**Performance Assessment: D206**

**Research Question**

1. For this assessment, the research question that I will be focusing on is as follows: are older customers (for this we will define “older” as greater than or equal to 50 years of age) being charged more for medical services (total charges)? One aspect that led to the develop of this research question is that older people may have preexisting conditions or be at an increased risk of infections, etc. that the healthcare service provider could use as a means to make more money off of their medical needs.
2. The variables in the data set and the type of data being described are as follows:
   1. CaseOrder- num, quantitative, e.g., 1, 2, 3, etc.
   2. Customer\_id- chr, qualitative, e.g., C412403, Z919181, etc.
   3. Interaction- chr, qualitative, e.g., 8cd49b13-f45a-4b47-a2bd-173ffa932c2f, etc.
   4. UID- chr, qualitative, e.g., 3a83ddb66e2ae73798bdf1d705dc0932, etc.
   5. City- chr, qualitative, e.g., Eva, Marianna, Sioux Falls, etc.
   6. State- chr, qualitative, e.g., AL, FL, SD, etc.
   7. County- chr, qualitative , e.g., Morgan, Jackson, etc.
   8. Zip- num, qualitative, e.g., 35621, 32446, etc.
   9. Lat- num, quantitative, e.g., 34.3496, 30.84513, etc.
   10. Lng- num, quantitative, e.g., -86.7251, -85.2291, etc.
   11. Population- int, quantitative, e.g.,- 2951, 11303, 17125, etc.
   12. Area- chr, qualitative, e.g., suburban, urban, rural, etc.

TimeZone- chr, qualitative, e.g., American/Central, American/New\_York, etc.

* 1. Job- chr, qualitative, e.g., Psychologist, Actuary, etc.
  2. Children- int, quantitative, e.g., 1, 3, 0, etc.
  3. Age- int, quantitative, e.g., 53, 51, 78, etc.
  4. Education- chr, qualitative, e.g., Some College, Master’s Degree, etc.
  5. Employment-chr, qualitative, e.g., Full Time, Retired, etc.
  6. Income- chr, quantitative, e.g., 86575.93, 46805.99, etc.
  7. Marital- chr, qualitative, e.g., Divorced, Married, etc.
  8. Gender- chr, qualitative, e.g., Male, Female, non-binary, prefer not to say.
  9. ReAdmis-chr, qualitative, e.g., Yes, No.
  10. VitD\_levels- num, quantitative, e.g., 17.80233, 18.99464, etc.
  11. Doc\_visits- int, quantitative, e.g., 6, 4, 1, etc.
  12. Full\_meals\_eaten- int, quantitative, e.g., 0, 2, 1, etc.
  13. VitD\_supp- int, quantitative, e.g., 0, 1, 2, etc.
  14. Soft\_drink- chr, qualitative, e.g., Yes, No.
  15. Initial\_admin- chr, qualitative, e.g., Emergency, Observation, Elective.
  16. HighBlood- chr, qualitative, e.g., Yes, No.
  17. Stroke- chr, qualitative, e.g., Yes, No.
  18. Complication\_risk- chr, qualitative, e.g., Medium, Low, High.
  19. Overweight- int, quantitative, e.g., Yes, No.
  20. Arthritis- chr, qualitative, e.g., Yes, No.
  21. Diabetes- chr, qualitative, e.g., Yes, No.
  22. Hyperlipidemia- chr, qualitative, e.g., Yes, No.
  23. BackPain- chr, qualitative, e.g., Yes, No.
  24. Anxiety- int, quantitative, e.g., Yes, No.
  25. Allergic\_rhinitis- chr, qualitative, e.g., Yes, No.
  26. Reflux\_esophagitis- chr, qualitative, e.g., Yes, No.
  27. Asthma- chr, qualitative, e.g., Yes, No.
  28. Services- chr, qualitative, e.g., Blood Work, CT scan, etc.
  29. Initial\_days- chr, e.g., 10.58577, 15.12956, etc.
  30. TotalCharge- num, quantitative, e.g., 3191.049, 4214.905, etc.
  31. Additional\_charges- num, quantitative, e.g., 17939.4, 17613, etc.
  32. Item1- int, quantitative, e.g., 1, 2, 3, etc.
  33. Item2- int, quantitative, e.g., 1, 2, 3, etc.
  34. Item3- int, quantitative, e.g., 1, 2, 3, etc.
  35. Item4- int, quantitative, e.g., 1, 2, 3, etc.
  36. Item5- int, quantitative, e.g., 1, 2, 3, etc.
  37. Item6- int, quantitative, e.g., 1, 2, 3, etc.
  38. Item7- int, quantitative, e.g., 1, 2, 3, etc.
  39. Item8- int, quantitative, e.g., 1, 2, 3, etc.

This information was obtained using the str(medical\_raw\_data) command in RStudio where “chr” represents the term “character” and “num” represents the term “number.” The command and results are showcased below.

Text

Description automatically generated

**C. Data-Cleaning Plan**

**1.** To start the process of cleaning the data, I will use the dplyr package in R to check for duplications within the data set and then set the data set equal to the distinct entries of that data set. This is the first step to cleaning the data as it removes all duplicate entries so there is nothing repeating that could distort your analysis. The next step in my plan is something called data wrangling, which is a little different from data cleaning as instead of cleaning dirty data, I will be re-expressing numerical data to represent categorical data. To be specific, the “Overweight” and “Anxiety” columns need to be re-expressed, as I noticed the inconsistency when creating the summary of the data. The reason for this step is re-expressing these numerical values as categories will keep the data consistent across all of the Yes/No columns of data. After this step of data wrangling, I will return to the process of cleaning the data by filling in null values (NAs) with the mean or median or mode (depending on the data type for each of the missing variables that show up). This type of imputation works because it does not reduce the sample size of the data while filling in any missing variables with the most appropriate value. Lastly, I will check all of the quantitative (numerical) data columns for outliers using boxplots. Boxplots are good for this step in data cleaning because they are capable of showing which entries in the data columns do not fit within the interquartile range while also showcasing just how extreme they are so you can decide if they were potentially the results of human error or could statistically have occurred. This concludes all of the steps that I will use to clean the data set. After the data set has been cleaning, I will conduct a principal component analysis of the cleaned, numerical data to determine which dimensions in the data frame are principal components by creating a scree plot and analyzing the eigenvalues. (Dr. Middleton 2022).

**2.** These data cleaning steps are necessary because they allow inconsistencies within the data package to be corrected so that the data can then be analyzed in a way that will not harm the results of said analysis. Duplications, missing variables, and outliers are among the most basic forms of data cleaning but are also some of the most important. Studies show that data cleaning is what a data analyst spends most of their time doing, sometimes as much as 80% of their time goes towards data cleaning (Larose, C. D., & Larose, D. T., 2019). It is imperative that a data analysis uses software capable of cleaning the data, such as R and Python, so that the data can be used for analysis to meet business needs in a way that does not hurt their analysis leading to potentially disastrous results that hurt the business.

**3.** To perform the data cleaning process, I will use R. I will use the dplyr package to remove duplications, the plyr package to re-express integers as categorical variables, and the modeest package to fill missing values with the mode for categorical data. I will also use the factoextra package for the PCA. My reason for choosing R for this assessment is that I personally think R is easier to use than Python, as I already had experience using R in my undergraduate studies and am already familiar with the interface and tools, whereas Python is completely new to me, and I am not yet comfortable using it.

**4/D4.** Code used for cleaning duplicates:

* med\_raw\_data <- distinct(med\_raw\_data)- this code uses the dplyr package to set the med\_raw\_data file as equal to the distinct (unique) entries within the file, i.e., removing all duplicates and setting the file equal to the data frame without duplication.

Code for re-expressing numerical data as categorical data:

* med\_raw\_data$colname <- as.character(med\_raw­\_data$colname)- this code coverts the integer column into a character vector
* med\_raw\_data$colname <- revalue(med\_raw\_data$colname, c(“0”=”No)”- this code coverts any data points in the column that are a 0 to a “No”.
* med\_raw\_data$colname <- revalue(med\_raw\_data$colname, c(“1”=”Yes”)- this code coverts any data points in the column that are a 1 to a “Yes”.

Code used for treating missing variables:

* colsSums(is.na(med\_raw\_data)- this code is used to detect all entries within the data frame that are null.
* Hist(df$colname)- this code will be used to create histograms of each of the columns with missing data in order to determine their distribution to discover what imputation method will be necessary to fill in the missing values.
* medical\_raw\_data$colname[is.na(medical\_raw\_data$colname)] <- median(medical\_raw\_data$colname, na.rm=TRUE)- this code is used to fill missing data in with the median
* medical\_raw\_data$colname[is.na(medical\_raw\_data$colname)] <- mean(medical\_raw\_data$colname, na.rm=TRUE)- this code is used to fill missing data in with the mean
* medical\_raw\_data$colname[is.na(medical\_raw\_data$colname)] <- mfv(medical\_raw\_data$colname, na\_rm=FALSE)- this code is used to fill missing data in with the mode

Code used for the detection of outliers using boxplots:

* bColname <- boxplot(med\_raw\_data$colname)- this code is used to create a box plot for each of the numerical values in the data set to detect for outliers

**D. Data Cleaning**

**1/2/3.** Below, you will find my descriptions, justifications, and summaries for each step in the data cleaning process:

* **Duplications: Text

  Description automatically generated with medium confidence**

As shown in the screenshot above, the code from **C4** has been entered into RStudio to remove all duplicated from the data set, however, as you can see from the duplicated command, there actually were not any duplicates that needed to be cleaned, as everything returned FALSE.

* **Re-Expression:** As we discovered earlier, the overweight and anxiety columns need to be converted from an integer to a character column. Afterwards, we need to re-express the 0s as Nos and the 1s as Yeses. This will make it much easier to fill in the missing values in the next data cleaning step as we will be able to replace with the mode.

**A picture containing text

Description automatically generated**

As shown above, by the code in R, the two categories in question have been converted from an integer into a character and changed to express as 0 as a “No” and a 1 as a “Yes.” This allows us to keep the same consistency and text format among the other Yes/No data columns within the data set, as well as allow us to replace missing data with the most frequent term.

* **Missing Data:** When entering the code from **C4** into R, the resulting missing data is as follows:
  + **Children**: missing 2588 entries
  + **Age**: missing 2414 entries
  + **Income:** missing 2464 entries
  + **Soft\_drink:** missing 2467 entries
  + **Overweight:** missing 982 entries
  + **Anxiety:** missing 984 entries
  + **Initial\_days:** missing 1056 entries

A screenshot of a computer

Description automatically generated with low confidence

This chart above showcases the categories and all the missing data entries within them. Now after we noted which columns have missing entries, we have to decide how to go about filling the data in each of these 7 columns. In order to determine which imputation method to use to fill the data, we first need to create a histogram to determine the distribution for the numerical data terms. For variables that are normally or uniform distributed, we will imputate by filling the missing value with the mean. For columns that have a skewed distribution, we will fill with the median. Lastly, for categorical data, we will fill the missing entries with the mode (Dr. Middleton, 2022).

* **Children:**

Chart, histogram

Description automatically generated

As you can see from the histogram above, the data for the children column is skewed towards the right. Since it is skewed, I used the median to fill in the missing data points.

After filling in the missing values with the median, the histogram now looks like this.

Chart

Description automatically generated

The histogram is still skewed to the right, but you can see that the frequency of the variables is now higher, indicating that the null values have been filled correctly.

* **Age:**

Chart, histogram

Description automatically generated

The histogram for age shows a uniform distribution so the missing variables will be filled with the mean, and the post-fill histogram will now look like this. It is now normally distributed with a clear peak in the center where the mean data was inserted into the null entries.

Chart

Description automatically generated with medium confidence

* **Income:**

Chart, histogram

Description automatically generated

The income histogram is skewed to the right so the missing data will be filled with the median, and afterwards the post-fill histogram will look like this. As expected, it is still skewed to the right, but with a higher frequency.

Chart, histogram

Description automatically generated

* **Soft\_drink/Overweight/Anxiety:** As these data column are categorical variable and not a numerical variable, it is not possible to create a histogram to examine for the skewedness of the data, and instead the mode was used to replace the missing data as from the code mentioned in part **C4**. This is demonstrated with the screenshot below:



* **Initial\_days:**

Chart, histogram

Description automatically generated

The resulting histogram is a non-symmetric bimodal distribution with two peaks, so we will fill in the missing data with the median. Afterwards, the filled histogram will look like this, with a new third peak where the median values have been entered:

Chart, histogram

Description automatically generated

So now that all of the missing values have been cleaned, we can confirm by reentering the code from before and there should be all 0s were there was once numbers.

A screenshot of a computer

Description automatically generated with low confidence

* **Outliers:** After filling in all of the missing data, the last step for cleaning the data is to detect for outliers for all of the numerical data columns. The box plots for all of the numerical variables are shown below. Note: for this assessment, I am checking for outliers for Zip, Lat, Lng, or Population as these vary greatly depending on the address of the patient, thus nothing can truly be considered an outlier in these cases. Also, the Item1-Item8 categories are all based on personal preference due to the patient’s rankings, so those cannot be considered outliers as well.
  + **Children:Chart, box and whisker chart

    Description automatically generated**

The Children boxplot does show a few outliers present within the dataset, however, I will acknowledge that some people do in fact have a lot more children than others, so I will note that these outliers do exist, but I will not be removing them.

* + **Age: Chart, box and whisker chart

    Description automatically generated**

The Age boxplot does not contain any outliers, so nothing needs to be cleaned from this column.

* + **Income: Chart, box and whisker chart

    Description automatically generated**

The Income boxplot does showcase a great number of outliers present, however, due to the varying degree of education and work experience among all of the patients, it can be assumed that there will be extreme degrees of variety within personal income levels, so I am choosing not to remove any of these outliers, but I will acknowledge that they exist.

* + **VitD\_levels: Chart, box and whisker chart

    Description automatically generated**

The boxplot for VitD\_levels does showcase a number of outliers but due to the varying conditions of the patients and their initial reason for admittance, variance is expected so I will not be removing any of these outliers.

* + **Doc\_visits: Chart, box and whisker chart

    Description automatically generated**

There are no outliers in this data column so nothing needs to be cleaned.

* + **Full\_meals\_eaten:Chart, box and whisker chart

    Description automatically generated**

Outliers are present in the boxplot for the Full\_meals\_eaten data column, but as I have mentioned with other columns, some degree of variance and outliers are expected simply due to different needs and conditions of the patients so again, I will not be removing these outliers.

* + **VitD\_supp:Chart, box and whisker chart

    Description automatically generated**

Outliers are present in the boxplot for the VitD\_supp data column, but as I have mentioned with other columns, some degree of variance and outliers are expected simply due to different needs and conditions of the patients so again, I will not be removing these outliers.

* + **Initial\_days: Chart, box and whisker chart

    Description automatically generated**

There are no outliers in this data column so nothing needs to be cleaned.

* + **TotalCharge: Chart, box and whisker chart

    Description automatically generated**

The TotalCharge boxplot does showcase a number of outliers that are present, however due to the fact that each patient was receiving different treatments for different issues and needs, it can be assumed that there will be variety within the amount that each patient would be required to pay. Due to this, I am not removing any outliers for the TotalCharge column as well as the next Additional\_charges column for the same reason.

* + **Additional\_charges:Chart, box and whisker chart

    Description automatically generated**

**4.** The code used to mitigate anomalies was included with the code used to identify anomalies in part **C4**.

**5.** A copy of the cleaned data set has been attached to the submission task alongside this written assessment.

**6.** The biggest limitation in the data cleaning process for this dataset is that I have no way to contact the original team behind entering all of the data. Outliers and missing data tend to be the result of human error (typos, etc.) so normally it would be possible to contact the team and get the correct information from there to fix a typo or other instance of human error. Such was not the case in regard to this dataset, so the data had to be cleaned as opposed to corrected.

**7.** Considering that my cleaned data concludes that all of the outliers are necessary given the circumstances of the data, it is possible that other data analysts may not see it the same way and the outliers that I chose to leave in place, could distort or harm their analyses. I also cleaned the data to address the most common issues, but those are not the only issues that can be present. Issues such as text formatting were not addressed, and there are others as well, but my cleaning purposes served to cover the most common issues.

**E. Principal Component Analysis**

**1/2.** To get started with the PCA, I first subsetted the data to exclude all of the categorical data. This left me with just the children, age, income, VitD\_levels, Doc\_visits, Full\_meals\_eaten, VitD\_supp, Initial\_days, TotalCharge, the Additional\_charges columns and the 8 item columns. I took these columns and using the factoextra package, made a scree plot to identify which principal components have eigenvalues over 1. Afterwards, using the rotation command and the scree plot as guides, the highest values in each PC are considered the principal components.

Chart, histogram

Description automatically generatedTable

Description automatically generated

As you can see from the screenshots above, every principal component had an eigenvalue greater than 1 so the highest component in each PC is the principal component for this analysis, but since they are all greater than 1, each of these components is a principal component (Larose, C. D., & Larose, D. T., 2019).

**3.** The benefit of doing a PCA is that the data goes to show what specific variables are statistically relevant with one another. In this case, all of the variables have an effect on one another, but in other data samples, the PCA can showcase which variables exactly are dependent on one another. This helps the business because it reduces the dimensions within your data set and showcases which dimensions are statistically significant, therefore the business can look for trends within the statistically significant dimensions to figure out with ones are the most crucial/important for various business needs or problems that they are attempting to analyze (Williams 2021).

**G.** References used in the making of this assessment is listed below:

Middleton, Dr. 2022. *Getting Started with Missing Data and Outliers* [PowerPoint Slides]. <https://westerngovernorsuniversity-my.sharepoint.com/personal/keiona_middleton_wgu_edu/_layouts/15/onedrive.aspx?id=%2Fpersonal%2Fkeiona%5Fmiddleton%5Fwgu%5Fedu%2FDocuments%2FDocuments%2FD206%2FD206%2DMissingDataandOutliersPresentationSlides%2Epdf&parent=%2Fpersonal%2Fkeiona%5Fmiddleton%5Fwgu%5Fedu%2FDocuments%2FDocuments%2FD206&ga=1>

Larose, Chantal D., and Daniel T. Larose. “Principal Component Analysis.” Data Science Using Python and R, Wiley, Hoboken, NJ, 2019.

Williams. “How Principal Component Analysis Helps Get the Data You Actually Need.” Search Discovery, 13 Oct. 2021, https://www.searchdiscovery.com/blog/principal-component-analysis-helps-get-data-actually-need/#elementor-action%3Aaction%3Dpopup%3Aclose%26settings%3DeyJkb19ub3Rfc2hvd19hZ2FpbiI6IiJ9.