WGU MSDA Program

Logistic Regression Prediction Analysis on Patient Data

Write Up

Christopher Kamper

April 23rd, 2023

**Part I: Research Question**

1. Research question:
   1. What variables best predict if a patient will be readmitted within a month after release from their initial visit?
   2. The goal of the analysis is to determine if a patient can be expected to be readmitted within one month of their initial visit based on variables that would be available during the admission process. This can be accomplished using tools within the Python language. The variables that will be used could logically influence if a patient is readmitted. These explanatory variables are:
      * 1. Age
        2. Gender
        3. VitD\_levels
        4. Soft\_drink
        5. Initial\_admin
        6. HighBlood
        7. Stroke
        8. Complication\_risk
        9. Overweight
        10. Arthritis
        11. Diabetes
        12. Hyperlipidemia
        13. BackPain
        14. Anxiety
        15. Allergic\_rhinitis
        16. Reflux\_esophagitis
        17. Asthma

**Part II: Method Justification**

1. Logistic Regression Method
   1. Four assumptions that must be made:
      1. The dependent variables must be binary. (“Assumptions of Logistic Regression”)
      2. Independent variables must be independent of each other. (“Assumptions of Logistic Regression”)
      3. Must be little to no multicollinearity between independent variables. (“Assumptions of Logistic Regression”)
      4. Requires a large sample size. (“Assumptions of Logistic Regression”)
   2. Python is a great choice to use for this analysis for this research as it is a comprehensive language allowing all steps to be run within the same environment. It also enables the analyst to provide strong visuals to help present the findings and visualize the data sets used.
   3. Logistic regression is the appropriate technique to answer this because the dependent variable referenced is a categorical variable. This method will properly handle what is needed for this dependent variable compared to linear regression.

**Part III: Data Preparation**

1. Data preparation process for logistic regression
   1. The goal of the cleaning process here will be to ensure that all columns have values present and that they are represented in a manner that can be used in the multiple logistic regression model if needed. This will ensure that the model is as accurate as possible.

First, a check will be conducted for duplicated values based on the unique ID, “Customer\_id,” provided in the dictionary. Example of the code used:

medical\_df = medical\_df.drop\_duplicates(subset='Customer\_id', keep='first')

Then, any null values will be detected. Example of the code used:

medical\_df.isnull().sum()

Next, the column headers provided with naming conventions such as “Item1” will be updated to reflect the survey notation provided in the dictionary. Example of the code used:

medical\_df.rename(columns = {'Item1':'Timely\_admission',

'Item2':'Timely\_treatment',

'Item3':'Timely\_visits',

'Item4':'Reliability',

'Item5':'Options',

'Item6':'Hours\_of\_treatment',

'Item7':'Courteous\_staff',

'Item8':'Evidence\_of\_active\_listening\_from\_doctor'},

inplace=True)

Then, the categorical values will be reviewed to see the unique values present for each. Then each unique value will be transformed into dummy variables. For any variables that only have two outcomes, one dummy variable (in this case the “No”) will be removed. This will allow the models to utilize these values while maintaining the integrity of their data. Example of the code used (repeated for each categorical variable in the dataset):

# Create dummies for the Soft\_drink variable

medical\_df\_prep = pd.get\_dummies(medical\_df\_prep, prefix="Soft\_drink", columns=['Soft\_drink'])

medical\_df\_prep = medical\_df\_prep.drop('Soft\_drink\_No', axis=1)

medical\_df\_prep = medical\_df\_prep.rename(columns={'Soft\_drink\_Yes': "Soft\_drink\_Yes"})

medical\_df\_prep.info()

* 1. Descriptions of variables to be used in the model are below.
     1. Dependent Variable
        1. ReAdmis\_Yes:

A picture containing text

Description automatically generated

* + 1. Independent Variables (Explanatory)
       1. Age

Text

Description automatically generated

* + - 1. Gender\_Female:

Text

Description automatically generated with medium confidence

* + - 1. Gender\_Male:

Text

Description automatically generated with medium confidence

* + - 1. Gender\_Nonbinary:

Text

Description automatically generated

* + - 1. VitD\_levels:

Text, letter

Description automatically generated

* + - 1. Soft\_drink\_Yes:

Text

Description automatically generated

* + - 1. Initial\_admin\_Elective:

Text

Description automatically generated

* + - 1. Initial\_admin\_Emergency:

Text

Description automatically generated

* + - 1. Initial\_admin\_Observation:

Text

Description automatically generated with medium confidence

* + - 1. HighBlood\_Yes:

A picture containing text

Description automatically generated

* + - 1. Stroke\_Yes:

Text

Description automatically generated

* + - 1. Complication\_risk\_High:

Text

Description automatically generated

* + - 1. Complication\_risk\_Medium:

Text

Description automatically generated

* + - 1. Complication\_risk\_Low:

A picture containing text

Description automatically generated

* + - 1. Overweight\_Yes:

Text

Description automatically generated with medium confidence

* + - 1. Arthritis\_Yes:

A picture containing text

Description automatically generated

* + - 1. Diabetes\_Yes:

A picture containing text

Description automatically generated

* + - 1. Hyperlipidemia\_Yes:

Text, letter

Description automatically generated with medium confidence

* + - 1. BackPain\_Yes:

Text

Description automatically generated with low confidence

* + - 1. Anxiety\_Yes:

Text

Description automatically generated

* + - 1. Allergic\_rhinitis\_Yes:

A picture containing text

Description automatically generated

* + - 1. Reflux\_esophagitis\_Yes:

Text

Description automatically generated

* + - 1. Asthma\_Yes:

A picture containing table

Description automatically generated

From these summary statistics, it can be noted that all variables in the data set look as expected. The ones that are categorical including all dummy variables, all have values that make logical sense. We can tell what category most of the data for each variable falls into from the mean. The mean values below 0.5 tell us the majority is “no” while the values between 0.5 and 1.0 are the majority “yes”. For the categorical variables that were split into more than two dummy variables, we can see which is the majority to keep for our model to solve the collinearity issue. Gender, for example, shows a mean of 0.502 for female, 0.477 for male, and 0.021 for non-binary. These values can be seen as a percentage of the total gender columns showing that the majority is female. The same process can then be applied to pick the dummy variables to keep for Initial admin and Complication risk.

* 1. Univariate and bivariate analysis visualizations are below.
     1. Dependent Variable
        1. ReAdmis\_Yes
           1. Box Plot:

Chart

Description automatically generated with medium confidence

* + - * 1. Histogram:

Chart, histogram

Description automatically generated

* + - * 1. Density Plot:

Chart, histogram

Description automatically generated

* + 1. Independent Variables (Explanatory)
       1. Age
          1. Box Plot

Chart, box and whisker chart

Description automatically generated

* + - * 1. Histogram

Chart, histogram

Description automatically generated

* + - * 1. Density Plot

Chart

Description automatically generated

* + - * 1. Linear Regression

Chart, rectangle

Description automatically generated

* + - 1. Gender\_Female:
         1. Box Plot

A picture containing chart

Description automatically generated

* + - * 1. Histogram

Chart, histogram

Description automatically generated

* + - * 1. Density Plot

Chart, histogram

Description automatically generated

* + - * 1. Linear Regression

Chart, histogram

Description automatically generated

* + - 1. Gender\_Male:
         1. Box Plot

A picture containing chart

Description automatically generated

* + - * 1. Histogram

Chart, histogram

Description automatically generated

* + - * 1. Density Plot

Chart, histogram

Description automatically generated

* + - * 1. Linear Regression

Chart, histogram

Description automatically generated

* + - 1. Gender\_Nonbinary:
         1. Box Plot

Graphical user interface

Description automatically generated with medium confidence

* + - * 1. Histogram

Chart, histogram

Description automatically generated

* + - * 1. Density Plot

Chart, histogram

Description automatically generated

* + - * 1. Linear Regression

Chart

Description automatically generated

* + - 1. VitD\_levels:
         1. Box Plot

Chart, box and whisker chart

Description automatically generated

* + - * 1. Histogram

Chart, histogram

Description automatically generated

* + - * 1. Density Plot

Chart

Description automatically generated

* + - * 1. Linear Regression

Chart

Description automatically generated

* + - 1. Soft\_drink\_Yes:
         1. Box Plot

Chart

Description automatically generated with medium confidence

* + - * 1. Histogram

Chart, histogram

Description automatically generated

* + - * 1. Density Plot

Chart, histogram

Description automatically generated

* + - * 1. Linear Regression

Chart, histogram

Description automatically generated

* + - 1. Initial\_admin\_Elective:
         1. Box Plot

Chart

Description automatically generated with low confidence

* + - * 1. Histogram

Chart, histogram

Description automatically generated

* + - * 1. Density Plot

Chart, histogram

Description automatically generated

* + - * 1. Linear Regression

Chart, histogram, box and whisker chart

Description automatically generated

* + - 1. Initial\_admin\_Emergency:
         1. Box Plot

A picture containing chart

Description automatically generated

* + - * 1. Histogram

Chart, histogram

Description automatically generated

* + - * 1. Density Plot

Chart, histogram

Description automatically generated

* + - * 1. Linear Regression

Chart

Description automatically generated

* + - 1. Initial\_admin\_Observation:
         1. Box Plot

Graphical user interface

Description automatically generated with medium confidence

* + - * 1. Histogram

Chart, histogram

Description automatically generated

* + - * 1. Density Plot

Chart, histogram

Description automatically generated

* + - * 1. Linear Regression

Chart, histogram

Description automatically generated

* + - 1. HighBlood\_Yes:
         1. Box Plot

Chart

Description automatically generated with medium confidence

* + - * 1. Histogram

Chart, histogram

Description automatically generated

* + - * 1. Density Plot

Chart, histogram

Description automatically generated

* + - * 1. Linear Regression

Chart, histogram

Description automatically generated

* + - 1. Stroke\_Yes:
         1. Box Plot

Graphical user interface

Description automatically generated with low confidence

* + - * 1. Histogram

Chart, histogram

Description automatically generated

* + - * 1. Density Plot

Chart, histogram

Description automatically generated

* + - * 1. Linear Regression

Chart, histogram, box and whisker chart

Description automatically generated

* + - 1. Complication\_risk\_High:
         1. Box Plot

Chart

Description automatically generated with low confidence

* + - * 1. Histogram

Chart, histogram

Description automatically generated

* + - * 1. Density Plot

Chart, histogram

Description automatically generated

* + - * 1. Linear Regression

Chart

Description automatically generated

* + - 1. Complication\_risk\_Medium:
         1. Box Plot

A picture containing chart

Description automatically generated

* + - * 1. Histogram

Chart, histogram

Description automatically generated

* + - * 1. Density Plot

Chart, histogram

Description automatically generated

* + - * 1. Linear Regression

Chart

Description automatically generated

* + - 1. Complication\_risk\_Low:
         1. Box Plot

Graphical user interface

Description automatically generated with medium confidence

* + - * 1. Histogram

Chart, histogram

Description automatically generated

* + - * 1. Density Plot

Chart, histogram

Description automatically generated

* + - * 1. Linear Regression

Chart, histogram

Description automatically generated

* + - 1. Overweight\_Yes:
         1. Box Plot

A picture containing chart

Description automatically generated

* + - * 1. Histogram

Chart, histogram

Description automatically generated

* + - * 1. Density Plot

Chart, histogram

Description automatically generated

* + - * 1. Linear Regression

Chart, histogram

Description automatically generated

* + - 1. Arthritis\_Yes:
         1. Box Plot

Chart

Description automatically generated with medium confidence

* + - * 1. Histogram

Chart, histogram

Description automatically generated

* + - * 1. Density Plot

Chart, histogram

Description automatically generated

* + - * 1. Linear Regression

Chart, histogram

Description automatically generated

* + - 1. Diabetes\_Yes:
         1. Box Plot

Chart

Description automatically generated with low confidence

* + - * 1. Histogram

Chart, histogram

Description automatically generated

* + - * 1. Density Plot

Chart, histogram

Description automatically generated

* + - * 1. Linear Regression

Chart, histogram

Description automatically generated

* + - 1. Hyperlipidemia\_Yes:
         1. Box Plot

Chart

Description automatically generated with medium confidence

* + - * 1. Histogram

Chart, histogram

Description automatically generated

* + - * 1. Density Plot

Chart, histogram

Description automatically generated

* + - * 1. Linear Regression

Chart

Description automatically generated

* + - 1. BackPain\_Yes:
         1. Box Plot

Chart

Description automatically generated with medium confidence

* + - * 1. Histogram

Chart, histogram

Description automatically generated

* + - * 1. Density Plot

Chart, histogram

Description automatically generated

* + - * 1. Linear Regression

Chart, histogram

Description automatically generated

* + - 1. Anxiety\_Yes:
         1. Box Plot

Chart

Description automatically generated with low confidence

* + - * 1. Histogram

Chart, histogram

Description automatically generated

* + - * 1. Density Plot

Chart, histogram

Description automatically generated

* + - * 1. Linear Regression

Chart

Description automatically generated

* + - 1. Allergic\_rhinitis\_Yes:
         1. Box Plot

Chart

Description automatically generated with medium confidence

* + - * 1. Histogram

Chart, histogram

Description automatically generated

* + - * 1. Density Plot

Chart, histogram

Description automatically generated

* + - * 1. Linear Regression

Chart, histogram

Description automatically generated

* + - 1. Reflux\_esophagitis\_Yes:
         1. Box Plot

A picture containing chart

Description automatically generated

* + - * 1. Histogram

Chart, histogram

Description automatically generated

* + - * 1. Density Plot

Chart, histogram

Description automatically generated

* + - * 1. Linear Regression

Chart

Description automatically generated

* + - 1. Asthma\_Yes:
         1. Box Plot

Chart

Description automatically generated with medium confidence

* + - * 1. Histogram

Chart, histogram

Description automatically generated

* + - * 1. Density Plot

Chart, histogram

Description automatically generated

* + - * 1. Linear Regression

Chart, histogram

Description automatically generated

* 1. The goal of transformation is simply to ensure everything is numeric and easier to understand as possible. Steps were taken to rename columns to ensure clarity and convert the categorical variables to numeric ones so they can be included in the analysis. These steps will allow the easiest and best way to create a model to answer the question.
  2. The cleaned and prepared dataset is attached as ‘medical\_df\_prep.csv’.

**Part IV: Model Comparison and Analysis**

* 1. Initial Multiple Logistical Regression model using all independent variables from C2 is below.

# Define the X and y variables

X\_initial = medical\_df\_prep[['Age', 'VitD\_levels', 'Gender\_Female', 'Soft\_drink\_Yes', 'Initial\_admin\_Emergency', 'HighBlood\_Yes', 'Stroke\_Yes', 'Complication\_risk\_Medium', 'Overweight\_Yes', 'Arthritis\_Yes', 'Diabetes\_Yes', 'Hyperlipidemia\_Yes', 'BackPain\_Yes', 'Anxiety\_Yes', 'Allergic\_rhinitis\_Yes', 'Reflux\_esophagitis\_Yes', 'Asthma\_Yes']]

y = medical\_df\_prep['ReAdmis\_Yes']

# Split X and y into training and testing sets (75% for training, 25% for testing)

X\_initial\_train, X\_initial\_test, y\_train, y\_test = train\_test\_split(X\_initial, y, test\_size=0.25, random\_state=9)

# Set up the initial model

model\_initial = LogisticRegression(random\_state=9)

# Fit the initial model

model\_initial.fit(X\_initial\_train, y\_train)

# Run the fit model

y\_pred\_initial = model\_initial.predict(X\_initial\_test)

print("The classes for the result of the initial model are: ", (model\_initial.classes\_))

print("The intercept of the intitial model equation is: ", (model\_initial.intercept\_))

print("The coefficients of the initial model equation for each variable respecively is: ", (model\_initial.coef\_))

model\_initial.predict\_proba(X\_initial)

model\_initial.predict(X\_initial)

print("The initial model accuracy score is: ", (model\_initial.score(X\_initial, y)))

Text, letter

Description automatically generated

The model is returning the class results that are expected to see of 0 for “No” and 1 for “Yes”. The result shows the intercept value along with all coefficients for each variable in the model. The model accuracy score was decent, showing confidence in its results most of the time. Logically, there appears to be room for improvement as the accuracy score could be higher and considering the number of variables in the model, one could expect it can be reduced.

* 1. The initial model was set up using variables that could be known at the time of check-in for a patient related to their health. No variables that would result from treatment or treatment itself were included. The reduced model was created using the Recursive Feature Elimination (RFE) process from a base set of the variables chosen in the initial model. This process reviews each variable (feature) and selects the most important ones for the model by assigning a weight to each as a method of measuring importance. The estimator trains on the initial set of variables and then repeats to select the most important n values of variables. In this case, n was set to 2 as this would be the most efficient model possible if the accuracy score matches. (“Sklearn.feature\_selection.RFE”)
     1. Variables removed from initial model:
        1. Age
        2. VitD\_levels
        3. Gender\_Female
        4. Soft\_drink\_Yes
        5. HighBlood\_Yes
        6. Stroke\_Yes
        7. Complication\_risk\_Medium
        8. Overweight\_Yes
        9. Arthritis\_Yes
        10. Diabetes\_Yes
        11. Hyperlipidemia\_Yes
        12. BackPain\_Yes
        13. Anxiety\_Yes
        14. Allergic\_rhinitis\_Yes
        15. Reflux\_esophagitis\_Yes
     2. Variables selected by the process to use in the reduced model:
        1. Initial\_admin\_Emergency
        2. Asthma\_Yes
  2. Reduced Multiple Logistical Regression model using the independent variables from the RFE process is below.

reduced\_vars = X\_initial

X\_reduced = [i for i in reduced\_vars if i not in y]

modelr = LogisticRegression()

rfe = RFE(modelr, n\_features\_to\_select=2, step=1)

rfe = rfe.fit(X\_initial, y)

variables =[]

print("The variables selected for the reduced model are: ")

for i in range (X\_initial.shape[1]):

if rfe.support\_[i] ==True:

variables.append(X\_initial.columns[i])

print("Column: %d, Rank: %.2f, Variable: %s" %

(i, rfe.ranking\_[i], X\_initial.columns[i]))

Text

Description automatically generated

# Define the X and y variables

X\_reduced = medical\_df\_prep[['Initial\_admin\_Emergency', 'Asthma\_Yes']]

y = medical\_df\_prep['ReAdmis\_Yes']

# Split X and y into training and testing sets (75% for training, 25% for testing)

X\_reduced\_train, X\_reduced\_test, y\_train, y\_test = train\_test\_split(X\_reduced, y, test\_size=0.25, random\_state=9)

# Set up the reduced model

model\_reduced = LogisticRegression(random\_state=9)

# Fit the reduced model

model\_reduced.fit(X\_reduced\_train, y\_train)

# Run the fit model

y\_pred\_reduced = model\_reduced.predict(X\_reduced\_test)

print("The classes for the result of the reduced model are: ", (model\_reduced.classes\_))

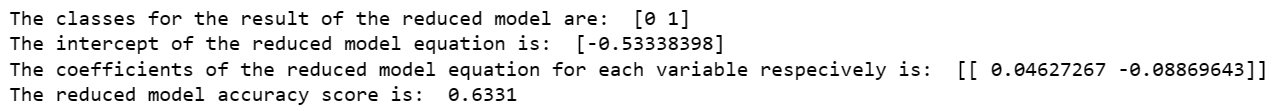
print("The intercept of the reduced model equation is: ", (model\_reduced.intercept\_))

print("The coefficients of the reduced model equation for each variable respecively is: ", (model\_reduced.coef\_))

model\_reduced.predict\_proba(X\_reduced)

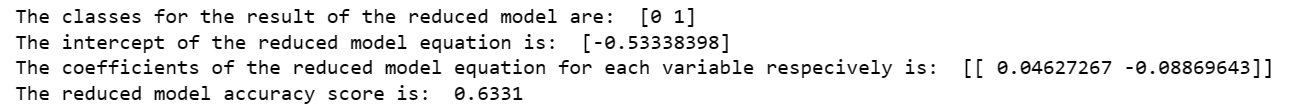
model\_reduced.predict(X\_reduced)

print("The reduced model accuracy score is: ", (model\_reduced.score(X\_reduced, y)))



The reduced model is showing exactly what was hoped. As previously mentioned, the n value of the RFE process was chosen to be the most efficient model by using as few variables as possible to still present the same quality of the model as the initial. This is achieved as the model accuracy score of the reduced model matches that of the initial but now only using two variables.

1. Analysis:
   1. The initial and reduced models were compared by looking at the model accuracy scores and confusion matrices. The initial model included all variables from C2 and ended with an accuracy score of 0.6331 (~63%). The reduced model was able to match the same accuracy but with only 2 variables showing an improvement in the efficiency without losing any predicting power. In looking at the confusion matrices, they do match between the two models showing that nothing has changed from the results of the initial and the reduced despite the change in variables.
   2. Reduced model calculations are below.
      1. Reduced model results:



The reduced model results show that the classes of the results are 0 for “No” and 1 for “Yes” as expected. The intercept and coefficients for each variable are clear. The model accuracy score does match the initial model showing that despite the lower count of variables, the model still is predicting results the same.

* + 1. Reduced model confusion matrix:

Chart, treemap chart

Description automatically generated

The confusion matrix for the reduced model actually matches the initial model’s. This is showing that of the 2500 test instances, the model is correctly predicting that 1600 are actually “No” and 0 are “Yes”. There are 900 instances were the model is incorrectly predicting “No”. The overall accurate predictions then are 1600 and 900 inaccurate. This equates to the ~63% accuracy score.

* + 1. Model accuracy: 0.6331

The reduced model accuracy score matches the initial model accuracy score.

* 1. Copy of code used is attached.

**Part V: Data Summary and Implications**

F. Summary of the findings:

1. The reduced logistic regression model equation is:

The coefficients for each variable in this model are shown above. These will show that the Asthma\_Yes variable has the largest effect on the model as it has the greatest absolute valued coefficient. Initial\_Admin\_Emergency is the variable with the smallest valued coefficient so this will have the smallest effect on the model.

The model above predicted a correct “No” value about 63% of the time. This is shown in the confusion matrix where one can see that there are 1600 instances of a correct “No” prediction versus the 900 incorrect “No” predictions. This means it will be correct at predicting a statistical majority of the time.

The analysis here is limited though as it does not consider other variables that could affect a patient’s readmittance after they have been checked in. This is focused on variables that are from information that would be available during a patient’s check-in. The model is also dependent on the preparation steps taken in this specific analysis. Other methods may represent some values differently after the data set is prepared. Outliers could also play a role if inputted as regression is a method that is not resistant to them.

1. Based on the results, the recommendation would be that this model can be used since it is correct a statistical majority of the time. It would not be recommended as the only metric for prediction, but one can be confident it would be accurate most of the time referring to the confusion matrix showing 1600 correct predictions versus 900 incorrect. Practically, it then could be used to predict if a patient will not be readmitted soon based on the variables used. This may be useful to use right after a patient has been admitted, with the knowledge that other factors between the admission of a patient and their release may affect whether or not they are readmitted with more significance.

**Part VI: Demonstration**

1. Here are the third-party sites used to aid in creating the code used in this analysis:

Li, Susan. “Building a Logistic Regression in Python, Step by Step.” *Medium*, Towards Data Science, 29 Sept. 2017, towardsdatascience.com/building-a-logistic-regression-in-python-step-by-step-becd4d56c9c8. Accessed 22 Apr. 2023.

Real Python. “Logistic Regression in Python.” *Realpython.com*, Real Python, 13 Jan. 2020, realpython.com/logistic-regression-python/. Accessed 22 Apr. 2023.

“Sklearn.feature\_selection.RFE.” *Scikit-Learn*, 2023, scikit-learn.org/stable/modules/generated/sklearn.feature\_selection.RFE.html. Accessed 22 Apr. 2023.

1. Here are the sources used to find information related to this analysis:

“Assumptions of Logistic Regression.” *Statistics Solutions*, 11 Aug. 2021, www.statisticssolutions.com/free-resources/directory-of-statistical-analyses/assumptions-of-logistic-regression/. Accessed 22 Apr. 2023.

Avinash Navlani. “Understanding Logistic Regression in Python Tutorial.” *Datacamp.com*, DataCamp, 16 Dec. 2019, www.datacamp.com/tutorial/understanding-logistic-regression-python. Accessed 22 Apr. 2023.

Bex T. “Powerful Feature Selection with Recursive Feature Elimination (RFE) of Sklearn | towards Data Science.” *Medium*, Towards Data Science, 11 May 2021, towardsdatascience.com/powerful-feature-selection-with-recursive-feature-elimination-rfe-of-sklearn-23efb2cdb54e. Accessed 22 Apr. 2023.

“Logistic Regression in Python.” *Realpython.com*, Real Python, 13 Jan. 2020, realpython.com/logistic-regression-python/. Accessed 22 Apr. 2023.