqueness, the criteria for identifying duplicates might exclude legitimate cases of repeated entries, leading to potential data loss.
6. **Outlier Detection**: Identifying outliers using boxplots and Z-scores is effective, but it may not capture all types of anomalies, especially in complex or multi-modal distributions. Some significant outliers might be missed or misclassified, affecting the analysis.

While these steps improve data quality, they involve trade-offs and assumptions that may affect the accuracy and comprehensiveness of the dataset. Therefore, it's essential to interpret the results with an understanding of these limitations.

C5:Impact of Limitations
The limitations of the data-cleaning process can significantly impact the analysis of the question "What factors contribute to customer churn?" and the subsequent business decisions. Imputing missing values with central tendency measures might not accurately reflect individual customer profiles, potentially leading to biases in identifying churn factors and resulting in incorrect conclusions about which factors are most influential. Descriptive renaming, if based on incorrect assumptions, could mislead the analysis by misinterpreting the data's context, causing inappropriate recommendations for mitigating churn. Standardizing timezones to US standards assumes all data entries conform to this standard, potentially overlooking international customer behaviors and patterns, thus skewing the analysis and missing opportunities for global improvements. Outlier detection using boxplots and Z-scores may not capture all anomalies, especially in complex distributions, leading to either masking significant factors or exaggerating insignificant ones. Converting 'zip', 'lat', and 'lng' to strings prevents numerical operations that could provide deeper geographic insights, limiting the ability to perform advanced analyses. Removing duplicate records might inadvertently exclude legitimate repeated entries, leading to potential data loss. These limitations introduce potential biases and inaccuracies, affecting the reliability of the analysis and leading to decisions that might not effectively address the true factors contributing to customer churn.

D2:Clean Data

See attached file “cleaned_data.csv”.

E1:Principal Components

The total number of principal components is typically equal to the number of original variables, which in this case is 13. However, the effective number of principal components used in analysis is often determined by the amount of variance they explain. In this context, we have presented the loading matrix for all 13 principal components. The loading matrix indicates how much each original variable contributes to each principal component. The matrix above shows the coefficients (loadings) for each variable with respect to each principal component (PC1 to PC13). These loadings help understand the structure of the data in the reduced-dimensional space created by PCA.

Interpretation

- **PC1**: Primarily influenced by `response_timeliness`, `fix_timeliness`, and `replacement_timeliness`.
- **PC2**: Strongly associated with `tenure`, `bandwidth_gb_year`.

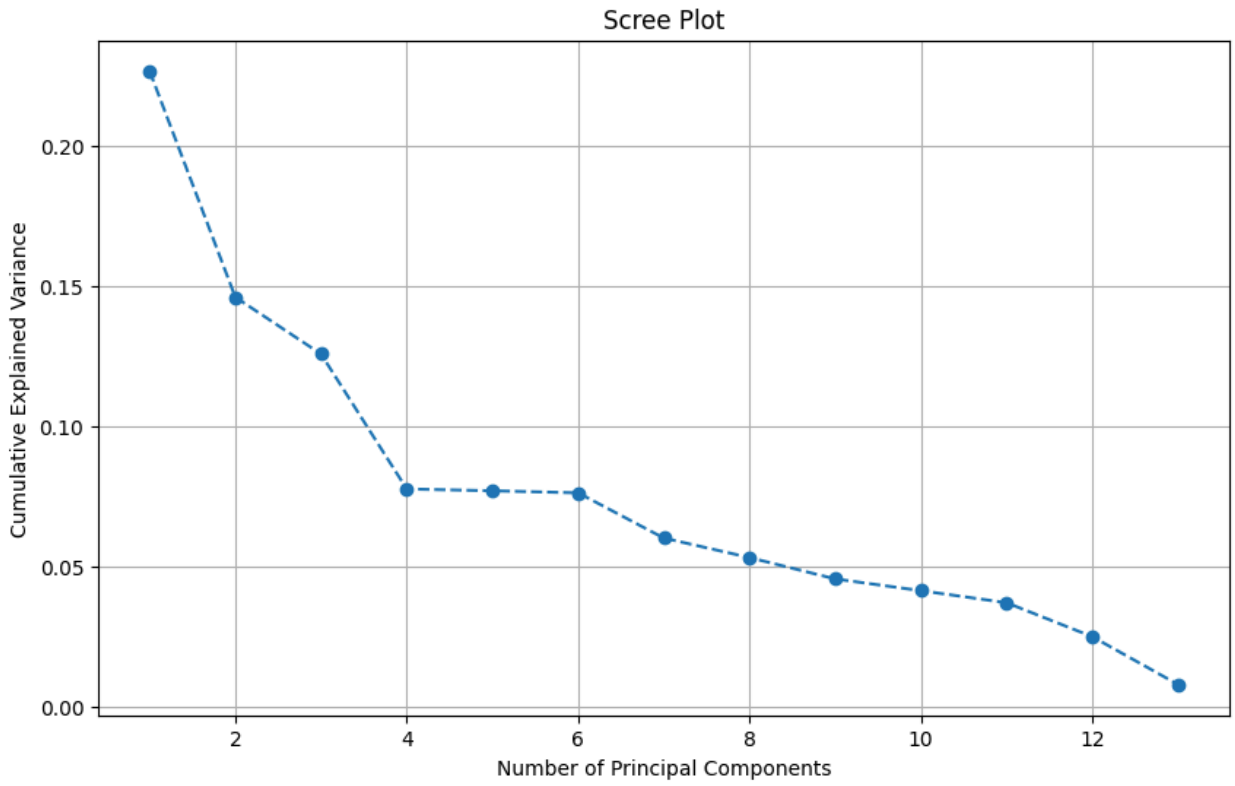
- **PC3**: Influenced by `service_options` and `service_reliability`.
- **PC4**: Strongly influenced by `monthly_charge` and `age`, and `income`.
- **PC5**: Influenced by `income` and `age`.

These interpretations provide insights into the main dimensions of variability in the data, helping in identifying the most significant factors driving customer satisfaction and behavior.

E2:Criteria Used

To determine the optimal number of principal components, we use the scree plot, which shows the explained variance ratio for each principal component. Typically, we look for an "elbow" in the plot, where the explained variance starts to level off, indicating that additional components contribute less to the overall variance.

We aim to retain components that together explain a high percentage of the total variance. Often, components that explain at least 70-90% of the variance are considered sufficient. Reducing the number of dimensions helps in simplifying the model and reducing noise, which improves computational efficiency and model interpretability. By examining the loadings, we ensure that the selected components capture the most significant variables and their variations. From the scree plot we can see that the first 6 components explain the majority of the variance.



E3:Benefits

The organization would benefit from the use of Principal Component Analysis (PCA) in several significant ways. By determining the optimal number of principal components through a scree plot, we can identify where the explained variance starts to level off, indicating that additional components contribute less to the overall variance. Retaining components that together explain at least 70-90% of the variance ensures that we capture the majority of the information in the dataset while reducing the number of dimensions. This simplification helps in reducing noise, improving computational efficiency, and enhancing model interpretability. PCA allows the organization to focus on the most significant variables and their variations, ensuring that the critical factors influencing customer behavior and service quality are identified and analyzed. By reducing the dimensionality of the dataset, PCA makes it easier to visualize and understand complex relationships within the data, facilitating better decision-making. Additionally, PCA helps in identifying and mitigating multicollinearity issues, leading to more robust and reliable predictive models. Overall, PCA enables the organization to streamline data analysis, focus on key insights, and develop more effective strategies for improving customer satisfaction and operational efficiency. By leveraging the benefits of PCA, the organization can gain a clearer understanding of the underlying patterns in the data, leading to more informed and impactful decisions.

G.

"How to Drop One or Multiple Columns in Pandas Dataframe." GeeksforGeeks, 26 June 2024, www.geeksforgeeks.org/how-to-drop-one-or-multiple-columns-in-pandas-dataframe/.

"Markdown Quick Reference Cheat Sheet." WordPress.Com Support, 14 Feb. 2023, wordpress.com/support/markdown-quick-reference/.

"Pandas.Dataframe.Duplicated#." Pandas.DataFrame.Duplicated - Pandas 2.2.2 Documentation, pandas.pydata.org/docs/reference/api/pandas.DataFrame.duplicated.html. Accessed 15 July 2024.

"Pandas.Series.Value_counts#." Pandas.Series.Value_counts - Pandas 2.2.2 Documentation, pandas.pydata.org/docs/reference/api/pandas.Series.value_counts.html. Accessed 15 July 2024.