Shane Boyce

WGU MSDA  
Data Cleaning – D206

**Part I: Background & Research Question**

A dataset was provided from a US telecom provider to begin building insights into their customer base. Understanding the what features contribute most toward knowledge in the dataset is essential in order to build models and make recommendations. The attached Jupyter Notebook and code seeks understand what features explain the first component using PCA.

The dataset provided is 10,000 observations by 50 features. The dataset came with a data dictionary which is attached. For this analysis, all categorical, binary and numerical data not corresponding to location was used. This means the only features removed are as follows:

* **CaseOrder** (categorical): A placeholder variable to preserve the original order of the raw data file
* **Customer\_id** (categorical): Unique customer ID
* **Interaction** (categorical): Unique IDs related to customer transactions, technical support, and sign-ups
* **City** (categorical): Customer city of residence as listed on the billing statement
* **State** (categorical): Customer state of residence as listed on the billing statement
* **County** (categorical): Customer county of residence as listed on the billing statement
* **Zip** (categorical): Customer zip code of residence as listed on the billing statement
* **Lat, Lng**(categorical): GPS coordinates of customer residence as listed on the billing statement
* **Area**(categorical): Area type (rural, urban, suburban), based on census data

The following features were used to weigh principal components in the analysis

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

The following variables represent response to an eight-question survey asking customers to rate the importance of various factors/surfaces on a scale of 1 to 8 (1 = most important, 8 = least important)

* **Item1** (categorical): Timely response
* **Item2** (categorical): Timely fixes
* **Item3** (categorical): Timely replacements
* **Item4** (categorical): Reliability
* **Item5** (categorical): Options
* **Item6** (categorical): Respectful response
* **Item7** (categorical): Courteous exchange
* **Item8** (categorical): Evidence of active listening

**Part II: Data-Cleaning Plan**

Phase 1 - Using common Python libraries for descriptive statistics and visualization, data will be assessed for:

* 1. Appropriate typing; using Pandas info instantiation will show feature typing.
  2. Missing Values; using Pandas Info instantiation, an overview of the original file’s columns will be displayed with how many observations each feature has.
  3. Duplicate Values; using the Pandas duplicated class, a sum of identical rows will be documented.
  4. Numerical Distribution; using matplotlib histograms alongside box and whisker plots, unclean data will be visualized.
  5. Categorical accuracy; assessment of categorical column inputs will be viewed and documented if a recommended update can be made. Also, noting what items should be encoded via Scikit-learn 1-hot Encoding (binary and multivariate).
  6. Ensure input values make sense. For example, there should be no negative numbers in this dataset based on the descriptions in the data dictionaries

These are standard data assessment procedures (Yıldırım, 2020).

Phase 2 – Cleaning data with Pandas and Scikit-learn by:

1. Imputing missing values or negative values is dependent on the type of data present and its distribution (Kumar, 2021).
   1. Binary data will be encoded to 0 and 1 to assist with analysis. Any missing values will be assumed to be a default “No” in the survey.
   2. Multivariate data will be 1 Hot Encoded and cleaned depending on its distribution.
      1. Normal or even distribution will impute randomly from the sample as a standard practice. (Kumar, 2021)
      2. Bimodal data will impute the median of each mode splitting the distribution in half. This works better than random sample as it minimizes risk and variance. (IBM, 2021)
      3. Negative values will be set to 0 in numerical datatypes.
      4. Note that K-Nearest-Neighbors may fill this dataset accurately as opposed to manual techniques, I chose not to do this due to the high dimensionality and large scale difference between
2. No duplicate values were detected in the dataset and therefore no treatment was necessary.
3. Scaling data via ScikitLearn RobustScaler
   1. Many of the items in this dataset were self-reported causing anomalies in data quality and mismatched categorization. For example, a customer in Yamhill Oregon is noted to be a CFO but income is listed $33000. Due to these outliers and the nature of this data being a self-reported survey, providing treatment corrections to this data without another similar sample is not recommended. Instead, RobustScaler limits the effects of outliers by assuming the interquartile range of data correctly corresponds to the data distribution (Hale, 2019)

111

1. Categorical accuracy; assessment of categorical column inputs will be viewed and documented if a recommended update can be made. Also, noting what items should be encoded via Scikit-learn 1-hot Encoding (binary and multivariate).

The code provided in the appendix notebook for this section is labeled Initial Exploration with charts and visuals. The documented anomalies appear in part 2 before the cleaning phase to give structure to the cleaning.

**Part III: Data Cleaning**

Below is an excerpt of text from the notebook documenting the types of issues in the dataset and the treatment method. They are split into 2 categories: quality and tidiness. Quality deals with missing and strange values in any individual feature while tidiness deals with the overall structure of the dataset.

**Quality**

* 1. ZIP/Lat/Lng are float when they should be strings as these are identifiers not numbers for mathematical operation.
     + Treated via Python’s built in astype function to cast as strings instead of np.number
  2. Outage time has some negative numbers. This doesn’t make sense for an outage time.
     + Set all values less than 0 to 0.
  3. Education has many categories that can be combined.
     + Using Pandas map functioning, categories were combined
  4. Some numerical columns with missing values: [‘Children’, ‘Income’, ‘Age’, ‘Tenure’, “Bandwidth\_Gb\_per\_year].
     + Children and Income were skewed, as IBM recommended I imputed the median to missing values (IBM, 2021).
     + Age was an even distribution and so a random imputation for the existing values was used to impute.
     + Bandwidth and Tenure had bimodal distributions. I found the median of the data and split the distribution into 2 sections. From these 2 sections, the median was chosen and missing data was randomly assigned using np.random\_choice. This ensured minimizing risk of increasing variance or introducing bias in either side of the distribution.
  5. Some qualitative columns have missing data [‘Area’,‘Education’, ‘Employment’, ‘Marital’, ‘Gender’, ‘Contract’, ‘InternetService’, ‘PaymentMethod’] that should be addressed alongside 1 hot encoding tidiness update.
     + Using SKlearn 1 hot encoding multivariate data mentioned above was mapped. This ensures the data is can be read by PCA to weigh features.
     + Binary data was mapped from ‘No/Yes’ to 0/1 respectively. 0 represents the default state and all missing data in binary columns were set to 0 during this step.

**Tidiness**

1. Many variables could use 1 hot encoding. This section will also clean the affected variables (multivariate with Nulls) and Binary columns of Yes/No can be made numeric.
   * + This was mentioned in quality as it was a dual issue.
2. Empty unnamed first column that corresponds to index should be dropped and Case Order should be set to index.
   * + The index and unused column were dropped. Case order was set to the index as no rows were dropped and case order is a sequential integer.

A copy of the cleaned data is included in the appendix and the code can be viewed in the notebook in the section labeled “Part 2: Cleaning.” The limitations of this dataset can be traced back to the data source. The data was voluntarily reported and therefore quality and accuracy can only be verified from more strenuous income matching via credit checks or employment verification. My imputation techniques also may be less effective than K-Nearest-Neighbors or decision tree classifiers such as random forest or a decision tree. Those methods however are more computationally expensive and may be impacted further by the quality of self-reported data. The effect this has on the current approach means that job title can’t be used meaningfully in PCA or we may introduce incorrect associations and covariance.

After the data was cleaned, RobustScaler was applied to the dataset to prepare it for SKlearn PCA. This scaled data is also supplied as a CSV in the appendix

**Results**

The top 3 components explain 29% of the available data revealing the most important features. Using PCA to reduce the dataset down can help with easier and accurate machine learning to cluster customers further on features using Churn Yes/No as the training criterion. The features reported below can be focused on as well to ensure a lowering of churn risk.

|  |  |  |
| --- | --- | --- |
| Component 0: 12% | Component 1: 10% | Component 2: 7% |
| Timely Response | Outage Time | Options |
| Timely Fixes | Monthly Charge | Timely Fixes |
| Timely Replacements | Active Listening | Active Listening |

# References

Hale, J. (2019, March 4). *Scale, Standardize, or Normalize with Scikit-Learn*. Retrieved from Towards Data Science: https://towardsdatascience.com/scale-standardize-or-normalize-with-scikit-learn-6ccc7d176a02

IBM. (2021, August 17). *Imputing or Filling Missing Values*. Retrieved from IBM SPSS Documentation: ibm.com/docs/en/spss-modeler/SaaS?topic=values-imputing-filling-missing

Kumar, A. (2021, October 3). *Python – Replace Missing Values with Mean, Median & Mode*. Retrieved from Vitaflux: https://vitalflux.com/pandas-impute-missing-values-mean-median-mode/

Yıldırım, S. (2020, March 2). *A Practical Guide for Data Analysis with Pandas*. Retrieved from Toward Data Science: https://towardsdatascience.com/a-practical-guide-for-data-analysis-with-pandas-e24e467195a9