Predictive Modeling – D208

Task 1

Western Governor’s University

Performance Assessment

Matthew Morgan

Student ID: 010471280

11/21/2022

**Part I: Research Question**

A1. What patient demographics and medical conditions lead to longer initial hospital stays?

A2. Ultimately, we want to identify the variables that cause patients to be readmitted. However, when running a bivariate analysis on patient readmissions nothing shows a relationship except initial days. By looking at other variables that might have a relationship with initial days we can find a trend or pattern in patients who end up back in the hospital then we can start looking for ways to reduce readmissions and ultimately reduce the chances of receiving a fine.

**Part II: Method Justification**

B1. Multiple linear regression makes several assumptions. The first being that there must be a linear relationship between the outcome variable and independent variables. It also assumes that the residuals are normally distributed and that the independent variables are not highly correlated with each other.

B2. I have decided to use Python for the entirety of the project. I am familiar with it, was provided with the necessary education on datacamp, and it’s versatile enough to produce all the output needed for this assessment.

B3. Multiple linear regression is used to estimate the relationship between a dependent variable and two or more independent variables using a straight line. It allows us to visualize the relationship between many different variables and determine possible relationships.

**Part III: Data Preparation**

C1. To prepare and manipulate data for this project first we check for missing and duplicate values. None were found. Then we try and convert categorical variables to numerical variables so they can be used for analysis. For this I used pd.get\_dummies to save time and ensure we mitigate multicollinearity.

C2. The target variable I chose was initial\_days, as it is a continuous variable and can be used for Multiple Linear Regression. The predictor variables I chose were the following; vitD\_supp, Children, Income, Full\_Meals\_Eaten, Additional\_charges, TotalCharge, VitD\_levels, Age, Doc\_Visits, HighBlood, Stroke, Arthritis, Diabetes, Hyperlipidemia, BackPain, Allergic\_rhinitis, Reflux\_esophagitis, Asthma, Marital, Services, Gender, Initial\_admin, and Complication\_risk.

I included screenshots of summary statistics below:

|  |
| --- |
| Summary statistics |
|  |

The summary stats show us that the dataset has many continuous variables, due to the mean/max/mode being outside of 0 and 1. This also gives us a plan for starting to re-express our categorical variables (variables that are simply yes/no, or have a defined group) in the next step so begin building models to test our dependent variable.

The summary statistics overall show us that our average patient lives in an area with a population of 9,965 people (with a standard deviation of 14,824) within a mile radius, has 2 children (with a standard deviation of 2.16), has an income of $40k/yr (with a standard deviation of $28,521), eats 1 full meal a day while in the hospital (with a standard deviation of 1), receives $12,934 in additional charges (with a standard deviation of $6,542), spends 34 days on their initial stay in the hospital (with a standard deviation of 26 days), receives $5,312 in total charges (with a standard deviation of $2180), has Vitamin D levels of 17.96 ng/mL upon admission (with a standard deviation of 2), is 53 years old (with a standard deviation of 21) , and is visited by their doctor 5 times during their stay (with a standard deviation of 1).

C3. To prepare the data for analysis I re-expressed some categorical variables on my own, and used pd.get\_dummies to automate one hot encoding of others. Code snippets are below.

|  |
| --- |
| Re-expression of categorical variables |
| #Data Wrangling; turn categorical values into quantitative data  df['ReAdmis\_numeric'] = df['ReAdmis']  dict\_ReAdmis = {"ReAdmis\_numeric": {"No": 0, "Yes": 1}}  df.replace(dict\_ReAdmis, inplace=True)  df['Soft\_drink\_numeric'] = df['Soft\_drink']  dict\_Soft\_drink = {"Soft\_drink\_numeric": {"No": 0, "Yes": 1}}  df.replace(dict\_Soft\_drink, inplace=True)  df['HighBlood\_numeric'] = df['HighBlood']  dict\_HighBlood = {"HighBlood\_numeric": {"No": 0, "Yes": 1}}  df.replace(dict\_HighBlood, inplace=True)  df['Stroke\_numeric'] = df['Stroke']  dict\_stroke = {"Stroke\_numeric": {"No": 0, "Yes": 1}}  df.replace(dict\_stroke, inplace=True)  df['Arthritis\_numeric'] = df['Arthritis']  dict\_arthritis = {"Arthritis\_numeric": {"No": 0, "Yes": 1}}  df.replace(dict\_arthritis, inplace=True)  df['Diabetes\_numeric'] = df['Diabetes']  dict\_diabetes = {"Diabetes\_numeric": {"No": 0, "Yes": 1}}  df.replace(dict\_diabetes, inplace=True)  df['Hyperlipidemia\_numeric'] = df['Hyperlipidemia']  dict\_hyperlipidemia = {"Hyperlipidemia\_numeric": {"No": 0, "Yes": 1}}  df.replace(dict\_hyperlipidemia, inplace=True)  df['BackPain\_numeric'] = df['BackPain']  dict\_backpain = {"BackPain\_numeric": {"No": 0, "Yes": 1}}  df.replace(dict\_backpain, inplace=True)  df['Allergic\_rhinitis\_numeric'] = df['Allergic\_rhinitis']  dict\_allergies = {"Allergic\_rhinitis\_numeric": {"No": 0, "Yes": 1}}  df.replace(dict\_allergies, inplace=True)  df['Reflux\_esophagitis\_numeric'] = df['Reflux\_esophagitis']  dict\_reflux = {"Reflux\_esophagitis\_numeric": {"No": 0, "Yes": 1}}  df.replace(dict\_reflux, inplace=True)  df['Asthma\_numeric'] = df['Asthma']  dict\_asthma = {"Asthma\_numeric": {"No": 0, "Yes": 1}}  df.replace(dict\_asthma, inplace=True) |

|  |
| --- |
| Pd.get\_dummies one-hot encoding |
| df = pd.get\_dummies(df, columns=["Marital", "Services", "Gender", "Initial\_admin", "Complication\_risk"]) |

I used both methods for the practice and to get experience with both.

C4. Univariate and Bivariate visualizations are below:

|  |
| --- |
| Univariate Visualizations |
|  |

|  |
| --- |
| Bivariate Visualizations |
|  |

C5. Prepared data set uploaded as past of submission.

**Part IV: Model Comparison and Analysis**

D1. Initial regression model with variables identified in C2

Graphical user interface, application

Description automatically generated with medium confidence

Notes:  
[1] Standard Errors assume that the covariance matrix of the errors is correctly specified.  
[2] The smallest eigenvalue is 3.06e-22. This might indicate that there are  
strong multicollinearity problems or that the design matrix is singular.

D2. I get an r-squared value of 1. I also have a multicollinearity issue. This means my model does explain the dependent variable, however, I should look at reducing the number of variables to make the model less complex and reduce multicollinearities. I can use VIF to look at variables that are producing multicollinearity.

Results of VIF below:

Table

Description automatically generated

Results of new model with variables of a VIF = infinity removed:

D3. Reducing the number of variables using VIF to produce an even smaller model with continuous and categorical variables. VIF results are below:

Graphical user interface, table

Description automatically generated

Continuous variables in this model include; Children, Income, vitD\_supp, Full\_meals\_eaten, Additional\_charges, TotalCharge, VitD\_levels, Age and Doc\_visits.

Categorical variables include all of the numeric columns.

**E.  Analyze the data set.**

E1. When building the model I chose to select patient demographics and patient health conditions to try and find a specific cohort of patients that were at risk of spending long initial days in the hospital. I chose Initial\_days as the dependent variable because we are supposed to choose a continuous variable for this task and I saw in previous assignments that Initial\_days has a very strong correlation for patient readmissions.

Both models I built had high R-squared and Adj. R-Squared scores. However, the F-Statistic is very high and P-value of the individual variables vary drastically. With that said, the models I built are very complex and need to be cleaned up more. More variables could also be chosen, though when I played with this data set I couldn’t find anything that had a large correlation to Initial\_days. I think this dataset just isn’t a great fit for answering this question.

Below is a screenshot of my Residuals Plot:

Chart

Description automatically generated

E2. Residual Standard Error:

Text

Description automatically generated

(Techhelpnotes, 2022)

E3.

|  |
| --- |
| Initial Model |
| mdl\_initial\_vs\_variables = ols("Initial\_days ~ vitD\_supp + Children + Income + Full\_meals\_eaten + Additional\_charges + TotalCharge + VitD\_levels + Age + Doc\_visits + HighBlood\_numeric + Stroke\_numeric + Arthritis\_numeric + Diabetes\_numeric + Hyperlipidemia\_numeric + BackPain\_numeric + Allergic\_rhinitis\_numeric + Reflux\_esophagitis\_numeric + Asthma\_numeric + Marital\_Divorced + Marital\_Married + Marital\_Never\_Married + Marital\_Separated + Marital\_Widowed + Services\_Blood\_Work + Services\_CT\_Scan + Services\_Intravenous + Services\_MRI + Gender\_Female + Gender\_Male + Gender\_Nonbinary + Initial\_admin\_Elective\_Admission + Initial\_admin\_Emergency\_Admission + Initial\_admin\_Observation\_Admission + Complication\_risk\_High + Complication\_risk\_Low + Complication\_risk\_Medium", data=df).fit()  mdl\_initial\_vs\_variables.summary() |

|  |
| --- |
| VIF to reduce model |
| # Checking for the VIF values of the variables.  from statsmodels.stats.outliers\_influence import variance\_inflation\_factor  X = df[['Initial\_days', 'vitD\_supp', 'Children', 'Income', 'Full\_meals\_eaten', 'Additional\_charges', 'TotalCharge', 'VitD\_levels', 'Age', 'Doc\_visits', 'HighBlood\_numeric', 'Stroke\_numeric', 'Arthritis\_numeric', 'Diabetes\_numeric', 'Hyperlipidemia\_numeric', 'BackPain\_numeric', 'Allergic\_rhinitis\_numeric', 'Reflux\_esophagitis\_numeric', 'Asthma\_numeric', 'Marital\_Divorced', 'Marital\_Married', 'Marital\_Never\_Married', 'Marital\_Separated', 'Marital\_Widowed', 'Services\_Blood\_Work', 'Services\_CT\_Scan', 'Services\_Intravenous', 'Services\_MRI', 'Gender\_Female', 'Gender\_Male', 'Gender\_Nonbinary', 'Initial\_admin\_Elective\_Admission', 'Initial\_admin\_Emergency\_Admission', 'Initial\_admin\_Observation\_Admission', 'Complication\_risk\_High', 'Complication\_risk\_Low', 'Complication\_risk\_Medium']]  # VIF dataframe  vif\_data = pd.DataFrame()  vif\_data["feature"] = X.columns    # calculating VIF for each feature  vif\_data["VIF"] = [variance\_inflation\_factor(X.values, i)  for i in range(len(X.columns))]    print(vif\_data) |

(GeeksforGeeks, 2019)

|  |
| --- |
| Reduced Model |
| #Running new model based on VIF results, removing variables with VIF = infinity  mdl\_initial\_vs\_variables = ols("Initial\_days ~ vitD\_supp + Children + Income + Full\_meals\_eaten + Additional\_charges + TotalCharge + VitD\_levels + Age + Doc\_visits + HighBlood\_numeric + Stroke\_numeric + Arthritis\_numeric + Diabetes\_numeric + Hyperlipidemia\_numeric + BackPain\_numeric + Allergic\_rhinitis\_numeric + Reflux\_esophagitis\_numeric + Asthma\_numeric", data=df).fit()  mdl\_initial\_vs\_variables.summary() |

**Part V: Data Summary and Implications**

F1. My linear regression equation:

Y = -26.85 + -0.03 (vitD\_supp) + -0.009 (Children) + 1.843x10^6 (Income) + -0.05 (Full\_meals\_eaten) + -0.0005 (Additional\_charges) + 0.012 (TotalCharge) + -0.0443 (VitD\_levels) + 0.1051 (Age) + -0.05 (Doc\_visits) + 2.6166 (HighBlood) + 0.252 (Stroke) + -0.7721 (Arthritis) + -0.7978 (Diabetes) + -1.2028 (Hyperlipidemia) + -1.0001 (BackPain) + -0.7978 (Allergic\_rhinitis) + -0.7123 (Reflux\_esophagitis) + 0.0840 (Asthma)

This line means for every 1 unit of:

vitD\_supp, Initial\_days will decrease 0.03 units

Children, Initial\_days will decrease 0.009 units

Income, Initial\_days will increase 1.843x10^6 units

Full\_meals\_eaten, Initial\_days will decrease 0.05 units

Additional\_charges, Initial\_days will decrease 0.0005 units

TotalCharge, Initial\_days will increase 0.012 units

VitD\_levels, Initial\_days will decrease -0.0443 units

Age, Initial\_days will increase 0.1051 units

Doc\_visits, Initial\_days will decrease 0.0523 units

HighBlood\_numeric, Initial\_days will increase 2.6166 units

Stroke\_numeric, Initial\_days will increase 0.2519 units

Arthritis\_numeric, Initial\_days will decrease 0.7721 units

Diabetes\_numeric, Initial\_days will decrease 0.7978 units

Hyperlipidemia\_numeric, Initial\_days will decrease 1.2028 units

BackPain\_numeric, Initial\_days will decrease 1.0001 units

Allergic\_rhinitis\_numeric, Initial\_days will decrease .7978 units

Reflux\_esophagitis\_numeric, Initial\_days will decrease 0.7123 units

Asthma\_numeric, Initial\_days will increase 0.0840 units

My model is statistically significant based on the R-squared value and p-values of the many different variables. However, this result is likely due to chance due to the complexity of the model and number of variables. Meaning that overall, this model carries no practical significance as well. This model will not produce repeatable results for there to be any real world impact that our hospital can act on.

This model is very limited. The task asked for us to choose a continuous variable, I chose Initial\_days, though the main problem we are trying to solve for is ReAdmis, which is a categorical variable. Because I was limited with choosing my dependent variable, the analysis was limited by the scope of the dataset itself. I am not sure that this dataset is robust enough to answer the main question at this point with my current level of knowledge. I have to imagine that I will have to apply a machine learning algorithm to this dataset to figure out what the best model is in the future.

F2. Based on my results, there really isn’t a course of action to be recommended. The model is too robust and complex and no predictions can really be made. The course of action would be to start back at square 1 and pick a different dependent variable, reducing our model with a different method, and maybe evaluating what data we capture moving forward.

**Part VI: Demonstration**

G.  https://wgu.hosted.panopto.com/Panopto/Pages/Viewer.aspx?id=a459c80e-2e2e-4a35-bbf4-af5401196114

H.

Techhelpnotes, 2022 <https://techhelpnotes.com/residual-standard-error-of-a-regression-in-python/>

GeeksforGeeks, 2019 https://www.geeksforgeeks.org/detecting-multicollinearity-with-vif-python/

I.  No sources used.