­

**Predictive Modeling**

Jason Willis

College of Information Technology,

Western Governors University

Dr. Eric Straw

January 9th, 2021

Table of Contents for Each Rubric

Part I: Research Question

[Describe Purpose, Summarize Research Question and Define Objectives:](#RubricA) 3

***Part II: Method Justification***

[Summarize Model Assumptions,](#RubricC1) Python Benefits and Why Choose Logistic Regression7

***Part III: Data Preparation***

[Describe](#RubricC4) Data Preparation, Summary Statistics, Visualizations and Code 8

Part IV: Model Comparison and Analysis

Compare Initial and Reduced Models, Justify Variable Selection 15

Provide Reduced Logistic Regression Models15

Analyze Data Using Reduced Mode, Explain with Residual Error15

Part V: Data Summary and Implications

Summarize 16

Part VI: Demonstration

[Video](#RubricF) 17

[Sources for Third-Party Code](#RubricG) 17

[Sources](#RubricH) 18

# Hospital Readmission Problem

For our chain of hospitals to lower readmission concerns, we need to identify patients who have increased risk of rehospitalization within a month of their release. According to Schuller (2020), non-obese adults were 21% less likely to be readmitted than obese adults. A readmission study by Gert, et. al. (2002) showed a correlation between longer initial hospital stays and readmission. Within the provided dataset, I’m leveraging these studies to help create my hypothetical question and shape my approach in finding potential patient groups with a statistically significant chance for readmission outcomes.

After viewing the provided medical\_clean.csv data set and accompanying data dictionary, there seems to be some patient groupings which are aligned with the research mentioned above. For instance, the following patient data fields: Initial\_days, Initial\_admin and Diabetes seem like they should correlate with a patient’s readmission. While my initial feelings towards these variables might make them feel related, are they?

## A1 - Research Question

Do the following three predictors: Initial Days, Initial Admin, and Diabetes have an influence on the probability of a patient’s readmission?

**A2 - Objectives and Goals**

The goal of our analysis is to logically investigate given data to support or reject the hypothesis. Some data will need to be converted from categorical to numerical data types in order to analyze and process. Our objective is to see how, if at all, a patient’s initial inpatient days, the reason for their initial admission or if they have diabetes correlate with their readmission within 30 days.

**B1 – Summary of Assumptions**

Applying logistic regression to our model will be based on some assumptions. This regression’s dependent variable is binary and therefore based on the Bernoulli distribution. Predicted values are restricted to a range of nominal (categorical) values like yes, no, pass, fail, win, lose, small, medium, large. Predictions will be a probability of a particular outcome rather than the outcome itself. Predictors will not have a high correlation (multicollinearity). The logistic regression approach is the logarithm of odds in achieving 1. Our regression model will help understand the probability of patient readmission (the dependent variable) based on the following independent predictor variables:

* Initial Days (as inpatient)
* Initial Reason for Admission (emergency, elective or observational)
* If the patient has diabetes

## B2 – Tool Benefits

Python and the chosen libraries (pandas, numpy, seaborn, matplotlib, statsmodels, sklearn) were all chosen to help structure, clean, transform, manipulate, analyze, and visualize the data set quickly, accurately, and efficiently.

**B3** – **Appropriate Technique**

Logistic regression was chosen to predict a binary response, e.g. the probability of patient readmission. An equation of ) will be utilized. Understanding chosen predictor variable relationships could help our hospital chain better predict potential patient care and readmissions.

Chart

Description automatically generated

Figure – Heat map of Data Field Correlation Tests (X)

As you can see above in the Figure 1 heatmap, the current set of data fields needs to be reduced. By viewing the display of significant correlation with other data fields, comparing to the hypothesis, and later viewing the series p-value, the data frame will by statistically reduced.

**C1** – **Data Goals**

From the original data set of 10,000 rows and 50 columns, attention was focused on discarding fields which did not address and/or correlate with the hypothesis. To understand the data, df.info() was run which provided range, shape, completeness (all columns contained Non-Null data) and field data types. After verifying data had no nulls (Figure 2), df.describe() was used to provide a descriptive statistical summary of the field’s central tendency and dispersion. I split the data frame into numeric and categorical subsets (df\_num and df\_cat respectively), and used pandas describe() method to help me get a feel of the data.

|  |  |
| --- | --- |
| A picture containing text  Description automatically generated | Text  Description automatically generated |

Figure - Looking for Nulls

Next, I started to drop what I felt was obviously not helpful data and unrelated to the hypotheses. Next, pd.get\_dummies() converted predictive categorical data to separate fields based on their category. For example, Diabetes became Diabetes\_No and Diabetes\_Yes, containing ones and zeros respectively. Then, for all binary categories, I dropped the ‘\_No’ columns to remove chance of multicollinearity (Figure 3).

Table

Description automatically generated with low confidence

Figure - Pruning Data, Convert Categorical Fields to Numerical, Address Multicollinearity

**A screenshot of a computer screen

Description automatically generated with medium confidence**

Figure - Summary Statistics for the Data Frame

C2 – Summary Statistics

As stated in the Data Goals, to answer the selected research question, the target variable and all predictor variables remained in the data frame to analyze. The pandas describe() method was used to provide summary statistics (Figure 4). From the describe() method, we can see 10,000 observations across all fields. For instance, when viewing the following target and predictor variable columns:

* Initial Days – 10,000 observations, a mean of 34.455299, a standard deviation of 26.309341, minimum value of 1.001981, 25% Percentile: 7.896215, 50% Percentile: 35.836244, 75% Percentile: 61.161020, and maximum value: 71.981490
* Diabetes – Count of 10,000 observations, 2 Unique elements (Yes/No), Top = ‘No’, Frequency of 7,262 No’s
* Initial Admin – Count of 10,000 observations, 3 Unique elements (Emergency, Elective and Observation), Top = ‘Emergency’, Frequency of 5060

**C3 – Steps to Prepare Data**

In addition to some cleaning, wrangling, removing and transformation of data (Figure 2 and 3), I wanted to view each data series p-values (Figure 5).

A picture containing table

Description automatically generated

Figure - Set up and Run the Logistic Regression Model, Compare P-Values to 0.05 Threshold

By comparing the p-value of each series to a standard cutoff threshold of 0.05, I removed the data fields that did not fit the model (Figure 5).

Text

Description automatically generated

Figure - Set up and Run the Logistic Regression Model, Only With Data Meeting the P-Values to 0.05 Threshold

Next, the pruned data set was ready to be set for the next phase of training and testing. A variable ‘X’ was set to the remaining predictor variables meeting the p-value threshold and a variable ‘y’ was set to the target variable. A constant variable was also added to the set of ‘X’ predictor variables, which creates a column with the value of 1.0. Then the logistic regression model was run (Figure 6) and split into a training size of 70% (7,000 rows) and a test size of 30% (3,000 rows).

A screenshot of a computer

Description automatically generated with medium confidence

Figure - Pruned Test Data Results

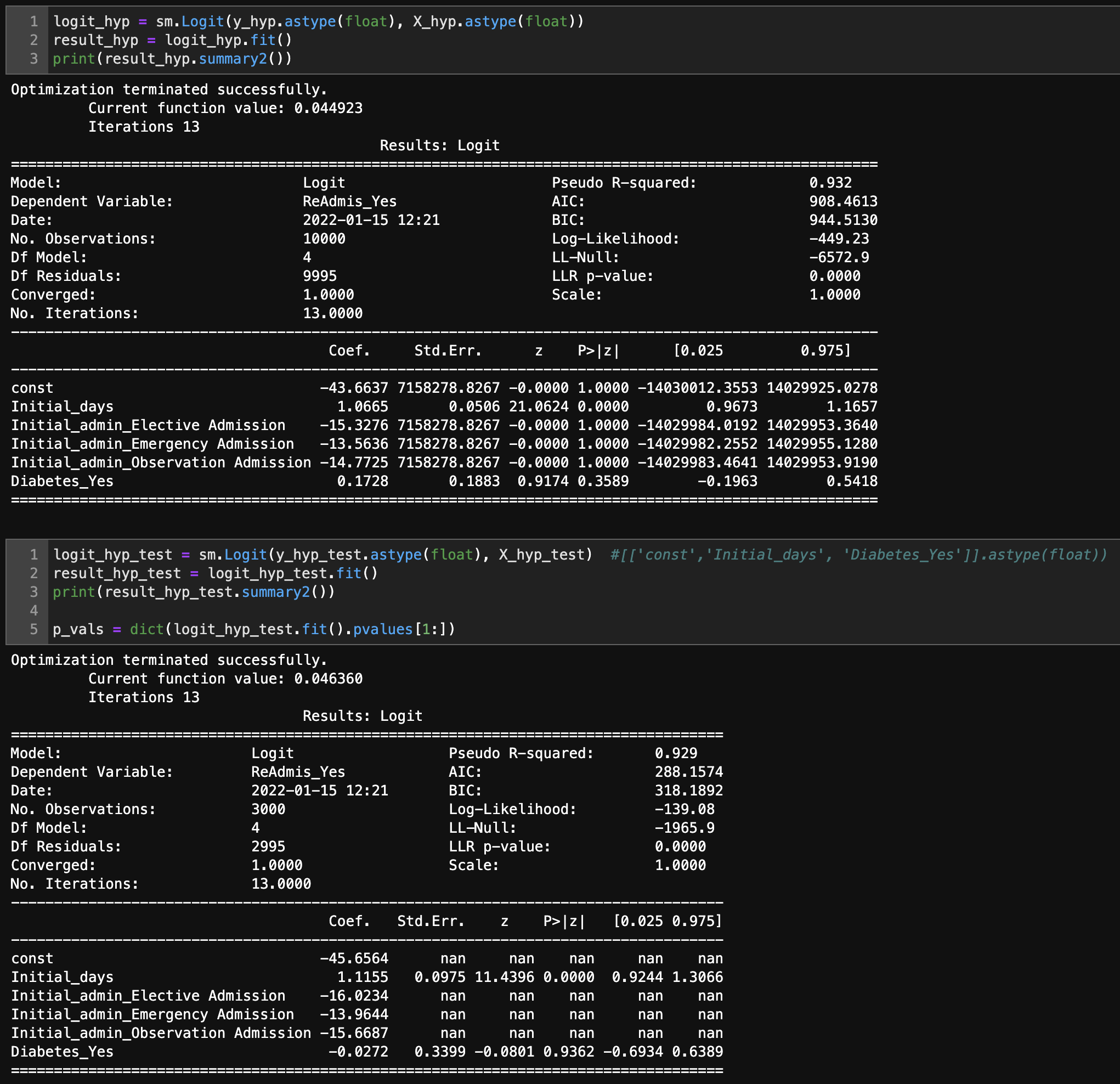
Once run, test results were compared to p-values to a threshold of 0.05 cutoff. As you can see in Figure 7, not all test results met the cutoff as in the previous logistic regression model. 

Figure - Logistic Regression Run with Hypothesis Variables and Compared to P-Value Threshold

C4 – Visualizations

The following univariate (Table 1) and bivariate (Table 2) visualizations are from the cleaned and reduced data set. The bivariate visualization includes the target variable.

|  |  |  |
| --- | --- | --- |
|  |  |  |

Table - Univariate Visualizations

|  |  |
| --- | --- |
|  |  |
|  |  |
|  | |

Table - Bivariate Visualizations

**C5: Prepared Data Set**

Saved within the provided folder is the “JWillis\_ D208\_ PA2\_LogReg.ipynb” file, demonstrating the steps taken to explore and analyze the provided data set from the “medical\_clean.csv” file. Once Jupyter Notebook or JupyterLab is running, navigate to the open the “JWillis\_ D208\_ PA2\_LogReg “ folder. From there, you will find the JWillis\_ D208\_ PA2\_LogReg.ipynb file and the “medical\_clean.csv” file. Run all cells by navigating to Cells 🡪 ‘Run All’ in Jupyter Notebook or View 🡪 ‘Run All Cells’ from within JupyterLab. The first step is to import libraries and read the raw data from the CSV. Once the libraries are imported, the data is read into the notebook. Ensure the medical\_raw\_data.csv file is kept in the same folder or provide the correct path when reading in the data.

**D1: Initial Model**

The submission provides an accurate initial logistic regression model from all predictors identified in the hypothesis.

**D2** – **Justification of Model Reduction**

The model’s variables were further reduced (Figure 6 and 7) after initially running the model, comparing variable p-values to a threshold of 0.05, retaining values that were lower than the cutoff, and rerunning the model.

**D3 –Reduced Logistic Regression Model**

Submission provides a reduced logistic regression model, including categorical and continuous variables, aligned with the hypothesis.

**E1 – Model Comparison**

Using stepwise regression helped logically choose which variables to keep and which variables to remove. As seen in Figure 6 and 7, while looking at the “P > |t|” column and comparing coefficients against a p-value of 0.05, I kept significant variables that were less than this threshold. Additionally, only Initial\_days made the training model threshold from our original hypothesis. Meaning, we would have to accept the null hypothesis as the other predictor variables (Initial\_admin and Diabetes\_Yes) were all above a p-value threshold of 0.05. Furthermore, I specifically ran and tested only variables from the hypothesis (Figure 8) showing out of the original variables, only Initial\_days met the model criteria in both the original run and the training/testing run.

**E2 – Output and Calculations**

**From Reading:**

**The parametric form of the model:**

**The descriptive form:**

From the file listed above and/or from Figure 9 or Figure 10, this data is inserted into the final regression equation here:

*y’* = dependent variable (estimated value of target y) = Total Charges (Total\_charge)

*X* = independent variable(s) (predictors) = Initial Days (Initial\_days), Readmission (transformed to ReAdmis\_Num), Diabetes (transformed to Diabetes\_Num)

*exp* =

*β* = *β0* = intercept*, while β1, β2, β3* … = known values of the regression coefficients

**E3 – Code**

Code is provided in the JWillis\_D208\_PA2\_LogReg.ipynb file.

**F1 – Results**

**From Data Science Using Python and R (reading):**

**The parametric form of the model:**

**The descriptive form:**

From the file listed above and/or from Figure 9 or Figure 10, this data is inserted into the final regression equation here:

*y’* = dependent variable (estimated value of target y) = Readmission

*X* = independent variable(s) (predictors) = Initial Days (Initial\_days), Initial\_admin (Elective, Emergency, Observation transformed using get\_dummies()), Diabetes (transformed to Diabetes\_Yes)

*exp* = exponential of the term

*β* = *β0* = intercept*, while β1, β2, β3* … = known values of the regression coefficients

Interpretation of most statistically significant coefficients: After running most variables through the model, p-values were compared to a commonly used cutoff threshold of 0.05. Fields with p-values lower than the cutoff were kept in the model. Since the model utilizes coefficients with respective p-values providing a significant statistical relationship to the target variable and some not providing a significant statistical relationship, we must accept the null hypothesis. Meaning, the probability needed to show a significant relationship between the predictor and target variables was not observed.

Statistical and practical significance of the reduced model (as related to the research question): A reduced model containing most data fields: Initial\_days, and Diabetes variables met the cutoff threshold in our initial model. Additionally, the category variable for Diabetes was transformed to numerical output and added to the model as they, too, passed the p-value cutoff threshold.

Limitations of study: While we have a mathematical cutoff threshold set for this logistic regression model, in reality, there may not be correlation between a, some or all predictor variables and the target variable. Correlation doesn’t always equal causation, especially in the case of correlated predictor variables.

Chosen predictor variables within the hypothesis did not provide significant statistical relation to the target variable. What the logistic model shows, since we need to accept the null hypothesis, is that we should reframe the question and rerun the tests. For instance, if a hypothesis was created with predictor variables focused on Initial\_days, Anxiety\_Yes and Stroke\_Yes, (Figure 9) then we could reject the null hypotheses since these variables do show a significant statistical relationship to a patient’s readmission probability.

Text

Description automatically generated

Figure - Alternative Predictor Variables

**F2 – Recommendations**

Looking at the Logistic model, based on the coefficients, the three predictor variables (Initial\_days, Initial\_admin (type: elective, emergency and observation), and Diabetes\_Yes) are not all positively and significantly (due to p-values) correlated with the response variable of Readmissions. Therefore, we did not observe there evidence where longer hospital stays, the initial reason for administration, and diabetes are all predictors of (and cannot determine) a patients readmission within 30 days of being discharged from the hospital.

Here is our recommended course of action: We should rework the hypothesis to focu on predictor variables that pass the threshold testing. For instance, Figure 9 shows an example of how a patients initial days, anxiety and history of stroke could help predict readmissions. Further subsets could also be explored to help predict readmissions, reduce the length of their hospital stays and overall costs while freeing up beds. While focusing on a new subset of patients, we should also recommend that patients reduce their inpatient stay in the hospital for as little time as possible, i.e. avoid getting readmitted if at all possible, and to manage, treat, or minimize their diabetes symptoms as much as possible. Prevention and continued/improved healthier living behaviors should be recommended as lifestyle implementations for this subset of patients in order to reduce additional charges through their initial inpatient days, diabetes, and the possibility of readmission.

**G – Panopto Demonstration**

Panopto video will be uploaded after this submission is reviewed

**H – Sources for Third-Party Code**

* Help using Markdown: <https://www.markdownguide.org/basic-syntax/>
* Help to see ALL columns: <https://stackoverflow.com/questions/24524104/pandas-describe-is-not-returning-summary-of-all-columns>
* Help to create a better histogram design: <https://mode.com/example-gallery/python_histogram/>
* Python Help: <https://docs.python.org/3.9/library/index.html>
* Pandas Help: <https://pandas.pydata.org/docs/user_guide/index.html#user-guide>
* Numpy Help: <https://numpy.org/doc/stable/>
* Seaborn Help: <https://seaborn.pydata.org/>
* Matplotlib Help: <https://matplotlib.org/>
* Sklearn Help: <https://scikit-learn.org/>

## References

Larose, D., C., & Larose, D., T. (2019). Data Science Using Python and R. Wiley. <https://www.wiley.com/en-us/Data+Science+Using+Python+and+R-p-9781119526810>

Gert P Westert, Ronald J Lagoe, Ilmo Keskimäki, Alastair Leyland, Mark Murphy,

An international study of hospital readmissions and related utilization in Europe and the USA,

Health Policy, Volume 61, Issue 3, 2002, Pages 269-278, ISSN 0168-8510, <https://doi.org/10.1016/S0168-8510(01)00236-6>. (<https://www.sciencedirect.com/science/article/pii/S0168851001002366>)

Schuller K. A. (2020). Is obesity a risk factor for readmission after acute myocardial infarction?  *Journal of healthcare quality research*, *35*(1), 4–11. <https://doi.org/10.1016/j.jhqr.2019.09.002>