**D206 Data Cleaning**

**Michael Rupert**

**Western Governors University**

**Table of Contents**

**Part I: Research Question and Variables**

**A: Research Question**

**B: Description of the Dataset**

**Part II: Detection**

**C1: Detection Methods**

**C2: Justification for Detection Models**

**C3: Justification for Program Language**

**C4: Detection Code**

**Part III: Treatment**

**D1: Discussion of Findings**

**D2: Treatment Methods and Justifications**

**D3: Summary of Treatment**

**D4: Treatment of Code**

**D5: Clean Dataset**

**D6: Limitations**

**D7: Implications**

**Part IV: PCA**

**E1: Variables and PCA Loadings**

**E2: PCs Selection**

**E3: Benefits**

**Part I**

**A: Research Question**

Based on the sample of historical data, develop a model that shows which patients are at risk of readmission.

**B: Description of the Dataset**

Below is an example of the data and the code that produced it.

Text

Description automatically generated

*Figure 1*

Below is listed every variable, data type of variable, description of the data, an example of the data, and if it is quantitative or qualitative data.



I wish to note at this point, in at least one instance the data definitions provided are inaccurate to what is produced when the code is executed. For example: the variable overweight is said to be a yes or no answer, but which would be qualitative data, when inspected we see clear it is a float 64 type and numeric so would be quantitative.

Below is an example of data types and the code the produced it.

Timeline

Description automatically generated with low confidence

*Figure 2*

**Part II**

**C1: Detection Methods**

We must use a few different methods to ensure the data has no duplicates, remove columns that are irrelevant, re-name variables whose names are unclear or ambiguous, and re-express qualitative data and convert to quantitative data.

**Find Duplicates**

Text

Description automatically generated

*Figure 3*

No duplicate data detected. This makes sense since each patient is given a unique id number as well as the transaction number.

**Remove Unneeded Columns**

Graphical user interface

Description automatically generated with low confidence

*Figure 4*

**Re-express Ambiguously Named Variables**

A number of variables have been re-named to make them more understandable by column name.

Text

Description automatically generated

*Figure 5*

**Re-index the data to begin with 1 as the first index.**

A screenshot of a computer

Description automatically generated with medium confidence

*Figure 6a Before Re-Indexing*

A screenshot of a computer

Description automatically generated with medium confidence

*6b After Re-Indexing*

**Re-express Qualitative data as Quantitative data**

The code used to achieve re-expression is below, *Figure 7*.

Text

Description automatically generated

*Figure 7*

The output of the above code is below for the first 5 and last 5 records, *Figure 8a and b*.

A screenshot of a computer

Description automatically generated with medium confidence

*Figure 8a*

A picture containing graphical user interface

Description automatically generated

*Figure 8b*

You can see in some of the records above NaN values appear. We wish to check for these values through out the Data as well as values that might erroneously show as zero, for example in the population data.

We do an initial check on the Data Frame as shown below in Figure 9. This identifies what columns contain NaN values.

A screenshot of a computer

Description automatically generated

*Figure 9*

Additionally, we want to check the Population variable column for zero population, since if a patient exists in a population area that would me the population was at least one so zero would be a false value. Below we check the population column. Figure 10.

A screenshot of a computer

Description automatically generated with medium confidence

*Figure 10*

The data shows there are 109 zeros listed in the Population column. And columns 35-45 show this is indeed true.

We will now replace the zeros with NaN in population. Shown in Figure 11 for records 35-45.

A screenshot of a computer

Description automatically generated with medium confidence

Figure 11

Just to verify the Data Frame has NaN in all the columns including the Population column will repeat the NaN value check. In figure 12

Graphical user interface, text, application

Description automatically generated

*Figure 12*

Next find the mean values for each of these columns. Figure 13

Text

Description automatically generated

*Figure 13*

Replace the NaN values with the Mean value. Figure 14 shows records 35-45

Text

Description automatically generated

*Figure 14*

Check to see if NaNs are removed. Figure 15.

A screenshot of a computer

Description automatically generated with medium confidence

*Figure 15*

**Box Plot and Grubbs testing for outlier detection of variables.**

**Area**

A screenshot of a computer

Description automatically generated with medium confidenceChart, bar chart

Description automatically generated

No Outliers

**Population**

Graphical user interface, text, application, website

Description automatically generated

*Figure 16a*

Chart, box and whisker chart

Description automatically generated

*Figure 16b*

**Timezone**

Text

Description automatically generated

Chart, box and whisker chart

Description automatically generated

These outliers will be standardized.

A screenshot of a computer

Description automatically generated with medium confidence

**Children**

Chart, box and whisker chart

Description automatically generated

*Figure 17a*

Text

Description automatically generated

*Figure 17b*

Text

Description automatically generated

*Figure 17c*

Below *Figure 18a*  is the Box plot of age. No outliers appear on this box plot.

Chart, box and whisker chart

Description automatically generated

Grubbs testing for Age, *Figure 18a*

A screenshot of a computer

Description automatically generated with medium confidence

*Figure 18b*

**Education**

Chart, box and whisker chart

Description automatically generated

Hypothesis: Extremely high and low values are outliers. The zero outlier could very well be an erroneous entry.

A screenshot of a computer

Description automatically generated with medium confidence

The very high and low education levels are outliers and will be normalized. To reduce the effect this might have on the model.

Text

Description automatically generated

**Household\_Income**

A screenshot of a computer

Description automatically generated with medium confidence

Chart, box and whisker chart

Description automatically generated

**These outliers are far from the mean and will be standardized.**

Graphical user interface, text

Description automatically generated

**VitD\_Levels**

**Chart, box and whisker chart

Description automatically generated**

**Text

Description automatically generated**

The grubbs test yields two higher outliers 53.019124  and 52.757599 hypothesis: high values 53.019124  and 52.757599 are outliers.These are located at indices 1963 and 1306 respectively.

**Gender**

**\** Chart, box and whisker chart

Description automatically generated

No Outliers for this variable.

**ReAdmis**

Chart, bar chart

Description automatically generated

No Outliers for this variable.

**Doctor Visits**

**Chart, box and whisker chart

Description automatically generated**

**A picture containing text

Description automatically generated**

**No outliers in this variable.**

**Full\_meals\_eaten**

**A screenshot of a computer

Description automatically generated with medium confidence**

**Full\_meals\_eaten**

**Chart, box and whisker chart

Description automatically generated**

**Text

Description automatically generated**

**Alternative hypothesis: highest value 7 is an outlier.**

**Full\_meals\_eaten Standardized values**

**Text

Description automatically generated**

**VitD\_supp**

Chart, box and whisker chart

Description automatically generated

A screenshot of a computer

Description automatically generated

**These outliers will be standardized**

Text

Description automatically generated

**Soft\_drink**

Chart, box and whisker chart

Description automatically generated

This is either yes or no so the values are not outliers and will be retained as is.

**Initial\_admin**

Chart, bar chart, line chart

Description automatically generated

There are no outliers for this variable.

**HighBlood**

Chart, bar chart

Description automatically generated

There are no outliers in this variable.

**Stroke**

Chart, scatter chart

Description automatically generated

The values of this variable are 0 or 1 these are not outliers and will be kept.

**Complication\_risk**

**A screenshot of a computer

Description automatically generated with medium confidence**

**Chart, line chart

Description automatically generated**

There are no outliers in this variable.

**Overweight**

Chart, bar chart, line chart

Description automatically generated

There are no outliers in this variable.

**Arthritis**

Chart, bar chart

Description automatically generated

There are not outliers for this variable.

**Diabetes**

Chart, bar chart

Description automatically generated

There are no outliers for this variable.

**Hyperlipidemia**

Chart, bar chart

Description automatically generated

There are no outliers for this variable.

**BackPain**

Chart, bar chart

Description automatically generated

There are no outliers for this variable.

**Anxiety**

Chart, bar chart

Description automatically generated

There are no outliers for this variable

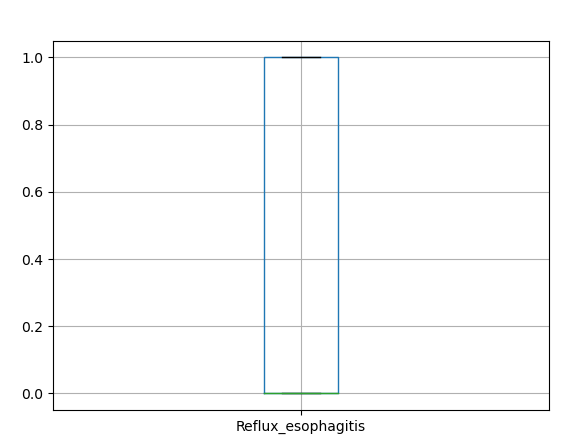
**Allergic\_rhinitis**

Chart, bar chart

Description automatically generated

There are no outliers for this variable

**Reflux\_esophagitis**



There are no outliers for this variable

**Asthma**

Chart, bar chart

Description automatically generated

There are no outliers for this variable

**Initial\_days**

**A screenshot of a computer

Description automatically generated with medium confidence**

**Chart, box and whisker chart

Description automatically generated**

There are no outliers within this variable.

**Daily\_Average\_Charge**

A screenshot of a computer

Description automatically generated with medium confidence

**Chart, box and whisker chart

Description automatically generated**

These outliers seem to have some correlation with ReAdmis data so they will not be standardized.

**Additional\_charges**

**Text

Description automatically generated**

**Chart, box and whisker chart

Description automatically generated**

This will be standardized.

Text

Description automatically generated

**Timely\_Admission**

**A screenshot of a computer

Description automatically generated with medium confidence**

**Chart, box and whisker chart

Description automatically generated**

**Timely\_Treatment**

**A screenshot of a computer

Description automatically generated with medium confidence**

**Boxplot**

**Chart, box and whisker chart

Description automatically generated**

**Timely\_Visits**

**Text

Description automatically generated**

**Chart, box and whisker chart

Description automatically generated**

**Reliability**

**Text

Description automatically generated**

**Chart, box and whisker chart

Description automatically generated**

**Options**

**Text

Description automatically generated**

**Chart, box and whisker chart

Description automatically generated**

**Treatment\_Hours**

**Text

Description automatically generated**

**Chart, box and whisker chart

Description automatically generated**

**Courteous\_Staff**

**A screenshot of a computer

Description automatically generated with medium confidence**

**Chart, box and whisker chart

Description automatically generated**

**Active\_Listening**

**Text

Description automatically generatedChart, box and whisker chart

Description automatically generated**

The above values are from a questionnaire and will be retained as is.

**C2: Justification for Detection Methods**

I decided to you the methods provided in the language I used to analyze the data. The Python Language has many powerful tools to correct, clean and process data. The ability to find nulls, zeros, replace data in a single column or multiple cells. Below are some of the sources to corroborate my decision.

**Finding Zeros in a Padas dataframe or column:**

**Finding Nulls in a Pandas dataframe**: <https://www.geeksforgeeks.org/working-with-missing-data-in-pandas/#:~:text=In%20order%20to%20check%20missing,null%20values%20in%20a%20series>.

**Replacing Zer0 with NaN values in a pandas dataframe:** https://www.geeksforgeeks.org/replace-nan-values-with-zeros-in-pandas-dataframe/

**Grubbs Testing Resources**: <https://www.real-statistics.com/students-t-distribution/identifying-outliers-using-t-distribution/grubbs-test/>

**C3: Justification for Program Language**

I decided to use Python programming language. It is a powerful language with rich libraries for both analysis and visualization as well as the ability to clean and process the data. I used Pandas, Numpy, Matplotlib, outliers from smirnov\_grubbs. These libraries contained the needed functions and libraries to process and visualize the data set.

**C4: Detection Code**

see code / script attached

D1. Discussion of Findings

After looking at the data for duplicate records none were detected. The following variables were found to have missing values or values of zero would prove to be incorrect values. For example the Population variable having a value of zero for any record would prove incorrect simply due to the fact if a person lives in a area then the population is at least 1. Other variables that had missing values were , 'Children', 'Age', 'Household\_Income', 'Soft\_drink', 'Overweight', 'Anxiety', 'Initial\_days'. All of these values were replaces using the mean of all other records of that variable.

A number of the variables present in the data had outliers, some of these outliers were kept, Population, Household\_Income, Total\_Average\_Daily\_Charge, were left unchanged. Other outliers were normalized.

D2. Treatment Methods and Justification

As discussed above missing variables were replaced using the mean of the values of that variable that did exist. This is a common way to handle missing data points but must be used with caution so as not to reduce the variability of the data. (<https://seleritysas.com/blog/2020/03/03/the-best-way-to-handle-missing-data//>)

The normalizing of data with outliers was used to deal with many of the variables that had outliers. This causes the data to be utilized across all records in the database. (<https://blog.insycle.com/normalize-data>)

The visualizations are shown above.in Section C.

D3. Summary of Treatment

As can be seen in the code provided and the snapshot sin Section C. I first removed, unneeded columns, Reindexed the data so it started at index 1 as opposed to index 0. Data that was Qualitative data was transformed to quantitative data. The data was then examined for duplicates, missing values or values that seemed to make no sense, Variable names were made to be more understandable if they were ambiguous.

D4. Treatment Code

see code / script attached

D5. Clean Dataset

Clean data Attached.

D6. Limitations

Some of the limitations of the methods I used are using the mean to fill missing vales could cause the data to vary less of the range of the actual true data. Normalizing the data that contains outliers doesn’t remove the outliers or handle them. It makes the data more consistent across the range.

D7. Implications

There are still some questions about the quality of this data. One thing of note is that Due to privacy and HIPAA laws it might be recommended the Longitude and Latitude variables not be used simple because it is personal information hat can be directly linked to the patient and would violate privacy laws.

The data is very broad, and the variables aren’t clearly defined. If they are trying to determine the cause of readmission rates being high, then it might not be by hospital since this data is collected by region or geographic area.

Part IV: PCA

E1: Variables

Graphical user interface, text

Description automatically generated Graphical user interface, text

Description automatically generated

**PCA Loadings**

We can see the PCA values of the first three in the sorted list equal over 96% of the variance. The Skree of the data supports this as well.

Chart, line chart

Description automatically generated

Text

Description automatically generated

A quick check of sums of the PC of interest below.

Text

Description automatically generated

An Eignen Value check shows below

Chart, line chart

Description automatically generated

E2: PCs Selection

According to the Scree plot above the principal components that most effect the variance of data are the variables with a cos2 over 0.4 and would be Daily\_Average\_Charge – Additional\_charges - Survey\_HoursTreatment - Survey\_TimelyVisits - Survey\_TimelyAdmin - Survey\_TimelyTreatment.

E3. Benefits

By analyzing the correlation between the PCA and Readmission rates the at-risk medical facilities could benefit and improve performance. The results of a deep exploration and analysis can provide the organization with actionable insights and that could result in a reduction readmission rates.