D208 Task 2 Performance Assessment

Hillary Osei (Student ID #011039266)

Western Governors University, College of Information Technology

Program Mentor: Dan Estes

January 12, 2025

Table of Contents

Part I: Research Question

Section A1) Research Question

The research question is, "Which factors contribute to patients being readmitted within one month after release?"

Section A2) Goals

This data analysis aims to identify key factors influencing patients being readmitted to the hospital within one month after release. This analysis will discover which factors are relevant and statistically significant in predicting patient readmissions by analyzing patient demographic information, medical conditions, and their initial hospital stay. Understanding this can help hospitals and other healthcare facilities develop strategies to reduce readmissions.

In addition, the analysis will measure the impact of each of the predictor variables. For example, based on the model's results, it is determined that medical conditions like high blood pressure or length of initial hospital stays can increase or decrease the chance of readmission.

Part II: Method Justification

Section B1) Summary of Assumptions

The four assumptions for the logistic regression model are as followed (Bobbitt, 2020):

1. The response/target variable is binary and has two possible outcomes, such as “Yes” and “No.”
2. The observations in the dataset are independent, meaning that they should not be related to one another.
3. No multicollinearity present in independent variables. This happens when independent variables are highly correlated, meaning they are not independent.
4. There are no extreme outliers or influential observations present in the dataset.

Section B2) Tool Benefits

Python is a helpful tool for data analysis because it has several libraries that can support different phases of the analysis process. The pandas library detects missing values (df.isnull.sum()) and transforms categorical variables. The numpy library is used for numerical calculations. Matplotlib is used to create data visualizations such as histograms and scatter plots.

For logistic regression modeling, sklearn.linear_model.LogisticRegression makes building and fitting models to predict patient readmissions easier. The statsmodel library's

variance_inflation_factor checks for multicollinearity, assessing if predictor variables are independent. Additionally, sklearn.model_selection.train_test_split simplifies splitting data for training purposes, and sklearn.metrics.confusion_matrix helps assess the accuracy of model predictions.

Section B3) Appropriate Technique

Logistic regression model is an effective method for analyzing the research question because the technique is meant to predict outcomes with a categorical dependent variable. In this instance, the dependent variable "ReAdmis," indicating whether a patient is readmitted within a month of release or not (Western Governors University, n.d.) is binary with two possible outcomes: "yes" (the patient is readmitted within a month of release) or "no" (the patient is not readmitted within a month of release). Logistic regression requires independent/predictor variables that could be either categorical or continuous, making it suitable for analyzing different factors such as demographics, medical conditions, and admission types.

Part III: Data Preparation

Section C1) Data Cleaning

The main goal of data cleaning is to prepare data for analysis by ensuring data is accurate, consistent, and well-structured. To do this, first, duplicated rows are detected using df.duplicated() to prevent redundant values. Missing values are also detected using df.isnull.sum(). Next, outliers are detected by first creating boxplots for the continuous independent variables. Additionally, irrelevant columns — 'CaseOrder', 'Customer_id', 'Interaction', 'UID', 'City', 'State', 'County', 'Zip', 'Lat', 'Lng', 'Population', 'Area' 'TimeZone', 'Job','Children', 'Income', 'Marital', 'VitD_levels', 'Full_meals_eaten', 'vitD_supp', 'Soft_drink', 'Additional_charges', 'Allergic_rhinitis', 'Reflux_esophagitis', 'Services', 'TotalCharge', 'Item1', 'Item2', 'Item3', 'Item4', 'Item5', 'Item6', 'Item7', and 'Item8'— are removed using df.drop() to focus solely on the variables that possibly contribute to patient readmissions. The variables that will be used for the regression analysis are: “Initial_admin,” “Gender,” “HighBlood,” “Stroke,” “Complication_risk,” “Overweight,” “Arthritis,” “Diabetes,” “BackPain,” “Asthma,” “Hyperlipidemia,” “Initial_days,” “Age,” “Doc_visits,” and “Anxiety.”

Categorical variables with “yes/no” values, including the target variable “ReAdmis,” are re-expressed into binary values (0 or 1) using dictionary mapping. For categorical variables with multiple categories like “Gender” and “Initial_admin,” dummy variables are created to represent each category as separate columns. Furthermore, Boolean values for these dummy variables are converted to binary values to work effectively with logistic regression. Column names are rewritten with underscores for readability purposes.

Ordinal encoding is applied to “Complication_risk” because it has ordered categories. The values are then converted to integers, again to make it compatible with logistic regression since numeric values are required.

Section C2) Summary Statistics

For the analysis, the dependent variable selected is “ReAdmis,” and the independent variables selected are “Initial_admin,” “HighBlood,” “Stroke,” “Complication_risk,” “Overweight,” “Arthritis,” “Diabetes,” “BackPain,” “Asthma,” “Hyperlipidemia,” “Initial_days,” “Gender,” “Age,” “Doc_visits,” and “Anxiety.”

To get the summary statistics for the dependent and independent variables, the describe() function is used.

For the “Initial_days” variable, the average is 34.455299. The standard deviation is 26.309341. The range for the “Initial_days” variable is from 1.001981 to 71.981490. The 25th percentile is at 7.896215, the 50th percentile/median is at 35.836244, and the 75th percentile is at 61.161020.

For the “Age” variable,” the mean is 23.511700 years old. The standard deviation is at 20.638538. The “Age” variable ranges from 18 to 89 years old. The 25th percentile is at 36 years old, the 50th percentile/median is at 53 years old, and the 75th percentile is at 71 years old.

For the “Docs_visits” variable, the average is 5.012200. The standard deviation is 1.045734. The “Docs_visits” variable ranges from 1 to 9. The 25th percentile is at 4, the 50th percentile/median is at 5, and the 75th percentile is at 6.

	Initial_days	Age	Doc_visits
count	10000.000000	10000.000000	10000.000000
mean	34.455299	53.511700	5.012200
std	26.309341	20.638538	1.045734
min	1.001981	18.000000	1.000000
25%	7.896215	36.000000	4.000000
50%	35.836244	53.000000	5.000000
75%	61.161020	71.000000	6.000000
max	71.981490	89.000000	9.000000

To get the summary statistics for the categorical independent variables, the values_count() function was used.

For the “Initial_admin” variable, the number of patients who were initially admitted to the hospital for emergency admission was 5,060. The number of patients who were initially admitted to the hospital for elective admission is 2,504. The number of patients who were initially admitted for observation admission is 2,436.

```
Initial_admin
Emergency Admission      5060
Elective Admission       2504
Observation Admission    2436
Name: count, dtype: int64
```

For the “HighBlood” variable, the number of patients who do not have high blood pressure is 5,910 and the number of patients who do have high blood pressure is 4,090.

```
HighBlood
No     5910
Yes    4090
Name: count, dtype: int64
```

For the “Stroke” variable, the number of patients who have not had a stroke is 8,007 and the number of patients who have had a stroke is 1,993.

```
Stroke
No     8007
Yes    1993
Name: count, dtype: int64
```

For the “Complication_risk” variable, the number of patients who were assessed to be at a medium-level complication risk is 4,517. The number of patients who were assessed to be at a high-level complication risk is 3,358. The number of patients who were assessed to be at a low-level complication risk is 2,125.

```
Complication_risk
Medium    4517
High      3358
Low       2125
Name: count, dtype: int64
```

For the “Overweight” variable, the number of patients who are considered overweight based on age, gender, and height are 7,094 and the number of patients who are not considered overweight are 2,906.

```
Overweight
Yes    7094
No     2906
Name: count, dtype: int64
```

For the “Arthritis” variable, the number of patients who do not have arthritis is 6,426 and the number of patients who do have arthritis is 3,574.

```
Arthritis
No     6426
Yes    3574
Name: count, dtype: int64
```

For the “Diabetes” variable, the number of patients who are not diabetic are 7,262 and the number of patients who are diabetic are 2,738.

```
Diabetes
No     7262
Yes    2738
Name: count, dtype: int64
```

For the “BackPain” variable, the number of patients who do not have chronic back pain are 5,886 and the number of patients who do have chronic back pain are 4,114.

```
BackPain
No     5886
Yes    4114
Name: count, dtype: int64
```

For the “Asthma” variable, the number of patients who do not have asthma are 7,107 and the number of patients who do have asthma are 2,893.

```
Asthma
No     7107
Yes    2893
Name: count, dtype: int64
```

For the “Hyperlipidemia” variable, the number of patients who do not have hyperlipidemia are 6,628 and the number of patients who do have hyperlipidemia are 3,372.

```
Hyperlipidemia
No     6628
Yes    3372
Name: count, dtype: int64
```

For the “Gender” variable, the number of female patients is 5,018. The number of male patients is 4,768. The number of nonbinary patients is 214.

```
Gender
Female       5018
Male         4768
Nonbinary     214
Name: count, dtype: int64
```

For the “Anxiety” variable, the number of patients who do not have an anxiety disorder are 6,785 and the number of patients who do have an anxiety disorder are 3,215.

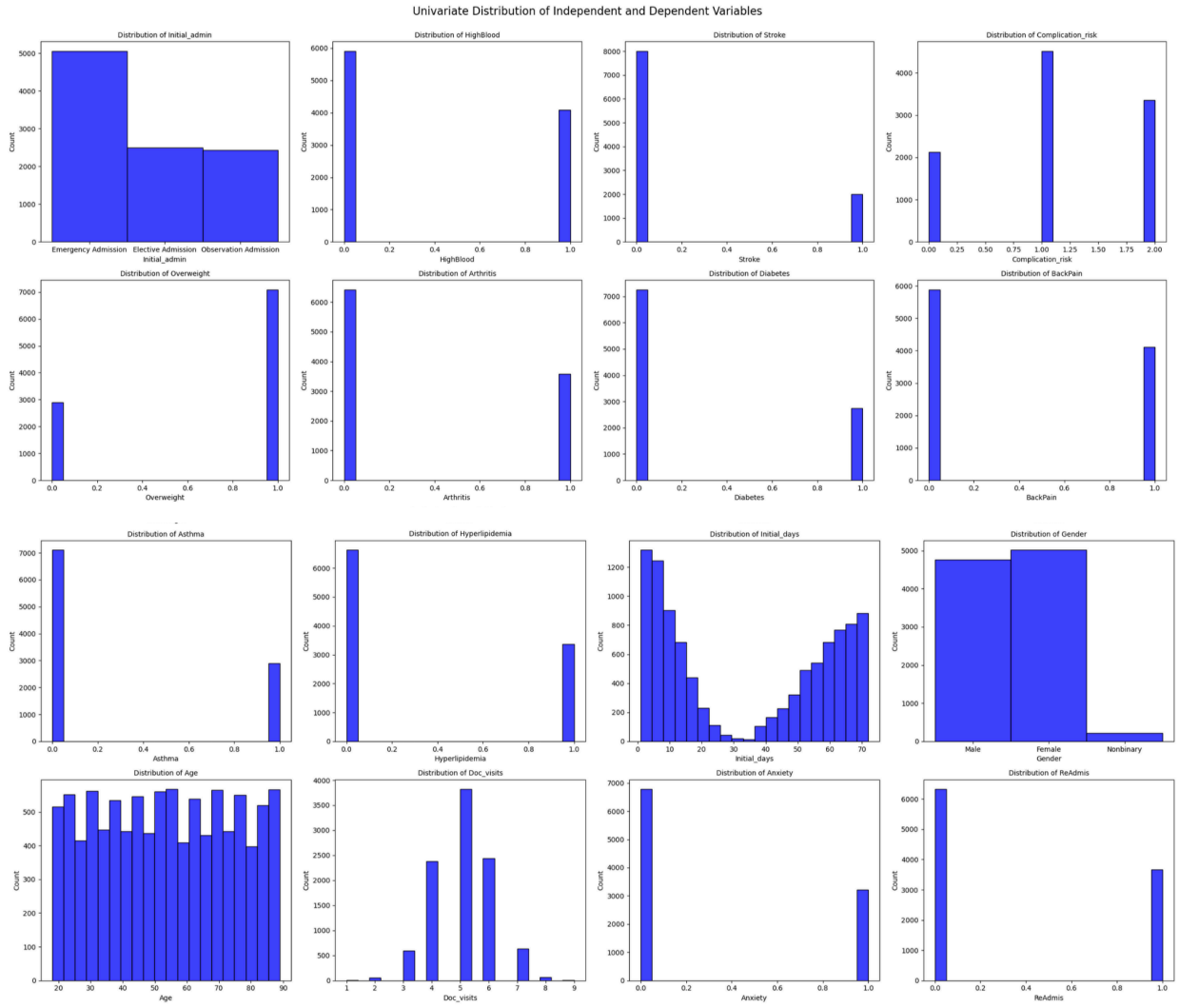
```
Anxiety
No     6785
Yes    3215
Name: count, dtype: int64
```

For the “ReAdmis” variable, the number of patients who were not readmitted within a month of release are 6,331 and the number of patients who were readmitted within a month of release are 3,669.

```
ReAdmis
No     6331
Yes    3669
Name: count, dtype: int64
```

Section C3) Visualizations

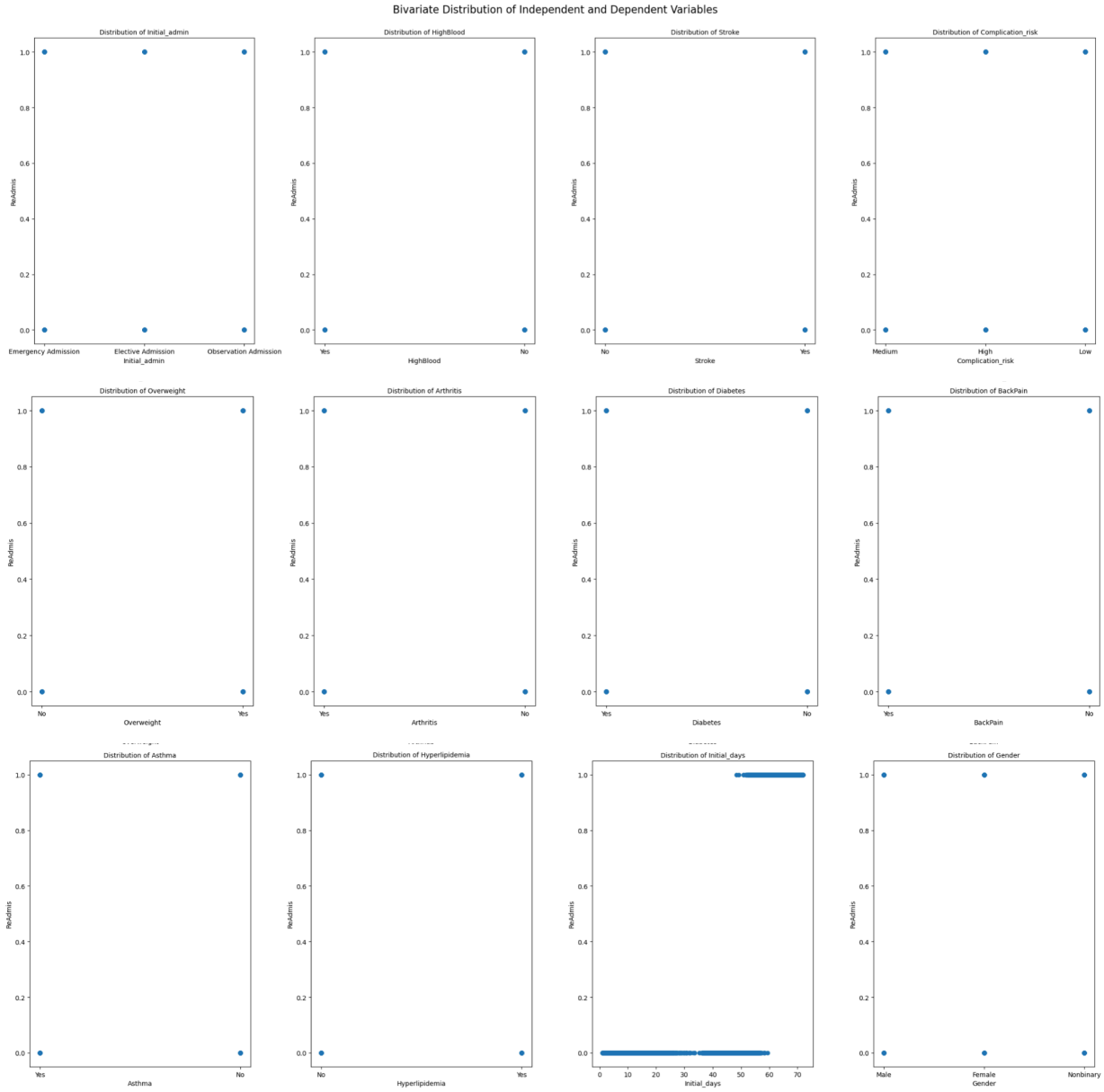
Univariate:

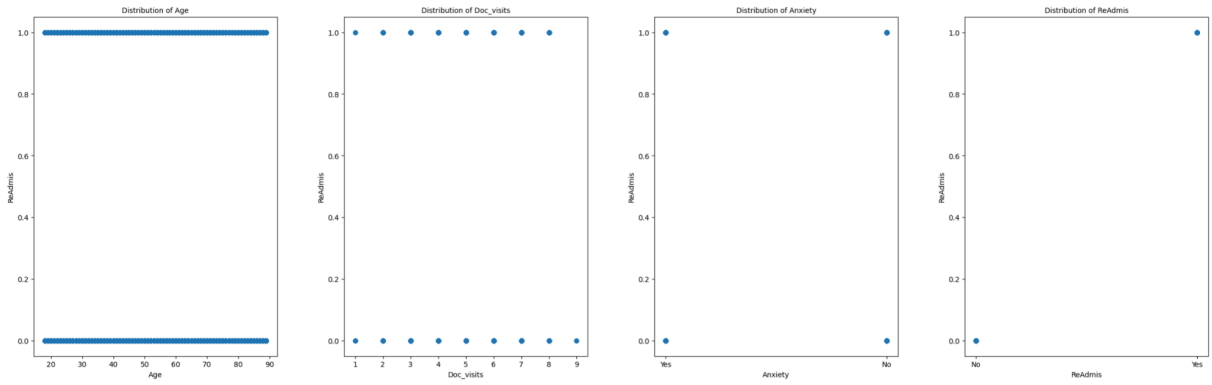


These histograms provide insights into the distribution of the independent and dependent variables needed for the data analysis. For “Initial_admin” there are significantly more patients admitted through emergency means compared to elective and observation. The “HighBlood” variable shows that there are more patients who do not have high blood pressure compared to patients who do have the condition. The “Stroke” variable shows that a significant number of patients have never had a stroke compared to patients who have. For “Complication_risk,” there are more patients who are classified as being at a medium-level complication risk, compared to patients who are classified as high-level or low-level. The “Overweight” variable shows that there are more patients who are overweight than patients who are not. In the “Arthritis” variable,

there are more patients who do not have arthritis compared to patients that do. The “Diabetes” variable reveals that most patients are not diabetic. For “BackPain,” it shows that most patients have not been diagnosed with chronic back pain. The distribution for the “Asthma” variable shows that most patients do not have an asthma diagnosis. Similar to “Asthma,” the “Hyperlipidemia” variable shows that most patients do not have hyperlipidemia. The “Initial_days” variable reveals a bimodal distribution, with peaks between the 0-10 markers and 60-70 markers. The “Gender” variable shows that there are more female patients than male and non-binary. The “Age” variable’s distribution is uniform. “Doc_visits” shows a roughly symmetric distribution, peaking at around 5 visits, suggesting this is the most common range for patients. For “Anxiety,” the distribution shows that most patients do not have an anxiety diagnosis.

Bivariate:


Bivariate Distribution of Independent and Dependent Variables
Distribution of Initial_admin
ReAdmis
Emergency Admission
Elective Admission
Observation Admission
Initial_admin
Distribution of HighBlood
Yes
No
HighBlood
Distribution of Stroke
No
Yes
Stroke
Distribution of Complication_risk
Medium
High
Low
Complication_risk
Distribution of Overweight
Overweight
Distribution of Arthritis
Arthritis
Distribution of Diabetes
Diabetes
Distribution of BackPain
BackPain
Distribution of Asthma
Asthma
Distribution of Hyperlipidemia
Hyperlipidemia
Distribution of Initial_days
Initial_days
Distribution of Gender
Male
Female
Nonbinary
Gender



The scatterplots are meant to visualize the relationship between the independent variables and the target variable. Based on all of these visualizations, it is determined that there is no relationship between "ReAdmis" and any of the independent variables.

Section C4) Data Transformation

Data transformation is an important part in preparing raw data to ensure that it is clean, structured, and ready for analysis (IBM, 2024). The primary goal of data transformation is to make sure the dataset clearly aligns with the research question "Which factors contribute to patients being readmitted within one month of release?" and that the data is in a format compatible with logistic regression, which is numeric. The data transformation process includes cleaning the data, re-expressing categorical variables, creating dummy variables, and encoding ordinal categories.

The first step in the data transformation process is to check for duplicates and missing values. The df.duplicated() function returns a Boolean value, either True or False, for each row to indicate where it is a duplicate of another row. For the dataset, all the rows returned False, meaning no duplicates were detected. Next, missing values are detected using df.isnull.sum(), which gives a count of missing values for each column. All columns returned 0, meaning that no missing values are found.

Then, outliers are detected by creating boxplots for the continuous independent variables "Initial_days," "Age," and "Doc_visits."

```
# Create a boxplot for 'Initial_days'
plt.figure(figsize=(6, 4))  # Adjust figure size
sns.boxplot(x=df["Initial_days"])

# Add title and labels
plt.title("Boxplot for Initial Days", fontsize=16)
```

```
plt.xlabel("Initial Days", fontsize=12)

# Show the plot
plt.show()

------------------------------------------------------------------------

# Create a boxplot for 'Age'
plt.figure(figsize=(6, 4))  # Adjust figure size
sns.boxplot(x=df["Age"])

# Add title and labels
plt.title("Boxplot for Age", fontsize=16)
plt.xlabel("Age", fontsize=12)

# Show the plot
plt.show()

------------------------------------------------------------------------

# Create a boxplot for 'Doc_visits'
plt.figure(figsize=(6, 4))  # Adjust figure size
sns.boxplot(x=df["Doc_visits"])

# Add title and labels
plt.title("Boxplot for Doc_visits", fontsize=16)
plt.xlabel("Doc_visits", fontsize=12)

# Show the plot
plt.show()
```

The boxplots show that there were no outliers present in the selected continuous independent variables because there were no data points found outside the upper and lower bounds of the plots. Therefore, there’s no need for outliers to be handled.

Next, irrelevant columns that are not aligned with the research question are removed to streamline the dataset. Columns such as “CaseOrder,” “Timezone,” and “Job” are dropped using df.drop().

```
# Drop irrelevant columns
df = df.drop(columns=[‘CaseOrder’, ‘Customer_id’, ‘Interaction’, ‘UID’, ‘City’, ‘State’, ‘County’, ‘Zip’, ‘Lat’, ‘Lng’, ‘Population’, ‘Area’,
                      ’TimeZone’, ‘Job’,’Children’, ‘Income’, ‘Marital’, ‘VitD_levels’, ‘Full_meals_eaten’, ‘vitD_supp’, ‘Soft_drink’,
                      ’Additional_charges’, ‘Allergic_rhinitis’, ‘Reflux_esophagitis’, ‘Services’, ‘TotalCharge’, ‘Item1’, ‘Item2’,
                      ’Item3’, ‘Item4’, ‘Item5’, ‘Item6’, ‘Item7’, ‘Item8’])
```

Categorical variables with “Yes” and “No” values are re-expressed into binary values (0 and 1) using dictionary mapping. This is essential because it ensures compatibility with logistic regression modeling since it requires numeric values.

```
# Re-express categorical variables w/ Yes/No values
yes_no = {”Yes”: 1, “No”: 0}
df[”HighBlood”] = df[”HighBlood”].replace(yes_no).infer_objects(copy=False)
df[”Stroke”] = df[”Stroke”].replace(yes_no).infer_objects(copy=False)
df[”Overweight”] = df[”Overweight”].replace(yes_no).infer_objects(copy=False)
df[”Arthritis”] = df[”Arthritis”].replace(yes_no).infer_objects(copy=False)
df[”Diabetes”] = df[”Diabetes”].replace(yes_no).infer_objects(copy=False)
df[”BackPain”] = df[”BackPain”].replace(yes_no).infer_objects(copy=False)
df[”Asthma”] = df[”Asthma”].replace(yes_no).infer_objects(copy=False)
df[”Hyperlipidemia”] = df[”Hyperlipidemia”].replace(yes_no).infer_objects(copy=False)
df[”Gender”] = df[”Gender”].replace(yes_no).infer_objects(copy=False)
df[”Anxiety”] = df[”Anxiety”].replace(yes_no).infer_objects(copy=False)
df[”ReAdmis”] = df[”ReAdmis”].replace(yes_no).infer_objects(copy=False)
```

For categorical variables with more than two categories (e.g. “Gender” and “Initial_admin”), dummy variables are created using the pd.get_dummies() function. This function is a simple way to one-hot encode categorical variables while also avoiding multicollinearity by dropping the first category for each variable. This then converts the categories to separate columns.

```
# Create dummy variables for “Gender” and “Initial_admin”
df = df.join(pd.get_dummies(df[”Gender”], prefix=”Gender”, drop_first=True))
df = df.join(pd.get_dummies(df[”Initial_admin”], prefix=”Initial_admin”, drop_first=True))
```

In the above code, the first category “Female” was dropped for the “Gender” variable and the first category “Elective Admission” was dropped for the “Initial_admin” variable.

These new columns made from the pd.get_dummies() function are rewritten to replace spaces with underscores for better readability.

```
# Replace spaces with ‘_’
df.columns = df.columns.str.replace(’ ‘, ‘_’)
```

Altogether, this results in the new columns “Gender_Male,” “Gender_Nonbinary,” “Initial_admin_Observation_Admission,” and “Initial_admin_Emergency_Admission.”

The Boolean values for these dummy variables are converted to binary values (0/1) using dictionary mapping to ensure the data is consistent.

```
# Convert dummy variables into binary value (0/1)
true_false = {True: 1, False: 0}
df[“Gender_Male”] = df[”Gender_Male”].replace(true_false).infer_objects(copy=False)
df[“Gender_Nonbinary”] =
df[“Gender_Nonbinary”].replace(true_false).infer_objects(copy=False)
df[“Initial_admin_Emergency_Admission”] =
df[“Initial_admin_Emergency_Admission”].replace(true_false).infer_objects(copy=False)
df[“Initial_admin_Observation_Admission”] =
df[“Initial_admin_Observation_Admission”].replace(true_false).infer_objects(copy=False)
```

The “Complication_risk” variable is re-expressed using ordinal encoding because its values—“Low,” “Medium,” and “High”—follow a natural, logical order. Using sklearn.preprocessing.OrdinalEncoder, the values are mapped to integers, where “Low” = 0, “Medium” = 1, and “High” = 2:

```
# Ordinal encoding for “Complication_risk”
# Code adapted from GeeksforGeeks:
# GeeksforGeeks. (n.d.). How to perform ordinal encoding using sklearn. Retrieved December 10, 2024, from https://www.geeksforgeeks.org/how-to-perform-ordinal-encoding-using-sklearn/

# Initialize and fit encoder
encoder = OrdinalEncoder(categories=[[’Low’, ‘Medium’, ‘High’]])
df[’Complication_risk’] = encoder.fit_transform(df[[’Complication_risk’]])

# Convert values to integer
df[‘Complication_risk’] = df[‘Complication_risk’].astype(int)
```

```
# Print result
print(df[[‘Complication_risk’]])
```

Section C5) Prepared Data Set

Cleaned and prepared data set attached and saved as “medical_logistic.csv”

Part IV: Model Comparison and Analysis

Section D1) Initial Model

Logistic regression is used to model the relationship between a categorical target variable and continuous explanatory variables (JMP, n.d.). Below is the code for the initial model:

```
# Create initial logistic regression model

X = df[[”Initial_admin_Emergency_Admission”,
“Initial_admin_Observation_Admission”,”Gender_Male”, “Gender_Nonbinary”, “HighBlood”,
“Stroke”, “Complication_risk”, “Overweight”, “Arthritis”, “Diabetes”, “BackPain”, “Asthma”,
“Hyperlipidemia”, “Initial_days”, “Age”, “Doc_visits”, “Anxiety”]]
y = df[“ReAdmis”]

from statsmodels.stats.outliers_influence import variance_inflation_factor
import statsmodels.api as sm

# Calculate VIFs for each independent variable
vif_data = pd.DataFrame({
        ’Variable’: X.columns,
        ‘VIF’: [variance_inflation_factor(X.values, i) for i in range(X.shape[1])]
})

# Display VIF
print(vif_data)

X = sm.add_constant(X)  # Add intercept
initial_model = sm.Logit(y,X).fit()
print(initial_model.summary())
```

```
                              Variable        VIF
0     Initial_admin_Emergency_Admission   2.842879
1   Initial_admin_Observation_Admission   1.892816
2                           Gender_Male   1.902442
3                    Gender_Nonbinary     1.042730
4                           HighBlood     1.677777
5                              Stroke     1.242838
6                    Complication_risk     3.194531
7                           Overweight     3.266795
8                            Arthritis     1.540834
9                             Diabetes     1.368604
10                            BackPain     1.683595
11                              Asthma     1.394448
12                      Hyperlipidemia     1.488792
13                        Initial_days     2.612350
14                                 Age     6.622019
15                          Doc_visits    11.987446
16                             Anxiety     1.460890
Optimization terminated successfully.
         Current function value: 0.036392
         Iterations 14
                           Logit Regression Results
==============================================================================
Dep. Variable:                 ReAdmis   No. Observations:                10000
Model:                           Logit   Df Residuals:                     9982
Method:                            MLE   Df Model:                           17
Date:                Sat, 04 Jan 2025   Pseudo R-squ.:                  0.9446
Time:                        02:25:54   Log-Likelihood:                -363.92
converged:                       True   LL-Null:                       -6572.9
Covariance Type:            nonrobust   LLR p-value:                     0.000
=========================================================================================================
                                            coef    std err          z      P>|z|      [0.025      0.975]
---------------------------------------------------------------------------------------------------------
const                                   -75.1001      4.021    -18.679      0.000     -82.980     -67.220
Initial_admin_Emergency_Admission         2.2111      0.257      8.591      0.000       1.707       2.716
Initial_admin_Observation_Admission       0.6636      0.266      2.497      0.013       0.143       1.185
Gender_Male                               0.1657      0.195      0.847      0.397      -0.217       0.549
Gender_Nonbinary                          0.7807      0.698      1.118      0.264      -0.588       2.149
HighBlood                                 0.8702      0.204      4.261      0.000       0.470       1.270
Stroke                                    1.5643      0.254      6.169      0.000       1.067       2.061
Complication_risk                         0.7559      0.136      5.559      0.000       0.489       1.022
Overweight                               -0.2384      0.215     -1.110      0.267      -0.659       0.182
Arthritis                                -1.2519      0.214     -5.846      0.000      -1.672      -0.832
Diabetes                                  0.4686      0.217      2.163      0.031       0.044       0.893
BackPain                                  0.2940      0.195      1.509      0.131      -0.088       0.676
Asthma                                   -1.2191      0.219     -5.561      0.000      -1.649      -0.789
Hyperlipidemia                            0.3008      0.204      1.472      0.141      -0.100       0.701
Initial_days                              1.3469      0.072     18.741      0.000       1.206       1.488
Age                                       0.0017      0.005      0.373      0.709      -0.007       0.011
Doc_visits                               -0.0171      0.089     -0.191      0.849      -0.192       0.158
Anxiety                                  -0.9078      0.211     -4.299      0.000      -1.322      -0.494
=========================================================================================================
```

Section D2) Justification of Model Reduction

The features in the initial model are reduced using backward stepwise elimination, a method that starts with all initial features and iteratively removes the least significant feature until no further improvement is needed (Hayes, 2022). Features are removed based on two criteria: the Variance Inflation Factor (VIF) and the p-value. The VIF threshold is set at 10 to address multicollinearity (Singh, 2024), and the p-value threshold is set at 0.05 because any value at or below the value signifies statistical significance. This process continues until no features exceed a VIF value of 10 or a p-value greater than 0.05, ensuring that the remaining predictors are statistically significant and independent.

The VIF is a tool for detecting multicollinearity in regression analysis by measuring how much the variance of a regression coefficient is inflated due to correlations among predictors (Singh, 2024). Multicollinearity occurs when independent variables are highly correlated, leading to unstable coefficient estimates and inflated standard errors. (Frost, n.d.) A VIF value exceeding 10 indicates severe multicollinearity and means that removing the variable is needed to stabilize the model (Singh, 2024). By eliminating variables with high VIF values, backward stepwise elimination ensures that the predictors independently contribute to explaining the variation in the patient readmissions.

Backward stepwise elimination using p-values ensures that statistically significant predictors remain in the model, avoiding potential misinterpretations from irrelevant variables. This aligns with the research question because by retaining the most important variables, major factors that impact patient readmissions can be identified and actionable insights can be made to address the issue.

Section D3) Reduced Logistic Regression Model

Removal of "Doc_visits" with VIF of 11.987446:

	Variable	VIF
0	Initial_admin_Emergency_Admission	2.622902
1	Initial_admin_Observation_Admission	1.781475
2	Gender_Male	1.855337
3	Gender_Nonbinary	1.041695
4	HighBlood	1.655902
5	Stroke	1.236072
6	Complication_risk	2.986180
7	Overweight	3.058621
8	Arthritis	1.522691
9	Diabetes	1.353329
10	BackPain	1.658780
11	Asthma	1.386048
12	Hyperlipidemia	1.477462
13	Initial_days	2.513965
14	Age	5.440358
15	Anxiety	1.446793

```
Optimization terminated successfully.
         Current function value: 0.036394
         Iterations 14
```

Logit Regression Results

Dep. Variable:	ReAdmis	No. Observations:	10000
Model:	Logit	Df Residuals:	9983
Method:	MLE	Df Model:	16
Date:	Sat, 04 Jan 2025	Pseudo R-squ.:	0.9446
Time:	02:26:22	Log-Likelihood:	-363.94
converged:	True	LL-Null:	-6572.9
Covariance Type:	nonrobust	LLR p-value:	0.000

	coef	std err	z	P>|z|	[0.025	0.975]
const	-75.1673	4.007	-18.760	0.000	-83.020	-67.314
Initial_admin_Emergency_Admission	2.2081	0.257	8.597	0.000	1.705	2.711
Initial_admin_Observation_Admission	0.6609	0.265	2.491	0.013	0.141	1.181
Gender_Male	0.1657	0.195	0.847	0.397	-0.217	0.549
Gender_Nonbinary	0.7765	0.699	1.112	0.266	-0.593	2.146
HighBlood	0.8675	0.204	4.259	0.000	0.468	1.267
Stroke	1.5650	0.254	6.173	0.000	1.068	2.062
Complication_risk	0.7558	0.136	5.558	0.000	0.489	1.022
Overweight	-0.2384	0.215	-1.111	0.267	-0.659	0.182
Arthritis	-1.2541	0.214	-5.863	0.000	-1.673	-0.835
Diabetes	0.4664	0.216	2.156	0.031	0.042	0.890
BackPain	0.2925	0.195	1.502	0.133	-0.089	0.674
Asthma	-1.2186	0.219	-5.559	0.000	-1.648	-0.789
Hyperlipidemia	0.3030	0.204	1.485	0.138	-0.097	0.703
Initial_days	1.3467	0.072	18.741	0.000	1.206	1.487
Age	0.0017	0.005	0.377	0.707	-0.007	0.011
Anxiety	-0.9063	0.211	-4.295	0.000	-1.320	-0.493

Removal of "Age" with highest p-value at 0.707:

```
                              Variable       VIF
0     Initial_admin_Emergency_Admission  2.472301
1   Initial_admin_Observation_Admission  1.712275
2                          Gender_Male  1.824265
3                       Gender_Nonbinary  1.040776
4                             HighBlood  1.636252
5                                Stroke  1.227306
6                     Complication_risk  2.817314
7                            Overweight  2.909919
8                              Arthritis  1.504324
9                               Diabetes  1.341704
10                              BackPain  1.630015
11                               Asthma  1.373604
12                       Hyperlipidemia  1.461612
13                         Initial_days  2.407506
14                              Anxiety  1.432755
Optimization terminated successfully.
         Current function value: 0.036401
         Iterations 14
                           Logit Regression Results
==============================================================================
Dep. Variable:                 ReAdmis   No. Observations:                10000
Model:                           Logit   Df Residuals:                     9984
Method:                            MLE   Df Model:                           15
Date:                 Sat, 04 Jan 2025   Pseudo R-squ.:                  0.9446
Time:                         02:43:07   Log-Likelihood:                -364.01
converged:                       True   LL-Null:                       -6572.9
Covariance Type:            nonrobust   LLR p-value:                     0.000
=========================================================================================================
                                            coef    std err          z      P>|z|      [0.025      0.975]
---------------------------------------------------------------------------------------------------------
const                                   -75.1282      4.007    -18.750      0.000     -82.981     -67.275
Initial_admin_Emergency_Admission         2.2049      0.257      8.594      0.000       1.702       2.708
Initial_admin_Observation_Admission       0.6612      0.265      2.494      0.013       0.142       1.181
Gender_Male                               0.1612      0.195      0.826      0.409      -0.221       0.544
Gender_Nonbinary                          0.7701      0.698      1.104      0.270      -0.598       2.138
HighBlood                                 0.8684      0.204      4.265      0.000       0.469       1.268
Stroke                                    1.5652      0.254      6.172      0.000       1.068       2.062
Complication_risk                         0.7563      0.136      5.561      0.000       0.490       1.023
Overweight                               -0.2387      0.215     -1.113      0.266      -0.659       0.182
Arthritis                                -1.2530      0.214     -5.858      0.000      -1.672      -0.834
Diabetes                                  0.4596      0.215      2.133      0.033       0.037       0.882
BackPain                                  0.2916      0.195      1.498      0.134      -0.090       0.673
Asthma                                   -1.2193      0.219     -5.560      0.000      -1.649      -0.789
Hyperlipidemia                            0.3025      0.204      1.483      0.138      -0.097       0.702
Initial_days                              1.3477      0.072     18.751      0.000       1.207       1.489
Anxiety                                  -0.9024      0.211     -4.282      0.000      -1.315      -0.489
=========================================================================================================
```

Removal of "Gender_Male" with highest p-value at 0.409:

	Variable	VIF
0	Initial_admin_Emergency_Admission	2.448298
1	Initial_admin_Observation_Admission	1.702296
2	Gender_Nonbinary	1.022622
3	HighBlood	1.630793
4	Stroke	1.226035
5	Complication_risk	2.777638
6	Overweight	2.873711
7	Arthritis	1.498773
8	Diabetes	1.339430
9	BackPain	1.627044
10	Asthma	1.370424
11	Hyperlipidemia	1.455257
12	Initial_days	2.381558
13	Anxiety	1.430621

Optimization terminated successfully.
Current function value: 0.036435
Iterations 14

Logit Regression Results

Dep. Variable:	ReAdmis	No. Observations:	10000
Model:	Logit	Df Residuals:	9985
Method:	MLE	Df Model:	14
Date:	Sat, 04 Jan 2025	Pseudo R-squ.:	0.9446
Time:	02:43:37	Log-Likelihood:	-364.35
converged:	True	LL-Null:	-6572.9
Covariance Type:	nonrobust	LLR p-value:	0.000

	coef	std err	z	P>|z|	[0.025	0.975]
const	-75.1220	4.008	-18.744	0.000	-82.977	-67.267
Initial_admin_Emergency_Admission	2.1947	0.256	8.578	0.000	1.693	2.696
Initial_admin_Observation_Admission	0.6597	0.265	2.491	0.013	0.141	1.179
Gender_Nonbinary	0.6925	0.691	1.002	0.317	-0.663	2.048
HighBlood	0.8592	0.203	4.234	0.000	0.461	1.257
Stroke	1.5832	0.253	6.258	0.000	1.087	2.079
Complication_risk	0.7632	0.136	5.628	0.000	0.497	1.029
Overweight	-0.2420	0.215	-1.128	0.259	-0.662	0.178
Arthritis	-1.2391	0.213	-5.822	0.000	-1.656	-0.822
Diabetes	0.4564	0.215	2.119	0.034	0.034	0.878
BackPain	0.2865	0.194	1.473	0.141	-0.095	0.668
Asthma	-1.2104	0.219	-5.536	0.000	-1.639	-0.782
Hyperlipidemia	0.3053	0.204	1.497	0.134	-0.094	0.705
Initial_days	1.3490	0.072	18.758	0.000	1.208	1.490
Anxiety	-0.9095	0.211	-4.320	0.000	-1.322	-0.497

Removal of "Overweight" with highest p-value at 0.277:

```
                              Variable       VIF
0     Initial_admin_Emergency_Admission  2.340344
1   Initial_admin_Observation_Admission  1.644760
2                              HighBlood  1.607600
3                                 Stroke  1.221415
4                     Complication_risk  2.660559
5                               Arthritis  1.484954
6                                Diabetes  1.333294
7                                BackPain  1.607282
8                                  Asthma  1.359260
9                         Hyperlipidemia  1.445340
10                          Initial_days  2.323750
11                               Anxiety  1.424245
Optimization terminated successfully.
         Current function value: 0.036544
         Iterations 14
                           Logit Regression Results
==============================================================================
Dep. Variable:                 ReAdmis   No. Observations:                10000
Model:                           Logit   Df Residuals:                     9987
Method:                            MLE   Df Model:                           12
Date:                 Sat, 04 Jan 2025   Pseudo R-squ.:                 0.9444
Time:                         02:52:01   Log-Likelihood:                -365.44
converged:                       True   LL-Null:                       -6572.9
Covariance Type:             nonrobust   LLR p-value:                     0.000
=========================================================================================================
                                            coef    std err          z      P>|z|      [0.025      0.975]
---------------------------------------------------------------------------------------------------------
const                                   -74.9043      3.985    -18.796      0.000     -82.715     -67.094
Initial_admin_Emergency_Admission         2.1616      0.254      8.514      0.000       1.664       2.659
Initial_admin_Observation_Admission       0.6365      0.264      2.408      0.016       0.118       1.155
HighBlood                                 0.8441      0.202      4.180      0.000       0.448       1.240
Stroke                                    1.5770      0.252      6.257      0.000       1.083       2.071
Complication_risk                         0.7601      0.136      5.608      0.000       0.494       1.026
Arthritis                                -1.2446      0.212     -5.861      0.000      -1.661      -0.828
Diabetes                                  0.4680      0.215      2.175      0.030       0.046       0.890
BackPain                                  0.3072      0.194      1.586      0.113      -0.072       0.687
Asthma                                   -1.2076      0.218     -5.536      0.000      -1.635      -0.780
Hyperlipidemia                            0.2927      0.203      1.441      0.150      -0.105       0.691
Initial_days                              1.3424      0.071     18.823      0.000       1.203       1.482
Anxiety                                  -0.9058      0.210     -4.309      0.000      -1.318      -0.494
=========================================================================================================
```

Removal of "Hyperlipidemia" with highest p-value at 0.150:

```
                              Variable       VIF
0     Initial_admin_Emergency_Admission  2.298192
1   Initial_admin_Observation_Admission  1.629105
2                             HighBlood  1.605090
3                                Stroke  1.221059
4                    Complication_risk  2.628924
5                             Arthritis  1.482614
6                              Diabetes  1.329747
7                              BackPain  1.603136
8                                Asthma  1.357951
9                          Initial_days  2.307534
10                              Anxiety  1.423247
Optimization terminated successfully.
         Current function value: 0.036649
         Iterations 14
                           Logit Regression Results
==============================================================================
Dep. Variable:                ReAdmis   No. Observations:                10000
Model:                          Logit   Df Residuals:                     9988
Method:                           MLE   Df Model:                           11
Date:                Sat, 04 Jan 2025   Pseudo R-squ.:                  0.9442
Time:                        02:58:56   Log-Likelihood:                -366.49
converged:                       True   LL-Null:                       -6572.9
Covariance Type:            nonrobust   LLR p-value:                     0.000
=====================================================================================================
                                        coef    std err          z      P>|z|      [0.025      0.975]
-----------------------------------------------------------------------------------------------------
const                                -74.6263      3.964    -18.824      0.000     -82.396     -66.856
Initial_admin_Emergency_Admission      2.1723      0.254      8.568      0.000       1.675       2.669
Initial_admin_Observation_Admission    0.6487      0.263      2.462      0.014       0.132       1.165
HighBlood                              0.8457      0.202      4.194      0.000       0.450       1.241
Stroke                                 1.5575      0.251      6.212      0.000       1.066       2.049
Complication_risk                      0.7708      0.135      5.697      0.000       0.506       1.036
Arthritis                             -1.2339      0.212     -5.827      0.000      -1.649      -0.819
Diabetes                               0.4566      0.215      2.126      0.033       0.036       0.877
BackPain                               0.3062      0.193      1.583      0.113      -0.073       0.685
Asthma                                -1.2076      0.218     -5.533      0.000      -1.635      -0.780
Initial_days                           1.3388      0.071     18.852      0.000       1.200       1.478
Anxiety                               -0.9091      0.210     -4.332      0.000      -1.320      -0.498
=====================================================================================================
```

Reduced logistic regression model, removing "BackPain" with highest p-value at 0.113:

```
                              Variable       VIF
0    Initial_admin_Emergency_Admission  2.247887
1  Initial_admin_Observation_Admission  1.602504
2                             HighBlood  1.598998
3                               Stroke  1.218943
4                    Complication_risk  2.594733
5                             Arthritis  1.480711
6                              Diabetes  1.328286
7                                Asthma  1.352582
8                          Initial_days  2.271846
9                               Anxiety  1.418649
Optimization terminated successfully.
         Current function value: 0.036775
         Iterations 14
                           Logit Regression Results
==============================================================================
Dep. Variable:                 ReAdmis   No. Observations:                10000
Model:                           Logit   Df Residuals:                     9989
Method:                            MLE   Df Model:                           10
Date:                 Sat, 04 Jan 2025   Pseudo R-squ.:                  0.9441
Time:                         03:00:13   Log-Likelihood:                -367.75
converged:                       True   LL-Null:                       -6572.9
Covariance Type:            nonrobust   LLR p-value:                     0.000
=========================================================================================================
                                            coef    std err          z      P>|z|      [0.025      0.975]
---------------------------------------------------------------------------------------------------------
const                                   -74.2729      3.940    -18.853      0.000     -81.994     -66.551
Initial_admin_Emergency_Admission         2.1793      0.253      8.624      0.000       1.684       2.675
Initial_admin_Observation_Admission       0.6659      0.262      2.537      0.011       0.152       1.180
HighBlood                                 0.8477      0.201      4.210      0.000       0.453       1.242
Stroke                                    1.5441      0.250      6.172      0.000       1.054       2.034
Complication_risk                         0.7693      0.135      5.700      0.000       0.505       1.034
Arthritis                                -1.2347      0.211     -5.845      0.000      -1.649      -0.821
Diabetes                                  0.4579      0.215      2.128      0.033       0.036       0.880
Asthma                                   -1.2084      0.218     -5.548      0.000      -1.635      -0.781
Initial_days                              1.3344      0.071     18.884      0.000       1.196       1.473
Anxiety                                  -0.8753      0.208     -4.214      0.000      -1.282      -0.468
=========================================================================================================
```

Section E1) Model Comparison

The initial logistic regression model contained 17 independent variables: “Initial_admin_Emergency_Admission,” “Initial_admin_Observation_Admission,” “Gender_Male,” “Gender_Nonbinary,” “HighBlood,” “Stroke,” “Complication_risk,” “Overweight,” “Arthritis,” “Diabetes,” “BackPain,” “Asthma,” “Hyperlipidemia,” “Initial_days,” “Age,” “Doc_visits,” and “Anxiety.” The first step in refining the model was addressing multicollinearity by removing predictors with high Variance Inflation Factor (VIF) values using backward stepwise elimination. The variable “Doc_visits” was removed first due to its VIF of 11.987446, which exceeded the threshold of 10, indicating severe multicollinearity.

Next, backward stepwise elimination was used to further refine the model by removing predictors based on their p-values. Variables with the highest p-values above the significance threshold of 0.05 were removed iteratively. This process led to the removal of the following variables: “Age,” “Gender_Male,” “Gender_Nonbinary,” “Overweight,” “Hyperlipidemia,” and “BackPain.” This process continued until all remaining predictors had p-values below 0.05, indicating statistical significance. The reduced logistic regression model retained the following predictors: “Initial_admin_Emergency_Admission,” “Initial_admin_Observation_Admission,” “HighBlood,” “Stroke,” “Complication_risk,” “Arthritis,” “Diabetes,” “Asthma,” “Initial_days,” and “Anxiety.”

The initial and reduced models were evaluated using pseudo r-squared values to assess their goodness-of-fit. A higher pseudo r-squared value indicates a better model fit (Western Governors University, n.d., slide 14). The pseudo r-squared value for the initial model was 0.9446, while the reduced model had a slightly lower value of 0.9441. Based on these results, the initial model is better than the reduced model.

Section E2) Output and Calculations

The analysis for the reduced logistic model includes creating a test dataset and evaluating the model's performance using a confusion matrix and accuracy score.

The test dataset is created using the train_test_split() function, which divides data into training and testing subsets. The test_size parameter is set to 0.3, meaning that 30% of the data is used for testing and the other 70% is used for training the model. The random_state parameter is set to 42 to ensure that the data split is reproducible which is needed for consistent results and to mitigate possible biases.

A confusion matrix is a table that maps a model's predictions versus the actual classes that the data belongs to (Kundu, 2022). The matrix shows how well the model predicts patients who are readmitted and patients who are not readmitted. The confusion matrix is separated into four components (Kundu, 2022):

1. True positive: cases where the model correctly predicts the positive class
2. True negative: cases where the model correctly predicts the negative class
3. False positive: cases where the model incorrectly predicts the positive class when it is actually negative
4. False negative: cases where the model incorrectly predicts the negative class when it is actually positive

```
# Create confusion matrix and accuracy

from sklearn.model_selection import train_test_split
from sklearn.linear_model import LogisticRegression
from sklearn.metrics import confusion_matrix

# Split data into training and testing sets
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.3, random_state=42)

# Initialize and train the logistic regression model
logreg = LogisticRegression()
logreg.fit(X_train, y_train)

# Predict the target variable for the test set
y_pred = logreg.predict(X_test)

# Print accuracy on the test set
```

```
print('Accuracy of logistic regression classifier on test set: {:.2f}'.format(logreg.score(X_test, y_test)))

# Compute the confusion matrix
final_matrix = confusion_matrix(y_test, y_pred)

# Print the confusion matrix
print("Confusion Matrix:")
print(final_matrix)
```

```
Accuracy of logistic regression classifier on test set: 0.98
Confusion Matrix:
[[1897   37]
 [  33 1033]]
```

From the confusion matrix, the model correctly predicts 1,897 cases where patients are not readmitted (true negatives) and 1,033 cases where patients are correctly predicted as readmitted (true positives). However, there are 33 cases where patients are classified as being readmitted but are not (false positives) and 37 cases where patients are inaccurately classified as not being readmitted but actually are (false negatives).

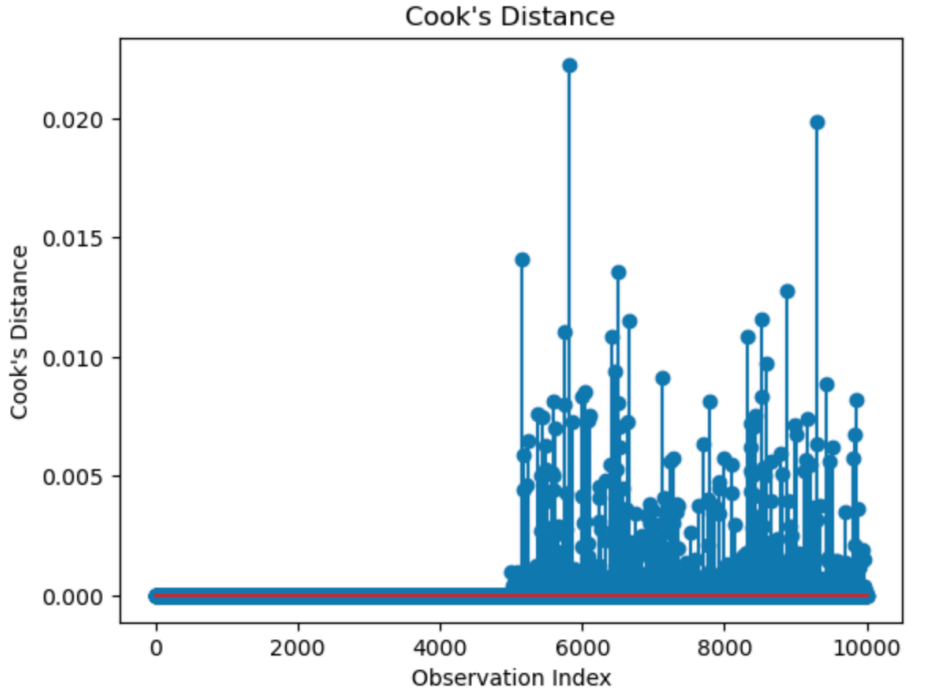
Then, the accuracy score evaluates the proportion of accurate predictions out of the total predictions made by the model. These metrics also show that the model's accuracy is at 98%, meaning it correctly predicts the outcomes for 98% of the test cases. This high accuracy score shows that the model predicts patient readmissions effectively.

Cook's Distance is used for finding extreme outliers and influential observations in logistic regression models (Bobbit, 2020). It is calculated to measure the influence of each individual observation on the regression coefficients.

```
# Cook's distance to check for outliers in reduced model
# Code adapted from Statology:
# Bobbit, Z. (2020). How to Calculate Cook's Distance in Python
# Retrieved January 8, 2025, from https://www.statology.org/cooks-distance-python/

# Suppress scientific notation
import numpy as np
np.set_printoptions(suppress=True)

#create instance of influence
influence = reduced_model.get_influence()
```

```
#obtain Cook’s distance for each observation
cooks = influence.cooks_distance

#display Cook’s distances
print(cooks)

plt.stem(np.arange(len(cooks[0])), cooks[0], markerfmt=”o”)
plt.title(”Cook’s Distance”)
plt.xlabel(”Observation Index”)
plt.ylabel(”Cook’s Distance”)
plt.show()

# Identify influential observations
influential_points = np.where(cooks[0] > 0.02)
print(”Indices of influential observations:”, influential_points[0])

# Inspect the influential observation
print(df.iloc[5812])
```

```
(array([0., 0., 0., ..., 0., 0., 0.]), array([1., 1., 1., ..., 1., 1., 1.]))
```



```
Indices of influential observations: [5812]
Age                                                42
Gender                                           Male
ReAdmis                                             1
Doc_visits                                          4
Initial_admin                  Emergency Admission
HighBlood                                           0
Stroke                                              0
Complication_risk                                   1
Overweight                                          1
Arthritis                                           1
Diabetes                                            0
Hyperlipidemia                                      1
BackPain                                            0
Anxiety                                             0
Asthma                                              0
Initial_days                                 48.43358
Gender_Male                                         1
Gender_Nonbinary                                    0
Initial_admin_Emergency_Admission                   1
Initial_admin_Observation_Admission                 0
Name: 5812, dtype: object
```

The analysis shows that most of the observations had values close to 0, meaning there is minimal influence on the model and does not have overly influential data points that could potentially distort the model. However, there is one observation (outlier), at index 5812, that has a noticeably higher Cook's Distance value, way above the threshold at 0.02. This threshold was selected because the standard threshold for Cook's Distance 4/n would flag too many minor influences in a dataset of 10,000 rows.

Upon inspecting the observation, the patient at index 5812 is a 42-year old male who was readmitted after an unusually long first hospital stay at 48.43358 days. The patient was initially admitted under emergency and is classified as being at a medium-level complication risk. The patient is overweight and has arthritis and hyperlipidemia. These findings do challenge one of the assumptions of logistic regression modeling, which is that there cannot be any extreme outliers.

Section E3) Code

Error-free code is attached, titled "d208-pa-task-2.ipynb"

Part V: Data Summary and Implications

Section F1) Results

The data analysis results in the following equation for the reduced model:
$ln\frac{p}{1-p}$= -74.2729 + (2.1793)*Initial_admin_Emergency_Admission +
(0.6659)*Initial_admin_Observation_Admission + (0.8477)*HighBlood + (1.5441)*Stroke +
(0.7693)*Complication_risk + (-1.2347)*Arthritis + (0.4579)*Diabetes + (-1.2084)*Asthma +
(1.3344)*Initial_days + (-0.8753)*Anxiety

The equation represents the log-odds of a patient being readmitted to the hospital within one month of release and it incorporates two elements: the intercept and the coefficients of the predictors. The intercept (-74.2729) represents the log-odds of being readmitted when the independent variables are zero. The coefficients represent the log odds ratio, meaning how much the log odds change when the predictor variable changes by one unit (*Interpreting Logit Results, What Do Coefficients Mean?*, n.d.). A positive coefficient means that there is an increase in log-odds, while a negative coefficient means a decrease in log-odds. (Athimala et al., n.d.)

The equation was created using the following code:

```
# Get regression equation

# Get coefficients from the reduced model
coefficients = reduced_model.params

# Build the log-odds regression equation
log_odds_equation = f"ln(p / (1-p)) = {coefficients['const']:.4f}"  # Start with intercept
for predictor, coef in coefficients.items():
        if predictor != 'const':  # Skip intercept after including it
        log_odds_equation += f" + ({coef:.4f})*{predictor}"

# Print the log-odds equation
print("Logistic Regression Equation (Log-Odds):")
print(log_odds_equation)
```

The odds ratio is the ratio of the odds of the event occurring in one group versus the odds of it occurring in another group (Jain, 2024) and is also used for interpreting coefficients in logistic regression models. The odds ratio represents the change in the outcome (e.g. patient readmissions) for a one unit increase in a predictor variable. This assumes that the other predictor variables are constant. If the odds ratio is equal to 1, then there is no association between the predictor and the outcome. If the odds ratio is greater than 1, there is a positive association between the predictor and the outcome, meaning that higher predictor values increases the outcome's odds. If the odds ratio is less than 1, then there is a negative association between the predictor and the outcome, meaning that higher predictor values decrease the outcome's odds.

Here is the code used for calculating the odds ratio:

```
# Odds ratio analysis

# Get coefficients and calculate odds ratios
coefficients = reduced_model.params
odds_ratios = np.exp(coefficients)

# Exclude intercept from analysis
odds_ratios = odds_ratios.drop('const', errors='ignore')

# Interpret each odds ratio
for predictor, or_value in odds_ratios.items():
    if or_value > 1:
    print(f"The odds ratio for {predictor} is {or_value:.2f}, meaning that as {predictor} increases by one unit, the odds of the outcome increase by {(or_value - 1) * 100:.2f}%.")
    elif or_value < 1:
    print(f"The odds ratio for {predictor} is {or_value:.2f}, meaning that as {predictor} increases by one unit, the odds of the outcome decrease by {(1 - or_value) * 100:.2f}%.")
    else:
    print(f"The odds ratio for {predictor} is {or_value:.2f}, meaning that {predictor} has no effect on the odds of the outcome.")
```

Output:

```
The odds ratio for Initial_admin_Emergency_Admission is 8.84, meaning that as Initial_admin_Emergency_Admission increases by one unit, the odds of the outcome increase by 784.05%.
The odds ratio for Initial_admin_Observation_Admission is 1.95, meaning that as Initial_admin_Observation_Admission increases by one unit, the odds of the outcome increase by 94.63%.
The odds ratio for HighBlood is 2.33, meaning that as HighBlood increases by one unit, the odds of the outcome increase by 133.42%.
The odds ratio for Stroke is 4.68, meaning that as Stroke increases by one unit, the odds of the outcome increase by 368.37%.
The odds ratio for Complication_risk is 2.16, meaning that as Complication_risk increases by one unit, the odds of the outcome increase by 115.82%.
The odds ratio for Arthritis is 0.29, meaning that as Arthritis increases by one unit, the odds of the outcome decrease by 70.91%.
The odds ratio for Diabetes is 1.58, meaning that as Diabetes increases by one unit, the odds of the outcome increase by 58.07%.
The odds ratio for Asthma is 0.30, meaning that as Asthma increases by one unit, the odds of the outcome decrease by 70.13%.
The odds ratio for Initial_days is 3.80, meaning that as Initial_days increases by one unit, the odds of the outcome increase by 279.78%.
The odds ratio for Anxiety is 0.42, meaning that as Anxiety increases by one unit, the odds of the outcome decrease by 58.33%.
```

Patients admitted for emergency reasons are 8.84 times more likely to be readmitted within one month compared to patients who were not admitted for an emergency. As "Initial_admin_Emergency Admission" increases by one unit, the odds of the outcome increases by 784.05%, meaning that emergency admissions are a significant risk factor for readmission.

Patients admitted for observation reasons are 1.95 times more likely to be readmitted within one month compared to patients who were not admitted for observation. As "Initial_admin_Observation_Admission" increases by one unit, the odds of the outcome increases by 94.63%. This increase is not as significant compared to "Initial_admin_Emergency_Admission," however, it suggests that observation admissions also carry a high risk of readmission.

Patients with high blood pressure are 2.33 times, or 133.42%, more likely to be readmitted compared to those who do not have high blood pressure. High blood pressure could contribute to complications or a higher likelihood of unresolved health issues, increasing the risk of readmission.

Patients with strokes are 4.68 times more likely, or 368.37%, to be readmitted compared to patients who have not had strokes. This indicates that strokes are a significant factor for readmission, as patients would need thorough follow-ups depending on complications that arise from said stroke.

The odds ratio for "Complication_risk" is 2.16, meaning that a one-unit increase in complication risk increases odds of readmission by 115.82%. This shows the importance of addressing complication risks during the discharge planning process to help reduce readmission rates.

"Arthritis" has an odds ratio of 0.29, meaning that patients with arthritis are 70.91% less likely to be readmitted compared to those without arthritis. This means that arthritis, although it is a chronic but manageable condition, does not significantly contribute to readmission rates.

The odds ratio for "Diabetes" is 1.58, indicating that patients with diabetes are 58.07% more likely to be readmitted. This shows that it is not a very significant contributor to readmission rates but still has some influence.

The odds ratio for "Asthma" is 0.30, showing that patients with asthma are 70.13% less likely to be readmitted. This could be due to effective management strategies and relatively fewer severe complications compared to other chronic conditions.

The odds ratio for “Initial_days” is 3.80, meaning that each additional day spent in the hospital during the initial stay increases the odds of readmission by 279.78%. Longer hospital stays often signify severe medical complications, which can increase the likelihood of readmission.

The odds ratio for “Anxiety” is 0.42, indicating that patients with anxiety are 58.33% less likely to be readmitted. This could suggest that anxiety does not have a significant impact on the likelihood of readmission compared to other medical conditions like stroke or high blood pressure.

The reduced logistic regression model shows there is strong statistical significance, with all predictor variables having p-values below the threshold of 0.05, indicating that they significantly contribute to explaining the likelihood of patient readmission. The predictor variables and their respective p-values are as follows: “Initial_admin_Emergency_Admission” ($p = 0.000$), “Initial_admin_Observation_Admission” ($p = 0.011$), “HighBlood” ($p = 0.000$), “Stroke” ($p = 0.000$), “Complication_risk” ($p = 0.000$), “Arthritis” ($p = 0.000$), “Diabetes” ($p = 0.033$), “Asthma” ($p = 0.000$), “Initial_days” ($p = 0.000$), and “Anxiety” ($p = 0.000$). These results indicate that each of these variables plays a meaningful role in predicting the patient readmission.

The reduced model has a pseudo R-squared value of 0.9441, indicating a strong model fit and suggesting that the predictors explain a substantial proportion of the variability in patient readmissions. The Variance Inflation Factor (VIF) values for all predictors are well below the threshold of 10, with the highest being 2.59 for “Complication_risk” and 2.27 for “Initial_days,” confirming that multicollinearity is not an issue in the reduced model and complying with the assumption for logistic regression.

The reduced logistic regression model also gives meaningful, real-world insights into factors that contribute to patient readmissions within one month after release. Key predictors such as “Initial_admin_Emergency_Admission,” with an odds ratio of 8.84, shows that patients admitted through emergency means are at a substantially higher risk of being readmitted. Similarly, “Initial_days,” with an odds ratio of 3.80, shows that longer hospital stays are strongly associated with increased readmission risk, likely because of the severity or complexity of the patient’s condition. Other predictors, such as “Stroke” (OR = 4.68), “HighBlood” (OR = 2.33), and “Complication_risk” (OR = 2.16), highlight the importance of managing chronic and acute conditions to reduce readmissions. On the other hand, factors such as Arthritis (OR = 0.29) and Asthma (OR = 0.30) show that patients with these conditions are less likely to be readmitted, possibly due to effective and easy long-term management practices.

These findings align closely with the research question “Which factors contribute to patients being readmitted one month after release?” by identifying specific predictors that significantly

impact readmission rates, the model provides actionable insights for healthcare providers to implement. For example, patients admitted under emergency conditions or those with longer initial hospital stays can undergo enhanced post-discharge care and comprehensive follow-ups.

The logistic regression model provides useful insights, but it has some limitations. It is sensitive to outliers, which can distort results despite using Cook's Distance to identify influential observations. Additionally, the model identifies correlations but cannot confirm causation, meaning factors like emergency admissions and longer hospital stays, while significant, may not directly cause readmissions.

Section F2) Recommendations

Based on the results from the data analysis, a targeted approach should be implemented to reduce patient readmissions rates within one month of release. The reduced model revealed key predictors, such as emergency admissions, long initial hospital stays, stroke, and high blood pressure, as strong contributors to readmission rates. To address these factors, healthcare providers should implement enhanced post-discharge care for high-risk patients. For example, patients admitted for emergency reasons or those with longer hospital stays should receive structured follow-up plans, including scheduled check-ins, care coordination, and access to home health services depending on the severity of their medical condition.

Chronic conditions such as stroke and high blood pressure should be closely monitored with customized care plans involving managing these conditions after discharge. Patients with asthma and arthritis could benefit from management strategies that reduce the likelihood of complications that would require readmission, further decreasing readmission rates. For example, asthma management can include long-term medications, inhalers, and education programs. Arthritis management can involve consistent treatment plans, such as anti-inflammatory medications and physical therapy. Additionally, mental health interventions, such as counseling and support programs, can help mitigate anxiety-related effects on physical health, given its association with lower readmission odds.

This recommended course of action allows for a more efficient allocation of resources to prevent unnecessary readmissions, potentially leading to improved patient outcomes and save both patients and hospitals money.

Part VI: Demonstration

Section G) Panopto Demonstration

https://wgu.hosted.panopto.com/Panopto/Pages/Viewer.aspx?id=c51b5fe4-d2af-411a-8fb3-b26401799822

Section H) Sources of Third-Party Code

GeeksforGeeks (n.d.). *How to perform ordinal encoding using sklearn?* GeeksforGeeks. Retrieved January 2, 2025, from https://www.statology.org/multiple-linear-regression-assumptions/

Bobbit, Z. (2020). *How to Calculate Cook's Distance in Python*. Statology Retrieved January 8, 2025, from https://www.statology.org/cooks-distance-python/

Section I) Sources

Athimala, J., Chiu, P., & Espinosa Torrés, J. (n.d.). *How can you interpret coefficients in logistic regression?* LinkedIn. https://www.linkedin.com/advice/0/how-can-you-interpret-coefficients-logistic-regression-t53gf

Bobbit, Z. (2020, December 23). *How to Calculate Cook's Distance in Python*. Statology. https://www.statology.org/cooks-distance-python/

Bobbitt, Z. (2020, October 13). *The 6 Assumptions of Logistic Regression (With Examples)*. Statology. Retrieved January 7, 2025, from https://www.statology.org/assumptions-of-logistic-regression/

Hayes, A. (2022, January 10). *Stepwise Regression: Definition, Uses, Example, and Limitations*. Investopedia. Retrieved January 9, 2025, from https://www.investopedia.com/terms/s/stepwise-regression.asp

IBM. (2024, June 19). *What is Data Transformation?* IBM. Retrieved January 8, 2025, from https://www.ibm.com/think/topics/data-transformation

Interpreting logit results, what do coefficients mean? (n.d.). Stack Exchange. https://stats.stackexchange.com/questions/523540/interpreting-logit-results-what-do-coefficients-mean

Jain, S. (2024, April 3). *How to interpret odds ratios in logistic regression*. GeeksforGeeks. Retrieved January 9, 2025, from https://www.geeksforgeeks.org/how-to-interpret-odds-ratios-in-logistic-regression/

JMP. (n.d.). *Simple Logistic Regression*. JMP. https://www.jmp.com/en_us/learning-library/topics/correlation-and-regression/simple-logistic-regression.html

Kundu, R. (2022, September 13). *Confusion Matrix: How To Use It & Interpret Results [Examples]*. V7 Labs. Retrieved January 8, 2025, from https://www.v7labs.com/blog/confusion-matrix-guide

Singh, V. (2024, November 18). *Variance Inflation Factor: How to Detect Multicollinearity*. DataCamp. Retrieved January 9, 2025, from https://www.datacamp.com/tutorial/variance-inflation-factor

Western Governors University. (n.d.). Getting Started D208, part II [Slide 14].

Western Governors University. (n.d.). *Medical Data Considerations and Dictionary*.