**RSIP Career Plus ML 044**

**PROJECT TITLE**

**Predicting Hospital Readmission for Patients with Diabetics**

**Category: Machine Learning**

**Skills Required:**  
Python,Python Web Frame Works,Python For Data Analysis,Python For Data Visualization,Data Preprocessing Techniques,Machine Learning,Classification Algorithms

**TEAM MEMBERS:**

**K.Indhumadhi**

**S.Bagavathi@ Bavatharani**

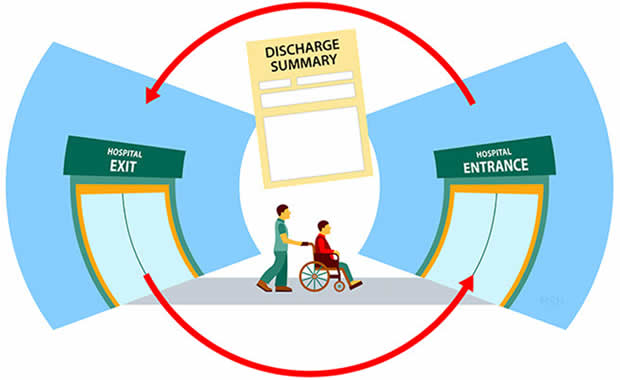
**K.Dharsini Prabha**



**Project Description:**

   A hospital readmission is when a patient who is discharged from the hospital, gets re-admitted again within a certain period of time. Hospital readmission rates for certain conditions are now considered an indicator of hospital quality, and also affect the cost of care adversely. For this reason, Centers for Medicare & Medicaid Services established the Hospital Readmissions Reduction Program which aims to improve quality of care for patients and reduce healthcare spending by applying payment penalties to hospitals that have more than expected readmission rates for certain conditions.Being able to determine factors that lead to higher readmission in such patients, and correspondingly being able to predict which patients will get readmitted can help hospitals save millions of dollars while improving quality of care.

**PREDICTING HOSPITAL READMISSION FOR PATIENTS WITH DIABETICS:**



**Introduction:**

   As the healthcare system moves toward value-based care, CMS has created many programs to improve the quality of care of patients. One of these programs is called the Hospital Readmission Reduction Program ([HRRP](https://www.cms.gov/Medicare/Quality-Initiatives-Patient-Assessment-Instruments/Value-Based-Programs/HRRP/Hospital-Readmission-Reduction-Program.html)), which reduces reimbursement to hospitals with above average readmissions. For those hospitals which are currently penalized under this program, one solution is to create interventions to provide additional assistance to patients with increased risk of readmission. But how do we identify these patients? We can use predictive modeling from data science to help prioritize patients.One patient population that is at increased risk of hospitalization and readmission is that of diabetes. Diabetes is a medical condition that affects approximately 1 in 10 patients in the United States. According to Ostling et al, patients with diabetes have almost double the chance of being hospitalized than the general population ([Ostling et al 2017](https://clindiabetesendo.biomedcentral.com/articles/10.1186/s40842-016-0040-x)). Therefore, in this article, I will focus on predicting hospital readmission for patients with diabetes.In this project we will demonstrate how to build a model predicting readmission in Python using the following steps

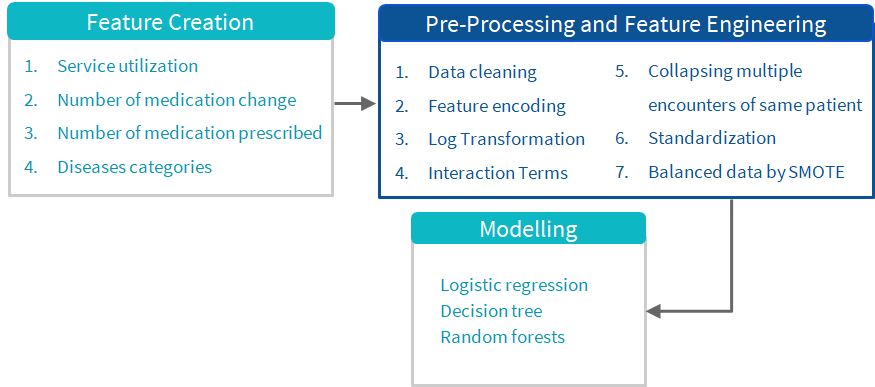
* data exploration
* feature engineering
* building training/validation/test samples
* model selection
* model evaluation

**Project Definition**  **:**

   Predict if a patient with diabetes will be readmitted to the hospital within 30 days.

**Overall process**  **:**

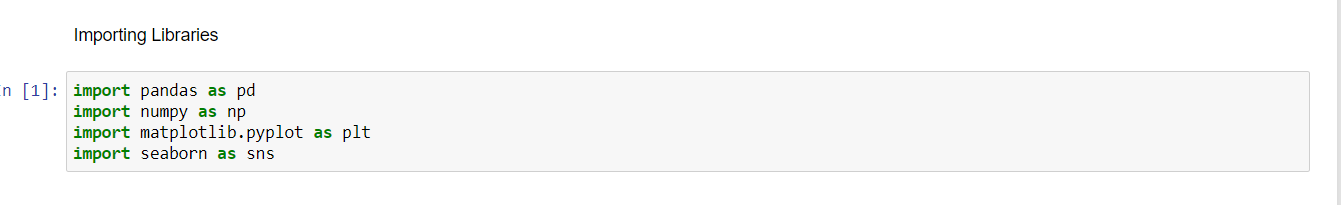
   From this point onward, we broadly followed the process as shown in the figure below. However, it is important to note that there isn’t much of a clean separation between steps, and there is a lot of back and forth iterations when trying different approaches to feature engineering and modelling. Also note that there is a quite a bit of overlap between terms like feature creation, feature engineering and pre-processing, depending on whom you talk to.



**Data Preparation and Exploration:**

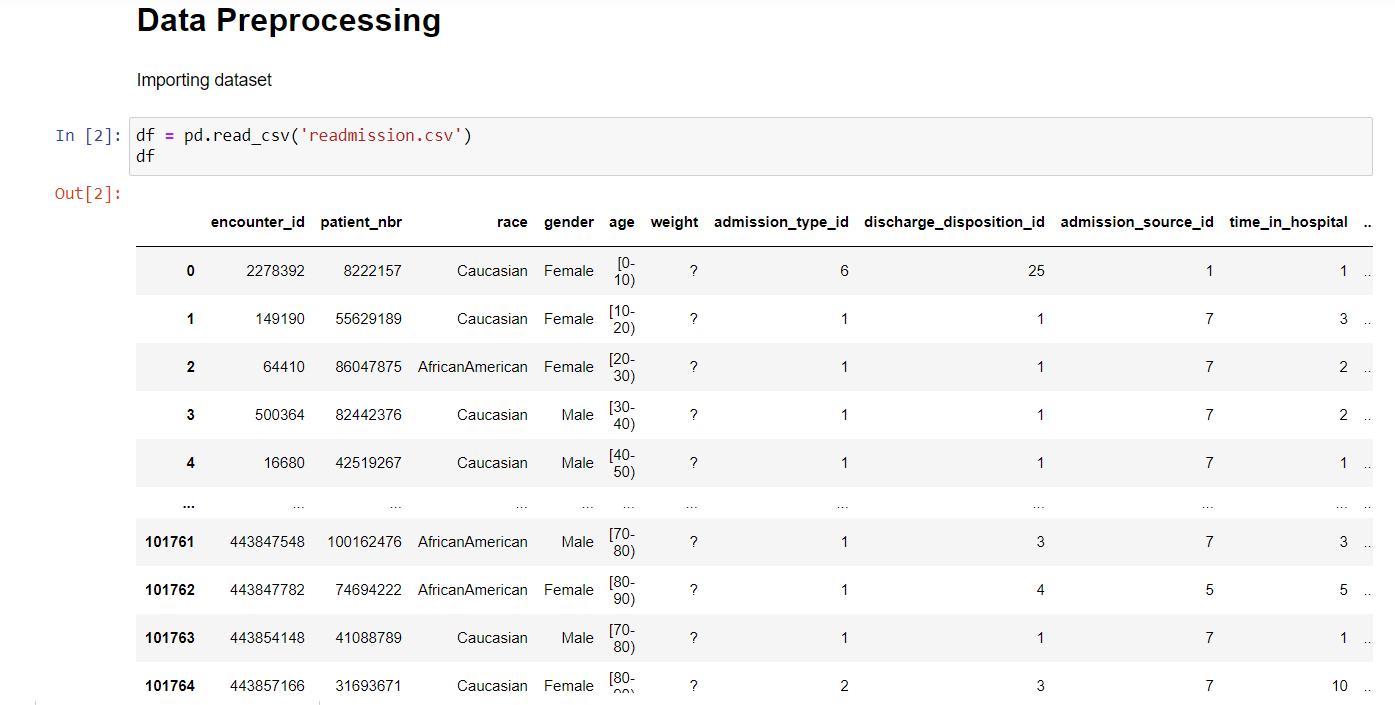
   Importing the Libraries:

     Pandas,Numpy,Matplotlib and Seaborn

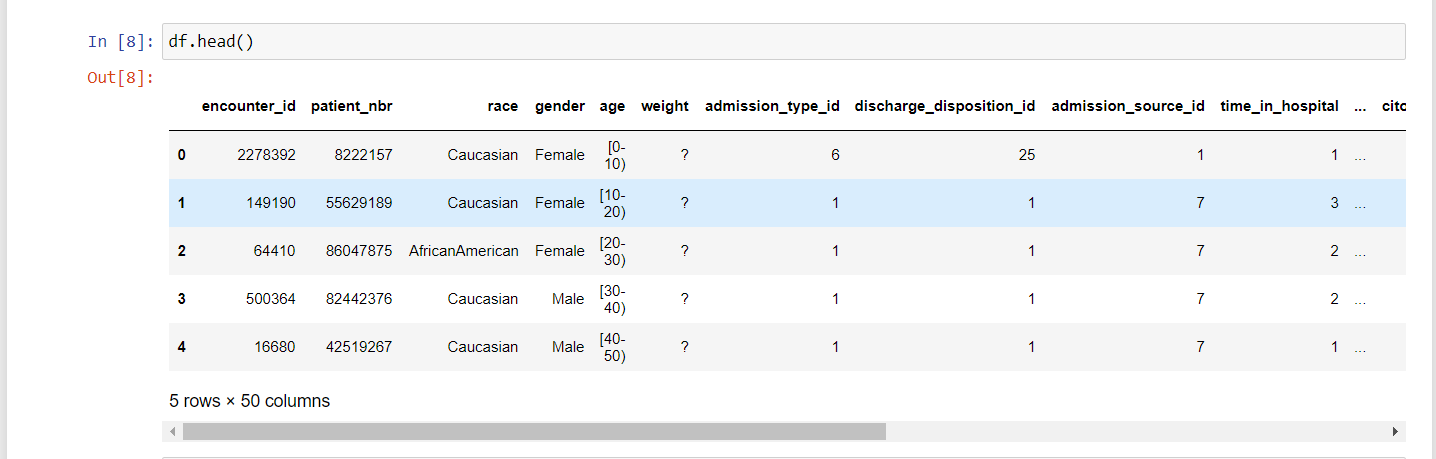


**Data Preprocessing:**

         Here the dataset named readmission.csv is used.



**Data Collection:**



**Dealing with Missing Values**  **:**

   Variable weight contains approximate 98% of the missing values so there is no significance in filling those missing values so we decided to drop these variables. Variable Payer code and medical specialty contains approximate 40% missing values so we also dropped these variables. Variables race, diag\_1, diag\_2, diag\_3 and gender contains very less missing values as compared to other attributes which we dropped so for these attributes we also decided to drop those where missing values contains.

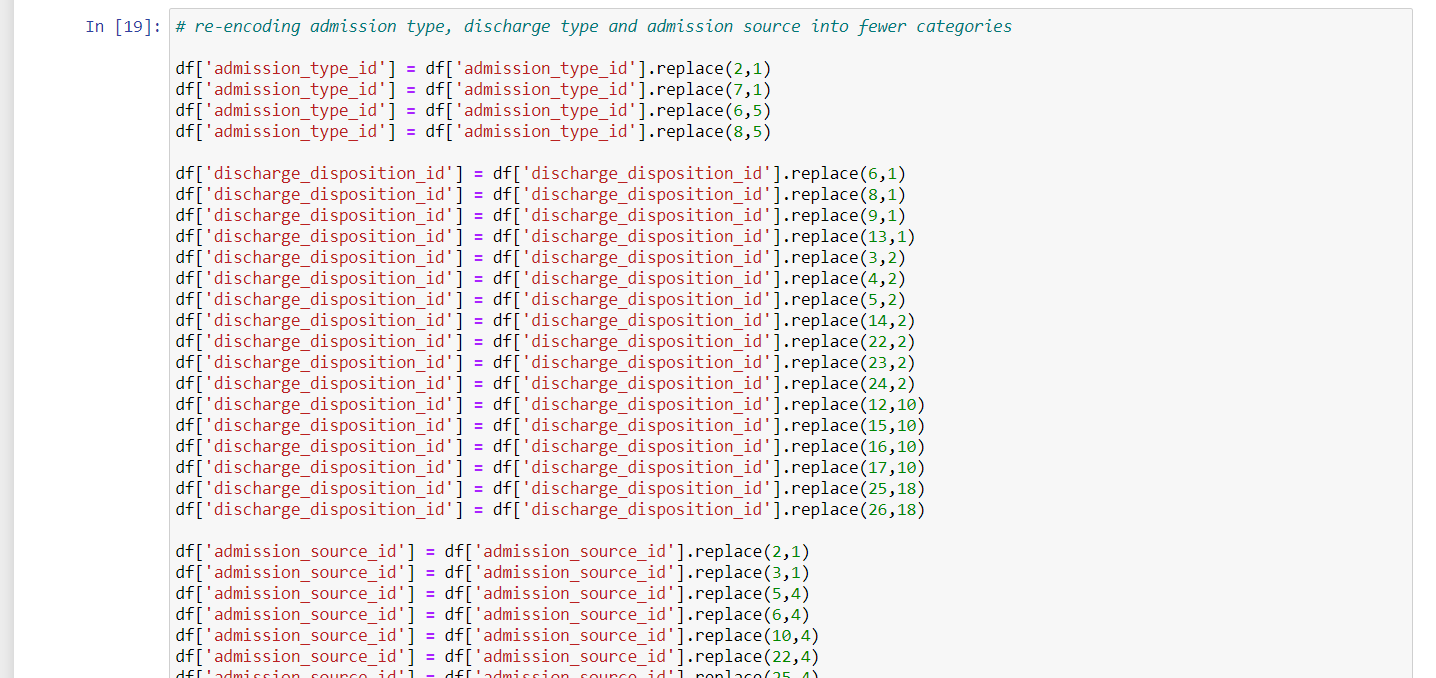


**Feature Engineering**  **:**

   This is highly subjective, and partly depends on a knowledge of health care services, and making sense of the potential relationships between features. There are perhaps thousands of ways to try here. We tried some...

* **Service utilization:** The data contains variables for number of inpatient (admissions), emergency room visits and outpatient visits for a given patient in the previous one year. These are (crude) measures of how much hospital/clinic services a person has used in the past year. We added these three to create a new variable called service utilization (see figure below). The idea was to see which version gives us better results. Granted, we did not apply any special weighting to the three ingredients of service utilization but we wanted to try something simple at this stage.
* **Number of medication changes:** The dataset contains 23 features for 23 drugs (or combos) which indicate for each of these, whether a change in that medication was made or not during the current hospital stay of patient. Medication change for diabetics upon admission has been shown by previous research to be associated with lower readmission rates. We decided to count how many changes were made in total for each patient, and declared that a new feature. The reasoning here was to both simplify the model and possibly discover a relationship with number of changes regardless of which drug was changed.

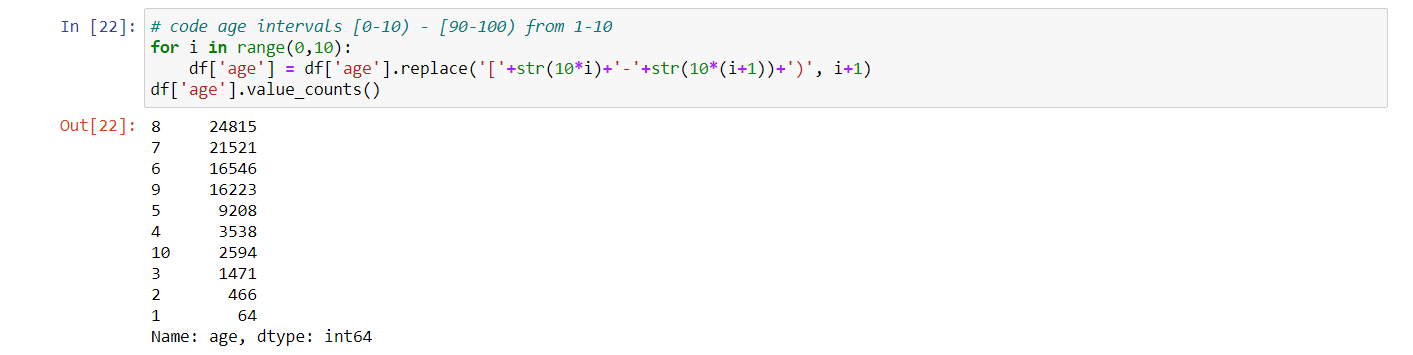




**Encoding some variables:** The original dataset used string values for gender, race, medication change, and each of the 23 drugs used. To better fit those variables into our model, we interpret the variables to numeric binary variables to reflect their nature. For example, we encoded the “ medication change ” feature from “No” (no change) and “Ch” (changed) into 0 and 1.

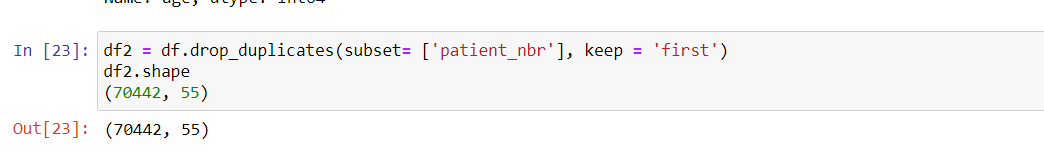


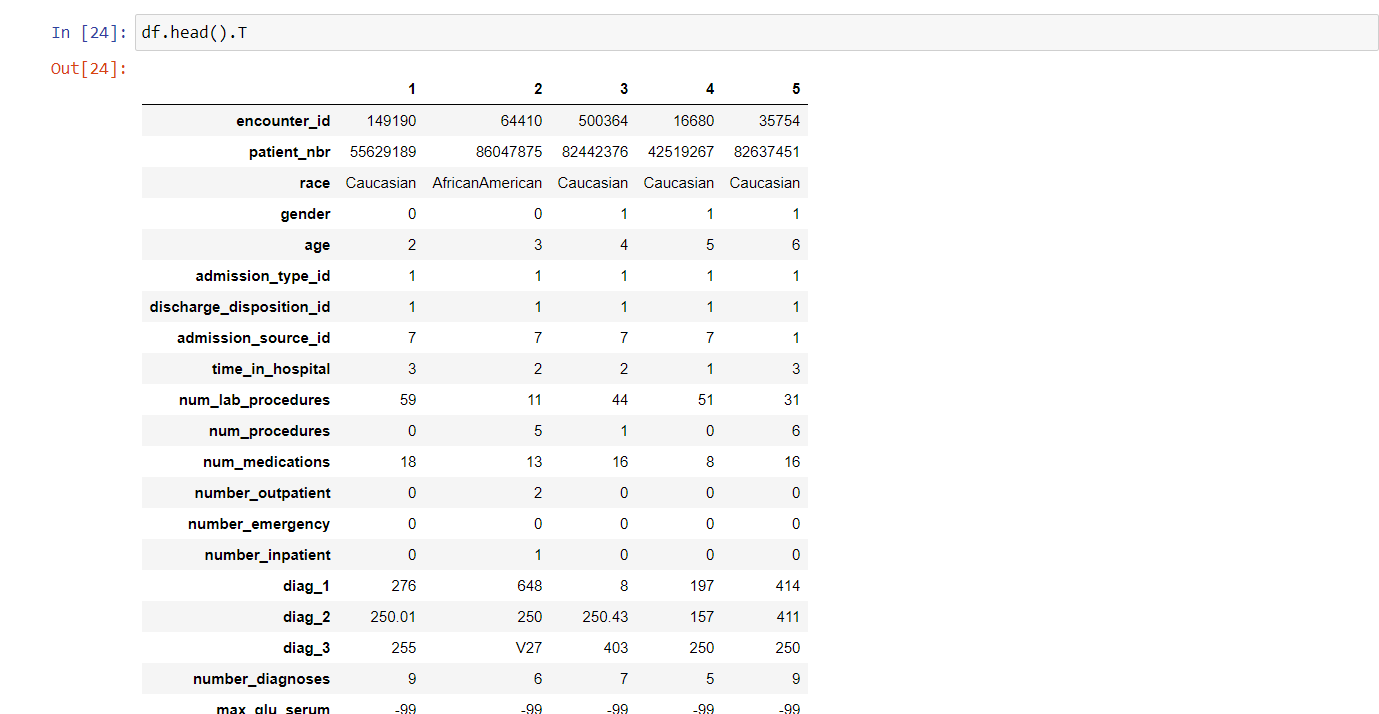
**Dealing with age:** There are different ways to deal with this. The dataset only gives us age as 10 year categories, so we don’t know the exact age of each patient. The previous study on this dataset used age categories as nominal variables, but we wanted to be able to see the effect of increasing age on readmission, even if in a crude way. To do that, we assume that age of the patient on average lies at the midpoint of the age category. For example, if the patient’s age category is 20–30 years, then we assume the age = 25 years. So we converted age categories to midpoints, resulting in a numeric variable:



Collapsing of Multiple Encounters for same patient Some patients in the dataset had more than one encounter.We could not count them as independent encounters because that bias the results towards those patients who had multiple encounters. Thus we tried multiple techniques to collapse and consolidate multiple encounters for same patient such as:

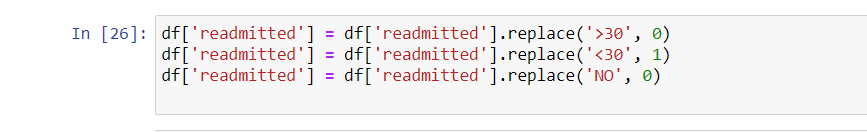
* Considering more than 2 readmissions across multiple encounters as readmission for collapsed record.
* Considering average stay at hospital across multiple encounters.
* Considering the percentage of the medication changes across multiple encounters
* Considering the total number of the encounters to replace the encounter unique ID
* Considering the combination of diagnoses across multiple encounters as a list However, taking the features such as “diagnosis”, for instance, we did not find it not meaningful to combine multiple categorical values into an array for building data model. We then considered first encounter and last encounter separately as possible representations of multiple encounters. However, last encounters gave extremely imbalanced data for readmissions (96/4 Readmissions vs No Readmissions) and thus, we decided to use first encounters of patients with multiple encounters. This resulted in dataset being reduced to about 70,000 encounters:





**Encoding the outcome variable:** The outcome we are looking at is whether the patient gets readmitted to the hospital within 30 days or not. The variable actually has < 30, > 30 and No Readmission categories. To reduce our problem to a binary classification, we combined the readmission after 30 days and no readmission into a single category:





**Categorization of diagnoses:** The dataset contained up to three diagnoses for a given patient (primary, secondary and additional). However, each of these had 700–900 unique ICD codes and it is extremely difficult to include them in the model and interpret meaningfully. Therefore, we collapsed these diagnosis codes into 9 disease categories in an almost similar fashion to that done in the original publication using this dataset. These 9 categories include Circulatory, Respiratory, Digestive, Diabetes, Injury, Musculoskeletal, Genitourinary, Neoplasms, and Others. Although we did this for primary, secondary and additional diagnoses, we eventually decided to use only the primary diagnosis in our model. Doing this in python was slightly cumbersome because, well, we are mapping the disease codes to certain category names. Below code should demonstrate this easily.

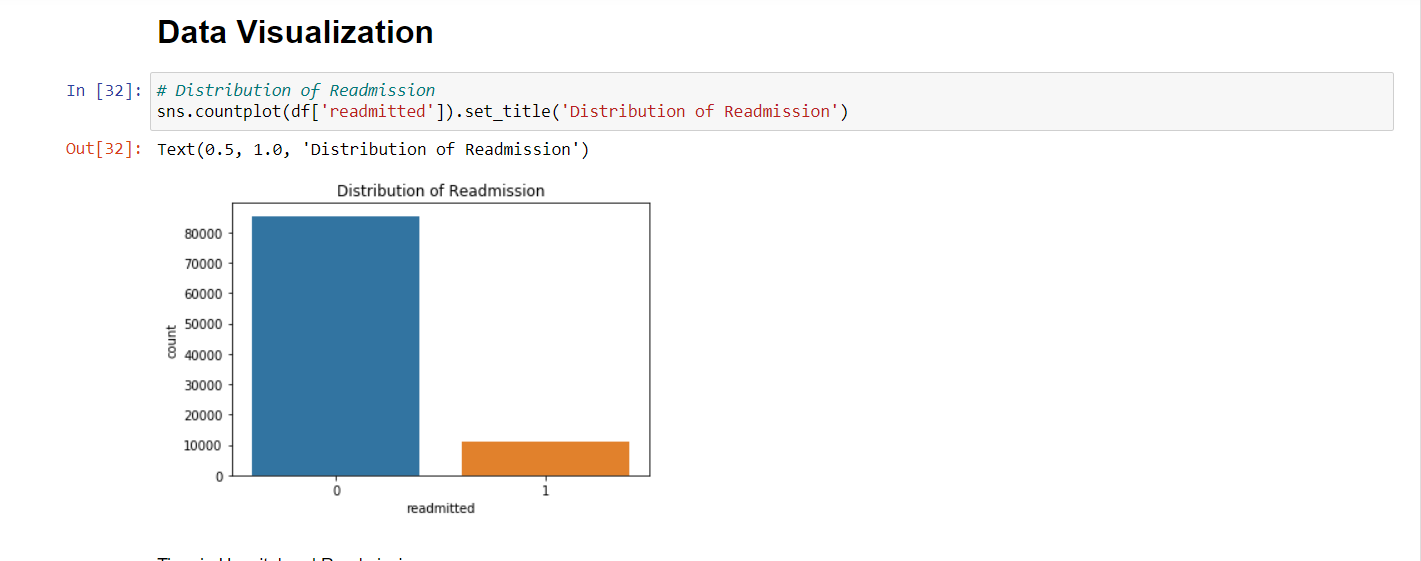




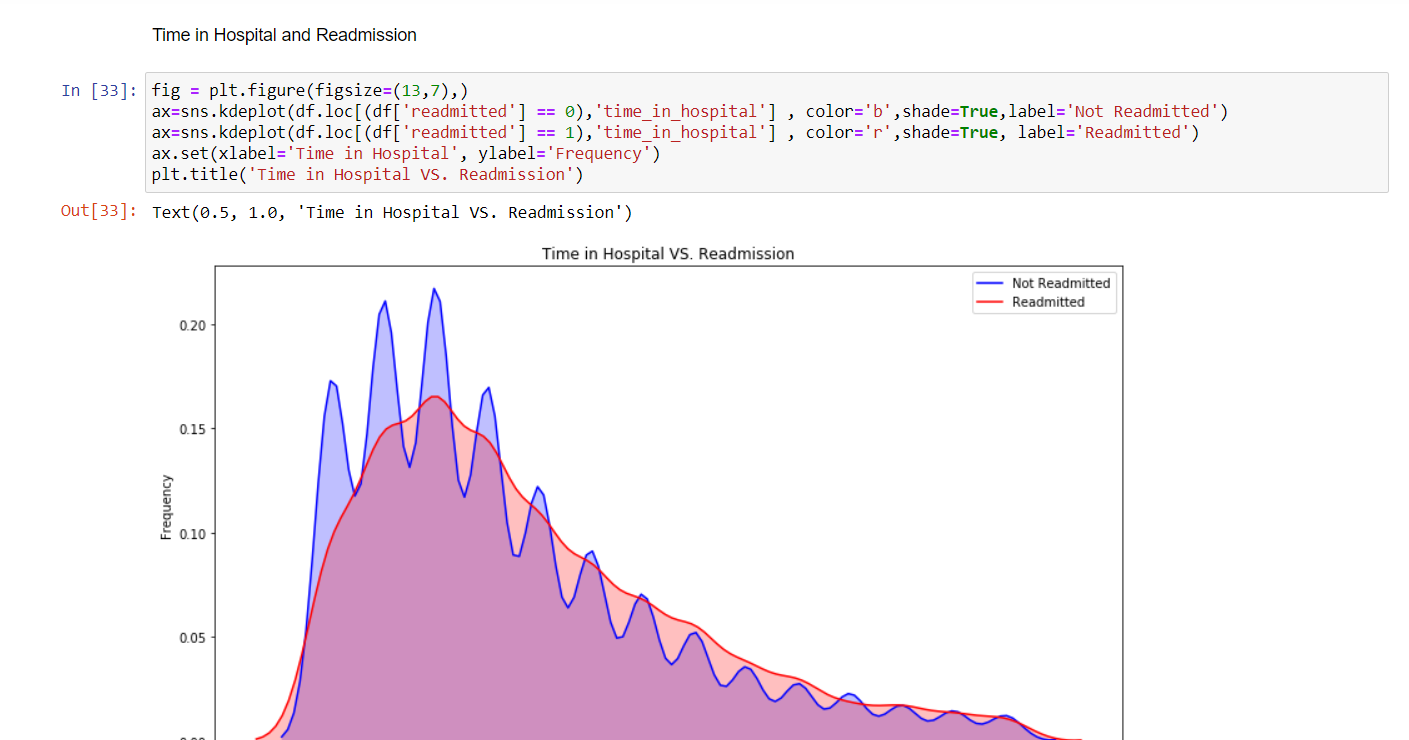
**Data Visualization:**

**1. Distribution of Readmission**  **:**

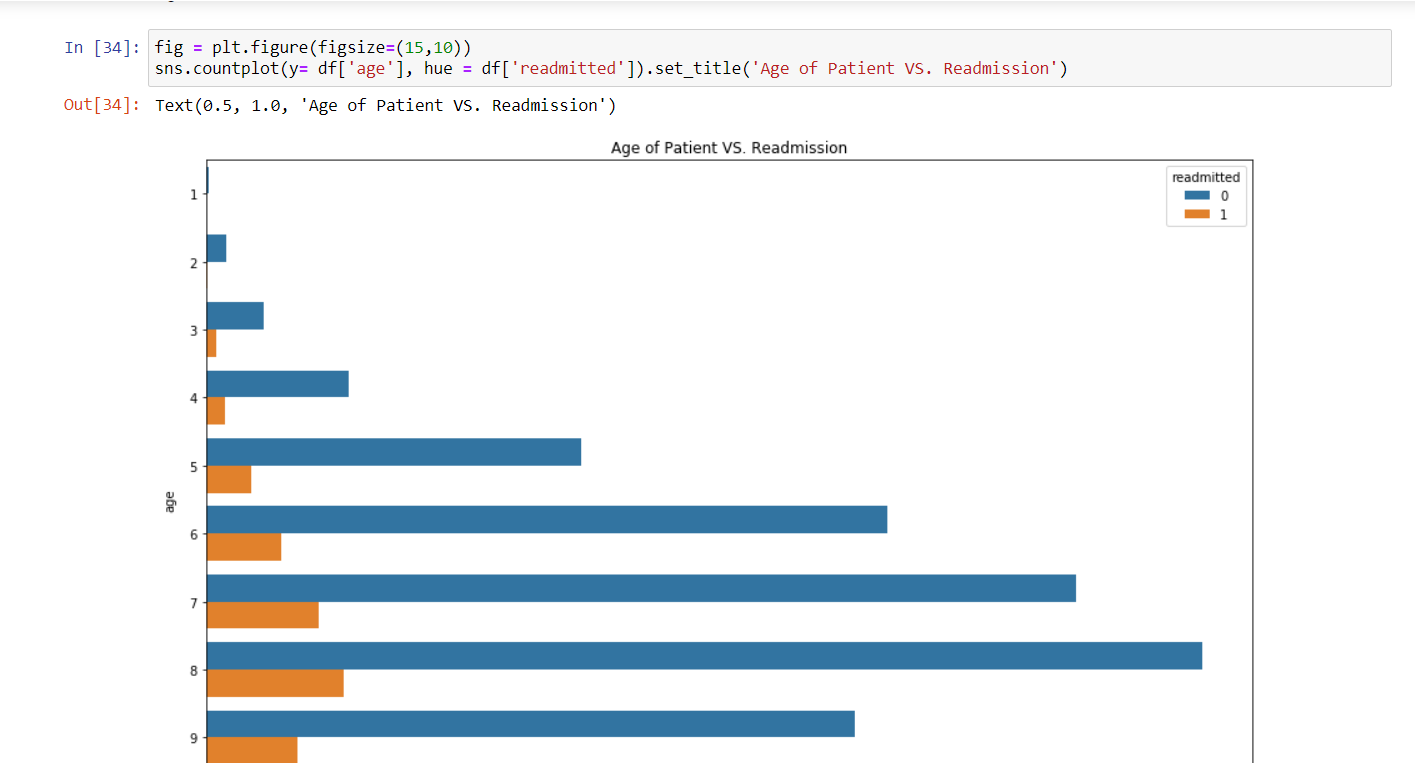
Our target variable is imbalance. Number of readmitted patient are quite less as compared to Not readmitted



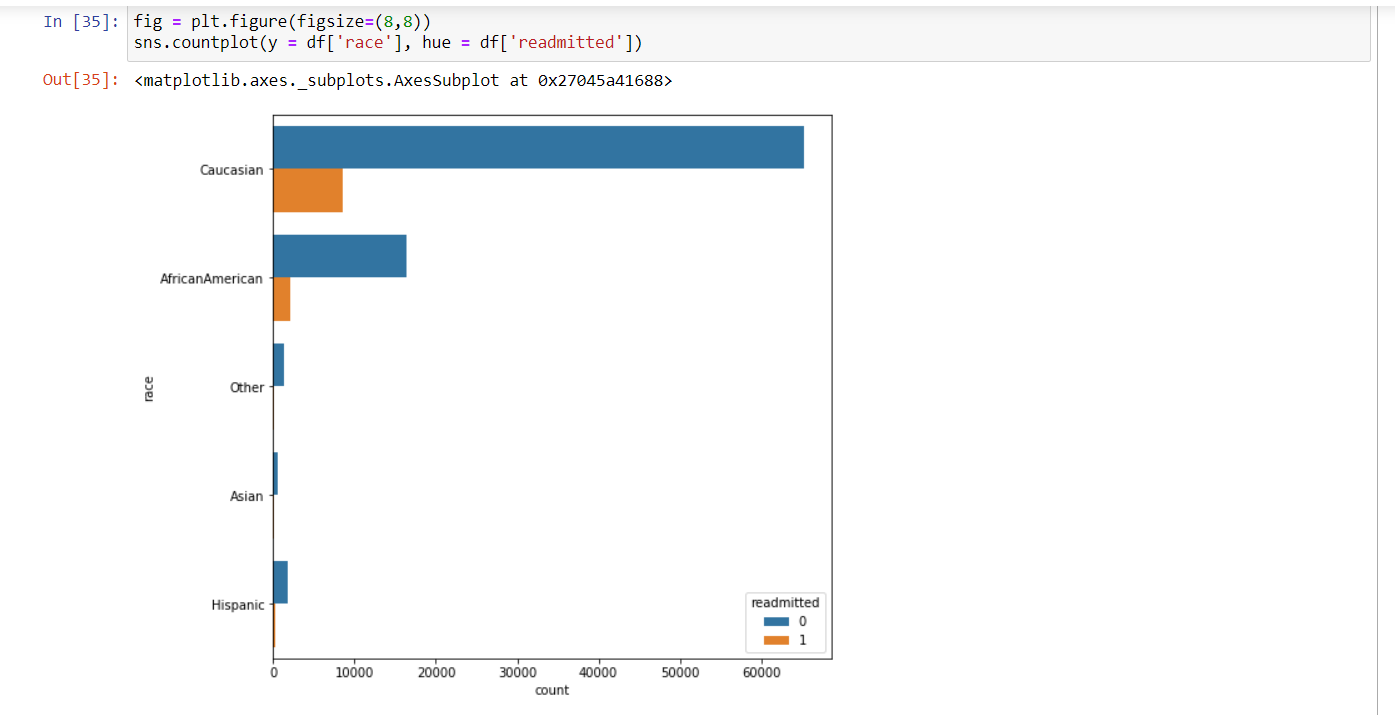
**2.Time in Hospital and Readmission:**



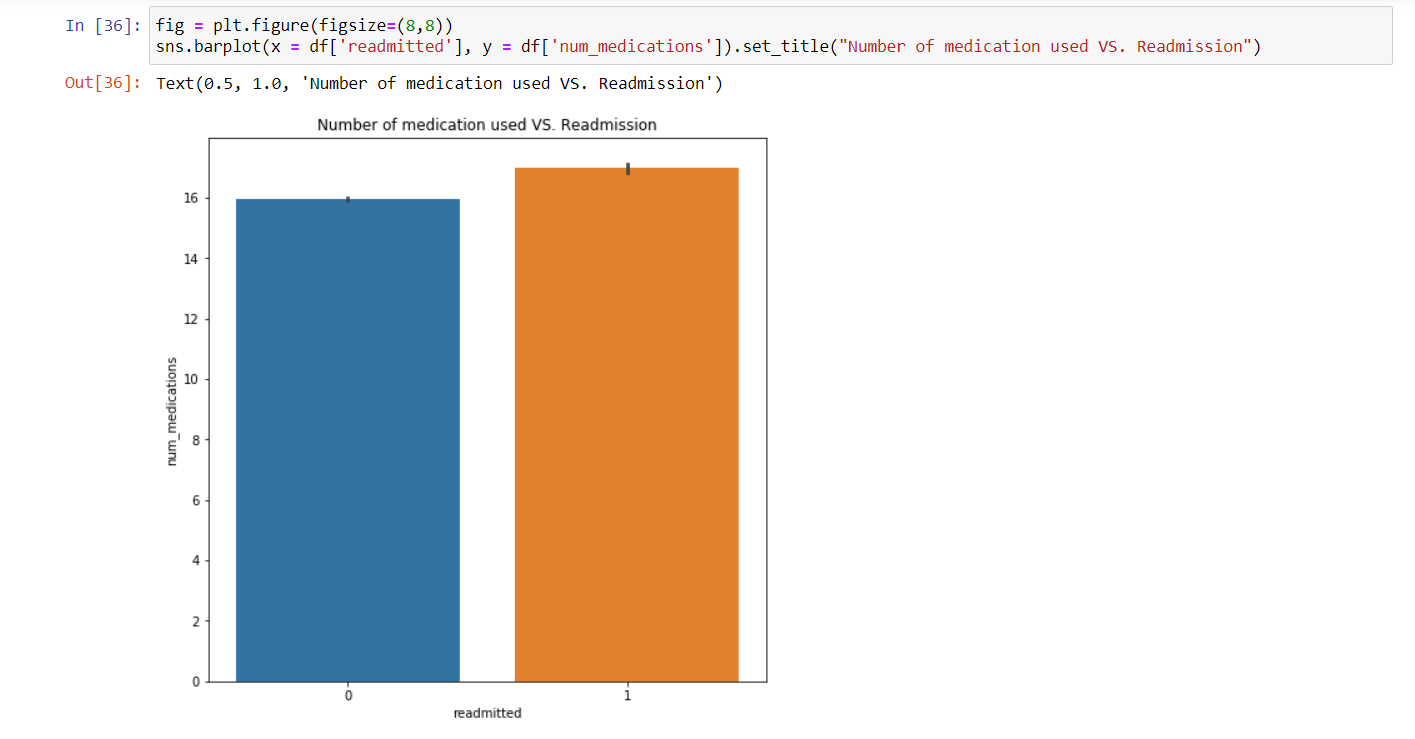
**3. Age and Readmission:**



**4. Ethnicity of patient and Readmission:**



**5. Number of Medications used and Readmission:**



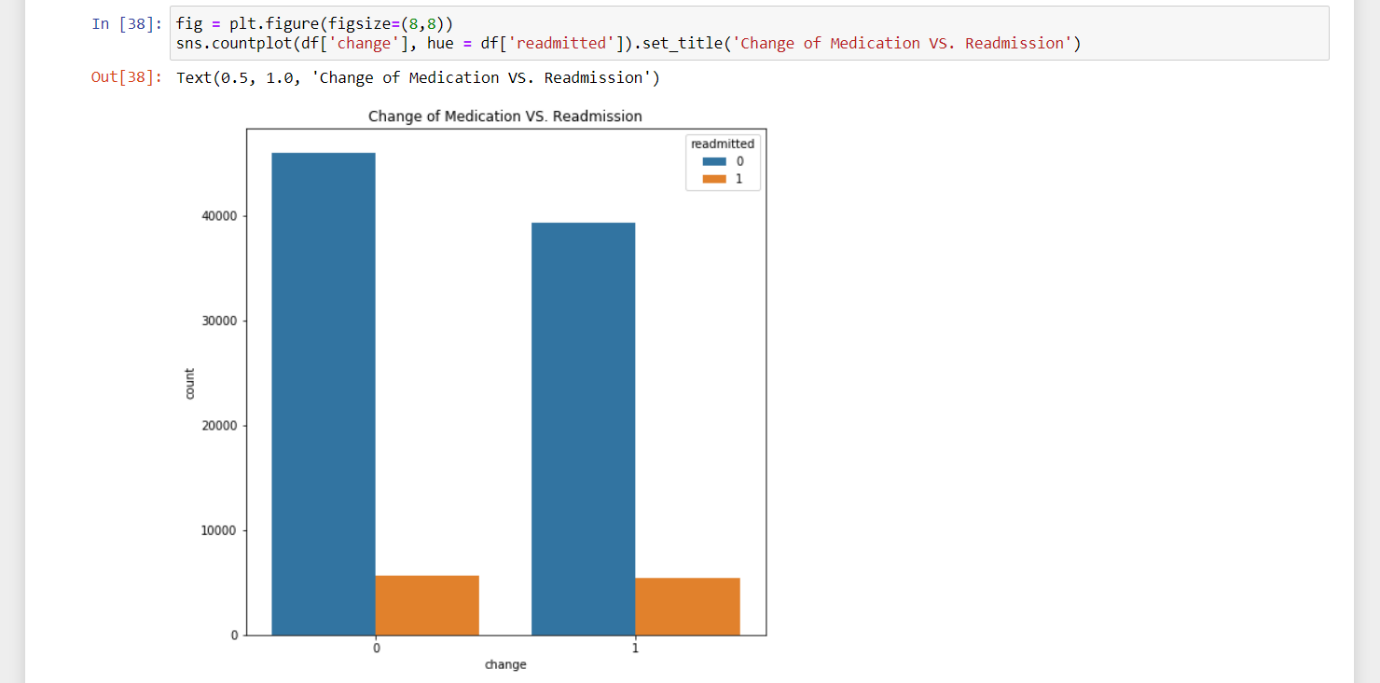
**6. Gender and Readmission:**

                      Male = 1 and Female = 0



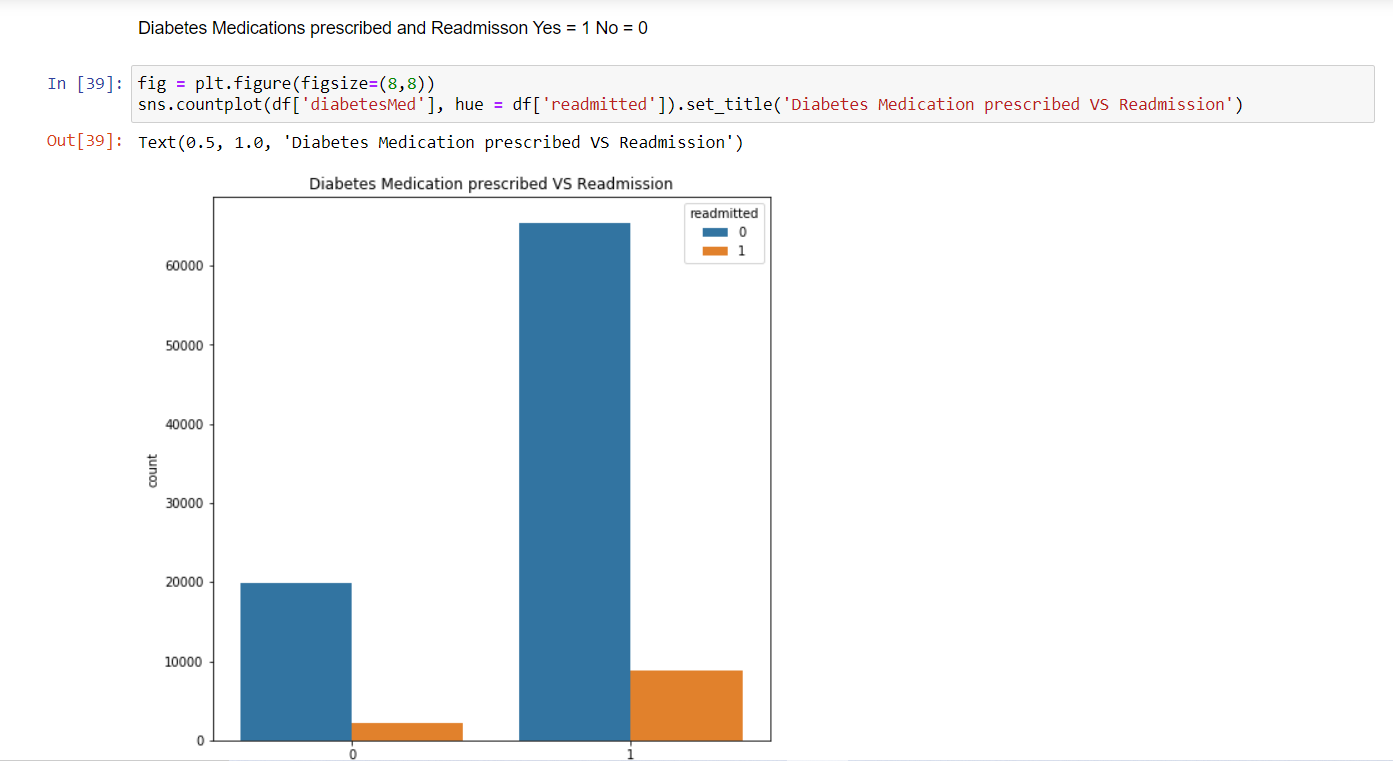
**7. Change of Medication and Readmisson:**

* Change = 1
* No Change = 0

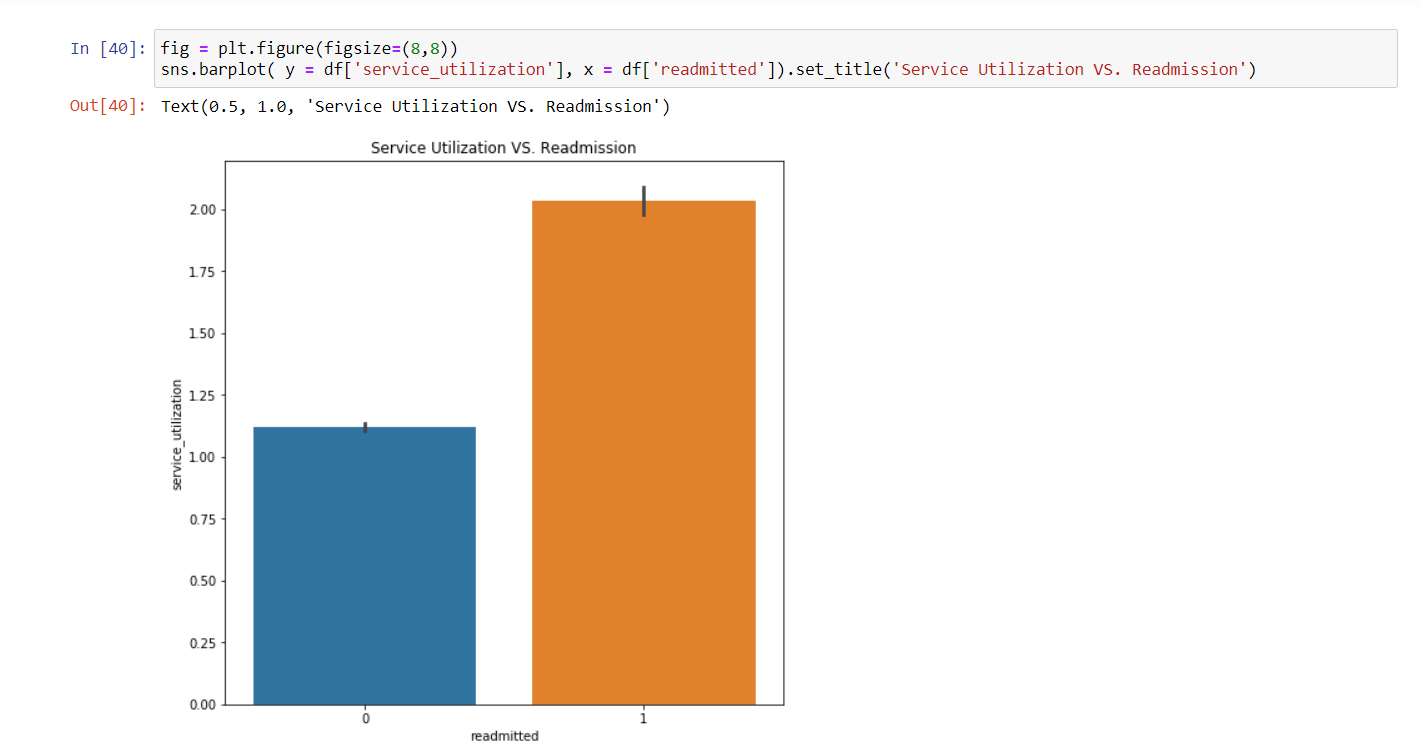


**8. Diabetics Medication Prescribed and Readmission:**

* Change = 1
* No Change = 0



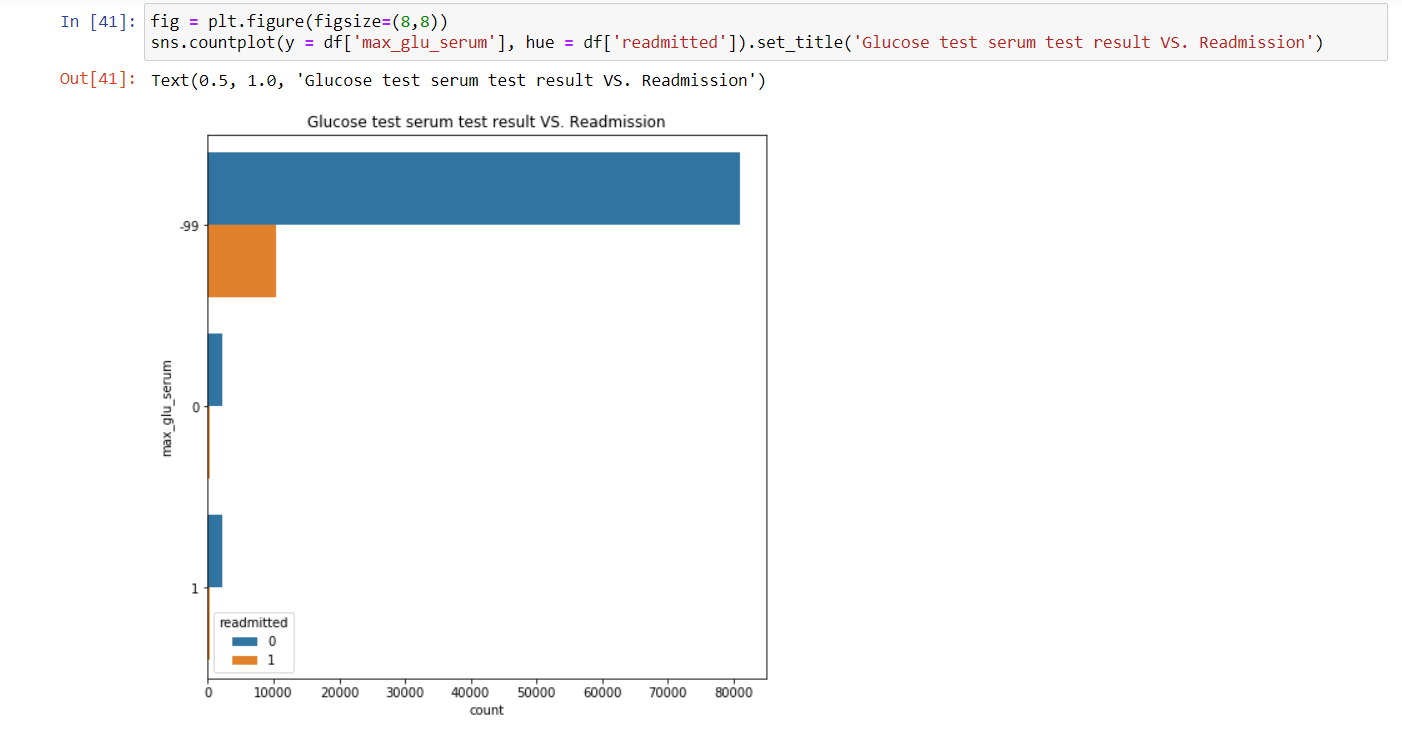
**9. Service utilization and Readmission:**



**10. Glucose serum test result and Readmission**  **:**

*Glucose Serum test* - A blood glucose test is used to find out if your blood sugar levels are in the healthy range. It is often used to help diagnose and monitor diabetes.

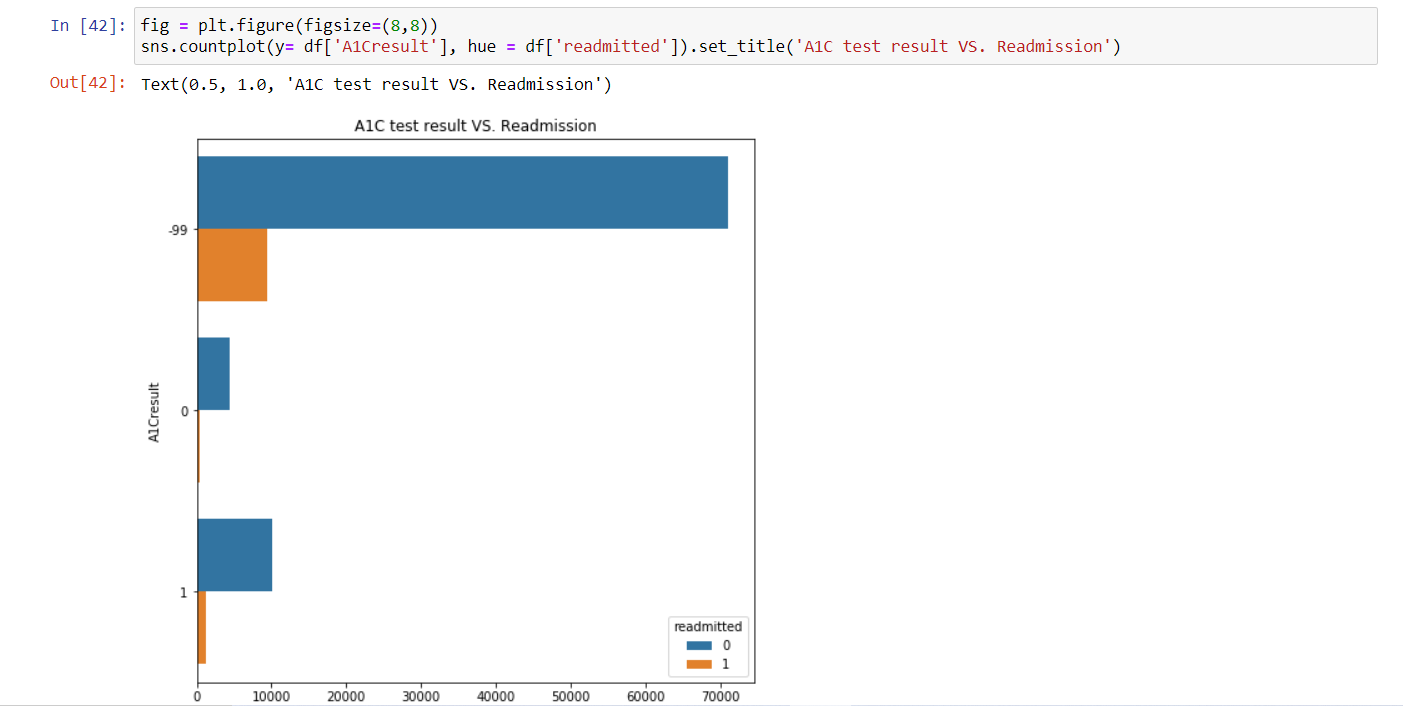
* '>200' : 1 = indicates diabetes
* '>300' : 1 = Indicates diabetes
* 'Norm' : 0 = Normal
* 'None' : -99 = test was not taken



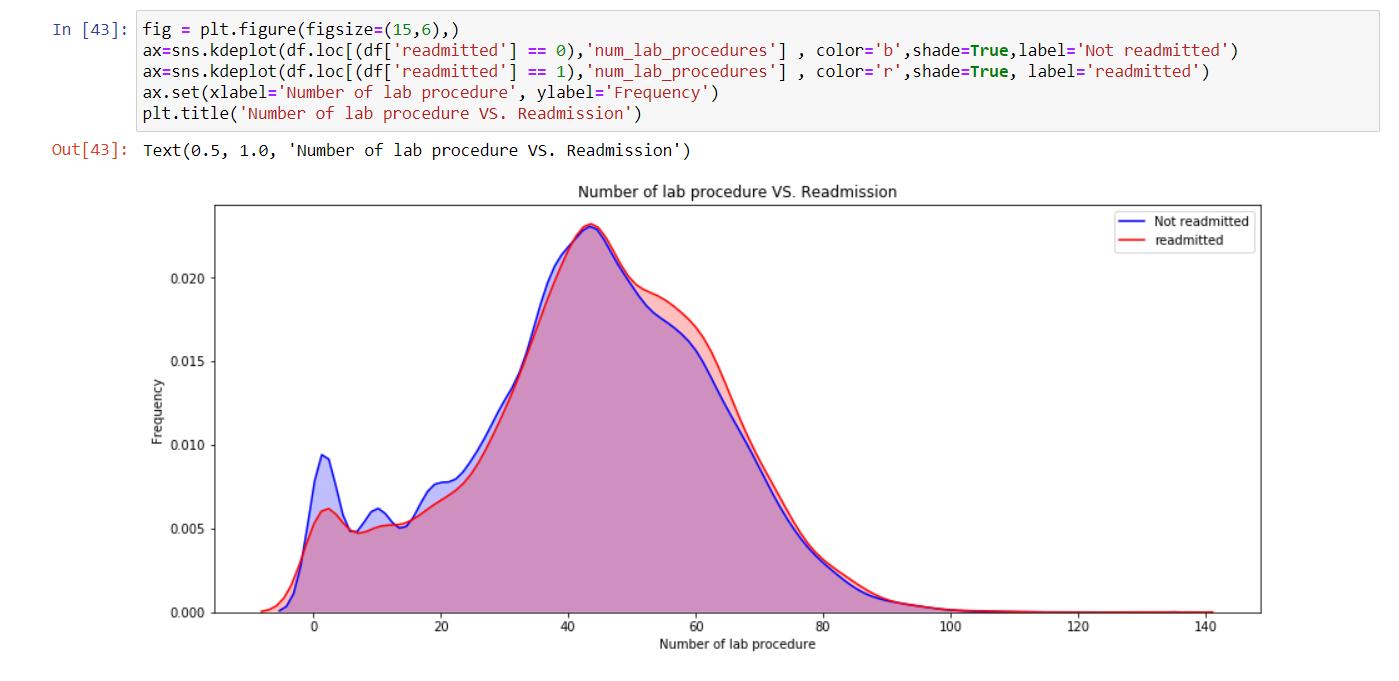
**11. AIC Result and Readmission:**

A1C result and Readmission *A1C test* - The A1C test is a blood test that provides information about your average levels of blood glucose, also called blood sugar, over the past 3 months.

* '>7' : 1
* '>8' : 1
* Norm : 0 = Normal
* None : -99 = Test was not taken

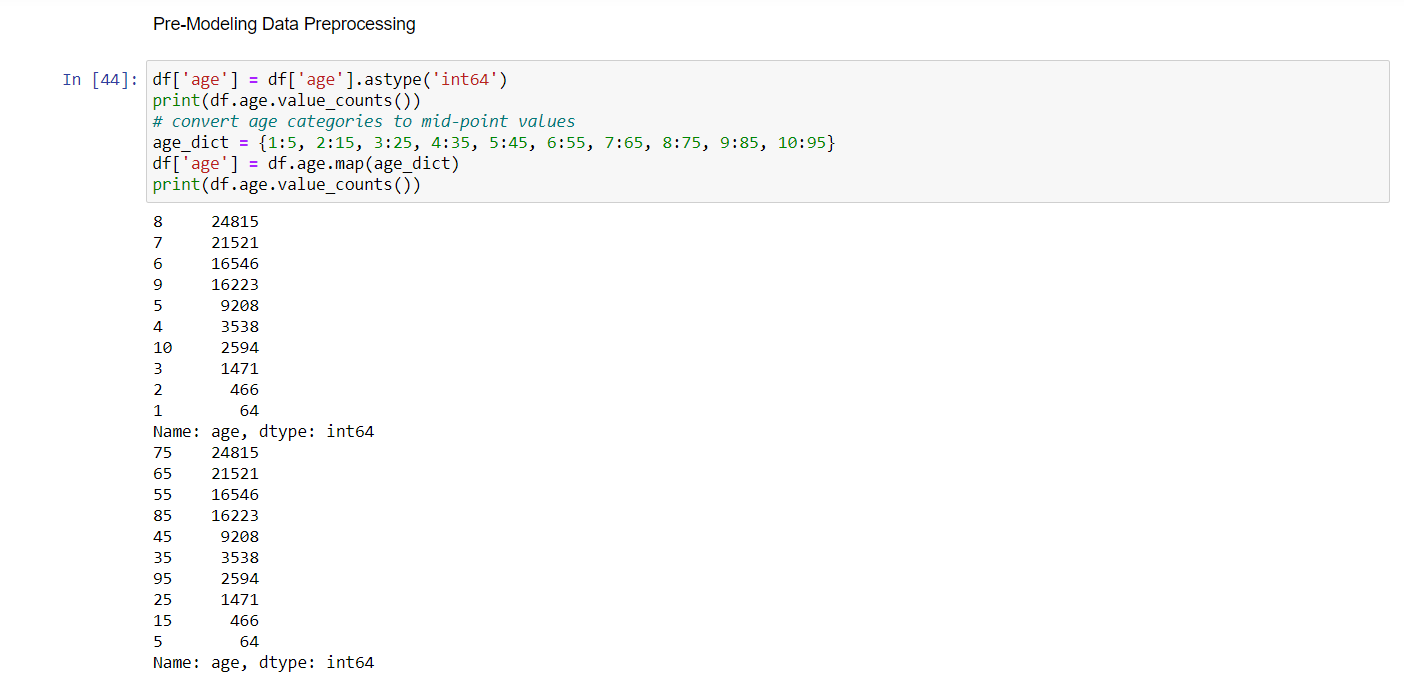


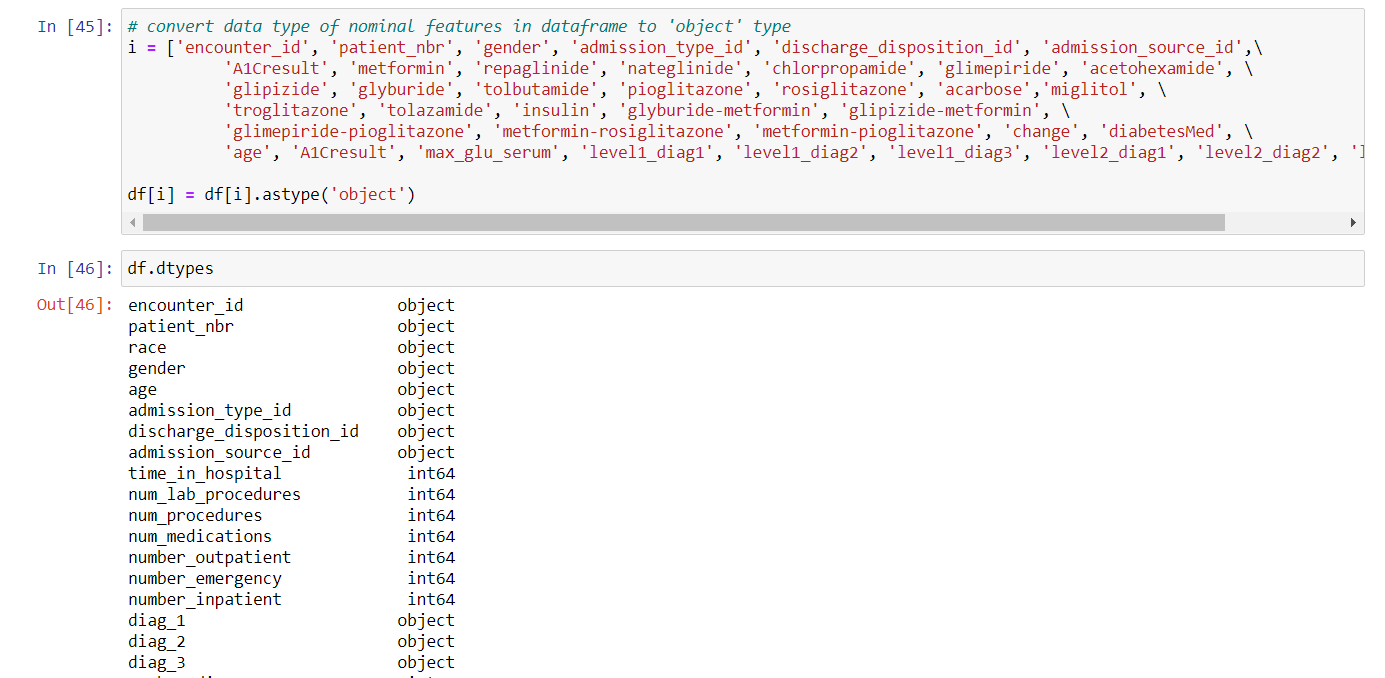
**12. Number of lab procedures and Readmission:**



**Pre-Modeling Data Preprocessing  :**

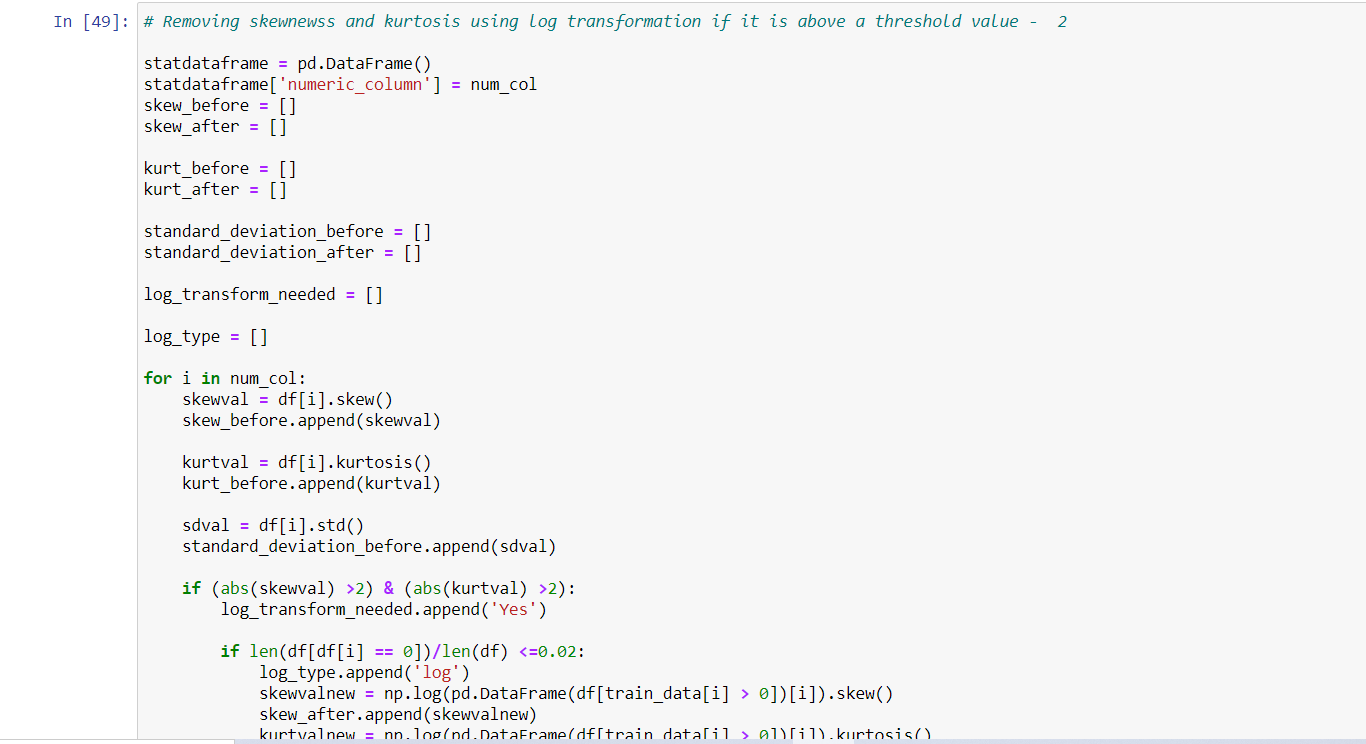
This code converts age as categorical variable to a continuous approximation by assuming mid-point of each age-category as the actual age value. This is done to avoid having to deal with age as a dummy variable in the models which makes interpretation very cumbersome. Also, since age category is not purely nominal but ordinal, we do not want to lose that information by treating it as a simple categorical variable

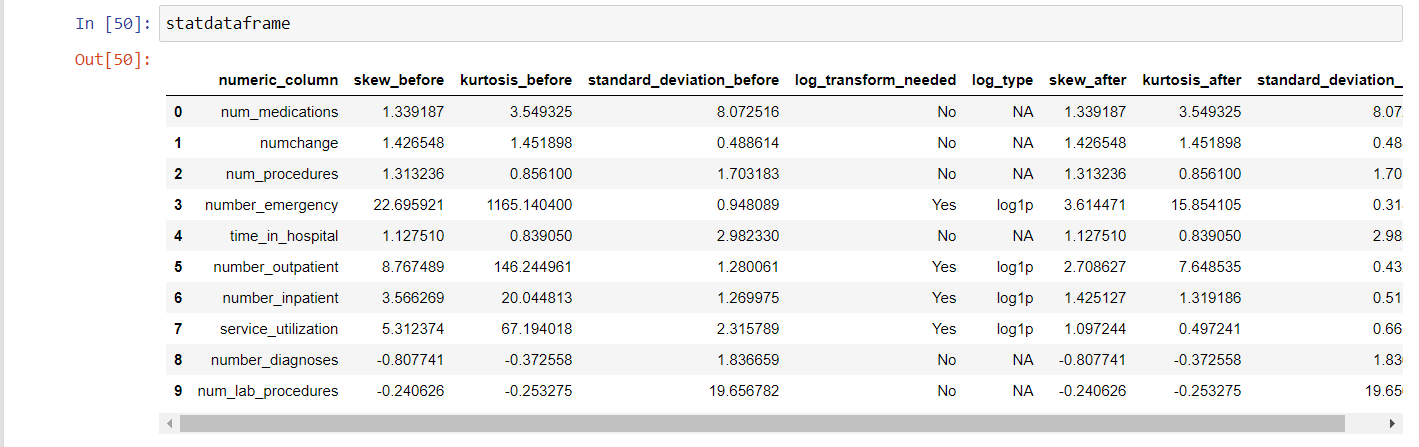




**Number of medication used:** Another possibly related factor could be the total number of medications used by the patient (which may indicate severity of their condition and/or the intensity of care). So we created another feature by counting the medications used during the encounter (keys variable in code below is continued from above):







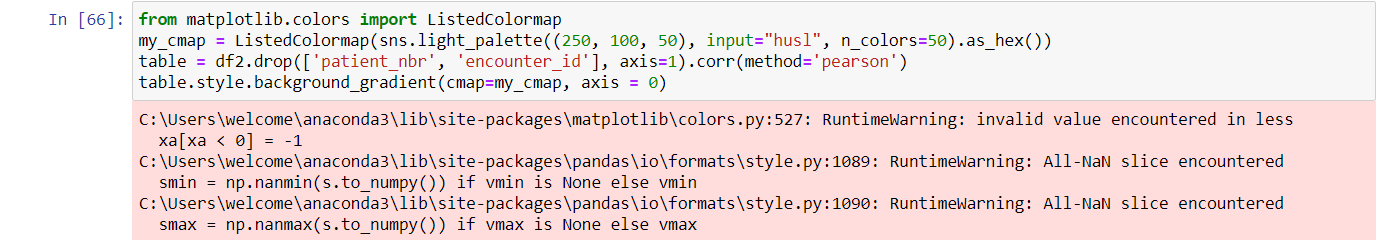


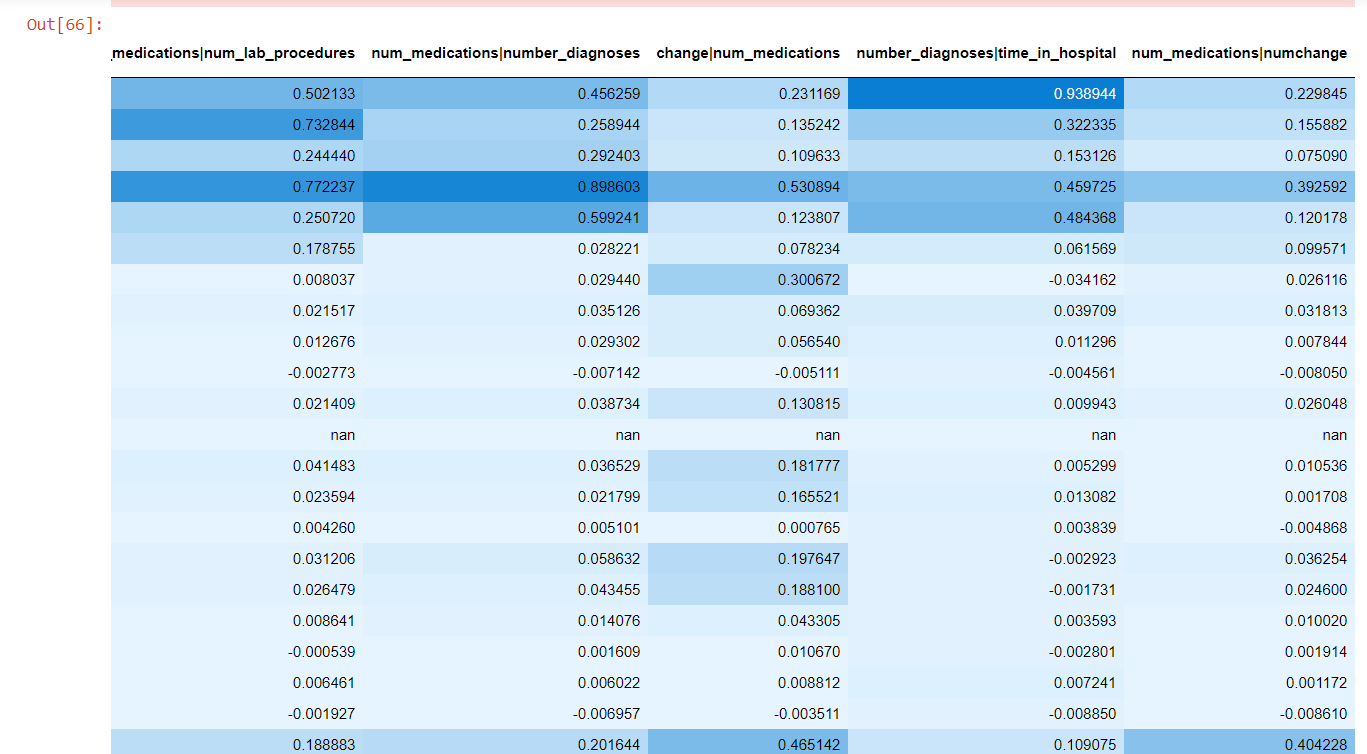




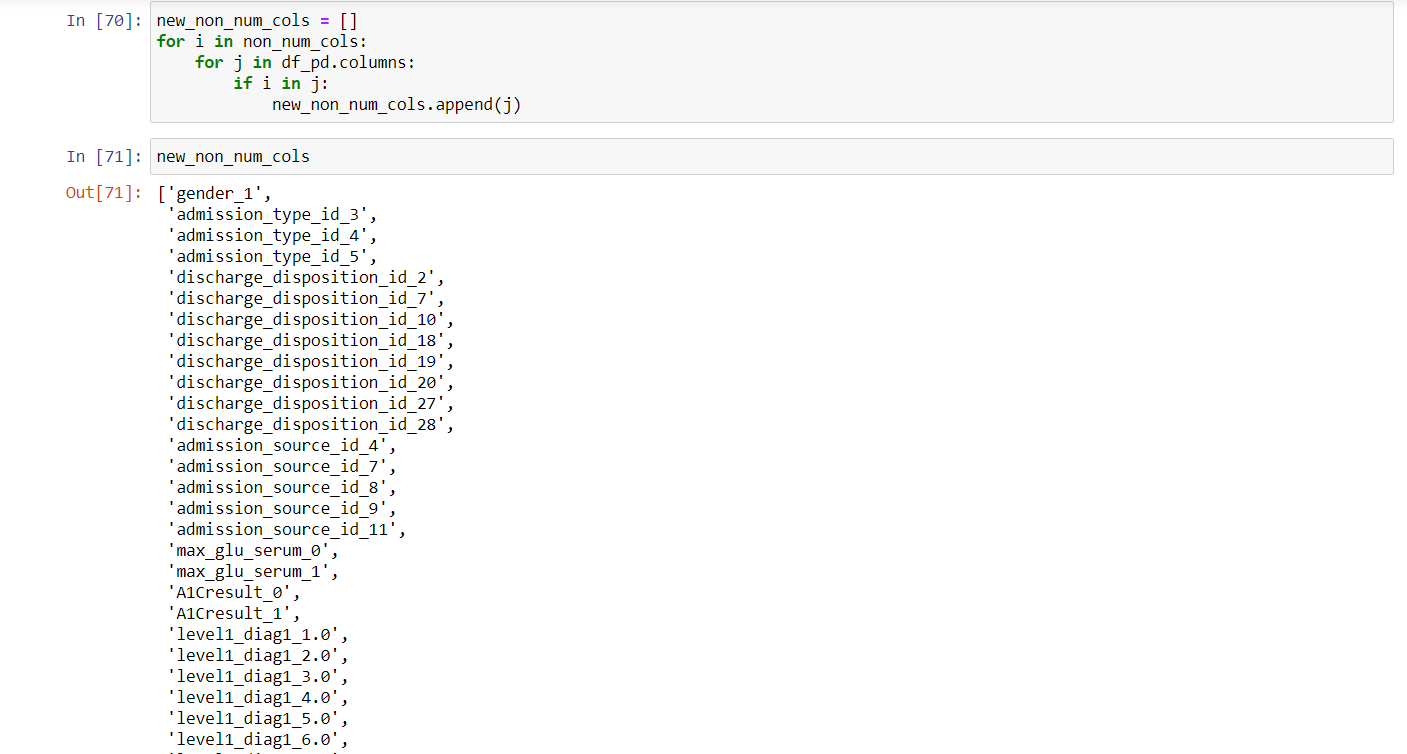
**Feature Scaling:**

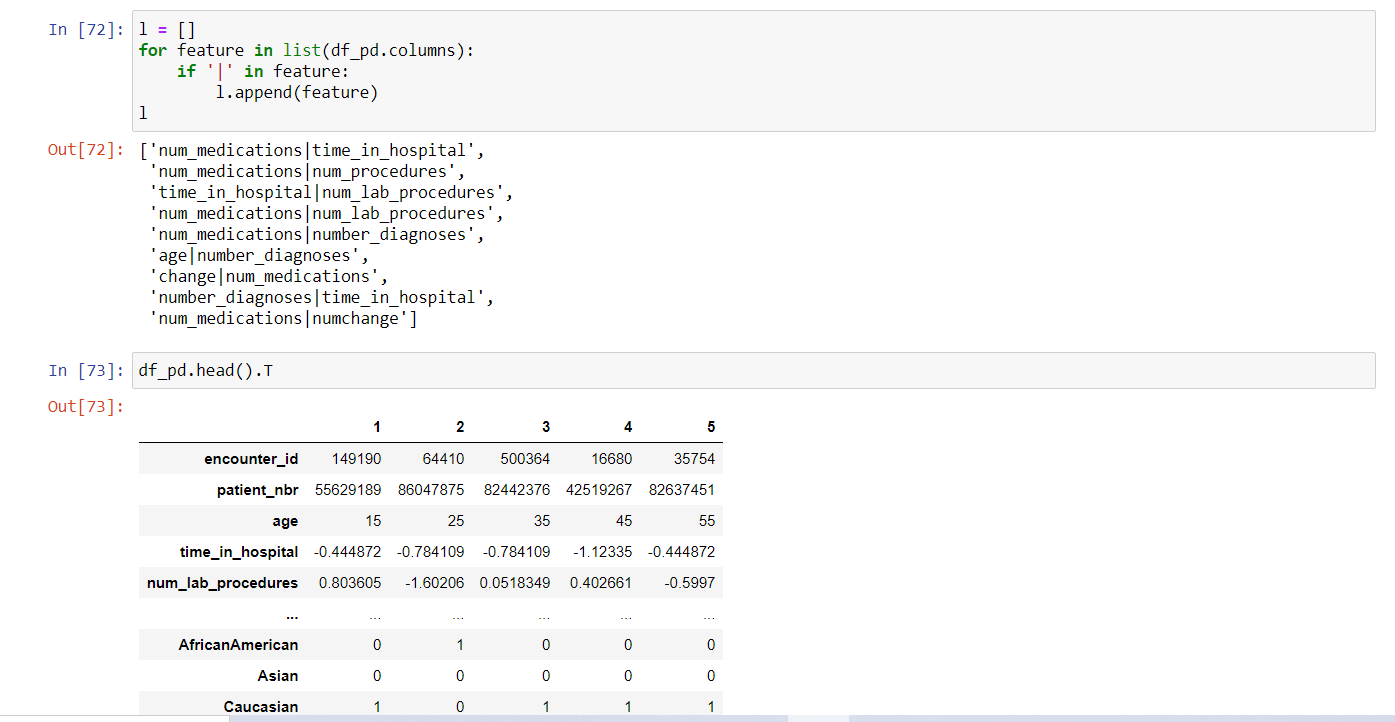










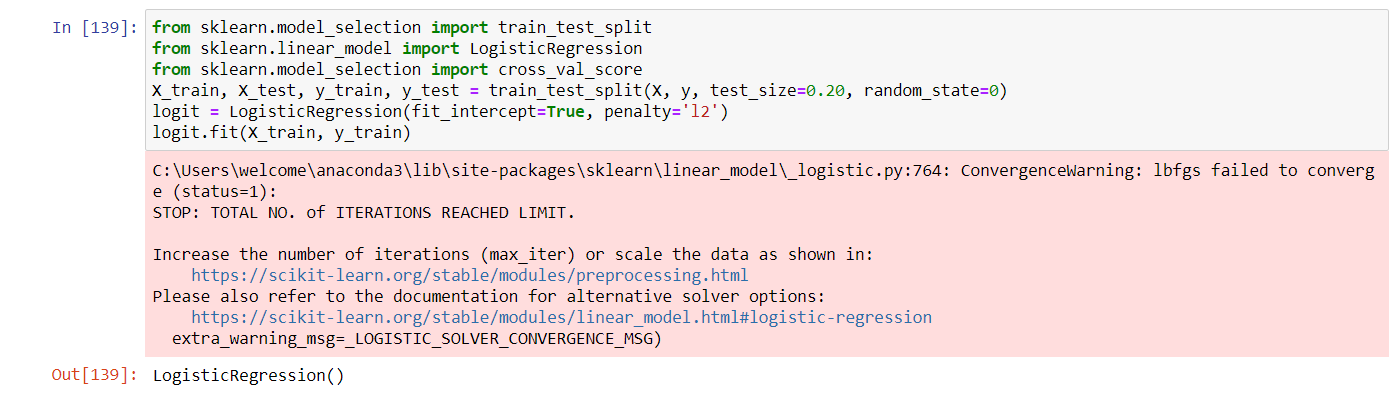


**Model Building:**





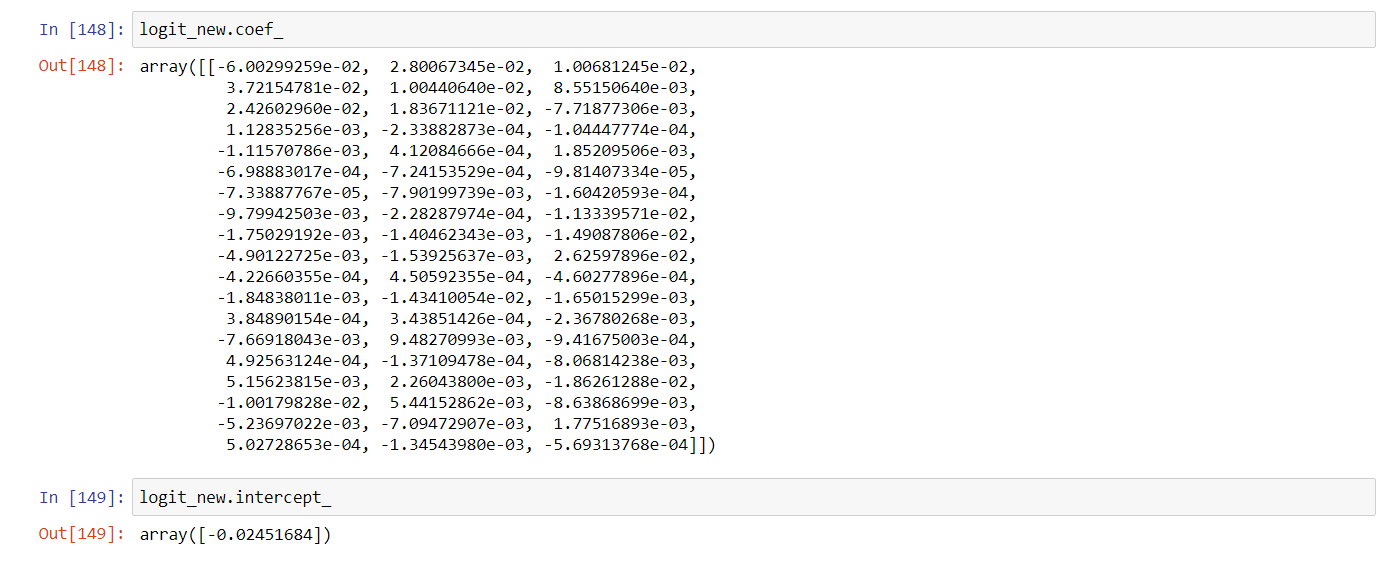
**Splitting Test and Train models:**





**Saving the Model using Pickle:**

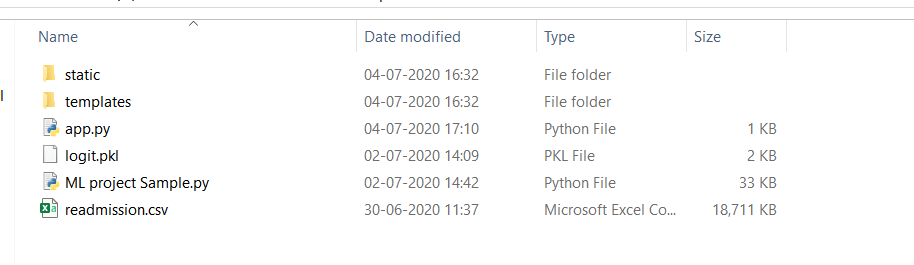




**Predicting  Accuracy:**

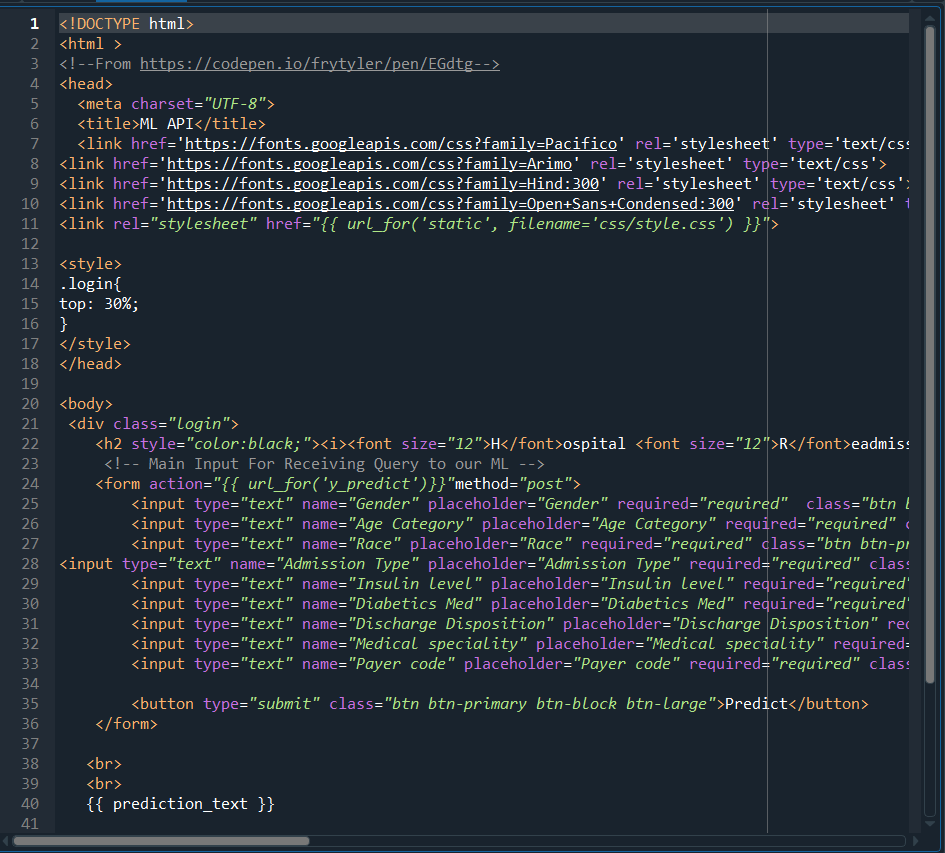


**File Structure:**

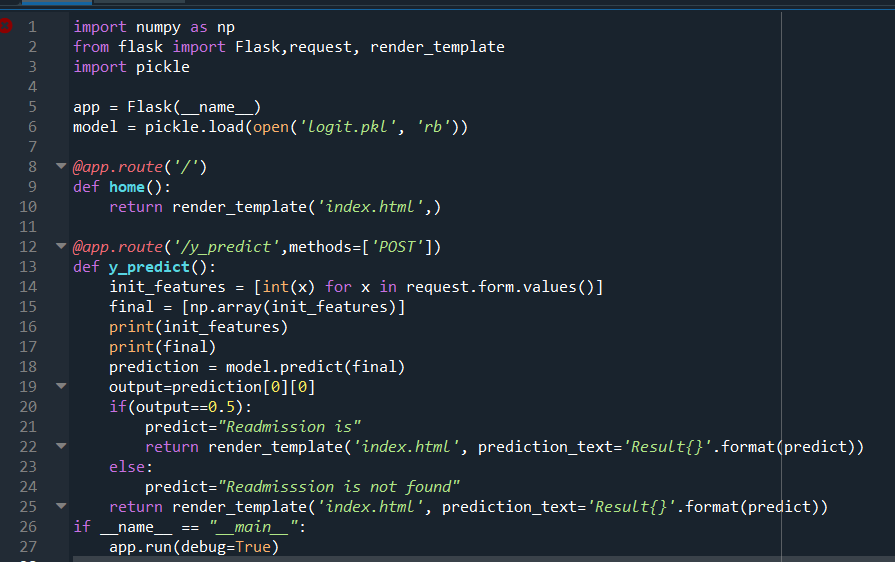


**Building HTML Page:**

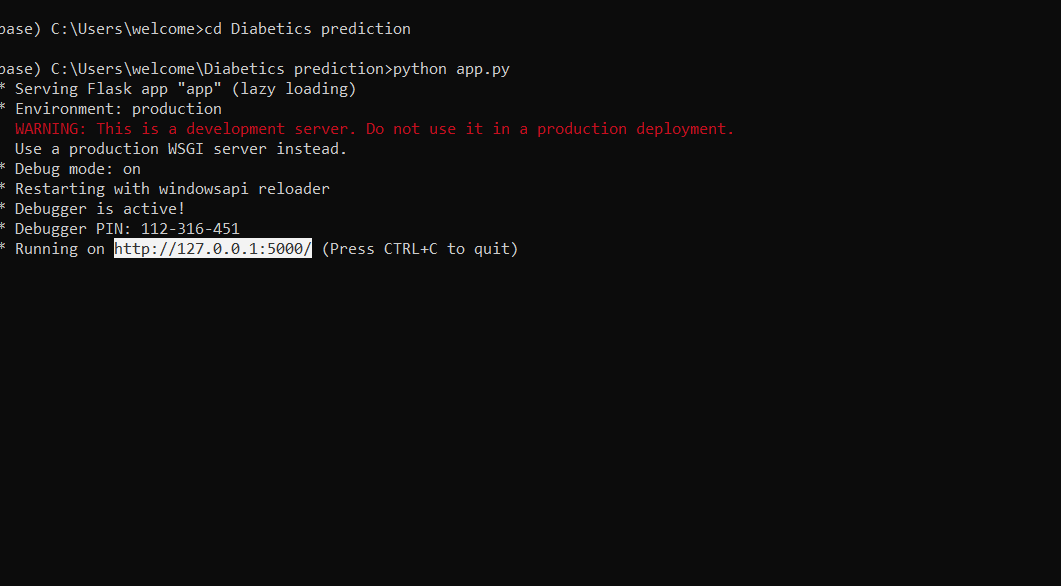
This is the basic HTML page for our project. H1 tag is used to give heading to the project. The user has to enter the details of and the output will be predicted in such a way that the particular patient has a chance of readmission or not.



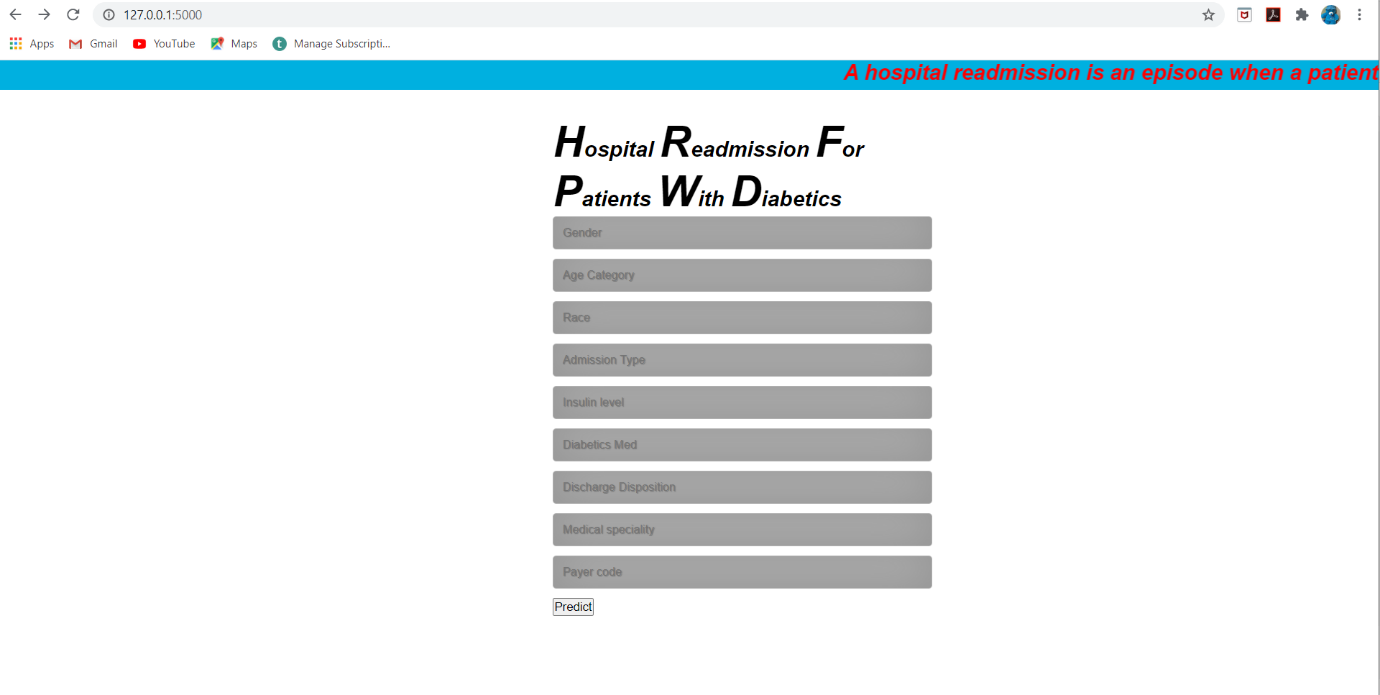
**Python Code:**



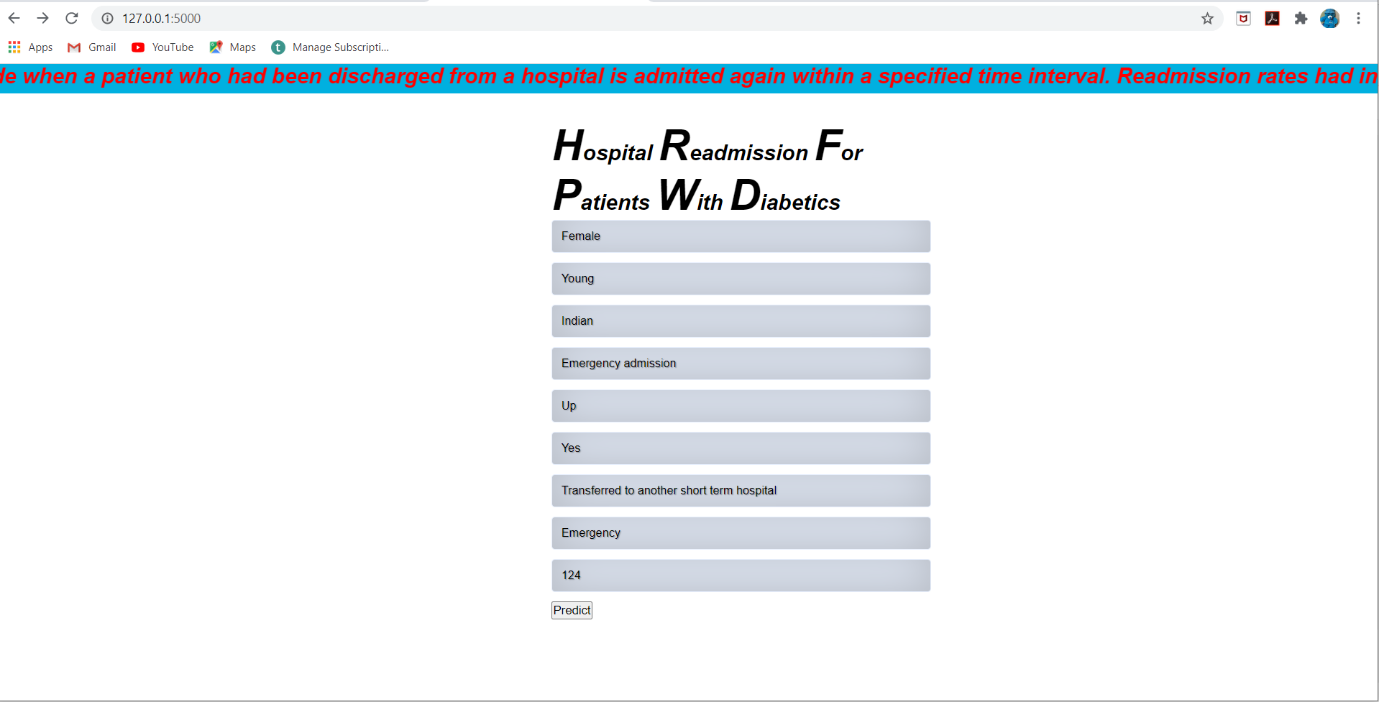
**Output:**



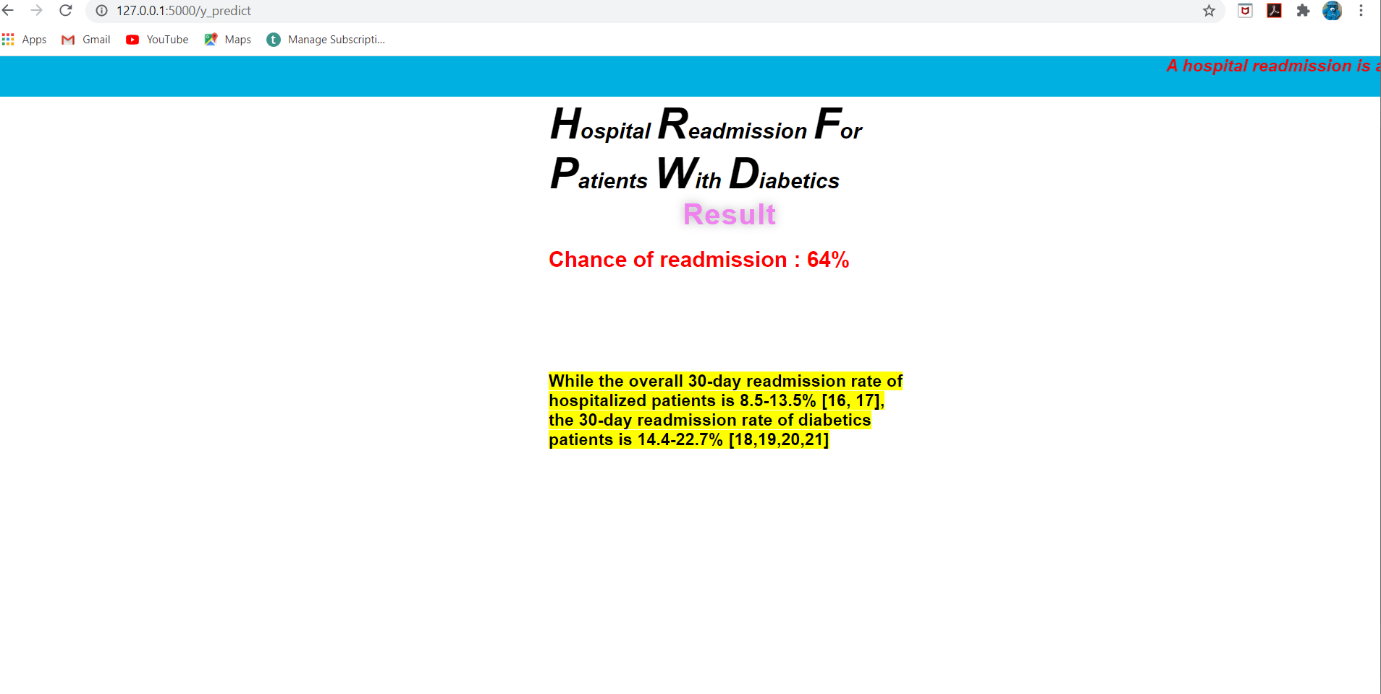
**Web Page:**



**After filling required details:**



**Output Page:**



**Result:**

Machine learning and artificial intelligence are powerful tools with the ability to improve understanding of predictive metrics in diabetics. We successfully trained and tested our model. This model was able to predict 0.91% of readmission rates accurately.

**Advantages:**

➤ We reviewed the recent literature on hospital readmissions and found that most of them are believed to be caused by patient frailty and progression of chronic disease.

➤ However, from 9% to 48% of all readmissions have been judged to be preventable because they were associated with indicators of substandard care during the index hospitalization, such as poor resolution of the main problem, unstable therapy at discharge, and inadequate postdischarge care.

➤ Furthermore, randomized prospective trials have shown that 12% to 75% of all readmissions can be prevented by patient education, predischarge assessment, and domiciliary aftercare. We conclude that most readmissions seem to be caused by unmodifiable causes, and that, pending an agreed-on method to adjust for confounders, global readmission rates are not a useful indicator of quality of care.

➤ However, high readmission rates of patients with defined conditions, such as diabetes and bronchial asthma, may identify quality-of-care problems. A focus on the specific needs of such patients may lead to the creation of more responsive health care systems for the chronically ill.

**Disadvantages:**

➤ When determining penalties, readmission that are unrelated to the initial hospitalization are included under the program. This means hospitals can be penalized for a readmission that is in no way connected to the hospital's patient care or transition planning.

➤ The time periods used to calculate excess readmissions are dated. The program bases hospital penalties on their performance during a three-year time period, meaning hospitals can incur penalties despite having achieved improvements in the 18 months prior to the penalty.

➤ Key sociodemographic factors that are outside of the hospital's control — such as race, ethnicity, education, income and payer — are not included in the program's risk adjustment, which could result in penalties that disproportionately affect hospitals that provide care to patients of low socioeconomic status.

**Conclusion:**

Reducing readmissions has become a mandate for hospitals across the United States, hastened by implementation of public reporting and financial penalties for excess readmissions. However, despite an immediate need to improve the quality of discharge planning and transitional care, there is limited high-quality evidence outlining how to best accomplish this. The scientific literature shows that individual interventions are unlikely to significantly reduce readmission rates. In contrast, some multifaceted interventions have been successful in this regard. Effective interventions share certain features: having multiple components that span both inpatient and outpatient settings and delivery by dedicated transitional care personnel. New evidence suggests that the number of components in a care transitions intervention is significantly related to its effectiveness, which strengthens the argument for more robust interventions.

**Future Scope:**

**Hospital readmission** is an important contributor to total medical expenditure and is an emerging indicator of quality of care. It is disruptive to patients and costly to healthcare systems. ... Out of these, 9,381 records were diabetic patient encounters. It holds approximately 7,100 patients diagnosed with diabetes.

**References:**

1. www.kaggle.com
2. www.geeksforgeeks.com
3. [www.github.com](http://www.github.com)
4. towardsdatascience.com