In [ ]:
```
NAME: MANE SHIVRAJ PANDURANG
COURSE: CL I
CLASS: BE AI&DS
```

In [ ]:
```
# Data Cleaning and Preparation
# Problem Statement: Analyzing Customer Churn in a Telecommunications Company
# Dataset: "Telecom_Customer_Churn.csv"
# Description: The dataset contains information about customers of a telecommunicat
#services). The datasetincludes various attributes of the customers, such as their
# account information.The goal is to perform data cleaning and preparation to gain
# Tasks to Perform:
# 1. Import the "Telecom_Customer_Churn.csv" dataset.
# 2. Explore the dataset to understand its structure and content.
# 3. Handle missing values in the dataset, deciding on an appropriate strategy.
# 4. Remove any duplicate records from the dataset.
# 5. Check for inconsistent data, such as inconsistent formatting or spelling varia
# 6. Convert columns to the correct data types as needed.
# 7. Identify and handle outliers in the data.
# 8. Perform feature engineering, creating new features that may be relevant to pre
# 9. Normalize or scale the data if necessary.
# 10. Split the dataset into training and testing sets for further analysis.
# 11. Export the cleaned dataset for future analysis or modeling.
```

In [39]:
```
import pandas as pd
import numpy as np
from scipy import stats
from sklearn.preprocessing import MinMaxScaler, StandardScaler
from sklearn.model_selection import train_test_split
```

In [6]:
```
# 1. Import the "Telecom_Customer_Churn.csv" dataset.
df = pd.read_csv("D:\DMV\dataset's\Telecom_customer_churn.csv")
```

In [30]:
```
#2. Explore the dataset to understand its structure and content.
print('OUTPUT=')
print(df.head())
# Get a summary of the dataset
print(df.info())

# Check for missing values
print(df.isnull().sum())
```

```
OUTPUT=
   customerID  gender  SeniorCitizen Partner Dependents  tenure PhoneService  \
0  7590-VHVEG  Female              0     Yes         No       1           No   
1  5575-GNVDE    Male              0      No         No      34          Yes   
2  3668-QPYBK    Male              0      No         No       2          Yes   
3  7795-CFOCW    Male              0      No         No      45           No   
4  9237-HQITU  Female              0      No         No       2          Yes   

      MultipleLines InternetService OnlineSecurity  ... DeviceProtection  \
0  No phone service             DSL             No  ...               No   
1                No             DSL            Yes  ...              Yes   
2                No             DSL            Yes  ...               No   
3  No phone service             DSL            Yes  ...              Yes   
4                No     Fiber Optic             No  ...               No   

  TechSupport StreamingTV StreamingMovies Contract PaperlessBilling  \
0          No          No              No      NaT              Yes   
1          No          No              No      NaT               No   
2          No          No              No      NaT              Yes   
3         Yes          No              No      NaT               No   
4          No          No              No      NaT              Yes   

               PaymentMethod MonthlyCharges  TotalCharges  Churn  
0           Electronic check          29.85         29.85     No  
1               Mailed check          56.95       1889.50     No  
2               Mailed check          53.85        108.15    Yes  
3  Bank transfer (automatic)          42.30       1840.75     No  
4           Electronic check          70.70        151.65    Yes  

[5 rows x 21 columns]
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 7043 entries, 0 to 7042
Data columns (total 21 columns):
 #   Column            Non-Null Count  Dtype         
---  ------            --------------  -----         
 0   customerID        7043 non-null   object        
 1   gender            7043 non-null   object        
 2   SeniorCitizen     7043 non-null   int64         
 3   Partner           7043 non-null   object        
 4   Dependents        7043 non-null   object        
 5   tenure            7043 non-null   int64         
 6   PhoneService      7043 non-null   object        
 7   MultipleLines     7043 non-null   object        
 8   InternetService   7043 non-null   object        
 9   OnlineSecurity    7043 non-null   object        
 10  OnlineBackup      7043 non-null   object        
 11  DeviceProtection  7043 non-null   object        
 12  TechSupport       7043 non-null   object        
 13  StreamingTV       7043 non-null   object        
 14  StreamingMovies   7043 non-null   object        
 15  Contract          0 non-null      datetime64[ns]
 16  PaperlessBilling  7043 non-null   object        
 17  PaymentMethod     7043 non-null   object        
 18  MonthlyCharges    7043 non-null   float64       
 19  TotalCharges      7043 non-null   float64       
 20  Churn             7043 non-null   category      
```

```
dtypes: category(1), datetime64[ns](1), float64(2), int64(2), object(15)
memory usage: 1.1+ MB
None
customerID             0
gender                 0
SeniorCitizen          0
Partner                0
Dependents             0
tenure                 0
PhoneService           0
MultipleLines          0
InternetService        0
OnlineSecurity         0
OnlineBackup           0
DeviceProtection       0
TechSupport            0
StreamingTV            0
StreamingMovies        0
Contract            7043
PaperlessBilling       0
PaymentMethod          0
MonthlyCharges         0
TotalCharges           0
Churn                  0
dtype: int64
```

In [29]:

```python
#3. Handle missing values in the dataset, deciding on an appropriate strategy.

print('OUTPUT=')
# Check for missing values
print("Missing values before handling:")
print(df.isnull().sum())

# Fill missing values with the mean for numerical columns only
numeric_cols = df.select_dtypes(include=['number']).columns
df[numeric_cols] = df[numeric_cols].fillna(df[numeric_cols].mean())

# Fill missing values with the mode for categorical columns
for column in df.select_dtypes(include=['object']).columns:
    df[column].fillna(df[column].mode()[0], inplace=True)

# Verify the changes
print("Missing values after handling:")
print(df.isnull().sum())
```

```
OUTPUT=
Missing values before handling:
customerID             0
gender                 0
SeniorCitizen          0
Partner                0
Dependents             0
tenure                 0
PhoneService           0
MultipleLines          0
InternetService        0
OnlineSecurity         0
OnlineBackup           0
DeviceProtection       0
TechSupport            0
StreamingTV            0
StreamingMovies        0
Contract            7043
PaperlessBilling       0
PaymentMethod          0
MonthlyCharges         0
TotalCharges          11
Churn                  0
dtype: int64
Missing values after handling:
customerID             0
gender                 0
SeniorCitizen          0
Partner                0
Dependents             0
tenure                 0
PhoneService           0
MultipleLines          0
InternetService        0
OnlineSecurity         0
OnlineBackup           0
DeviceProtection       0
TechSupport            0
StreamingTV            0
StreamingMovies        0
Contract            7043
PaperlessBilling       0
PaymentMethod          0
MonthlyCharges         0
TotalCharges           0
Churn                  0
dtype: int64
```

In [28]:

```python
# 4. Remove any duplicate records from the dataset.
# Check for duplicate rows
print('OUTPUT=')
duplicate_rows = df.duplicated()
print(f"Number of duplicate rows: {duplicate_rows.sum()}")

# Remove duplicate rows
df.drop_duplicates(inplace=True)
```

```
# Verify that duplicates have been removed
duplicate_rows = df.duplicated()
print(f"Number of duplicate rows after removal: {duplicate_rows.sum()}")
```

```
OUTPUT=
Number of duplicate rows: 0
Number of duplicate rows after removal: 0
```

In [27]:
```
#5. Check for inconsistent data, such as inconsistent formatting or spelling variat
print('OUTPUT=')
df['gender'] = df['gender'].str.strip().str.lower().replace({'male': 'Male', 'femal

# Standardize 'InternetService' column
df['InternetService'] = df['InternetService'].str.strip().str.lower().replace({'dsl

# Convert 'TotalCharges' to numeric, handling errors
df['TotalCharges'] = pd.to_numeric(df['TotalCharges'], errors='coerce')

# Standardize 'Contract' column if it contains dates
df['Contract'] = pd.to_datetime(df['Contract'], errors='coerce')

# Verify the changes
print(df.head())
```

```
OUTPUT=
   customerID  gender  SeniorCitizen Partner Dependents  tenure PhoneService  \
0  7590-VHVEG  Female              0     Yes         No       1           No
1  5575-GNVDE    Male              0      No         No      34          Yes
2  3668-QPYBK    Male              0      No         No       2          Yes
3  7795-CFOCW    Male              0      No         No      45           No
4  9237-HQITU  Female              0      No         No       2          Yes

      MultipleLines InternetService OnlineSecurity  ... DeviceProtection  \
0  No phone service             DSL             No  ...               No
1                No             DSL            Yes  ...              Yes
2                No             DSL            Yes  ...               No
3  No phone service             DSL            Yes  ...              Yes
4                No     Fiber Optic             No  ...               No

  TechSupport StreamingTV StreamingMovies Contract PaperlessBilling  \
0          No          No              No      NaT              Yes
1          No          No              No      NaT               No
2          No          No              No      NaT              Yes
3         Yes          No              No      NaT               No
4          No          No              No      NaT              Yes

              PaymentMethod MonthlyCharges  TotalCharges  Churn
0          Electronic check          29.85         29.85     No
1              Mailed check          56.95       1889.50     No
2              Mailed check          53.85        108.15    Yes
3 Bank transfer (automatic)          42.30       1840.75     No
4          Electronic check          70.70        151.65    Yes

[5 rows x 21 columns]
```

In [26]:
```
# 6. Convert columns to the correct data types as needed.
print('OUTPUT=')
print("Data types before conversion:")
print(df.dtypes)

# Convert 'TotalCharges' to numeric, handling errors
df['TotalCharges'] = pd.to_numeric(df['TotalCharges'], errors='coerce')

# Convert 'Churn' to a categorical type
df['Churn'] = df['Churn'].astype('category')

# If there are any date columns, convert them to datetime
# Example: df['Contract'] = pd.to_datetime(df['Contract'], errors='coerce')

# Display updated data types
print("Data types after conversion:")
print(df.dtypes)
```

```
OUTPUT=
Data types before conversion:
customerID                  object
gender                      object
SeniorCitizen                int64
Partner                     object
Dependents                  object
tenure                       int64
PhoneService                object
MultipleLines               object
InternetService             object
OnlineSecurity              object
OnlineBackup                object
DeviceProtection            object
TechSupport                 object
StreamingTV                 object
StreamingMovies             object
Contract            datetime64[ns]
PaperlessBilling            object
PaymentMethod               object
MonthlyCharges             float64
TotalCharges               float64
Churn                     category
dtype: object
Data types after conversion:
customerID                  object
gender                      object
SeniorCitizen                int64
Partner                     object
Dependents                  object
tenure                       int64
PhoneService                object
MultipleLines               object
InternetService             object
OnlineSecurity              object
OnlineBackup                object
DeviceProtection            object
TechSupport                 object
StreamingTV                 object
StreamingMovies             object
Contract            datetime64[ns]
PaperlessBilling            object
PaymentMethod               object
MonthlyCharges             float64
TotalCharges               float64
Churn                     category
dtype: object
```

In [25]:

```
# 7. Identify and handle outliers in the data.

# Select only numeric columns
print('OUTPUT=')
numeric_cols = df.select_dtypes(include=[np.number])

# Calculate Q1 (25th percentile) and Q3 (75th percentile)
Q1 = numeric_cols.quantile(0.25)
```

```
Q3 = numeric_cols.quantile(0.75)
IQR = Q3 - Q1

# Identify outliers
outliers = ((numeric_cols < (Q1 - 1.5 * IQR)) | (numeric_cols > (Q3 + 1.5 * IQR))).
print(f"Number of outliers: {outliers.sum()}")

# Handle outliers by removing them
df_cleaned = df[~outliers]
print(f"Number of rows after removing outliers: {df_cleaned.shape[0]}")

# Handle outliers by capping them
df_capped = df.copy()
for col in numeric_cols.columns:
    Q1 = df[col].quantile(0.25)
    Q3 = df[col].quantile(0.75)
    IQR = Q3 - Q1
    lower_bound = Q1 - 1.5 * IQR
    upper_bound = Q3 + 1.5 * IQR
    df_capped[col] = np.where(df[col] < lower_bound, lower_bound, df[col])
    df_capped[col] = np.where(df[col] > upper_bound, upper_bound, df_capped[col])

# Verify the changes
print(df_cleaned.describe())
print(df_capped.describe())
```

```
OUTPUT
Number of outliers: 1142
Number of rows after removing outliers: 5901
       SeniorCitizen       tenure Contract  MonthlyCharges  TotalCharges
count         5901.0  5901.000000        0     5901.000000   5890.000000
mean             0.0    32.192171      NaT       61.847441   2181.089550
min              0.0     0.000000      NaT       18.250000     18.800000
25%              0.0     9.000000      NaT       25.600000    365.575000
50%              0.0    28.000000      NaT       65.800000   1295.775000
75%              0.0    55.000000      NaT       86.700000   3566.362500
max              0.0    72.000000      NaT      118.750000   8684.800000
std              0.0    24.628639      NaN       30.316041   2233.217848
       SeniorCitizen       tenure Contract  MonthlyCharges  TotalCharges
count         7043.0  7043.000000        0     7043.000000   7032.000000
mean             0.0    32.371149      NaT       64.761692   2283.300441
min              0.0     0.000000      NaT       18.250000     18.800000
25%              0.0     9.000000      NaT       35.500000    401.450000
50%              0.0    29.000000      NaT       70.350000   1397.475000
75%              0.0    55.000000      NaT       89.850000   3794.737500
max              0.0    72.000000      NaT      118.750000   8684.800000
std              0.0    24.559481      NaN       30.090047   2266.771362
```

In [31]:

```
# 8. Perform feature engineering, creating new features that may be relevant to pre
print('OUTPUT=')
df['TotalServices'] = df[['PhoneService', 'InternetService', 'OnlineSecurity', 'Onl

# Create 'AvgMonthlyCharges' feature
df['AvgMonthlyCharges'] = df['TotalCharges'] / df['tenure']

# Convert 'SeniorCitizen' to categorical
```

```python
df['SeniorCitizen'] = df['SeniorCitizen'].map({1: 'Yes', 0: 'No'})

# Create 'TenureGroup' feature
def tenure_group(tenure):
    if tenure <= 12:
        return '0-1 year'
    elif tenure <= 24:
        return '1-2 years'
    elif tenure <= 48:
        return '2-4 years'
    elif tenure <= 60:
        return '4-5 years'
    else:
        return '5+ years'

df['TenureGroup'] = df['tenure'].apply(tenure_group)

# Verify the new features
print(df.head())
```

```
OUTPUT=
   customerID  gender SeniorCitizen Partner Dependents  tenure PhoneService  \
0  7590-VHVEG  Female            No     Yes         No       1           No
1  5575-GNVDE    Male            No      No         No      34          Yes
2  3668-QPYBK    Male            No      No         No       2          Yes
3  7795-CFOCW    Male            No      No         No      45           No
4  9237-HQITU  Female            No      No         No       2          Yes

      MultipleLines InternetService OnlineSecurity  ... StreamingMovies  \
0  No phone service             DSL             No  ...              No
1                No             DSL            Yes  ...              No
2                No             DSL            Yes  ...              No
3  No phone service             DSL            Yes  ...              No
4                No     Fiber Optic             No  ...              No

  Contract PaperlessBilling              PaymentMethod MonthlyCharges  \
0      NaT              Yes           Electronic check          29.85
1      NaT               No               Mailed check          56.95
2      NaT              Yes               Mailed check          53.85
3      NaT               No  Bank transfer (automatic)          42.30
4      NaT              Yes           Electronic check          70.70

  TotalCharges Churn                TotalServices  AvgMonthlyCharges  \
0        29.85    No          NoDSLNoYesNoNoNoNo          29.850000
1      1889.50    No        YesDSLYesNoYesNoNoNo          55.573529
2       108.15   Yes        YesDSLYesYesNoNoNoNo          54.075000
3      1840.75    No        NoDSLYesNoYesYesNoNo          40.905556
4       151.65   Yes  YesFiber OpticNoNoNoNoNoNo          75.825000

   TenureGroup
0     0-1 year
1    2-4 years
2     0-1 year
3    2-4 years
4     0-1 year

[5 rows x 24 columns]
```

In [38]:
```python
# 9. Normalize or scale the data if necessary.
print('OUTPUT=')
# Replace infinite values with NaN
df.replace([np.inf, -np.inf], np.nan, inplace=True)

# Fill NaN values in numeric columns with the mean of the column
numeric_cols = df.select_dtypes(include=[np.number])
df[numeric_cols.columns] = numeric_cols.fillna(numeric_cols.mean())

# Min-Max Scaling
min_max_scaler = MinMaxScaler()
df_min_max_scaled = df.copy()
df_min_max_scaled[numeric_cols.columns] = min_max_scaler.fit_transform(numeric_cols

# Z-Score Scaling
standard_scaler = StandardScaler()
df_standard_scaled = df.copy()
df_standard_scaled[numeric_cols.columns] = standard_scaler.fit_transform(numeric_co
```

```
# Verify the changes
print("Min-Max Scaled Data:")
print(df_min_max_scaled.head())

print("\nZ-Score Scaled Data:")
print(df_standard_scaled.head())
```

```
OUTPUT=
Min-Max Scaled Data:
   customerID  gender SeniorCitizen Partner Dependents    tenure PhoneService  \
0  7590-VHVEG  Female            No     Yes         No  0.013889           No   
1  5575-GNVDE    Male            No      No         No  0.472222          Yes   
2  3668-QPYBK    Male            No      No         No  0.027778          Yes   
3  7795-CFOCW    Male            No      No         No  0.625000           No   
4  9237-HQITU  Female            No      No         No  0.027778          Yes   

      MultipleLines InternetService OnlineSecurity  ... StreamingMovies  \
0  No phone service             DSL             No  ...              No   
1                No             DSL            Yes  ...              No   
2                No             DSL            Yes  ...              No   
3  No phone service             DSL            Yes  ...              No   
4                No     Fiber Optic             No  ...              No   

  Contract PaperlessBilling              PaymentMethod MonthlyCharges  \
0      NaT              Yes           Electronic check       0.115423   
1      NaT               No               Mailed check       0.385075   
2      NaT              Yes               Mailed check       0.354229   
3      NaT               No  Bank transfer (automatic)       0.239303   
4      NaT              Yes           Electronic check       0.521891   

  TotalCharges Churn               TotalServices  AvgMonthlyCharges  \
0     0.001275    No          NoDSLNoYesNoNoNoNo           0.149361   
1     0.215867    No        YesDSLYesNoYesNoNoNo           0.388372   
2     0.010310   Yes        YesDSLYesYesNoNoNoNo           0.374448   
3     0.210241    No        NoDSLYesNoYesYesNoNo           0.252084   
4     0.015330   Yes  YesFiber OpticNoNoNoNoNoNo           0.576539   

   TenureGroup  
0     0-1 year  
1    2-4 years  
2     0-1 year  
3    2-4 years  
4     0-1 year  

[5 rows x 24 columns]

Z-Score Scaled Data:
   customerID  gender SeniorCitizen Partner Dependents    tenure PhoneService  \
0  7590-VHVEG  Female            No     Yes         No -1.277445           No   
1  5575-GNVDE    Male            No      No         No  0.066327          Yes   
2  3668-QPYBK    Male            No      No         No -1.236724          Yes   
3  7795-CFOCW    Male            No      No         No  0.514251           No   
4  9237-HQITU  Female            No      No         No -1.236724          Yes   

      MultipleLines InternetService OnlineSecurity  ... StreamingMovies  \
0  No phone service             DSL             No  ...              No   
1                No             DSL            Yes  ...              No   
2                No             DSL            Yes  ...              No   
3  No phone service             DSL            Yes  ...              No   
4                No     Fiber Optic             No  ...              No   

  Contract PaperlessBilling              PaymentMethod MonthlyCharges  \
0      NaT              Yes           Electronic check      -1.160323   
```

```
1      NaT              No              Mailed check      -0.259629
2      NaT             Yes              Mailed check      -0.362660
3      NaT              No  Bank transfer (automatic)      -0.746535
4      NaT             Yes          Electronic check       0.197365

  TotalCharges Churn               TotalServices  AvgMonthlyCharges  \
0    -0.994971    No          NoDSLNoYesNoNoNoNo          -1.158794
1    -0.173876    No        YesDSLYesNoYesNoNoNo          -0.305897
2    -0.960399   Yes        YesDSLYesYesNoNoNoNo          -0.355582
3    -0.195400    No        NoDSLYesNoYesYesNoNo          -0.792233
4    -0.941193   Yes  YesFiber OpticNoNoNoNoNoNo           0.365568

   TenureGroup
0     0-1 year
1    2-4 years
2     0-1 year
3    2-4 years
4     0-1 year

[5 rows x 24 columns]
```

In [41]:

```
# 10. Split the dataset into training and testing sets for further analysis.
print('OUTPUT=')
# Define features and target variable
X = df.drop('Churn', axis=1)  # Features
y = df['Churn']  # Target variable

# Split the dataset into training and testing sets
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_sta

# Verify the split
print(f"Training set size: {X_train.shape[0]}")
print(f"Testing set size: {X_test.shape[0]}")
```

```
OUTPUT=
Training set size: 5634
Testing set size: 1409
```

In [47]:

```
# 11. Export the cleaned dataset for future analysis or modeling.
print('OUTPUT=')
print("Dataset loaded successfully.")
print(df.head())

# Remove duplicates
df.drop_duplicates(inplace=True)
print("Duplicates removed.")
print(df.head())

# Replace infinite values with NaN
df.replace([np.inf, -np.inf], np.nan, inplace=True)
print("Infinite values replaced with NaN.")

# Fill NaN values in numeric columns with the mean of the column
numeric_cols = df.select_dtypes(include=[np.number])
df[numeric_cols.columns] = numeric_cols.fillna(numeric_cols.mean())
print("NaN values filled with column mean.")
```

```
print(df.head())

# Min-Max Scaling
min_max_scaler = MinMaxScaler()
df_min_max_scaled = df.copy()
df_min_max_scaled[numeric_cols.columns] = min_max_scaler.fit_transform(numeric_cols
print("Min-Max Scaling applied.")
print(df_min_max_scaled.head())

# Z-Score Scaling
standard_scaler = StandardScaler()
df_standard_scaled = df.copy()
df_standard_scaled[numeric_cols.columns] = standard_scaler.fit_transform(numeric_co
print("Z-Score Scaling applied.")
print(df_standard_scaled.head())

# Define features and target variable
X = df.drop('Churn', axis=1)  # Features
y = df['Churn']  # Target variable

# Split the dataset into training and testing sets
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_sta
print("Dataset split into training and testing sets.")
print(f"Training set size: {X_train.shape[0]}")
print(f"Testing set size: {X_test.shape[0]}")

# Export the cleaned dataset
df.to_csv('Telecom_Customer_Churn_Cleaned.csv', index=False)
print("Cleaned dataset exported successfully.")
```

```
OUTPUT=
Dataset loaded successfully.
   customerID  gender SeniorCitizen Partner Dependents  tenure PhoneService  \
0  7590-VHVEG  Female            No     Yes         No       1           No
1  5575-GNVDE    Male            No      No         No      34          Yes
2  3668-QPYBK    Male            No      No         No       2          Yes
3  7795-CFOCW    Male            No      No         No      45           No
4  9237-HQITU  Female            No      No         No       2          Yes

      MultipleLines InternetService OnlineSecurity  ... StreamingMovies  \
0  No phone service             DSL             No  ...              No
1                No             DSL            Yes  ...              No
2                No             DSL            Yes  ...              No
3  No phone service             DSL            Yes  ...              No
4                No     Fiber Optic             No  ...              No

  Contract PaperlessBilling              PaymentMethod MonthlyCharges  \
0      NaT              Yes           Electronic check          29.85
1      NaT               No               Mailed check          56.95
2      NaT              Yes               Mailed check          53.85
3      NaT               No  Bank transfer (automatic)          42.30
4      NaT              Yes           Electronic check          70.70

  TotalCharges Churn               TotalServices  AvgMonthlyCharges  \
0        29.85    No          NoDSLNoYesNoNoNoNo          29.850000
1      1889.50    No        YesDSLYesNoYesNoNoNo          55.573529
2       108.15   Yes        YesDSLYesYesNoNoNoNo          54.075000
3      1840.75    No        NoDSLYesNoYesYesNoNo          40.905556
4       151.65   Yes  YesFiber OpticNoNoNoNoNoNo          75.825000

   TenureGroup
0     0-1 year
1    2-4 years
2     0-1 year
3    2-4 years
4     0-1 year

[5 rows x 24 columns]
Duplicates removed.
   customerID  gender SeniorCitizen Partner Dependents  tenure PhoneService  \
0  7590-VHVEG  Female            No     Yes         No       1           No
1  5575-GNVDE    Male            No      No         No      34          Yes
2  3668-QPYBK    Male            No      No         No       2          Yes
3  7795-CFOCW    Male            No      No         No      45           No
4  9237-HQITU  Female            No      No         No       2          Yes

      MultipleLines InternetService OnlineSecurity  ... StreamingMovies  \
0  No phone service             DSL             No  ...              No
1                No             DSL            Yes  ...              No
2                No             DSL            Yes  ...              No
3  No phone service             DSL            Yes  ...              No
4                No     Fiber Optic             No  ...              No

  Contract PaperlessBilling              PaymentMethod MonthlyCharges  \
0      NaT              Yes           Electronic check          29.85
1      NaT               No               Mailed check          56.95
```

```
2      NaT              Yes            Mailed check          53.85
3      NaT               No  Bank transfer (automatic)       42.30
4      NaT              Yes          Electronic check        70.70

  TotalCharges Churn              TotalServices  AvgMonthlyCharges  \
0        29.85    No          NoDSLNoYesNoNoNoNo          29.850000
1      1889.50    No        YesDSLYesNoYesNoNoNo          55.573529
2       108.15   Yes        YesDSLYesYesNoNoNoNo          54.075000
3      1840.75    No        NoDSLYesNoYesYesNoNo          40.905556
4       151.65   Yes  YesFiber OpticNoNoNoNoNoNo          75.825000

   TenureGroup
0     0-1 year
1    2-4 years
2     0-1 year
3    2-4 years
4     0-1 year

[5 rows x 24 columns]
Infinite values replaced with NaN.
NaN values filled with column mean.
   customerID  gender SeniorCitizen Partner Dependents  tenure PhoneService  \
0  7590-VHVEG  Female            No     Yes         No       1           No
1  5575-GNVDE    Male            No      No         No      34          Yes
2  3668-QPYBK    Male            No      No         No       2          Yes
3  7795-CFOCW    Male            No      No         No      45           No
4  9237-HQITU  Female            No      No         No       2          Yes

      MultipleLines InternetService OnlineSecurity  ... StreamingMovies  \
0  No phone service            DSL             No  ...              No
1                No            DSL            Yes  ...              No
2                No            DSL            Yes  ...              No
3  No phone service            DSL            Yes  ...              No
4                No     Fiber Optic             No  ...              No

  Contract PaperlessBilling              PaymentMethod MonthlyCharges  \
0      NaT              Yes          Electronic check         29.85
1      NaT               No              Mailed check         56.95
2      NaT              Yes              Mailed check         53.85
3      NaT               No  Bank transfer (automatic)        42.30
4      NaT              Yes          Electronic check         70.70

  TotalCharges Churn              TotalServices  AvgMonthlyCharges  \
0        29.85    No          NoDSLNoYesNoNoNoNo          29.850000
1      1889.50    No        YesDSLYesNoYesNoNoNo          55.573529
2       108.15   Yes        YesDSLYesYesNoNoNoNo          54.075000
3      1840.75    No        NoDSLYesNoYesYesNoNo          40.905556
4       151.65   Yes  YesFiber OpticNoNoNoNoNoNo          75.825000

   TenureGroup
0     0-1 year
1    2-4 years
2     0-1 year
3    2-4 years
4     0-1 year
```

```
[5 rows x 24 columns]
Min-Max Scaling applied.
   customerID  gender SeniorCitizen Partner Dependents    tenure PhoneService  \
0  7590-VHVEG  Female            No     Yes         No  0.013889           No
1  5575-GNVDE    Male            No      No         No  0.472222          Yes
2  3668-QPYBK    Male            No      No         No  0.027778          Yes
3  7795-CFOCW    Male            No      No         No  0.625000           No
4  9237-HQITU  Female            No      No         No  0.027778          Yes

      MultipleLines InternetService OnlineSecurity  ... StreamingMovies  \
0  No phone service             DSL             No  ...              No
1                No             DSL            Yes  ...              No
2                No             DSL            Yes  ...              No
3  No phone service             DSL            Yes  ...              No
4                No     Fiber Optic             No  ...              No

  Contract PaperlessBilling              PaymentMethod MonthlyCharges  \
0      NaT              Yes           Electronic check       0.115423
1      NaT               No               Mailed check       0.385075
2      NaT              Yes               Mailed check       0.354229
3      NaT               No  Bank transfer (automatic)       0.239303
4      NaT              Yes           Electronic check       0.521891

  TotalCharges Churn               TotalServices  AvgMonthlyCharges  \
0     0.001275    No          NoDSLNoYesNoNoNoNo           0.149361
1     0.215867    No        YesDSLYesNoYesNoNoNo           0.388372
2     0.010310   Yes        YesDSLYesYesNoNoNoNo           0.374448
3     0.210241    No        NoDSLYesNoYesYesNoNo           0.252084
4     0.015330   Yes  YesFiber OpticNoNoNoNoNoNo           0.576539

   TenureGroup
0     0-1 year
1    2-4 years
2     0-1 year
3    2-4 years
4     0-1 year

[5 rows x 24 columns]
Z-Score Scaling applied.
   customerID  gender SeniorCitizen Partner Dependents    tenure PhoneService  \
0  7590-VHVEG  Female            No     Yes         No -1.277445           No
1  5575-GNVDE    Male            No      No         No  0.066327          Yes
2  3668-QPYBK    Male            No      No         No -1.236724          Yes
3  7795-CFOCW    Male            No      No         No  0.514251           No
4  9237-HQITU  Female            No      No         No -1.236724          Yes

      MultipleLines InternetService OnlineSecurity  ... StreamingMovies  \
0  No phone service             DSL             No  ...              No
1                No             DSL            Yes  ...              No
2                No             DSL            Yes  ...              No
3  No phone service             DSL            Yes  ...              No
4                No     Fiber Optic             No  ...              No

  Contract PaperlessBilling              PaymentMethod MonthlyCharges  \
0      NaT              Yes           Electronic check      -1.160323
1      NaT               No               Mailed check      -0.259629
```

```
2      NaT              Yes               Mailed check      -0.362660
3      NaT               No  Bank transfer (automatic)      -0.746535
4      NaT              Yes           Electronic check       0.197365

  TotalCharges Churn               TotalServices  AvgMonthlyCharges  \
0    -0.994971    No          NoDSLNoYesNoNoNoNo          -1.158794
1    -0.173876    No        YesDSLYesNoYesNoNoNo          -0.305897
2    -0.960399   Yes        YesDSLYesYesNoNoNoNo          -0.355582
3    -0.195400    No        NoDSLYesNoYesYesNoNo          -0.792233
4    -0.941193   Yes  YesFiber OpticNoNoNoNoNoNo           0.365568

   TenureGroup
0     0-1 year
1    2-4 years
2     0-1 year
3    2-4 years
4     0-1 year

[5 rows x 24 columns]
Dataset split into training and testing sets.
Training set size: 5634
Testing set size: 1409
Cleaned dataset exported successfully.
```