Telecom Churn Analysis

Javier Lopez

D206, Data Cleaning

Table of Contents

Part I: Research Question

A. Research Question

Which variables are the most important in predicting which customers are at a high risk of churn?

B. Variables

Zendesk defines customer satisfaction (CSAT) as “a measure of how well a company’s products, services, and overall customer experience meet customer expectations” (Alaina Franklin, 2022). Using Franklin’s definition of CSAT and our data set, we seek to understand how our current and churned customers rate the importance of the variables in the table below.

Variable	Data Type	Description	Example
CaseOrder	Int	Integer related to the order of original data file	Range: 0 to 10,000
Customer_id	String	Character string unique to each customer	K409198
Interaction	String	Character string unique to each customer transaction, interaction, or sign-up	aa90260b-4141-4a24-8e36-b04ce1f4f77b
City	String	Character string indicating customer’s city of residence	Point Baker
State	String	Character string indicating customer’s state of residence	AK
County	String	Character string indicating customer’s county of residence	Prince of Wales-Hyder

Variable	Data Type	Description	Example
Zip	Int	Integer indicating the customer's zip code of residence	99927
Lat	Float	Decimal indicating the GPS latitude coordinate of the customer's residence	56.25100
Lng	Float	Decimal indicating the GPS longitude coordinate of the customer's residence	-133.37571
Population	Int	Integer indicating the population within a mile radius of the customer, based on census data	38
Area	String	Character string indicating the are type, based on census data	rural, urban, suburban
TimeZone	String	Character string indicating the time zone of customer's residence	America/Sitka
Job	String	Character string indicating the job of the customer	Environmental health practitioner
Children	Float	Decimal indicating the number of children in customer's household	2.0
Age	Float	Decimal indicating the age of customer as reported at sign-up	27.0
Education	String	Character string indicating customer's highest degree earned	Master's Degree
Employment	String	Character string indicating customer's employment status at sign-up	Full Time
Income	Float	Float indicating customer's annual income	67,000.0
Marital	String	Character string indicating customer's marital status at sign-up	Married
Gender	String	Character string indicating customer's gender self-identification	Male, Female, Nonbinary
Churn	String	Character string indicating if the customer discontinued service within the last month	yes, no
Outage_sec_per week	Float	Float indicating the average number of seconds of system outage in customer's neighborhood	9.265392

Variable	Data Type	Description	Example
Email	Int	Integer indicating number of emails sent to customer in the last year	3
Contacts	Int	Integer indicating number of time customer contacted technical support	2
Yearly_equip_failure	Int	Integer indicating number of times customer's equipment failed and was reset/replaced in the last year	1
Techie	String	Character string indicating whether customer considers themselves technically inclined	yes, no
Contract	String	Character string indicating the customer's contract term	month-to-month, one year, two year
Port_modem	String	Character string indicating whether customer has a portable modem	yes, no
Tablet	String	Character string indicating whether customer owns a tablet	yes, no
InternetService	String	Character string indicating customer's internet service provider	DSL, fiber optic, none
Phone	String	Character string indicating whether customer has a phone service	yes, no
Multiple	String	Character string indicating whether customer has multiple lines	yes, no
OnlineSecurity	String	Character string indicating whether customer has an online security add-on	yes, no
OnlineBackup	String	Character string indicating whether customer has an online backup add-on	yes, no
DeviceProtection	String	Character string indicating whether customer has a device protection add-on	yes, no
TechSupport	String	Character string indicating whether customer has a technical support add-on	yes, no
StreamingTV	String	Character string indicating whether customer has streaming TV	yes, no
StreamingMovies	String	Character string indicating whether customer has streaming movies	yes, no

Variable	Data Type	Description	Example
PaperlessBilling	String	Character string indicating whether customer has paperless billing	yes, no
PaymentMethod	String	Character string indicating customer’s payment method	electronic check
Tenure	Float	Decimal indicating number of months customer has remained with the provider	12.0
MonthlyCharge	Float	Decimal indicating the amount charged to the customer monthly	174.076305
Bandwidth_GB_Year	Float	Decimal indicating the average amount of data used in GB per year used by customer	3398.842752
Item1: Timely response	Int	Level of importance timely responses have to the customer	Scale: 1 = most important, 8 = least important
Item2: Timely fixes	Int	Level of importance timely fixes have to the customer	Scale: 1 = most important, 8 = least important
Item3: Timely replacements	Int	Level of importance timely device replacements have to the customer	Scale: 1 = most important, 8 = least important
Item4: Reliability	Int	Level of importance reliability of service has to the customer	Scale: 1 = most important, 8 = least important
Item5: Options	Int	Level of importance the variety of service options has to the customer	Scale: 1 = most important, 8 = least important
Item6: Respectful response	Int	Level of importance receiving respectful responses has to the customer	Scale: 1 = most important, 8 = least important
Item7: Courteous exchange	Int	Level of importance having a courteous exchange has to the customer	Scale: 1 = most important, 8 = least important
Item8: Evidence of active listening	Int	Level of importance signs of evidence of active listening has to the customer	Scale: 1 = most important, 8 = least important

Part II: Data-Cleaning Plan

C1. Plan Proposal

To assess the quality of our data, we will use Python. We will conduct our data-cleaning through the following actions:

- Load data and remove added index
- Observe data frame
- Rename non-descriptive variables, i.e., item1, item2, item3
- Check for and remove duplicates using Pandas
- Check for and treat missing values using Pandas
- Check for and treat outliers through standardization and visualization
- Standardize variables for analyzing

C2. Justification: Plan

The churn data set we are working with contains fifty variables and ten-thousand records. Using the Pandas info function, we can see a list of all fifty variables, how many non-null values exist in each variable, and their data types. Inspecting this list will bring forward any duplicate variable's existence and simplify the duplicate identification process. The info function also simplifies our task of ensuring each variable has the best data type for the records it contains.

Following the info function, the Pandas ‘isna’ function lets us quickly identify all null values and their location within the data set. Lastly, plotting our data into a box plot allows us to visualize the data's distribution and identify outliers. By calculating the z-scores for each variable, we can identify and treat any outliers that fall more than 3 points below or above zero.

C3. Justification: Programming Language

Our data cleaning process uses three Python libraries, Numpy, Pandas, and Matplotlib. We consider Python the best programming language for the data-cleaning task because it easily imports libraries as needed, the number of libraries available, and its straightforward syntax. The NumPy library supports quantitative data processing alongside Pandas, while Matplotlib supports data visualization efforts. Numpy and Pandas assist us in identifying duplicate variables, data types, and null values. Matplotlib then aids us in identifying outliers through visual means, while Scipy allows us to identify them by calculating the z-scores.

C4. Annotated Code: Data Quality Assessment

```
## Import libraries/packages
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
from scipy.stats import zscore
import warnings
warnings.filterwarnings('ignore')
plt.rcParams['figure.figsize'] = (18,10)
plt.rcParams['figure.max_open_warning'] = False

## Import data
df = pd.read_csv('churn_raw_data.csv', index_col=0).reset_index().drop('index',
axis=1)
```

```
## Rename survey columns
df.rename({
    'item1':'timely_response',
    'item2':'timely_fixes',
    'item3':'timely_replacements',
    'item4':'reliability',
    'item5':'options',
    'item6':'respectful_response',
    'item7':'courteous_exchange',
    'item8':'active_listening'
}, axis=1, inplace=True)

## Detect duplicates
print(df.duplicated().value_counts())

## Count all null values
print(df.isnull().sum())

## Assign zscores
for col in df.columns:
    if df[col].dtype == int or df[col].dtype == float:
        df['zscore_' + col] = zscore(df[col])

## Identify outliers
outliers = pd.DataFrame(columns=df.columns)
for col in df.columns:
    if 'zscore' in col:
        outliers = pd.concat([outliers, df.query('' + col + ' > 3 | ' + col + ' < -3')])

## Review unique outlier values
for col in df.columns:
    if 'zscore' in col:
        temp = df.query('' + col + ' > 3 | ' + col + ' < -3')[col[7:]].sort_values()
        print(temp.value_counts())
        print(col[7:] + ' length:', len(temp))
        print(col[7:] + ' percent of values:', (len(temp)/len(df))*100, end='\n\n')
```

Part III: Data Cleaning

D1. Data Quality Issues

The first data quality issue we found when reviewing the variables was the non-descriptive names of variables labeled item1 through item8. We then found several missing values in multiple variables, including Children, Age, Income, Techie, Phone, TechSupport, Tenure, and Badwidth_GB_Year. When checking the data, we identified several outliers across multiple variables, including Lat, Lng, Population, Children, Income, and many others.

D2. Data Quality Mitigation and Justification

We resolved the first data quality issue with multiple missing values by imputing them. Imputing the missing data allowed us to keep as much of our data as possible while maintaining data integrity. We first visualized the distribution of each variable and imputed the missing values for each variable based on their distribution. We imputed variables with a skewed or bi-modally distribution with the median, uniformly distributed variables with the mean, and categorical variables with the mode. Following the imputations, we checked the distribution for each variable to ensure no significant change.

We addressed the identified outliers by comparing the distribution of each variable with and without them to detect any significant change. We also reviewed the z-scores and values of each outlier to ensure that it was not a mistake. After reviewing the distribution of each variable and the value of each outlier, we determined they were not considered outliers and opted to keep them in the data.

D3. Data Quality Mitigation Outcome

After imputing the missing values for each variable, we visualized their distributions and noticed no significant change. The fact that there was no change meant that the imputed values would not cause any great statistical errors in future analyses. The same was the case with the detected outliers. Ultimately, after following all data-cleaning steps, we had a clean data set ready for PCA and further exploration.

D4. Annotated Code: Data Quality Mitigation

```
## Import libraries/packages
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
from scipy.stats import zscore
import warnings
warnings.filterwarnings('ignore')
plt.rcParams['figure.figsize'] = (18,10)
plt.rcParams['figure.max_open_warning'] = False

## Import data
df = pd.read_csv('churn_raw_data.csv', index_col=0).reset_index().drop('index',
axis=1)

## Rename survey columns
df.rename({
    'item1':'timely_response',
    'item2':'timely_fixes',
    'item3':'timely_replacements',
    'item4':'reliability',
    'item5':'options',
```

```
    'item6':'respectful_response',
    'item7':'courteous_exchange',
    'item8':'active_listening'
}, axis=1, inplace=True)

## Impute missing values with median (skewed/bi-modally distributed variables)
cols = ['Children', 'Income', 'Tenure', 'Bandwidth_GB_Year']
for col in cols:
    print(col + ': ', df[col].median())
    df[col].fillna(df[col].median(), inplace=True)
    print(col, ': ', df[col].median())

## Impute missing values with mean (uniformally distributed variables)
print(col, ': ', df.Age.mean())
df.Age.fillna(df.Age.mean(), inplace=True)
print(col, ': ', df.Age.mean())

## Impute missing values with mode (categorical variables)
cols = ['Techie', 'Phone', 'TechSupport']
for col in cols:
    print(col, ': ', df[col].mode()[0])
    df[col].fillna(df[col].mode()[0], inplace=True)
    print(col, ': ', df[col].mode()[0])

## Verify all values were imputed
print(df.isnull().sum())

## Remove z-score
for col in df.columns:
    if 'zscore' in col:
        df.drop(col, axis=1, inplace=True)

## Store data
df.to_csv('cleaned_data.csv')
```

D5. CSV File Attached

D6. Data-Cleaning Limitations

The biggest limitation when cleaning our data was the amount of missing data in certain columns. For example, nearly a quarter of the data for the Children and Age variables was missing. Our solution of imputing the missing data assisted us in retaining as much of the data as possible. However, it does have the potential to affect prediction models.

D7. Limitation Effects

Our research question seeks to identify which variables are the most important in predicting which customers are at a high risk of churn. The fact that a quarter of the data in the Children and Age variables was imputed can impact the accuracy of our results if it is determined that either of the two variables are the best predictors of churn.

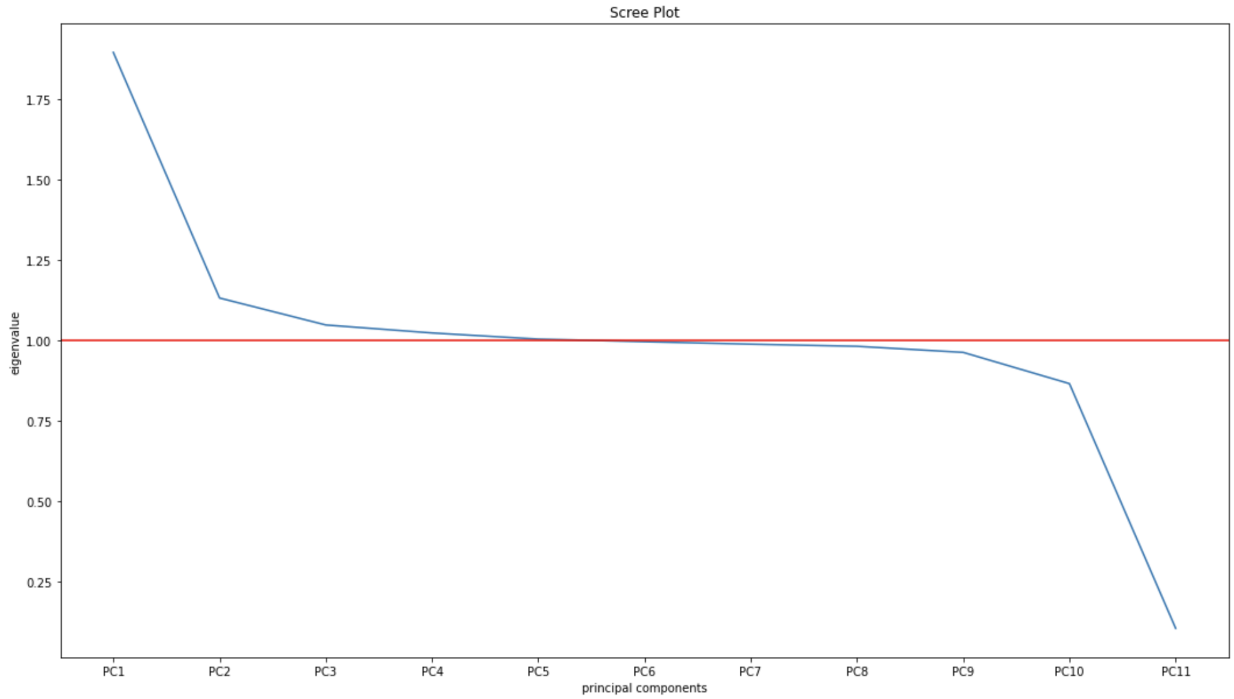
E1. Principal Components

There were eleven principal components in our data analysis. The image below depicts the loading matrix for the principal components.

	PC1	PC2	PC3	PC4	PC5	PC6	PC7	PC8	PC9	PC10	PC11
Population	-0.000410	-0.001876	-0.012267	0.006198	0.022597	-0.021291	0.004537	0.015837	0.704917	0.045223	0.706838
Children	-0.055144	0.023430	-0.047665	-0.004274	0.706395	0.057725	-0.007793	0.058193	-0.058210	0.696327	-0.009356
Age	-0.317635	0.553952	-0.362835	0.242582	0.021588	-0.336139	-0.433136	0.302881	-0.018146	-0.092825	0.002078
Income	-0.384314	-0.201651	0.520750	0.176956	-0.010086	-0.525970	0.329767	0.348825	-0.004487	0.040855	-0.016865
Outage_sec_perweek	-0.038206	0.051537	-0.103670	0.767615	0.014752	-0.058629	0.248663	-0.573845	-0.003246	0.033243	0.001718
Email	0.659805	0.207991	0.198053	0.415412	0.057735	0.169178	0.092719	0.515929	-0.000755	-0.053430	-0.004004
Contacts	0.431825	-0.491198	-0.443304	-0.003951	0.052749	-0.603274	-0.087308	0.027852	-0.018042	0.011890	-0.011402
Yearly_equip_failure	-0.054080	0.258891	-0.477952	-0.211643	0.015576	-0.002957	0.789294	0.168915	-0.015835	-0.069105	0.004992
Tenure	-0.349526	-0.545223	-0.323598	0.313605	0.051933	0.455452	-0.049867	0.376616	0.010936	-0.151217	-0.007306
MonthlyCharge	0.000900	0.009899	0.120987	-0.069471	0.700467	-0.055727	0.005655	-0.127722	0.038102	-0.684630	-0.012708
Bandwidth_GB_Year	-0.000975	-0.018410	0.021556	0.001166	0.000611	0.005590	-0.002975	-0.002464	-0.705121	-0.048334	0.706835

E2. Justification: Principal Components

We decided on eleven principal components for our analysis: Population, Children, Age, Income, Outage_sec_perweek, Email, Contacts, Yearly_equip_failure, Tenure, MonthlyCharge, Bandwidth_GB_Year. The main reason for the decision is attributed to the data type of the variables within the data set. Of the fifty total variables, thirty were of the object data type, and eleven were integer or float data types that best described the customer. We decided to use the eleven non-object descriptor variables because it would give us the best analysis results by not excluding any potential churn predictor. The image below depicts the scree plot for the eigenvalues of each principal component. Following Kaiser's rule, we kept principal components with an eigenvalue over one: PC1, PC2, PC3, PC4, and PC5. The principal components included Population, Children, 'Age, Income, and Outage_sec_perweek.



E3. Organizational Benefits

As time goes by and organizations grow so does the amount of data they have. The biggest benefit of PCA is being able to take in all of that data and determine what would be the best variables to use in a model. This benefit means that the organization can significantly reduce the time and cost of running their predictive models by using PCA.

Part IV: Supporting Documents

G. Third-Party Code References

We did not use third-party code in this project.

H. Sources

1. “Kaiser Rule.” *Displayr*, https://docs.displayr.com/wiki/Kaiser_Rule.
2. “A Step-by-Step Explanation of Principal Component Analysis (PCA).” *Built In*, https://builtin.com/data-science/step-step-explanation-principal-component-analysis.