Fall 2020

Using Machine Learning to Predict Readmissions of Diabetes Patients

Kishor Kumar Sridhar

Using Machine Learning to Predict Readmissions of Diabetes Patients

by

Kishor Kumar Sridhar

A Creative Component submitted to the graduate faculty in fulfillment
of the requirements for the degree of
MASTER OF SCIENCE

Major: Information Systems
Minor: Statistics

Program of Study Committee:
Major Professor: Dr. Anthony M Townsend
Minor Professor: Dr. Kris M De Brabanter

Ivy College of Business
Iowa State University
Ames, Iowa
2020

TABLE OF CONTENTS

LIST OF FIGURES & TABLES

ACKNOWLEDGEMENT

Foremost, I would like to express my sincere gratitude to my Major Professor Dr. Anthony M Townsend for his continued guidance, immense knowledge, and support throughout my graduate studies and related research for the Creative Component.

My sincere thanks to my Minor Professor Dr. Kris M De Brabanter for his motivation, guidance, and insightful comments.

I feel extremely grateful to my grandfather and my parents for their love, affection, care, and sacrifices for making me the person I am today. I would like to extend my thanks to my friends, teachers, and peers for the timely help and encouragement.

ABSTRACT

We live in an era where machine learning and data science play a pivotal role in almost all of the fields. Healthcare is one such field where the implementation of cutting-edge machine learning tools are used to predict, prevent, and cure diseases in a timely manner. Readmission of patients after their discharge from a medical facility has a significant impact on the cost and patient health. In this scenario, this project ventures out to utilize the historic data of diabetes patients to predict their re-admission based on a variety of diagnostic tests performed over the course of the time that the patient is in the hospital. The methodology is to employ machine learning classification algorithms such as Logistic Regression, Decision Trees, Random Forest, K-Nearest Neighbors (KNN), Linear Discriminant Analysis, and Stochastic Gradient Descent to classify a patient as to whether he/she would be readmitted or not. This project uses Recursive Feature Elimination technique to figure out the most important features that can be used as predictors to predict the readmission of patients. This information could be utilized on new patients such that based on the few diagnostic test results performed on the patient while he/she is treated in the hospital, we would be able to get a clearer picture of the patient concerning re-admissions. The model evaluation metrics that were used are Training Accuracy, Testing Accuracy, Precision, Recall, F-1 score, and Confusion Matrix.

1. INTRODUCTION:

In the mid-1980s, the hospital 30-day readmission rates were greater than 20% [11]. The identification of high-risk patients who are likely to be readmitted can provide significant benefits for both patients and medical providers. Unplanned readmission of a hospitalized patient is an indicator of patients' exposure to risk and avoidable waste of medical resources [1]. In order to overcome this, the use of novel technologies that predict the readmission of patients becomes imperative [2]. The goal of this project is to use Machine Learning to predict the readmission of diabetes patients. In this project, the data that is used comes from the Health Facts database, a national data warehouse that collects comprehensive clinical records across hospitals throughout the United States.[8]. The database consists of medical records of patients who have been admitted to the hospitals that include encounter data (emergency, outpatient, and inpatient), the tests that have been performed on the patients during their time of stay at the hospital, and the associated electronic medical records. Our dataset particularly deals with the data collected regarding the diabetes patients and the tests performed on them with an indicator variable that tracks whether the patient was re-admitted or not. In this project, we predict whether the readmission of patients using Machine Learning classification algorithms such as Logistic Regression, Decision Trees, Random Forest, K-Nearest Neighbors, and Stochastic Gradient Descent. The approach is to create a classification model and perform feature selection to reduce the number of variables required to predict the readmissions. I used Recursive Feature Elimination (RFE) with Linear Discriminant Analysis (LDA) to obtain the most important features.

The research questions I tried to answer are:

1. How best can we predict the readmission of diabetes patients?
2. What are the most important features responsible for readmissions?

The model evaluation metrics that were used for this project are Training Accuracy, Testing Accuracy, Precision, Recall, F-1 score, and Confusion Matrix.

2. LITERATURE REVIEW:

Hospitals are more than just centers for the treatment of diseases. The readmission of patients has a significant impact on the cost incurred by both patients and the medical facilities. A study suggests that two-thirds of patients who reported that they had good discharge experiences were still readmitted, one-third of patients discharged had a post-discharge doctor appointment scheduled; half of the patients were readmitted before that scheduled appointment [10]. Poor follow-ups with patients after their discharge is considered to be one of the major reasons for the high rates of readmission of patients [12]. However, this is subjective and depends on other demographic and socio-economic factors such as age – elder people are at a higher risk of readmissions as compared to the people of a younger age [13], gender - females have higher readmission rates than males [13], the income status of the patient – the higher-income population has lower hospital readmissions [15]. This project tries to find some of the most important features that could better predictors of the readmissions.

The readmission rates in general are important for the financial performance of the hospitals. On studying the readmission rates for acute myocardial infarction (AMI), pneumonia (PN), and heart failure (HF) against operating revenues per patient, operating expenses per patient, and operating margin, it is seen that readmissions could be reduced by increasing the operating revenues and expenses by rightly managing the costly treatment procedures. [14]. But, in the diabetes space, instead of concentrating on improving the operating revenues, the focus is on specialty care, better discharge instructions, coordination of care, and post-discharge support [20]. The study suggests that these kinds of post-discharge interventions and regular follow-ups are essential in reducing the readmission of patients.

In this scenario, in order to better follow up with the patients after discharge, an effective strategy that identifies which patients are more likely to be readmitted so that the hospitals can utilize their resources to provide better post-discharge aid to those patients and hence reduce the readmission rates. This is where Machine Learning comes into play and various Artificial Intelligence techniques are used to analyze various aspects of diabetes such as glycemic control,

prediction of glycemic events, diagnosis of complications [23]. More specifically, in the scenario of analyzing the readmission of patients, by utilizing the ML classification algorithms such as Linear Discriminant Analysis, Random Forest, k–Nearest Neighbor, and so on [23] are used to predict the readmission rates of patients based on historic data.

3. METHODOLOGY:

3.1. Data:

The data used in this project was downloaded from the UCI Machine Learning repository [4]. The data was collected under a voluntary program called Health Facts which was intended to maintain a database of the organizations that use the Cerner Electronic Health Record System [3]. The database encompasses a plethora of information related to the patients attending the participating hospitals (emergency, outpatient, and inpatient). The information that was recorded includes the patient's unique identification number, demographics such as age, sex, and race, diagnoses, time spent in the hospital and in-hospital lab procedures and test results, etc. Our dataset primarily deals with the 100,766 diabetes patients containing the information related to the diagnostic lab procedures and test results. The data has 50 features with the response feature "Readmitted" column indicating whether the patient was re-admitted within 30 days, after 30 days, or did not get re-admitted at all. The different features in the dataset are tabulated as follows characteristics.

3.1.1. Features of the dataset:

S.No.	Feature	Data Type	Description
1	Encounter ID	Integer	Unique identifier for an encounter
2	**Patient number**	Integer	Unique identifier of a patient
3	**Race**	**String**	Values: Caucasian, Asian, African American, Hispanic, and other
4	**Gender**	**String**	Values: male, female, and unknown/invalid
5	**Age**	**Integer**	Grouped in 10-year intervals: [0, 10], [10, 20], …, [90, 100]
6	**Weight**	**Integer**	Weight in pounds
7	**Admission type**	**Integer**	Integer corresponding to 9 distinct values
8	**Discharge disposition**	**Integer**	Integer identifier corresponding to 29 distinct values
9	**Admission source**	**Integer**	Integer identifier corresponding to 21 distinct values
10	**Time in hospital**	**Integer**	Integer number of days between admission and discharge
11	**Payer code**	**String**	Integer identifier corresponding to 23 distinct values.
12	**Medical specialty**	**String**	Integer identifier of a specialty of the admitting physician, corresponding to 84 distinct values, for example, cardiology, internal medicine, family/general practice, and surgeon
13	**Number of lab procedures**	**Integer**	Number of lab tests performed
14	**Number of procedures**	**Integer**	Numeric Number of procedures (excluding lab tests) performed during the encounter
15	**Number of medications**	**Integer**	Number of distinct generic medications administered during the encounter
16	**Number of outpatient visits**	**Integer**	Total Number of outpatient visits of the patient in the year preceding the encounter
17	**Number of emergency visits**	**Integer**	Total Number of emergency visits of the patient in the year preceding the encounter
18	**Number of inpatient visits**	**Integer**	Total Number of inpatient visits of the patient in the year preceding the encounter
19	**Diagnosis 1**	**String**	The primary diagnosis (coded as first three digits of ICD9)
20	**Diagnosis 2**	**String**	Secondary diagnosis (coded as first three digits of ICD9
21	**Diagnosis 3**	**String**	Additional secondary diagnosis (coded as first three digits of ICD9)
22	**Number of diagnoses**	**Integer**	Total Number of diagnoses entered into the system
23	**Glucose serum test result**	**String**	Indicates the range of the result or if the test was not taken. Values: ">200," ">300," "normal," and "none" if not measured
24	**A1c test result**	**String**	Indicates the range of the result or if the test was not taken.
25	**Change of medications**	**String**	Indicates if there was a change in diabetic medications Values: "change" and "no change"
26	**Diabetes medications**	**String**	Indicates if there was any diabetic medication prescribed. Values: "yes" and "no"
27	24 features for medications For the generic names:	String	**metformin, repaglinide, nateglinide, chlorpropamide, glimepiride, acetohexamide, glipizide, glyburide, tolbutamide, pioglitazone, rosiglitazone, acarbose, miglitol, troglitazone, tolazamide, examide, sitagliptin, insulin, glyburide-metformin, glipizide-metformin, glimepiride- pioglitazone, metformin-rosiglitazone, and metformin- pioglitazone.** Values: "up" if the dosage was increased during the encounter, "down" if the dosage was decreased, "steady" if the dosage did not change, and "no" if the drug was not prescribed
28	**Readmitted**	**String**	Days to inpatient readmission. Values: "<30" if the patient was readmitted in less than 30 days, ">30" if the patient was readmitted in more than 30 days, and "No" for no record of readmission

Table 1. Features of dataset [3]

3.2. Data Cleaning and pre-processing:

As with any real-world dataset, this data also has messy data that needs to be cleaned and curated in order to be used for our analysis. The following are the data cleaning and feature engineering steps that was performed on the dataset

Data preprocessing:

- Eliminating duplicates
- Cleaning garbage values
- Dropping unnecessary columns

Feature Engineering:

- Label encoding the categorical variables
- Combining values of columns
- Merging multiple columns

3.2.1. Eliminating duplicates

The original dataset contains information for some patients who made multiple visits to the hospitals and hence had multiple records in the data. These records could not be considered as statistically independent which is one of the assumptions of a logistic regression model. Hence, we remove the duplicate observations based on the unique patient ID and keep only the records pertaining to the patient's first admission into the hospital. Upon examination, there were only 71,518 records that had a unique patient ID out of the 101,766 records. We keep only these 71,518 records for our further analysis.

3.2.2. Cleaning garbage values

I performed some preliminary data exploration to find out the garbage values that were present in the dataset.

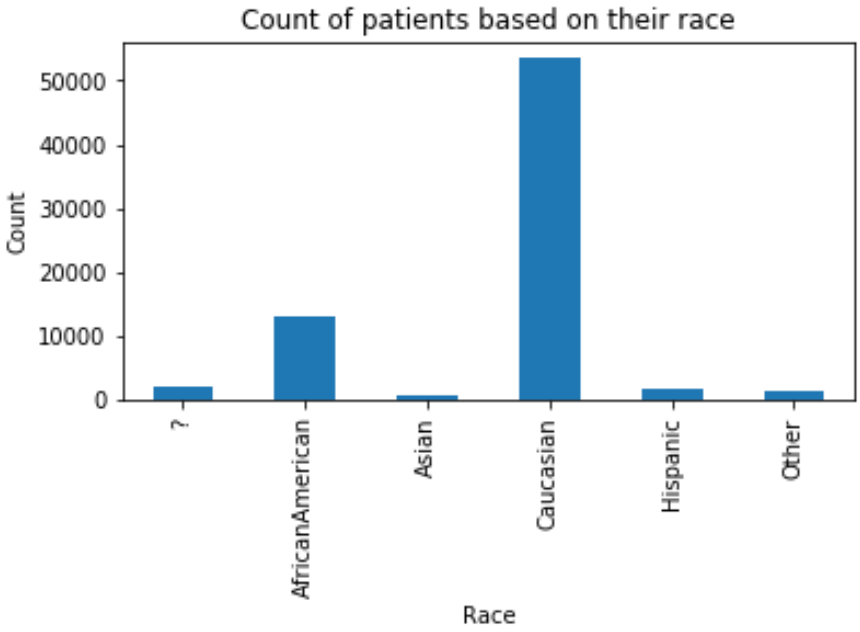

Figure 1. Bar chart showing the distribution of patients based on race

```
In [10]: # count of patients belonging to different races

         dataset.groupby(['race']).count()['patient_nbr']

Out[10]: race
         ?                  1948
         AfricanAmerican   12887
         Asian               497
         Caucasian         53491
         Hispanic           1517
         Other              1178
         Name: patient_nbr, dtype: int64
```

Figure 2. Count of patients based on Race

Based on the above counts, it is apparent that there are about 1,948 values in the "Race" column. Since this could potentially mean that the people did not fill out that particular column or it could be the case that this could be a data entry error. So, instead of getting rid of these records, I converted "?" into "Other".

The diag_1, diag_2, and diag_3 columns denote the ICD-9-CM codes [5] of the diagnostic tests performed on the patients. The values of the feature must be numeric or a combination of alphabets and numbers. However, some values had symbols such as "?" denoting garbage values. I removed these values from my analysis since there is no way to conclusively know the actual correct values for these patient records.

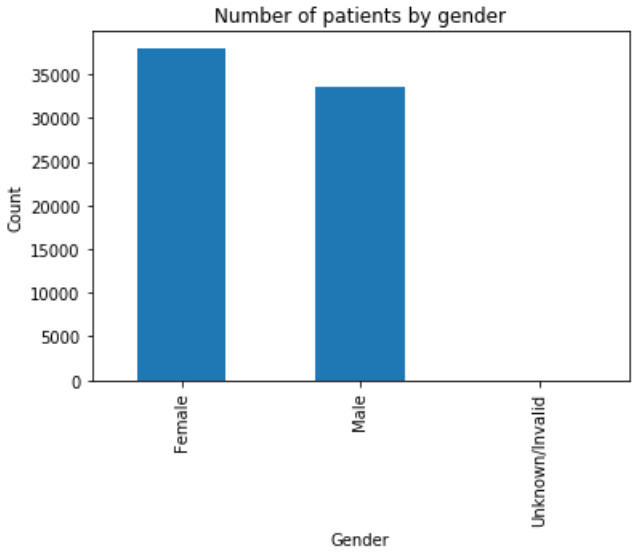

Figure 3. Bar chart showing the distribution of patients based on gender

```
# count of different categories in the 'gender' columns

dataset.groupby('gender').count()['patient_nbr']
```

```
gender
Female             38025
Male               33490
Unknown/Invalid        3
Name: patient_nbr, dtype: int64
```

Figure 4. Count of patients based on gender

The above chart shows that there 3 values that are Unknown/Invalid values. I have removed these records from my analysis.

The discharge disposition ID column tracks the information of the patients who were discharged from the hospital. Some of the patients were expired and hence need not be included in our analysis and I eliminated them.

3.2.3. Dropping unnecessary columns

```
# viewing the count of values for different weights

dataset.groupby(['weight']).count()['patient_nbr']
```

```
weight
>200            3
?           68662
[0-25)         46
[100-125)     566
[125-150)     131
[150-175)      33
[175-200)       9
[25-50)        89
[50-75)       781
[75-100)     1195
Name: patient_nbr, dtype: int64
```

Figure 5. Count of patients based on weight

From the above summary statistics, it can be seen that about 68,662 records do not have any “Weight” values for them. This could be attributed to the fact that before the HITECH legislation of the American Reinvestment and Recovery Act in 2009 hospitals and clinics were not required to capture it in a structured format.[6] Hence, I removed this column from my analysis. The dataset has 50 columns out of which some columns are not required for our analysis such as Encounter ID, Patient number, Payer code, Medical specialty. These columns arbitrary and do not hold much significance to the readmission of patients and hence can be removed from our analysis. Moreover, upon examination, the columns “citoglipton” and “examide" contain the same values and hence can be removed from our analysis.

3.2.4. Label encoding categorical variables:

A1c test result:

The A1C test is a blood test that provides information about average levels of blood glucose over the past 3 months. The A1C test can be used to diagnose type 2 diabetes and prediabetes [7]. It is represented as a percentage of red blood cells that have sugar-coated hemoglobin. The normal is below 5.7%, prediabetes is 5.7% to 6.4% and diabetes is 6.5% or above. I coded values above 7% as "1" as in diabetic, less than 7% as "0" as in not-diabetic, and if the test was not administered, I coded it as "-99"

Glucose serum test result:

This test measures the blood sugar level in mg/dL. 200 mg/dL or above is considered diabetic [9]. I coded values above 200 mg/dL as "1" as in diabetic, less than 200 mg/dL as "0" as in not-diabetic, and if the test was not administered, I coded it as "-99"

There were 21 features that indicate whether a drug was administered and whether or not there was a change in the prescription over the course of the time the patient was in the hospital. Values: "up" if the dosage was increased during the encounter, "down" if the dosage was decreased, "steady" if the dosage did not change, and "no" if the drug was not prescribed. I have coded them as

"No" = -99 (If the drug was not prescribed)

"Steady" = 0 (If there was no change in the drug prescription through the time)

"Up" = 1 (If the drug prescription increased)

"Down" = -1 (If the drug prescription decreased)

Finally, the target variable which is "readmitted" denotes whether the person got readmitted within 30 days or after 30 days or did not get readmitted. For simplification purposes, I combined the readmitted within 30 days or after 30 days values in one "Yes" for readmitted and "No" for not re-admitted.

I used a label-encoder to encode the categorical variables race, age, diagnosis 1, diagnosis 2, and diagnosis 3.

3.2.5. Merging columns:

The number of lab procedures and other required procedures that were performed on the patient were denoted in two different columns. I combined them into a single column by the mere addition of the values as "Total procedures"

The number of inpatient visits, outpatient visits, and emergency visit a patient made to any hospital in the past year is denoted in 3 different columns. I merged them all into a single column called "Total Visits"

I then converted the “Total procedures” and “Total Visits” into quintiles and encoded them to reduce the number of distinct values in the columns and to improve my analysis.

3.3. Data exploration and visualization:

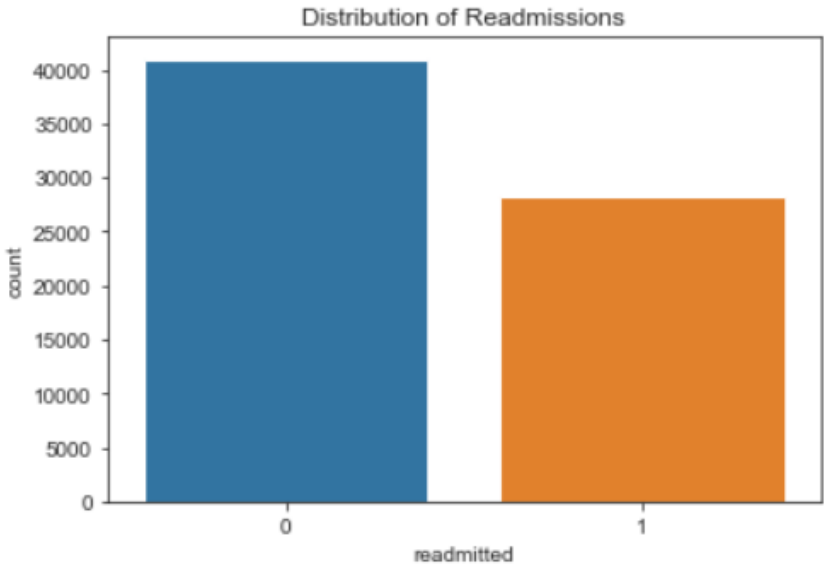

Figure 6. Distribution of Readmissions

The above bar chart shows the distribution of readmission of patients. There are 28189 patients who for readmitted and 42041 people who did not get readmitted.

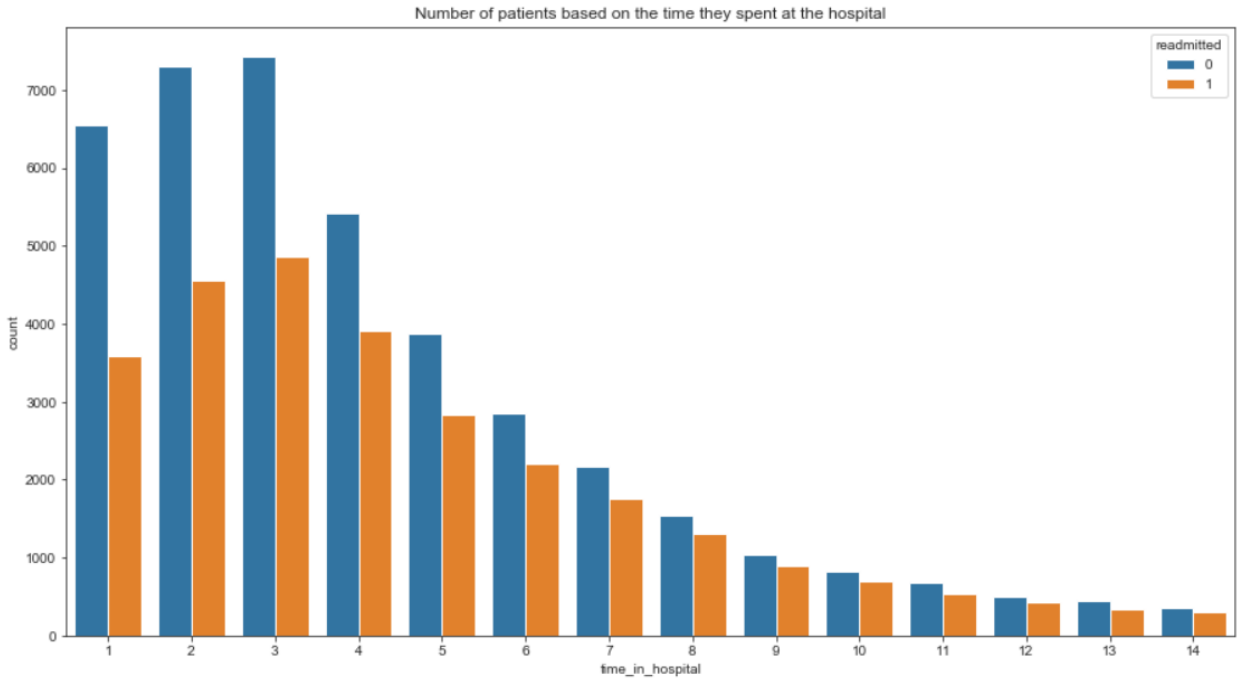

Figure 7. Distribution of patients based on time spent in the hospital

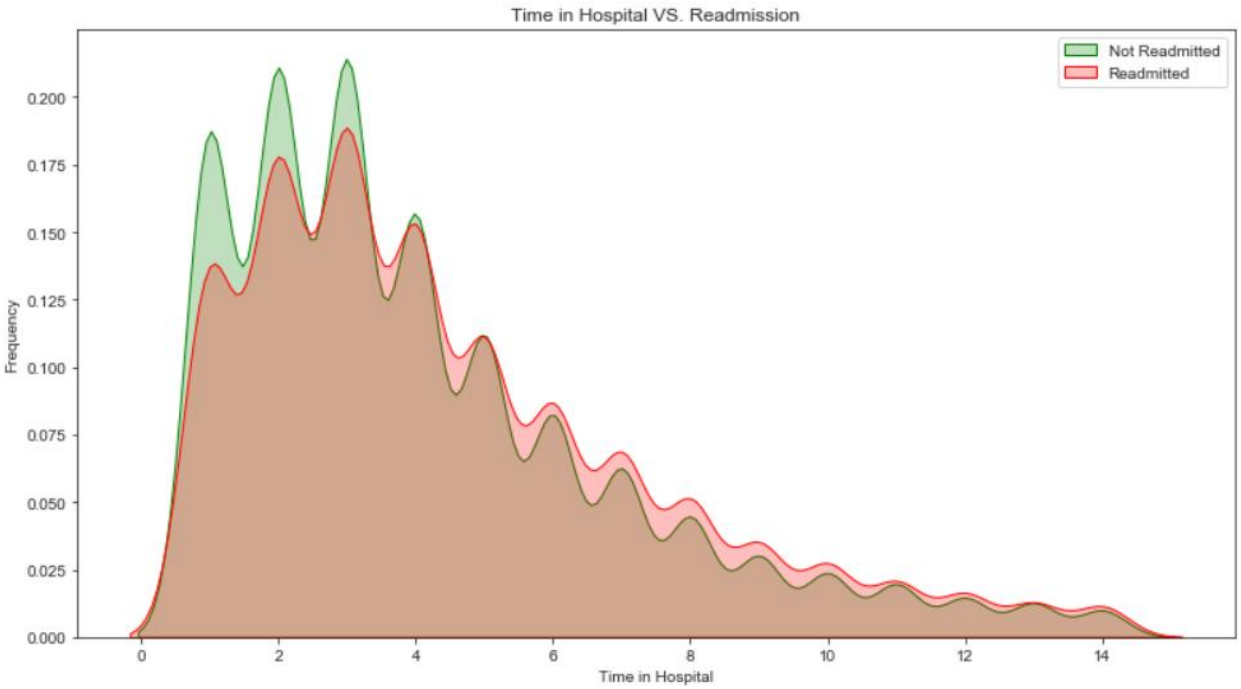

Figure 8. Kernel density plot of patients based on time spent in the hospital

The above bar chart shows the time that a patient spent in the hospital and the rates of readmission of the patient. The length of stay is one of the major predictors when it comes to the readmission of patients [16]. This chart above shows that there is a clear trend of higher readmission rates with the people who spent lesser time at a medical facility as compared to people who stayed longer. This is especially apparent with the people who stay in hospitals for less than 4 days. This could be because the people who stay longer in the hospitals to complete the course of the medication and required procedures before getting discharged and hence have a lesser risk of readmissions. However, this shortened period of stay could also be attributed to the hospital costs involved that may restrict people to be admitted to the hospital for the entire duration of their treatment.

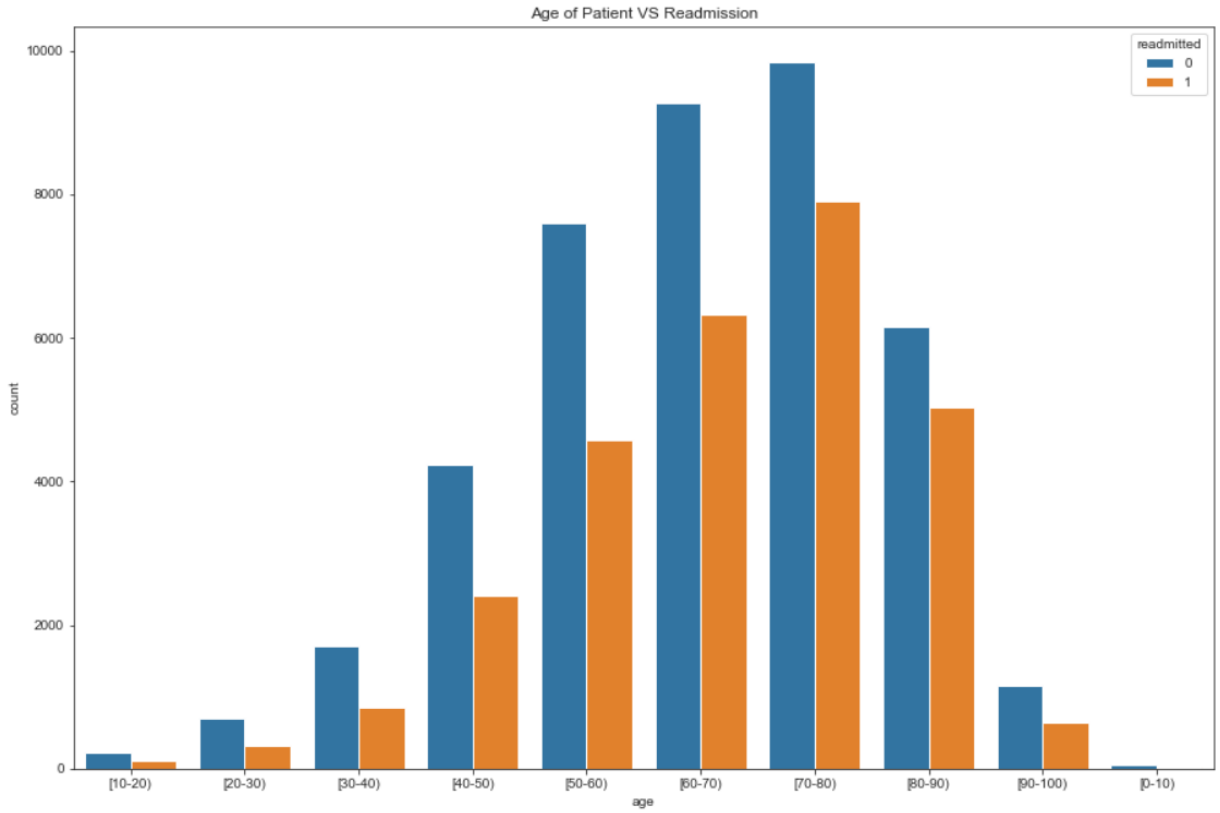

Figure 9. Distribution of patients based on age

The elderly people generally run a higher chance of readmissions. The bar chart shows that patients in the age group of 50 years to 90 years of age have the highest remissions.

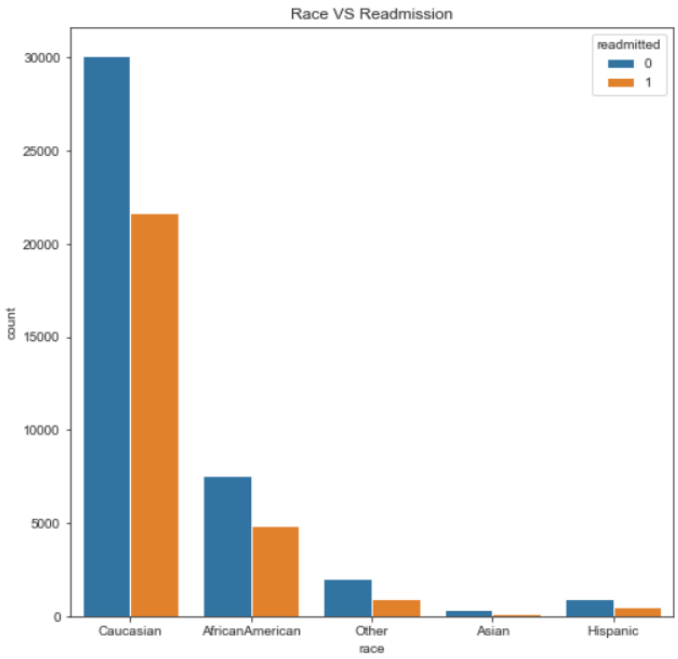

Figure 10. Distribution of patients based on race and readmission

The above bar chart shows that Caucasians are at the highest risk of readmissions as compared to the Hispanic, Asian, and African American populations.

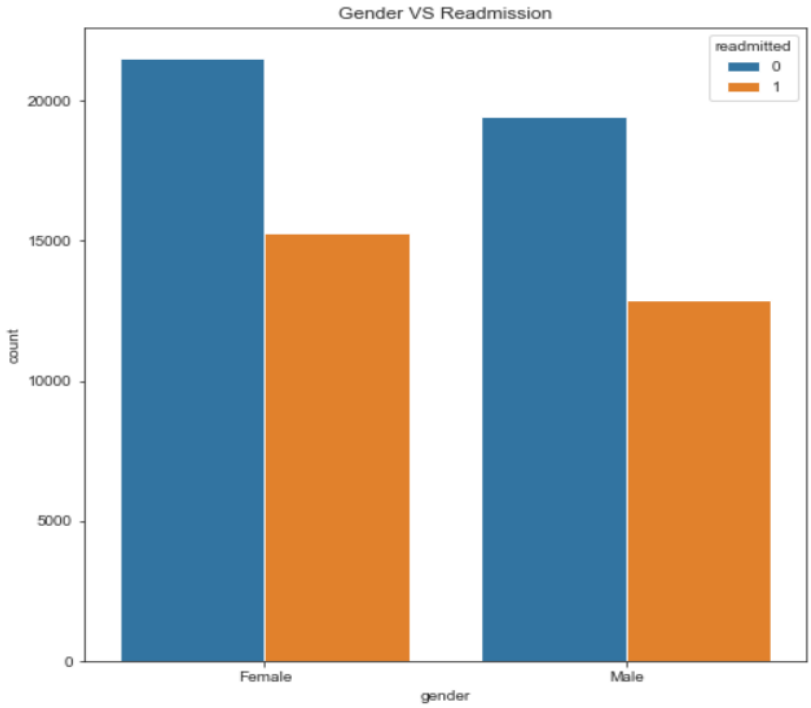

Figure 11. Distribution of patients based on gender and readmission

The study suggests that females have a higher risk of readmissions. Our bar chart above shows that the readmission of females is much higher as compared to the readmission of male patients. This is particularly true for the age groups of females and males between 50 years of old and 65 years old. Women in these age groups have a higher chance of readmission than men.

From our exploratory data analysis, we can notice that elderly people, females, and Caucasians have a higher risk of readmission.

4. ANALYSIS AND RESULTS:

4.1. Splitting Training and Testing Sets:

Before running the base model, I converted the dataset into a training and testing dataset in the ratio of 80% and 20% respectively using the train_test_split function available in the Scikit-Learn library of Python.

4.2. Base Model:

The logistic regression was used as a base model. It is a supervised classification algorithm that uses the logit function to calculate the logarithm of the odds. The logit function that models the probability of a class is

$$\text{logit}(p) = \log(p/1\text{-}p)$$

4.3. Cross-validation (CV):

Cross-Validation is a technique used for evaluating ML models where the models are trained on a subset of the data and are evaluated on the remaining data. Then the average value of the evaluation metrics such as accuracy is considered. Since there is an imbalance in the classes, I used the "RepeatedStratifiedKFold" cross-validation function that is available as a part of the Sci-kit learn library. This cross-validation method makes sure that each fold contains the approximately same percentage of samples of each target class thereby eliminating any inherent bias arising out of class imbalance. I used n_splits = 10 for 10-fold CV and n_repeats = 100 for repeating the CV 100 times

4.4. Evaluation metrics:

For all the algorithms that I implemented in this project, I used the following evaluation metrics:

- Training Accuracy
- Testing Accuracy
- Precision
- Recall
- Confusion Matrix

```
Training Accuracy for Logistic Regression is 0.60
Test Accuracy for Logistic Regression is 0.60
Precision for Logistic Regression is 0.54
Recall for Logistic Regression is 0.19
The confusion Matrix for Logistic Regression is
```

Predict / Actual	0	1	All
0	1372	202	1574
1	1095	195	1290
All	2467	397	2864

Figure 12. Evaluation metrics for the Logistic Regression Model

4.5. Feature selection:

4.5.1. Most important features based on RFE using LDA:

To improve upon the base model, I implemented Recursive Feature Elimination (RFE) technique along with Linear Discriminant Analysis (LDA) to obtain the 10 most important features associated with the readmission of patients. Recursive Feature Elimination (RFE) is a feature selection model that fits a given model and as the name suggests recursively removes the features from the data until it reaches a specific set of important and optimal features that give good accuracy. The RFE also attempts to eliminate the feature dependencies and collinearity if it is present in the model.

```
# feature selection using Recursive Feature Elimination and LDA

from sklearn.feature_selection import RFE

model = LinearDiscriminantAnalysis()
rfe = RFE(model, 10)
fit = rfe.fit(X_train, y_train)
print("Num Features: %d" % fit.n_features_)
print("Selected Features: %s" % fit.support_)
print("Feature Ranking: %s" % fit.ranking_)
```

```
Num Features: 10
Selected Features: [ True  True  True  True False  True False  True False False False  True
 False False False False False False False False False False False False
 False False False False False False False False False False False False
  True  True  True]
Feature Ranking: [ 1  1  1  1  7  1  3  1 25 23 26  1 18 14 13 12 16 27 19  2 24 20  9 22
 17 10  5  8 11 15 21  6 28 29 30  4  1  1  1]
```

```
# The features found using RFE with LDA

X.columns
```

```
Index(['race', 'gender', 'age', 'admission_type_id', 'admission_source_id',
       'num_medications', 'number_diagnoses', 'diabetesMed',
       'total_procedures', 'total_visits'],
      dtype='object')
```

Figure 13. List of important features using RFE with LDA

The list of top 10 features found by RFE using LDA are as follows:

1. Race
2. Gender
3. age
4. Admission type
5. Admission source
6. Number of medications
7. Number of diagnoses
8. Diabetes medications prescribed
9. Total procedures
10. Total visits

Alternatively, I also used the feature_importances_method available as a part of the Random Forest classifier and Decision Trees classifiers in Scikit-learn to figure out the top 10 most important features.

4.5.2. Most important features using Random Forest:

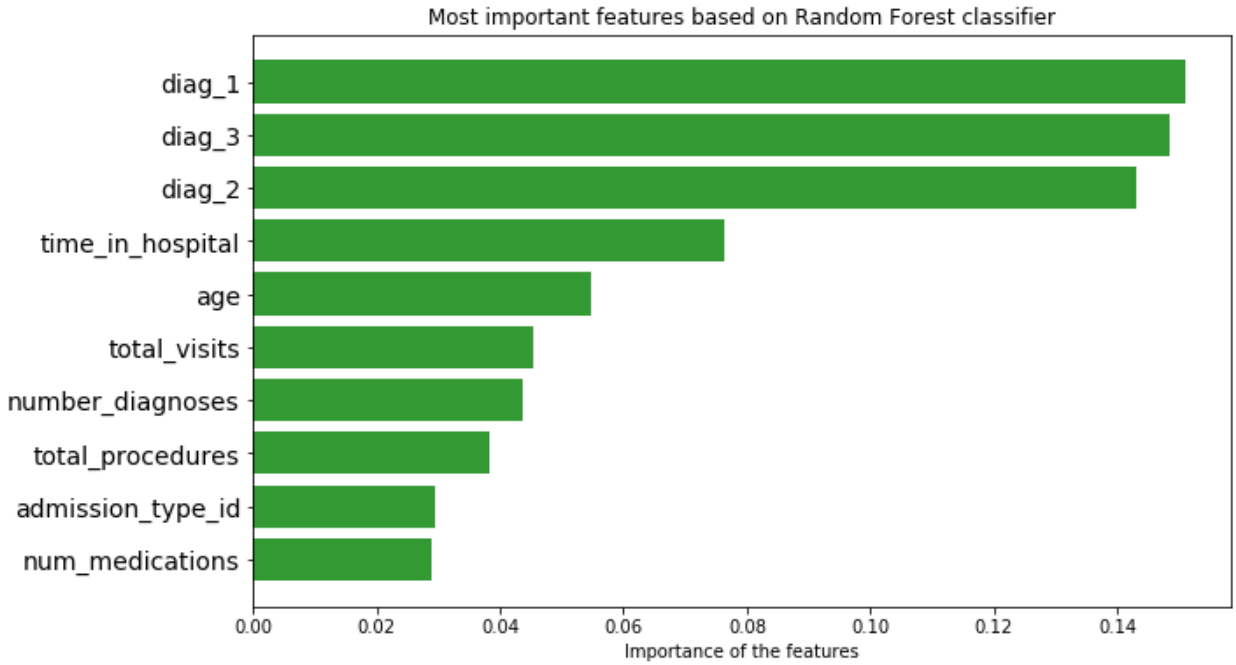

Figure 14. Bar chart showing important features using Random Forest

Feature	Importance
num_medications	0.029024
admission_type_id	0.029548
total_procedures	0.038245
number_diagnoses	0.043609
total_visits	0.045372
age	0.054851
time_in_hospital	0.076342
diag_2	0.143130
diag_3	0.148422
diag_1	0.150845

Figure 15. List of important features using Random Forest with importance score

4.5.3. Most important features using Decision Trees:

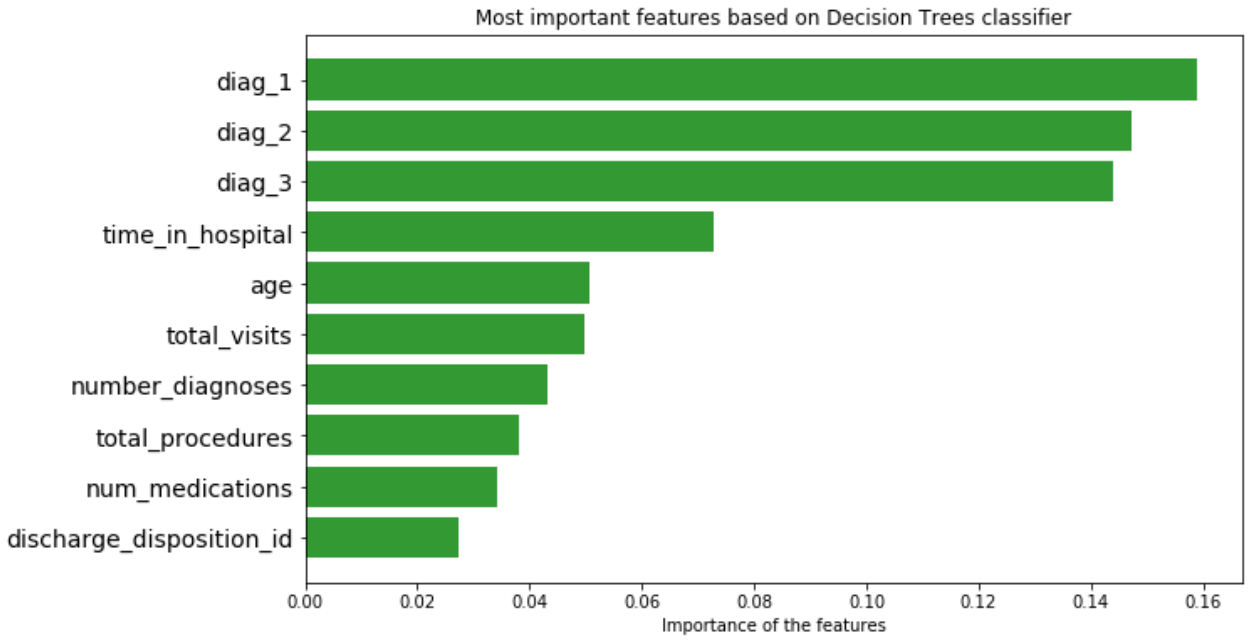

Figure 16. Bar chart showing important features using Decision Trees

Feature	Importance
discharge_disposition_id	0.027400
num_medications	0.034331
total_procedures	0.038008
number_diagnoses	0.043183
total_visits	0.049804
age	0.050735
time_in_hospital	0.072766
diag_3	0.143845
diag_2	0.147346
diag_1	0.158882

Figure 17. List of important features using Decision Trees with importance score

4.6. Model Comparison:

Using the above set of the most important features found by 3 different algorithms, I then implemented the following different classification algorithms:

- Linear Discriminant Analysis classifier
- Decision Trees classifier
- Random Forest classifier

- K-Nearest Neighbors classifier
- Stochastic Gradient Descent classifier

The evaluation metrics for these algorithms are as follows

4.6.1. Linear Discriminant Analysis:

The Linear Discriminant Analysis (LDA) classifier is majorly used as a dimensionality reduction technique. However, it can also be used for classification problems. It works by calculating the 'separability' between the classes known as the between-class variance. The extension of LDA is Quadratic Discriminant Analysis (QDA) in which each class uses its estimate of variance. In our example, we use LDA as a classifier. The evaluation metrics for LDA is given below,

```
Training Accuracy for LDA is 0.61
Test Accuracy for LDA is 0.61
Precision for LDA is 0.57
Recall for LDA is 0.18
The confusion Matrix for LDA is
```

Predict	0	1	All
Actual			
0	1385	189	1574
1	1108	182	1290
All	2493	371	2864

Figure 18. Evaluation metrics for the LDA Model

4.6.2. Decision Trees:

Decision Tree is one of the classification algorithms that work by recursive binary splitting as each node based on a test condition on the feature. In simpler terms, it uses a set of if-else conditions like True or False at each of the nodes and then classifies

according to the conditions. Decision Trees are non-parametric supervised learning methods which means that algorithms do not make strong assumptions about the data. The evaluation metrics for Decision Trees is given below,

```
Training Accuracy for Decision Tree is 0.82
Test Accuracy for Decision Tree is 0.58
Precision for Decision Tree is 0.47
Recall for Decision Tree is 0.43
The confusion Matrix for Decision Tree is
```

Predict	0	1	All
Actual			
0	1014	563	1577
1	743	464	1207
All	1757	1027	2784

Figure 19. Evaluation metrics for the Decision Tree Model

4.6.3. Random Forest:

Random Forest is an extension of the Decision Trees in the sense that it contains a large number of individual Decision Trees that work as an ensemble. Each of the trees in the forest classifies the outcome of the target class variable based on the features and the class with the greatest number of votes becomes the model's prediction. In our case, we have used it as a binary classifier. The evaluation metrics for Random Forest is given below,

```
Training Accuracy for Random Forest is 0.98
Test Accuracy for Random Forest is 0.61
Precision for Random Forest is 0.52
Recall for Random Forest is 0.30
The confusion Matrix for Random Forest is
```

Predict	0	1	All
Actual			
0	1188	389	1577
1	938	269	1207
All	2126	658	2784

Figure 20. Evaluation metrics for the Random Forest Model

4.6.4. K-Nearest Neighbors (KNN):

One of the assumptions of the K-Nearest Neighbors algorithm is that similar things exist in close proximity. KNN is a non-parametric classification algorithm and it does not require any assumptions about the data distribution. The steps involved implementing in the KNN algorithm are,

1. Initialize the "K" as the chosen number of neighbors
2. Calculate the distance Euclidean distance between each data point and the test data.
3. Sorts the ordered collection of distances in ascending order and pick the first K entries
4. Returns the mode of the K labels

The evaluation metrics for KNN is given below,

```
Training Accuracy for KNN is 0.65
Test Accuracy for KNN is 0.58
Precision for KNN is 0.46
Recall for KNN is 0.23
The confusion Matrix for KNN is
```

Predict	0	1	All
Actual			
0	1263	311	1574
1	1000	290	1290
All	2263	601	2864

Figure 21. Evaluation metrics for the KNN Model

4.6.5. Stochastic Gradient Descent (SGD) classifier:

Stochastic Gradient Descent classifier is an approach to fit linear classifiers and regressors using loss functions such as (linear) Support Vector Machines. Gradient means a slope. So gradient descent essentially keeps decreasing the slope to reach the lowest point on that surface. Mathematically, it works by finding the parameters of a function

that minimize the cost function. The evaluation metrics for the Stochastic Gradient Descent classifier is given below,

```
Training Accuracy for SGD is 0.51
Test Accuracy for SGD is 0.51
Precision for SGD is 0.43
Recall for SGD is 0.66
The confusion Matrix for SGD is
```

Predict	0	1	All
Actual			
0	617	957	1574
1	491	799	1290
All	1108	1756	2864

Figure 22. Evaluation metrics for the SGD Model

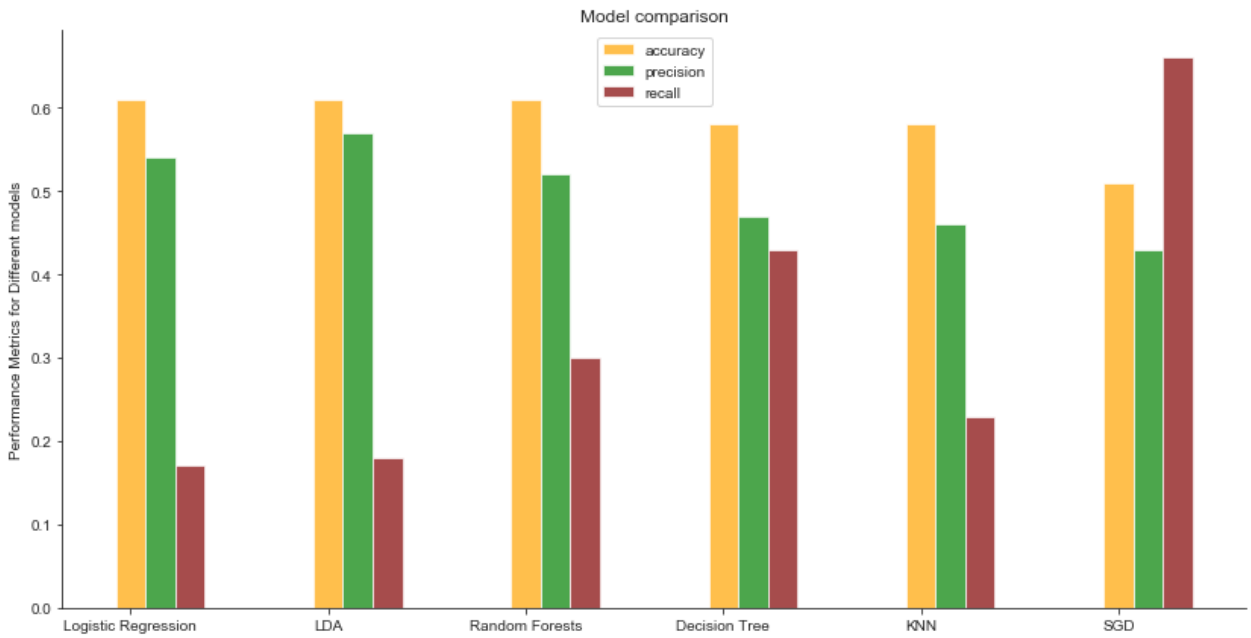

Figure 23. Model comparison

Model	F-1 Score
Logistic Regression	0.2810
Linear Discriminant Analysis	0.2736
Random Forest	0.3805
Decision Tree	0.4492
KNN	0.3667
SGD	0.5207

Table 2. F-1 Scores for each of the models

From the above charts and tables, it can be seen that while the accuracy is pretty much the same with all the models, Logistic Regression and LDA has the highest precision and Stochastic Gradient Descent classifier has the highest recall. This would mean that Logistic Regression and LDA produce a better percentage of relevant classifications, but SGD has the highest percentage of total relevant results correctly classified by the algorithm. Even with the F-1 score metric, the SGD algorithm performs much better than the other algorithms.

5. CONCLUSION

We tried to answer two of our research questions in this project. One of which is figuring out the most important features for predicting the readmission of diabetes patients. The most important features based on three different approaches of RFE using LDA, Random Forest, and Decision Trees are Race, Gender, Age, Admission type, Admission source, Number of medications, Number of diagnoses, Diabetes medications prescribed, Total procedures, Total visits, Diagnosis 1, Diagnosis 2, Diagnosis 3, Discharge Disposition.

As for the choice of the classification algorithm to classify the readmission of diabetes patients, based on the business situation at hand, we may use the algorithm that has the most precision or the most recall. We could use LDA with higher precision – if the use case is to be more confident on the True Positives or SGD with a higher recall where not missing out on capturing a diabetic patient by the algorithm is of the highest priority.

6. REFERENCES

[1] Lin, Y. W., Zhou, Y., Faghri, F., Shaw, M. J., & Campbell, R. H. (2019). Analysis and prediction of unplanned intensive care unit readmission using recurrent neural networks with long short-term memory. PloS one, 14(7), e0218942. https://doi.org/10.1371/journal.pone.0218942

[2] "Accurate and reproducible prediction of ICU readmissions" Dinh-Phong Nguyen, Nicolas Paris, Adrien Parrot medRxiv 2019.12.26.19015909; doi: https://doi.org/10.1101/2019.12.26.19015909

[3] "Impact of HbA1c Measurement on Hospital Readmission Rates: Analysis of 70,000 Clinical Database Patient Records" Beata Strack, Jonathan P. DeShazo, Chris Gennings, Juan L. Olmo, Sebastian Ventura, Krzysztof J. Cios, and John N. Clore

[4] Diabetes Data Set – UCI Machine Learning Repository – https://archive.ics.uci.edu/ml/datasets/diabetes+130-us+hospitals+for+years+1999-2008

[5] ICD-9-CM Diagnosis and Procedure Codes: Abbreviated and Full Code Titles https://www.cms.gov/Medicare/Coding/ICD9ProviderDiagnosticCodes/codes

[6] Burke T. (2010). The health information technology provisions in the American Recovery and Reinvestment Act of 2009: implications for public health policy and practice. Public health reports (Washington, D.C. : 1974), 125(1), 141–145. https://doi.org/10.1177/003335491012500119

[7] A1C Test Measure https://www.cdc.gov/diabetes/managing/managing-blood-sugar/a1c.html#:~:text=Get%20an%20A1C%20test%20to,over%20the%20past%203%20months .

[8] DeShazo, J. P., & Hoffman, M. A. (2015). A comparison of a multistate inpatient EHR database to the HCUP Nationwide Inpatient Sample. BMC health services research, 15, 384. https://doi.org/10.1186/s12913-015-1025-7

[9] Diabetes Tests https://www.cdc.gov/diabetes/basics/getting-tested.html#:~:text=Glucose%20Tolerance%20Test&text=At%202%20hours%2C%20a%20blood,higher%20indicates%20you%20have%20diabetes.

[10] Felix, H. C., Seaberg, B., Bursac, Z., Thostenson, J., & Stewart, M. K. (2015). Why do patients keep coming back? Results of a readmitted patient survey. Social work in health care, 54(1), 1–15. https://doi.org/10.1080/00981389.2014.966881

[11] Anderson GF, Steinberg EP. Hospital readmissions in the Medicare population. N Engl J Med. 1984 Nov 22;311(21):1349-53. doi: 10.1056/NEJM198411223112105. PMID: 6436703.

[12] Hernandez AF, Greiner MA, Fonarow GC, Hammill BG, Heidenreich PA, Yancy CW, Peterson ED, Curtis LH. Relationship between early physician follow-up and 30-day readmission among Medicare beneficiaries hospitalized for heart failure. JAMA. 2010 May 5;303(17):1716-22. doi: 10.1001/jama.2010.533. PMID: 20442387.

[13] Robinson S, Howie-Esquivel J, Vlahov D. Readmission risk factors after hospital discharge among the elderly. Popul Health Manag. 2012 Dec;15(6):338-51. doi: 10.1089/pop.2011.0095. Epub 2012 Jul 23. PMID: 22823255.

[14] Upadhyay, S., Stephenson, A. L., & Smith, D. G. (2019). Readmission Rates and Their Impact on Hospital Financial Performance: A Study of Washington Hospitals. Inquiry : a journal of medical care organization, provision and financing, 56, 46958019860386. https://doi.org/10.1177/0046958019860386

[15] Jasti H, Mortensen EM, Obrosky DS, Kapoor WN, Fine MJ. Causes and risk factors for rehospitalization of patients hospitalized with community-acquired pneumonia. Clin Infect Dis. 2008 Feb 15;46(4):550-6. doi: 10.1086/526526. PMID: 18194099.

[16] Carey K, Lin MY. Hospital length of stay and readmission: an early investigation. Med Care Res Rev. 2014 Feb;71(1):99-111. doi: 10.1177/1077558713504998. Epub 2013 Oct 16. PMID: 24132581.

[17] Costs for Hospital Stays in the United States, 2012 Brian Moore, Ph.D., Katharine Levit, B.A., and Anne Elixhauser, Ph.D.

[18] Dreyer, R. P., Ranasinghe, I., Wang, Y., Dharmarajan, K., Murugiah, K., Nuti, S. V., Hsieh, A. F., Spertus, J. A., & Krumholz, H. M. (2015). Sex Differences in the Rate, Timing, and Principal Diagnoses of 30-Day Readmissions in Younger Patients with Acute Myocardial Infarction. Circulation, 132(3), 158–166. https://doi.org/10.1161/CIRCULATIONAHA.114.014776

[19] Diabetes 130 US hospitals for years 1999-2008 https://www.kaggle.com/brandao/diabetes

[20] Rubin, D.J. Correction to: Hospital Readmission of Patients with Diabetes. Curr Diab Rep 18, 21 (2018). https://doi.org/10.1007/s11892-018-0989-1

[21] Bansal, V., Mottalib, A., Pawar, T. K., Abbasakoor, N., Chuang, E., Chaudhry, A., Sakr, M., Gabbay, R. A., & Hamdy, O. (2018). Inpatient diabetes management by specialized diabetes team versus primary service team in non-critical care units: impact on 30-day readmission rate and hospital cost. BMJ open diabetes research & care, 6(1), e000460. https://doi.org/10.1136/bmjdrc-2017-000460

[22] Alloghani, M., Aljaaf, A., Hussain, A., Baker, T., Mustafina, J., Al-Jumeily, D., & Khalaf, M. (2019). Implementation of machine learning algorithms to create diabetic patient re-admission profiles. BMC medical informatics and decision making, 19(Suppl 9), 253. https://doi.org/10.1186/s12911-019-0990-x

[23] Singla, R., Singla, A., Gupta, Y., & Kalra, S. (2019). Artificial Intelligence/Machine Learning in Diabetes Care. Indian journal of endocrinology and metabolism, 23(4), 495–497. https://doi.org/10.4103/ijem.IJEM_228_19