Georgia State University

CSC 4780/6780 & DSCI 4780 – Fundamentals of Data Science

Fall 2024

Final Project Report

Hospital Readmission Prediction for Diabetes Patients

RecallMe

Pranjal Patil
Natnael Alemayehu
Kushal Sarkar
Kevin Gallardo

Table of Contents

1 Business Understanding

There have been many advancements in the realm of diabetes treatment, but most patients are not taking advantage of those treatments. This could be exacerbated by the hospital's management of patients with diabetes which could be improved to allow patients to reach a stable condition before being discharged and prevent unnecessary suffering of patients and even deaths. By being able to better predict the early readmission of patients within 30 days of discharge we can improve the management of hospitals by allowing more time for the patients to stabilize and avoid the costs of readmissions, and any complications that could happen from being discharged too early. We choose the dataset of 130-US hospitals for the years 1999 to 2008 available at UC Irvine machine learning repository. The dataset has a total of 47 descriptive features, with some of them being admission type, number of lab procedures, weather certain drugs were prescribed or not, and the target feature of readmitted. The dataset has a total of 101,766 instances occurring within a span of 10 years. One of the proposed analytics solutions is a readmission prediction model. Our solution is to use a supervised learning model to classify if a patient will be readmitted within 30 days. An additional goal would be to determine the most impactful factors that lead to an early readmission within 30 days of discharge to provide possible new insights into the proper discharge of patients with diabetes.

1.1 Business Problem

Managing diabetes effectively remains a significant challenge, with many patients not leveraging advancements in treatment. This situation may be exacerbated by hospitals discharging patients prematurely, leading to avoidable suffering, complications, or even deaths. One way to address this issue is to predict early readmissions within 30 days of discharge, which would allow hospitals to manage patients better. By ensuring that patients have stabilized before being discharged, healthcare facilities can prevent unnecessary readmissions, reduce associated costs, and improve patient outcomes.

1.2 Dataset

The dataset consists of patient records from 130 U.S. hospitals and integrated delivery networks, spanning a 10-year period (1999–2008). It includes over 50 features representing various aspects of patient and hospital outcomes. The dataset specifically focuses on inpatient encounters for patients diagnosed with diabetes, including those who had laboratory tests performed, and medications administered during their stay.

Key Characteristics of the Dataset:

I. Inpatient Encounters: The data is restricted to inpatient encounters where the patient was admitted to the hospital, fulfilling the criteria for hospitalization, including a length of stay between 1 and 14 days.

II. Diabetic Encounters: Each record represents an encounter where a diabetes diagnosis (of any

type) was entered into the system.

III. Medical Interventions: The dataset includes details of medical procedures such as laboratory tests and medications administered during the hospitalization. These features are crucial in understanding the clinical care provided to the patients.

IV. Demographic Features: It contains sensitive information like age, gender, and race, which are important factors in diabetes care but must be handled carefully due to privacy concerns.

V. Hospital and Treatment Details: The dataset also includes information on the admission type, medical specialty of the admitting physician, number of procedures and lab tests, medications, and patient health outcomes like HbA1c test results.

Features of the Dataset:

I. Patient Demographics: Age, gender, race, and patient number.

II. Admission Information: Type of admission, time spent in the hospital, and admitting physician.

III. Medical Data: Number of lab tests performed, results of key medical tests, number of medications, and diabetic medications used.

IV. Historical Data: Information about prior hospital visits (outpatient, emergency, inpatient) within the past year.

This dataset is crucial for analyzing patient outcomes and predicting early readmissions for diabetic patients. The data spans a substantial time frame, which allows for historical analysis of changes in hospital practices and patient outcomes over the years.

P Sensitive Data:

The dataset includes sensitive personal information such as age, gender, and race, which are protected under healthcare data privacy regulations such as HIPAA in the U.S.

1.3 Proposed Analytics Solution

The proposed solution aims to develop a supervised machine learning model that predicts whether a diabetic patient will be readmitted to the hospital within 30 days of discharge. This predictive model will enable hospitals to optimize patient management, reduce readmission rates, and improve hospital resource allocation.

Key Objective:

Predict Early Readmissions: The primary objective is to predict whether a diabetic patient will be readmitted within 30 days. This will help identify patients who are at higher risk of complications and early discharge.

Impact on Hospital Management:

1. Improved Management: By predicting readmissions, hospitals can take timely action to ensure that patients have fully recovered and stabilized before being sent home, thus reducing unnecessary readmissions and associated costs.
2. Enhance Patient Care: Ensuring patients are stable before discharge will reduce the risk of readmissions and adverse outcomes, improving overall patient care.
3. Optimize Resource Allocation: Predictive models can help hospitals manage their resources more effectively, ensuring that the right number of staff and medical equipment are available for high-risk patients.

2 Data Exploration and Preprocessing

2.1 Data Quality Report

When exploring our attributes we saw that we had 13 continues attributes and 37 categorical attributes. Three of the continues attributes (admission_type_id, discharge_disposition_id, admission_source_id) were nominal attributes so we changed them to objects in the data frame. So, after that we have 10 continues attributes and 40 categorical attributes. The continues attributes include two id types (encounter_id and patient_nbr) and nine types of metrics (time in hospital, number of: lab procedures, procedures (not including lab procedures), medications, outpatient (visit), emergency, inpatient, and diagnosis(count). For the categorical features there is information about the patient like race, gender, age, and weight. There is also information about the payer code, medical specialty, admission type id, discharge_disposition_id, admission source id, and 3 types of diagnosis. Then there are 23 types of medicine that could be given to the patient. Then we have features about the patient's care like the glucose serum test result, A1c test result, if there was a change in medication, and if diabetes medication was prescribed or not. Lastly, we have our target feature which is the 'readmitted' feature. The readmitted feature which can take three types of values: <30 (readmitted within 30 days), >30(readmitted after 30 days), and No. When we first upload the data and view the data frame, it becomes apparent that some fields are filled in with '?'. So, the first thing we did is replace the question marks with a 'nan' value. We also noticed that the gender had a 'Unknown/Invalid' value, so we replaced that with a 'nan' value as well.

Table 1. Data Quality Report for Continuous Features

	count	mean	std	min	0.25	0.5	0.75	max
encounter_id	101766	1.65E+08	1.03E+08	1.25E+04	8.50E+07	1.52E+08	2.30E+08	4.44E+08
patient_nbr	101766	5.43E+07	3.87E+07	1.35E+02	2.34E+07	4.55E+07	8.75E+07	1.90E+08
time_in_hospital	101766	4.4	2.99	1	2	4	6	14
num_lab_procedures	101766	43.1	19.67	1	31	44	57	132
num_procedures	101766	1.34	1.71	0	0	1	2	6
num_medications	101766	16.02	8.13	1	10	15	20	81
number_outpatient	101766	0.37	1.27	0	0	0	0	42
number_emergency	101766	0.2	0.93	0	0	0	0	76
number_inpatient	101766	0.64	1.26	0	0	0	1	21
number_diagnoses	101766	7.42	1.93	1	6	8	9	16

In the continues features we noticed that encounter ID and patient NBR have a wide range of values when you look at the standard deviation. So, we will investigate that to see if we can drop those features.

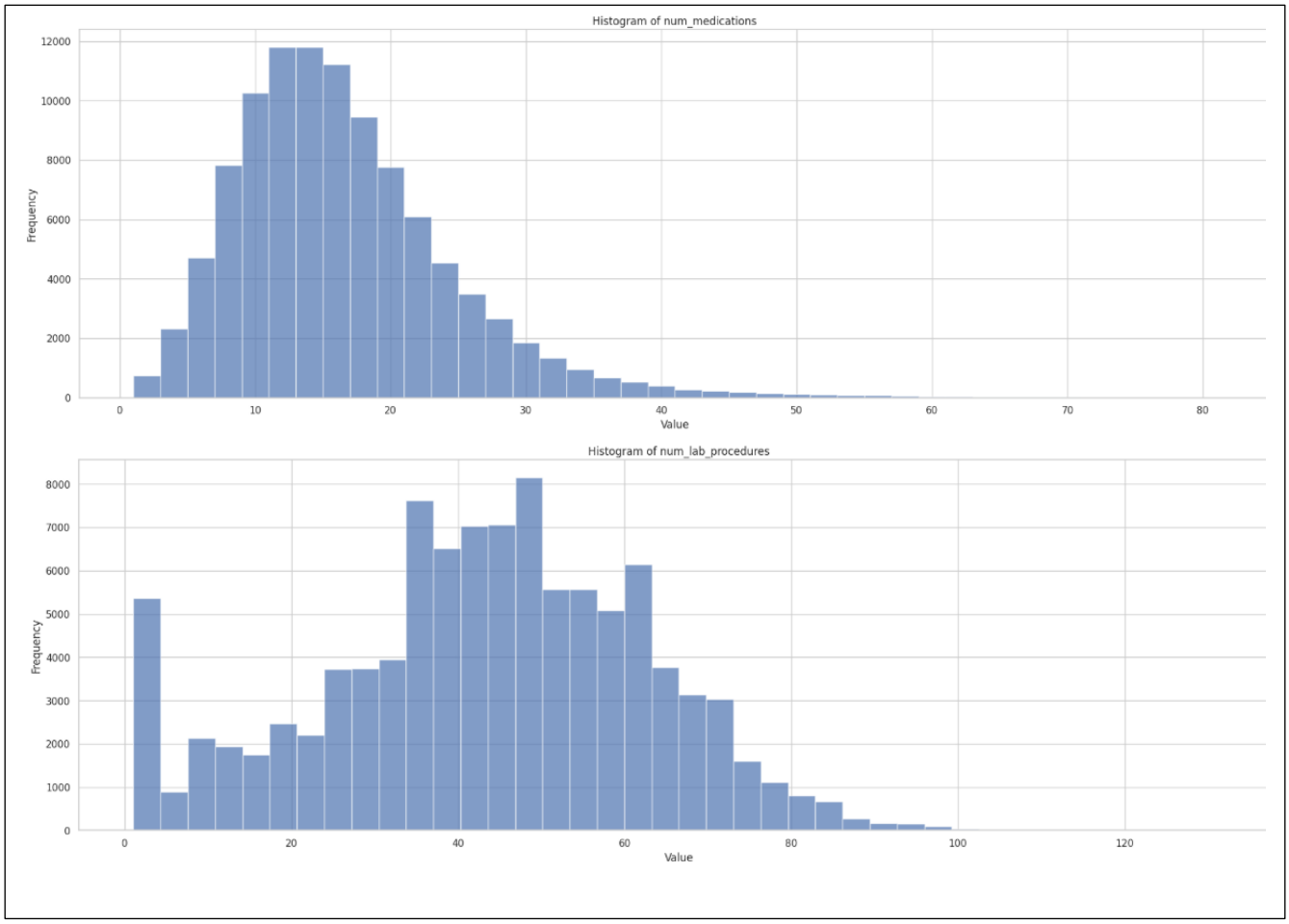
Table 2. Data Quality Report for Categorical Features

Feature	Count	% Miss	Card.	Mode	Mode Frec	Mode %	2nd Mode	2nd Mode	2nd Mode
weight	101766	96.85848	10	[75-100)	1336	1.312816	[50-75)	897	0.88
max_glu_serum	101766	94.74677	4	Norm	2597	2.551933	>200	1485	1.46
A1Cresult	101766	83.27732	4	>8	8216	8.073423	Norm	4990	4.9
medical_specialty	101766	49.08221	73	InternalMe	14635	14.38103	Emergenc	7565	7.43
payer_code	101766	39.55742	18	MC	32439	31.87607	HM	6274	6.17
race	101766	2.233555	6	Caucasian	76099	74.77841	AfricanAm	19210	18.88
diag_3	101766	1.398306	790	250	11555	11.35448	401	8289	8.15
diag_2	101766	0.351787	749	276	6752	6.634829	428	6662	6.55
diag_1	101766	0.020636	717	428	6862	6.74292	414	6581	6.47
gender	101766	0.002948	3	Female	54708	53.75862	Male	47055	46.24
metformin-rosiglitazon	101766	0	2	No	101764	99.99803	Steady	2	0
insulin	101766	0	4	No	47383	46.56074	Steady	30849	30.31
troglitazone	101766	0	2	No	101763	99.99705	Steady	3	0
tolazamide	101766	0	3	No	101727	99.96168	Steady	38	0.04
examide	101766	0	1	No	101766	100	na	na	na
citoglipton	101766	0	1	No	101766	100	na	na	na
glipizide-metformin	101766	0	2	No	101753	99.98723	Steady	13	0.01
glyburide-metformin	101766	0	4	No	101060	99.30625	Steady	692	0.68
metformin-pioglitazone	101766	0	2	No	101765	99.99902	Steady	1	0
glimepiride-pioglitazon	101766	0	2	No	101765	99.99902	Steady	1	0
diabetesMed	101766	0	2	Yes	78363	77.00312	No	23403	23
acarbose	101766	0	4	No	101458	99.69734	Steady	295	0.29
change	101766	0	2	No	54755	53.80481	Ch	47011	46.2
miglitol	101766	0	4	No	101728	99.96266	Steady	31	0.03
glipizide	101766	0	4	No	89080	87.53415	Steady	11356	11.16
rosiglitazone	101766	0	4	No	95401	93.74546	Steady	6100	5.99
pioglitazone	101766	0	4	No	94438	92.79917	Steady	6976	6.85
tolbutamide	101766	0	2	No	101743	99.9774	Steady	23	0.02
glyburide	101766	0	4	No	91116	89.53482	Steady	9274	9.11
acetohexamide	101766	0	2	No	101765	99.99902	Steady	1	0
glimepiride	101766	0	4	No	96575	94.89908	Steady	4670	4.59
chlorpropamide	101766	0	4	No	101680	99.91549	Steady	79	0.08
nateglinide	101766	0	4	No	101063	99.3092	Steady	668	0.66
repaglinide	101766	0	4	No	100227	98.48771	Steady	1384	1.36
metformin	101766	0	4	No	81778	80.35886	Steady	18346	18.03
admission_source_id	101766	0	17	7	57494	56.49628	1	29565	29.05
discharge_disposition_	101766	0	26	1	60234	59.18873	3	13954	13.71
admission_type_id	101766	0	8	1	53990	53.05308	3	18869	18.54
age	101766	0	10	[70-80)	26068	25.61563	[60-70)	22483	22.09
readmitted	101766	0	3	NO	54864	53.91192	>30	35545	34.93

For the categorical features we can see that in total there are a total of 10 features with missing data. We can also see that for the race feature the mode is Caucasian with 74% of the data consisting of it and second mode being African American with 18.88% of the data. Another interesting thing that we saw is that the examide and citoglipton have a cardinality of one with their mode taking 100% of the data.

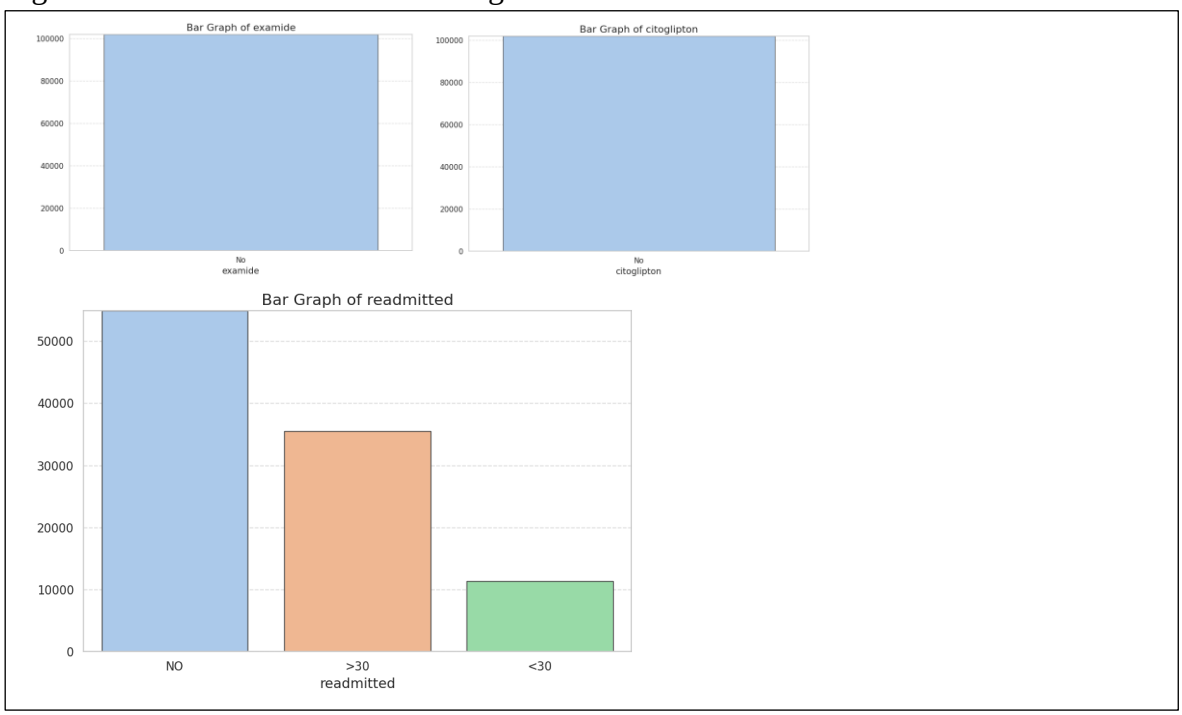
Next, we moved exploring some of the data. In figure 1 we can see that Number of medications has a unimodal skewed right distribution and number of lab procedures is a normal distribution.

Figure 1. Visualizations of some Continuous Features in Dataset



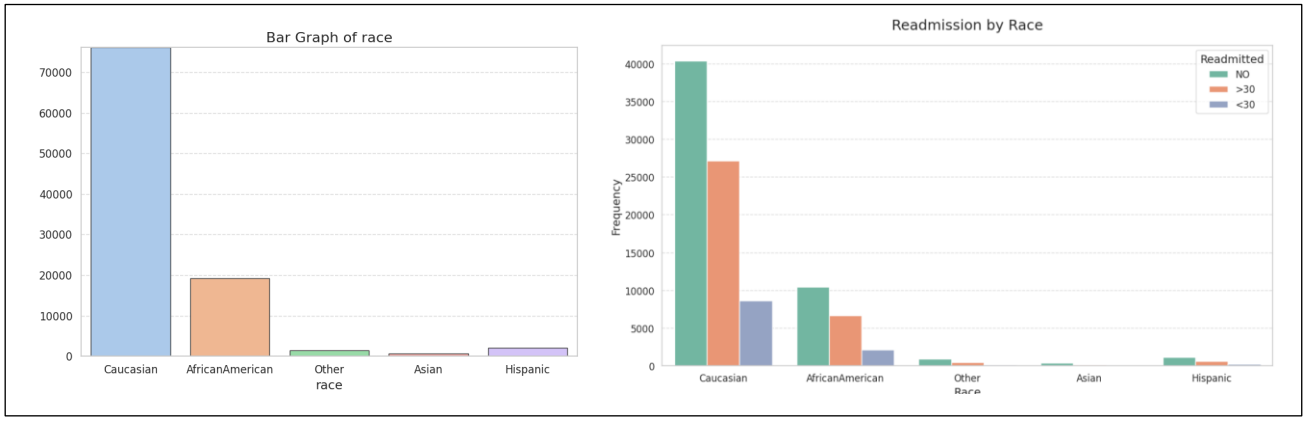
In figure 2 we visualized some of the categorical features and examide and citoglipton caught our eyes once again since they stay the same for every instance of the data. We also added our target feature to the graphs to show the target feature is present in the data.

Figure 2. Visualizations of some Categorical Features in Dataset



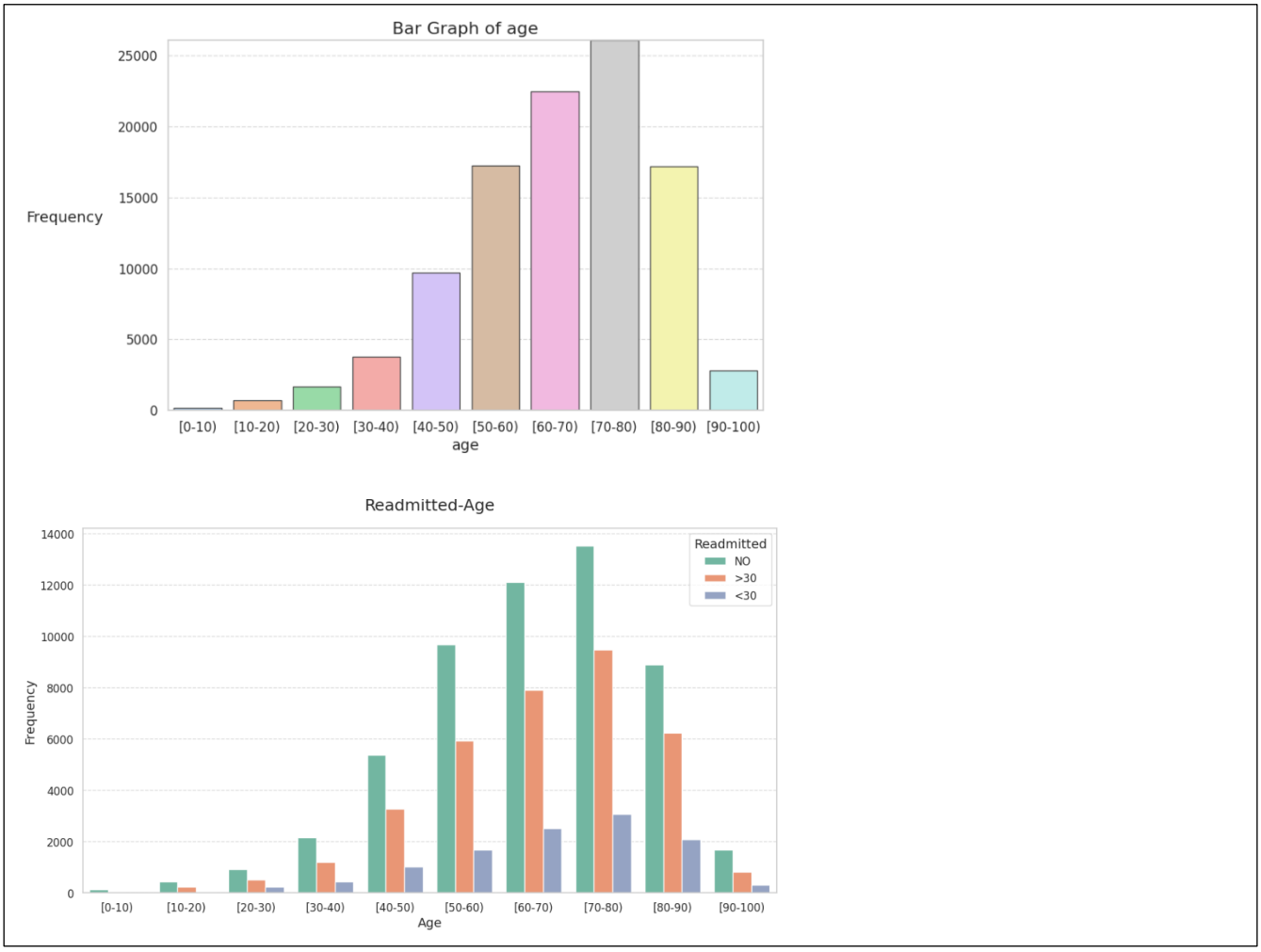
For figure 3 we can see that the distribution of race with the readmittance of each specific race. We can conclude that any readmissions pattern will follow across race since the readmittance rate seems to be proportional across the race categories.

Figure 3. Visualizations of race and readmitted split by race



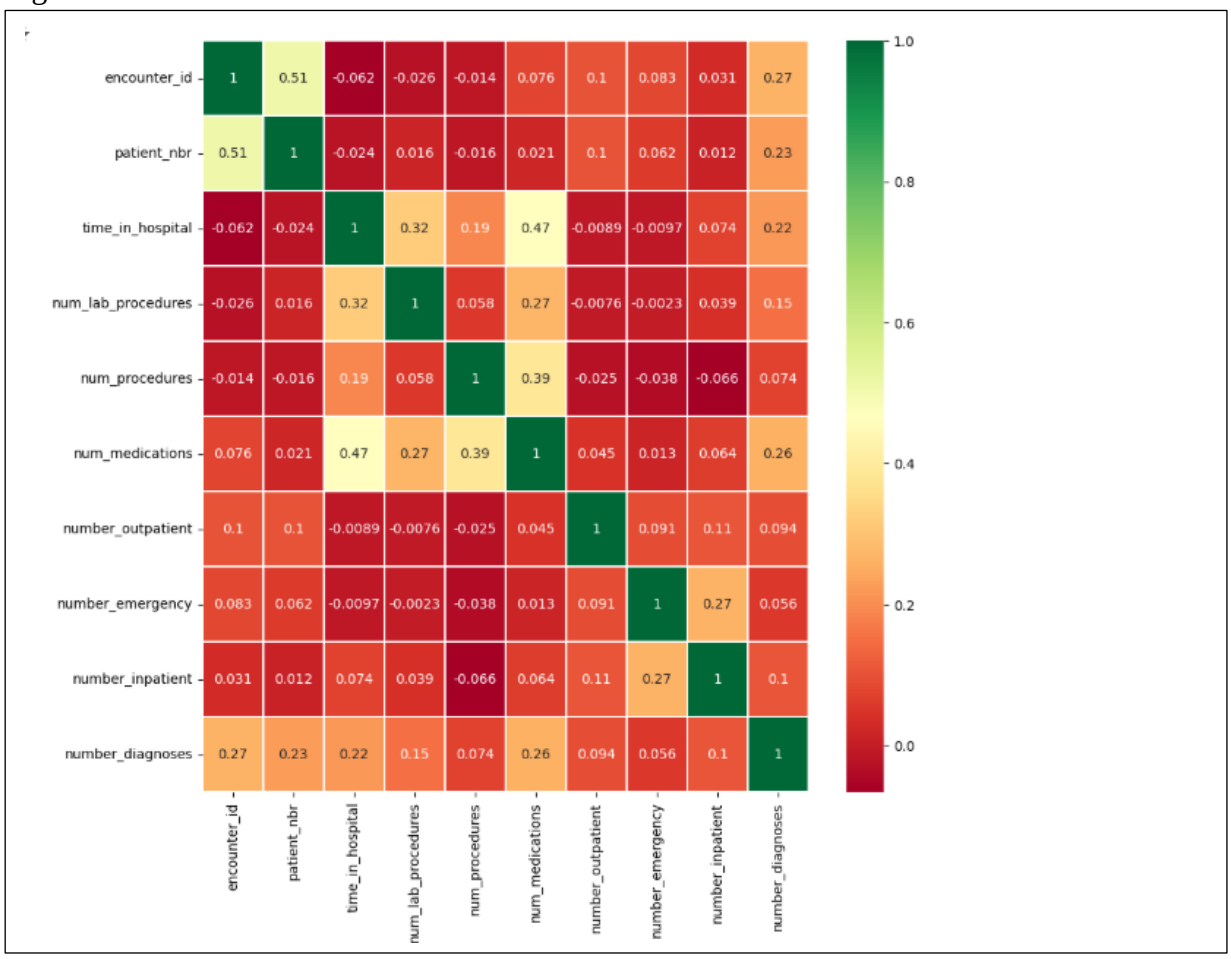
For figure 4 we can see that the distribution of age with the readmittance of each specific age. We can conclude that any readmissions pattern will follow across age groups since the readmittance rate seems to be proportional across the race age category as well.

Figure 4. Visualizations of age and readmitted split by age



In figure 5 we can see the highest correlation between encounter ID and patient NBR which might get dropped from the dataset once we investigate the meaning and purpose of each feature.

Figure 5.



2.2 Missing Values and Outliers

2.2.1 Data Cleaning and Dropping Features

1. Drop irrelevant:

The irrelevant columns that we decided to drop were encounter_id and patient_id. These were considered irrelevant because we can use index to identify each patient, and the ids do not add any benefit.

2. Drop high percentage missing:

The columns which had a high percentage of data missing were: weight, medical specialty, and payer code. Weight was missing 96.86%, medical specialty was missing 49.08%, and payer code was missing 39.56%. Max_glu_serum and A1Cresult showed that there were missing a big percentage of data but later we figured out that if the test was not taken then it would be recorded as “None” so that was why it was being read as missing.

3. Drop constant columns:

The columns that did not change at all where examide and citoglipton. For examide and citoglipton the Mode percentage was 100%. For the medicines that had the mode at 99% we decided to drop them as well as they would contribute little information. Mode at 99% were the following: metformin-rosiglitazone, troglitazone, tolazamide, glipizide-metformin, glyburide-metformin, metformin-pioglitazone, glimepiride-pioglitazone, acarbose, miglitol, tolbutamide, acetohexamide, chlorpropamide , nateglinide.

2.2.2 Handling Missing Data

1. Fill missing values in 'race':

We replace missing values in the 'race' column with the mode (most common category) to minimize information loss.

2. Remove gender entries with "unknown" values:

Ensures only valid gender data is retained, removing rows with missing gender information for better model integrity.

3. Drop rows where any diagnosis (diag_1, diag_2, diag_3) is missing:

Ensures that rows with complete lack of diagnostic data are removed, as these are critical to analysis.

2.2.3 Duplicate Removal:

1. Drop duplicate patient records:

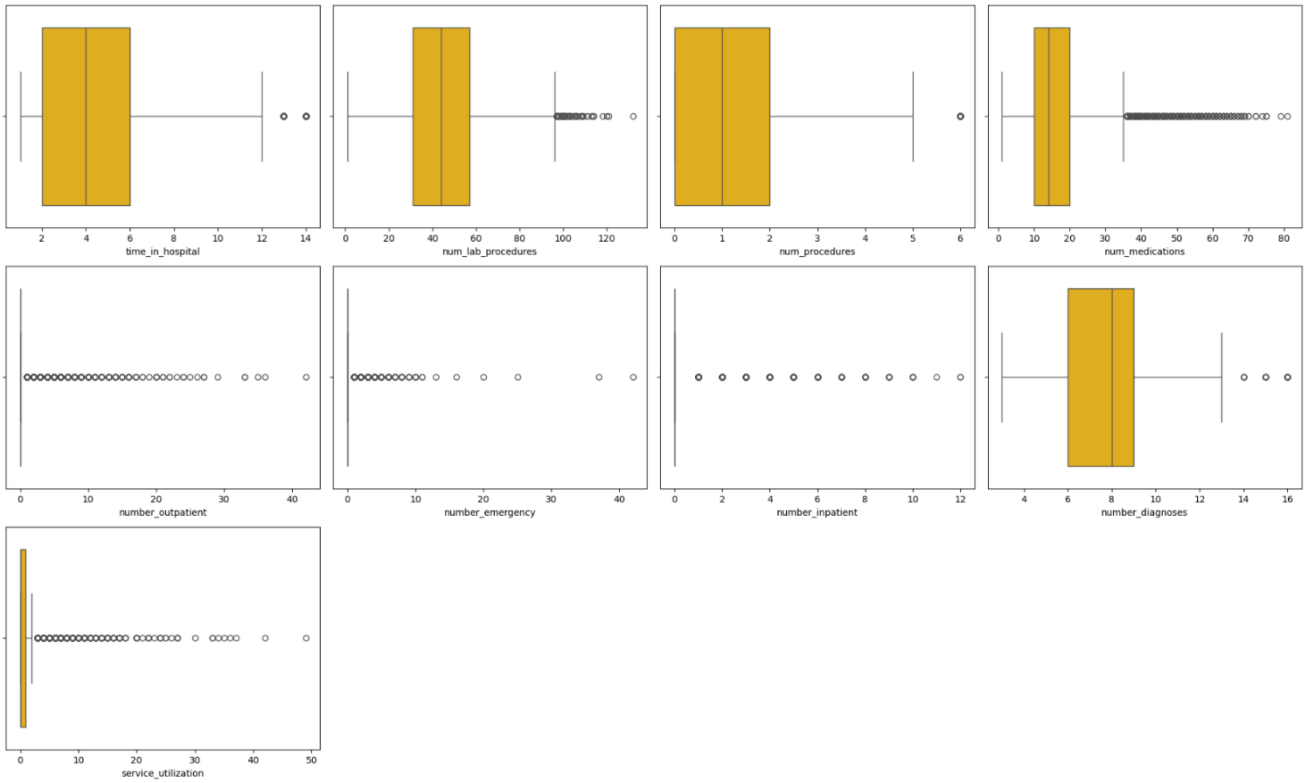
Retains only the first encounter of each patient to prevent data leakage or bias from multiple encounters. Drops patient identifiers to prevent models from inadvertently "learning" patient-specific patterns, focusing on generalizable features.

2.2.4 Outlier Detection

1. Boxplot visualization for numeric columns:

Boxplots for numeric features to visually identify outliers for possible handling, ensuring data consistency.

Figure 6.



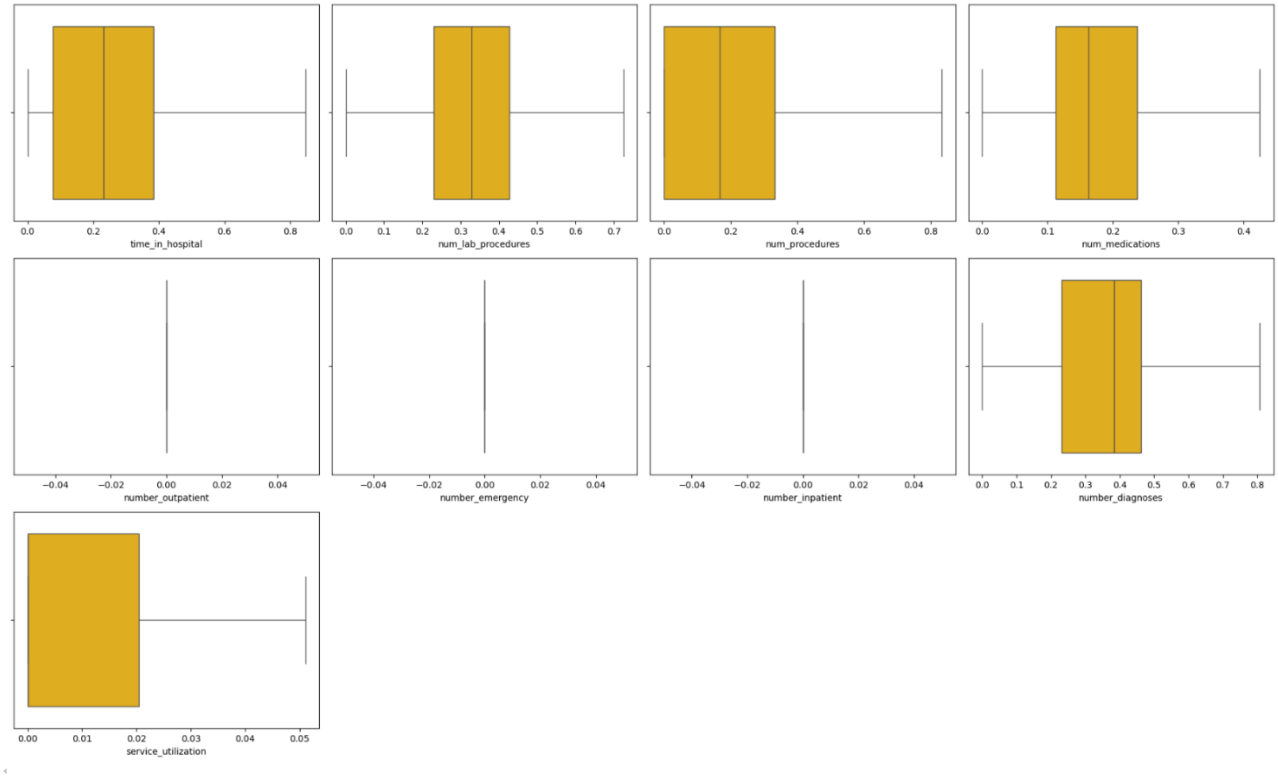
2.2.5 Handling Outliers

1. Outlier treatment using IQR:

Clamps extreme values in numeric features to IQR bounds, minimizing the impact of outliers while preserving data range consistency.

Visual validation to ensure outlier handling was applied effectively and the data is now within reasonable bound

Figure 7.



2.3 Normalization

After cleaning and initial transformations, we normalized the numerical features in our dataset to ensure all variables were on a comparable scale, which is crucial for many machine learning algorithms to perform optimally. We implemented a min-max normalization approach to scale all numeric features to a range between 0 and 1.

2.3.1 Normalization Method

We employed min-max normalization using the following formula:

*x_normalized = (x - x_min) / (x_max - x_min) * (new_max - new_min) + new_min*

2.3.2 Normalized Features

The following numeric features were normalized:

- **Demographic**: *gender*
- **Hospital Visit Information**: *time_in_hospital*

- **Medical Procedures and Tests**:
  - *num_lab_procedures*
  - *num_procedures*
  - *num_medications*
- **Patient History**:
  - *number_outpatient*
  - *number_emergency*
  - *number_inpatient*
  - *number_diagnoses*

2.3.3 Impact of Normalization

The normalization process ensures that:

- All numeric features now fall within the range [0,1]
- No feature range dominates the others due to different scales
- The relative relationships between values within each feature are preserved
- The model can treat all numeric features equally during training

Binary categorical features (such as *gender, diabetesMed, medication indicators, drugs administered and readmitted*) were left as 0/1 encoding since they were already effectively normalized after doing transformations and mapping the ones we could to 0s and 1s.

2.4 Transformations

In the dataset, we have several Nominal attributes. We transformed these attributes to Numerical values. The following transformations took place for the attributes:

- Readmitted: This column is our target feature. It is about "days to inpatient readmission". If the patient was readmitted in less than 30 days "<30". If the patient was readmitted in more than 30 days ">30". If there is no record "NO". We decided to reduce these values to two and map them according to the following rule: 0 for “NO” and “>30”, 1 for “<30”.

- Age: The attribute was given in range values like [0,10), [10,20) , ..., up to [90,100). These values were mapped to the mid values of the range starting at 5 and going all the way to 95.

- Diag_1, Diag_2, Diag_3: These attributes represented the primary diagnosis (coded as first three digits of ICD9); 848 distinct values, Secondary diagnosis (coded as first three digits of ICD9); 923 distinct values, Additional secondary diagnosis (coded as first three digits of ICD9); 954 distinct values respectively. We used the research paper (link here) associated with the dataset to map out these distinct numerical values for each diagnosis into fewer hierarchical group names. The mapping took places referencing this table:

Figure 8.

Table 2: Values of the primary diagnosis in the final dataset. In the analysis, groups that covered less than 3.5% of encounters were grouped into "other" category.

Group name	icd9 codes	Number of encounters	% of encounter	Description
Circulatory	390–459, 785	21,411	30.6%	Diseases of the circulatory system
Respiratory	460–519, 786	9,490	13.6%	Diseases of the respiratory system
Digestive	520–579, 787	6,485	9.3%	Diseases of the digestive system
Diabetes	250.xx	5,747	8.2%	Diabetes mellitus
Injury	800–999	4,697	6.7%	Injury and poisoning
Musculoskeletal	710–739	4,076	5.8%	Diseases of the musculoskeletal system and connective tissue
Genitourinary	580–629, 788	3,435	4.9%	Diseases of the genitourinary system
Neoplasms	140–239	2,536	3.6%	Neoplasms
Other (17.3%)	780, 781, 784, 790–799	2,136	3.1%	Other symptoms, signs, and ill-defined conditions
	240–279, without 250	1,851	2.6%	Endocrine, nutritional, and metabolic diseases and immunity disorders, without diabetes
	680–709, 782	1,846	2.6%	Diseases of the skin and subcutaneous tissue
	001–139	1,683	2.4%	Infectious and parasitic diseases
	290–319	1,544	2.2%	Mental disorders
	E–V	918	1.3%	External causes of injury and supplemental classification
	280–289	652	0.9%	Diseases of the blood and blood-forming organs
	320–359	634	0.9%	Diseases of the nervous system
	630–679	586	0.8%	Complications of pregnancy, childbirth, and the puerperium
	360–389	216	0.3%	Diseases of the sense organs
	740–759	41	0.1%	Congenital anomalies

Figure 9. Diagnosis 1, 2, and 3 after transformation with readmission

- 23 features for Medications: The dataset contains 23 distinct medicine names as its attributes. 2 of these attributes: examide and citoglipton have only "No" values for all instances of the data. Hence these features were dropped. There were 13 medicines where the mode "Steady" represented 99% of the values so those were dropped as well. That left us with 8 medicines staying in our data. So the values that medicines could take were "down", "up", "steady", and "no". So we decided to map the "no" values to 0 and all the other values to 1.

- Gender: Gender was mapped to numerical value using the rule: 0 for 'Female' and 1 'Male'.

- Change: Change attribute was mapped using the rule: 0 for 'No' and 1 for other.

- DiabetesMed: DiabetesMed attribute was mapped using the rule: 0 for 'No' and 1 for other.

- Max_glu_serum: Max_glu_serum was mapped using the rule: 0 for 'None', 1 for 'Norm', and 2 for '>200' and '>300'.

- A1Cresult: A1Cresult was mapped using the rule: 0 for 'None', 1 for 'Norm', and 2 for '>7' and '>8'.6

- Admission_type_id: Admission type we did a mapping reduction to similar categories. We mapped categories 2 and 7 to category 1 which was ‘emergency’. We mapped categories 6 and 8 to category 5 which was ‘not available’.

- Discharge_disposition_id: Discharge disposition we did a mapping reduction as well down to similar categories. Categories 6, 8, 9, and 13 were mapped to category 1 which was ‘discharged to home’. Categories 3, 4, 5, 14, 22, 23, and 24 were mapped to category 2 which was ‘discharge/transferred to another short-term hospital’. Categories 12, 15, 16, and 17 were mapped to category 10 which was ‘Neonate discharged to another hospital for neonatal aftercare’. And categories 25 and 26 were mapped to category 18 which was ‘Null’.

- Admission_source_id: Admission source id was also reduced in mapping by reducing to similar categories. Categories 2 and 3 reduced to 1 which was ‘physical referral’. Categories 5, 6, 10, 22, and 25 to category 4 which was ‘transferred from a hospital’. Categories 15, 17, 20, and 21 to category 9 which was ‘not available’. And categories 13 and 14 to category 11 which was ‘normal delivery’.

3. Model Selection and Evaluation

3.1 Evaluation Metrics

Our performance metrics directly address the business problem of predicting early readmissions of diabetic patients within 30 days of discharge, enabling hospitals to optimize patient management, improve outcomes, and reduce costs. The selected metrics provide insights into the model's ability to meet the business objectives and ensure the solution's reliability and utility. Here's how each metric aligns with the problem:

- Accuracy : Accuracy measures the proportion of correct predictions across all instances, providing a general assessment of the model’s effectiveness. In the context of our business problem, accuracy helps evaluate the model's overall reliability in predicting readmissions. While high accuracy is desirable, it needs to be interpreted alongside other metrics, especially in datasets with class imbalances like ours, where the majority of patients might not be readmitted.
- Precision : Precision calculates the proportion of patients predicted to be readmitted who were actually readmitted. For hospitals, high precision ensures resources are effectively allocated to patients genuinely at risk, minimizing unnecessary interventions for those unlikely to return. This directly supports the goal of efficient hospital management, optimizing care delivery while avoiding wasted effort.
- Recall (Sensitivity) : Recall measures the model's ability to correctly identify all patients who were actually readmitted. This metric is critical in our business problem since missing at-risk patients could result in severe health complications or additional costs due to unmanaged conditions. A high recall rate ensures that most at-risk patients are flagged for closer observation or extended stabilization before discharge.

- F1-Score : The F1-score provides a balanced measure of the trade-off between precision and recall. It is particularly useful for our problem because false negatives (missed at-risk patients) and false positives (patients incorrectly flagged) carry significant costs. A high F1-score indicates that the model effectively balances these concerns, making it a dependable tool for hospital decision-making.
- AUC-ROC (Area Under the Receiver Operating Characteristic Curve): AUC-ROC evaluates the model's ability to distinguish between classes (readmitted vs. not readmitted) across all possible classification thresholds. It provides a comprehensive view of the trade-off between the true positive rate (recall) and the false positive rate. A high AUC score ensures that the model is effective in ranking patients by their risk of readmission, which is crucial for prioritizing intervention efforts and managing hospital resources efficiently. This metric is particularly helpful when working with imbalanced datasets, as it highlights the model's discrimination capability independent of specific thresholds.
- Macro and Weighted Averages: Given the potential imbalance in the dataset (fewer patients may be readmitted compared to those who are not), macro and weighted averages summarize performance across both classes. These metrics ensure the model performs consistently for readmitted and non-readmitted patients, providing a holistic view of its effectiveness. This helps hospitals understand whether the model is biased towards the majority class and ensures fair treatment of all patient outcomes.

The classification report metrics, supplemented with AUC-ROC, collectively validate whether the model balances the trade-offs between false positives and false negatives to deliver actionable insights for improving diabetes patient management.

3.2 Models

3.2.1 Error-based Learning

For one of our models, we choose to do a logistic regression. We chose logistic regression since our target feature got mapped to 1 being readmitted and 0 being not readmitted. All our continuous data got normalized and our categorical features got one-hot encoding. The max_iter value was set to 1,000 which is essentially how many times we will run the gradient descent algorithm trying to get convergence. For training the input would be the training data which consists of 80% of the preprocessed data. The training data is further split into 'X_train' which contains features that do not include the target and 'y_train' the target feature. These two variables 'X_train' and 'y_train' will be the inputs to train the model. Once the model is trained you will test it with the testing data. The testing data consists of 20% of the preprocessed data which split into 'X_test' and 'y_test'. Again 'X_test' contains features that do not include the target, and 'y_test' is the target. The trained model is given 'X_test' as the input variable and as an output it gives a list of predictions which take the values 0 or 1. Then those predictions would be compared with the 'y_test' target feature to test for accuracy.

3.2.2 Information-based Learning

For one of our models, we chose **CatBoost Classifier**, an information-based learning approach utilizing gradient boosting on decision trees. This model was selected for its ability to handle categorical features natively.

The target feature was mapped to 1 (readmitted) and 0 (not readmitted). The model was configured with the following parameters:

- **iterations=500**: Specifies the number of boosting iterations, which determines how many trees are built during training.
- **learning_rate=0.1**: Controls the step size in updating model weights, balancing the trade-off between training speed and convergence.
- **depth=6**: Sets the depth of the decision trees, capturing interactions between features while avoiding overfitting.
- **loss_function='Logloss'**: Optimizes the model for binary classification tasks by minimizing log loss.
- **eval_metric='Logloss'**: Measures model performance on validation data, ensuring consistent optimization.

The dataset was split into 80% training data (X_train, y_train) and 20% testing data (X_test, y_test). The model was trained to minimize the loss function, learning patterns from the input features in X_train. Predictions on X_test were compared with y_test to calculate metrics such as accuracy, precision, recall, and F1-score, providing insights into its ability to predict patient readmissions effectively.

3.2.3 Similarity-based Learning

For the similarity-based learning approach, we used the k-Nearest Neighbors (k-NN) classifier. This method was chosen for its simplicity and effectiveness in utilizing proximity based decision making, which is particularly useful for datasets with clear distinctions in feature space.

We tuned the k-NN classifier using GridSearchCV to identify the best hyperparameters. The optimal configuration included:

- **n_neighbors:** 7, meaning the algorithm looks at the 7 closest data points to decide the class based on which class is most common among them.
- **metric**: Euclidean distance, to measure the similarity between data points.
- **weights**: Distance-based, assigning higher importance to closer neighbors in the voting process.

The target variable was encoded as 1 for readmitted patients and 0 for non-readmitted patients. Data preprocessing, consistent across all models, included normalization of continuous features and one-hot encoding for categorical features.

The dataset was split into 80% for training and 20% for testing. The training set was used to fit the k-

NN model, while predictions were made on the testing set to evaluate performance.

3.2.4 Probability-based Learning

This section discusses the probabilistic-based learning approach, specifically using a **Naive Bayes classifier**. Here's an overview:

1. Data Preprocessing and Splitting: The dataset is split into 80% training and 20% testing data to ensure the model is trained on one set and validated on another.
2. Feature Selection:
    a. Chi-Square Test identifies statistically significant features by evaluating the dependency between features and the target. This is giving us very low accuracy (0.20)
    b. Recursive Feature Elimination (RFE) ranks features iteratively using a Random Forest model. This method is giving us moderate accuracy(0.54).
    c. An ANOVA F-test is applied to select the top 15 features based on statistical significance. This method is giving us perfect score. But this seems to be overfitting the model.

3. Probabilistic Learning:
    a. Gaussian Naive Bayes (GNB), which assumes normally distributed features, is trained on the selected features.
    b. GNB calculates probabilities of each class using Bayes' theorem and selects the most probable class.
4. Evaluation:
    a. Metrics like accuracy, F1 score, and a classification report are generated to evaluate the model’s performance, emphasizing its probabilistic decision-making on unseen test data.

This approach focuses on the probabilistic underpinnings of Naive Bayes while leveraging feature selection and data balancing to improve performance.

3.3 Evaluation

3.3.1 Evaluation Settings and Sampling

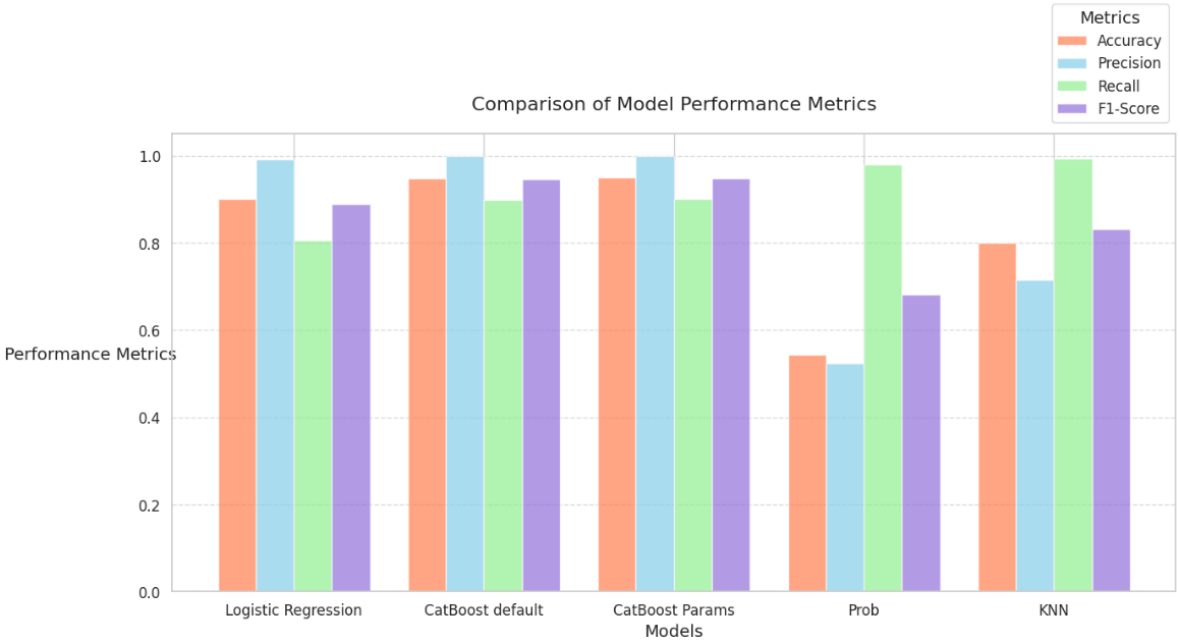
Our dataset was highly imbalanced 64163 cases where patients did not return, 6250 cases where patients returned within 30 days. To address the issue of this class imbalance in our dataset, we used **SMOTE (Synthetic Minority Oversampling Technique)**. Since our data had been one-hot encoded to represent categorical variables as binary columns, SMOTE was a more appropriate choice for generating synthetic samples. SMOTE works by interpolating between existing samples of the minority class, creating new synthetic data points that help balance the dataset. Using the SMOTE-sampled data, we built and evaluated various models, for all our models. This approach ensured that each model had

access to a balanced dataset, improving their ability to predict the minority class effectively and yielding a robust comparison of their performance across different metrics.

3.3.2 Evaluation

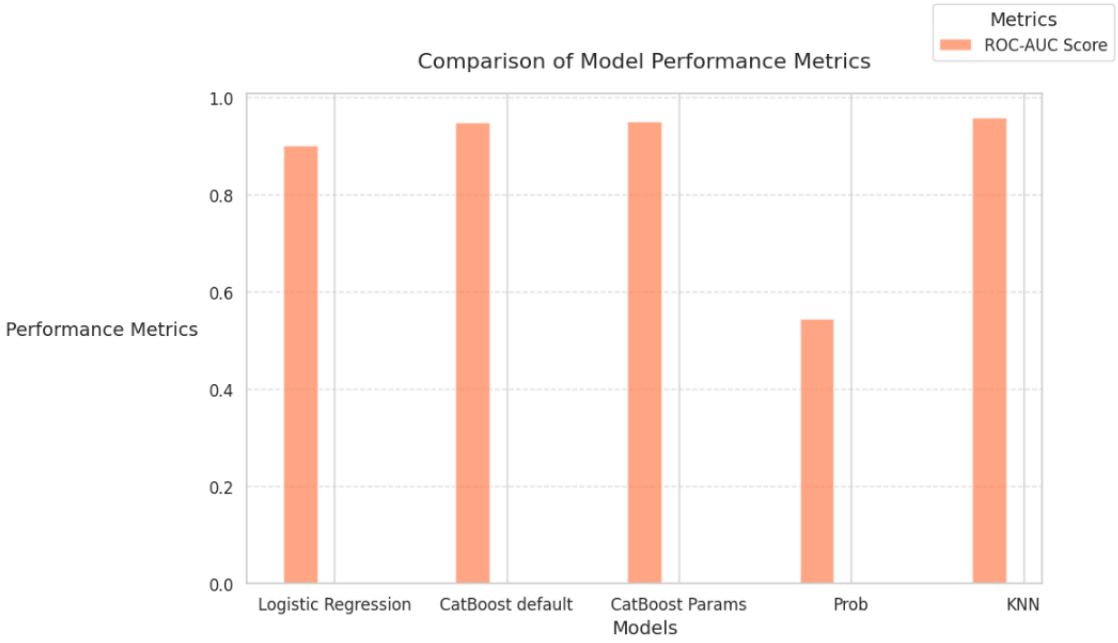
Based on figure 10 when comparing the accuracy, precision, recall, and F1-score for each model. We can see that CatBoost with default parameters has the highest overall accuracy and f1-score which ended up being very similar values to CatBosst with hyper-parameter optimization. CatBoost classifier had an accuracy of 94.88%, and f1-score of 95%,. While CatBoost with hyper-parameter optimization had an accuracy of 94.97%, and f1-score of 95%. So, we were able to improve the performance of CatBoost with hyper-parameter optimization by a small margin.

Figure 10.



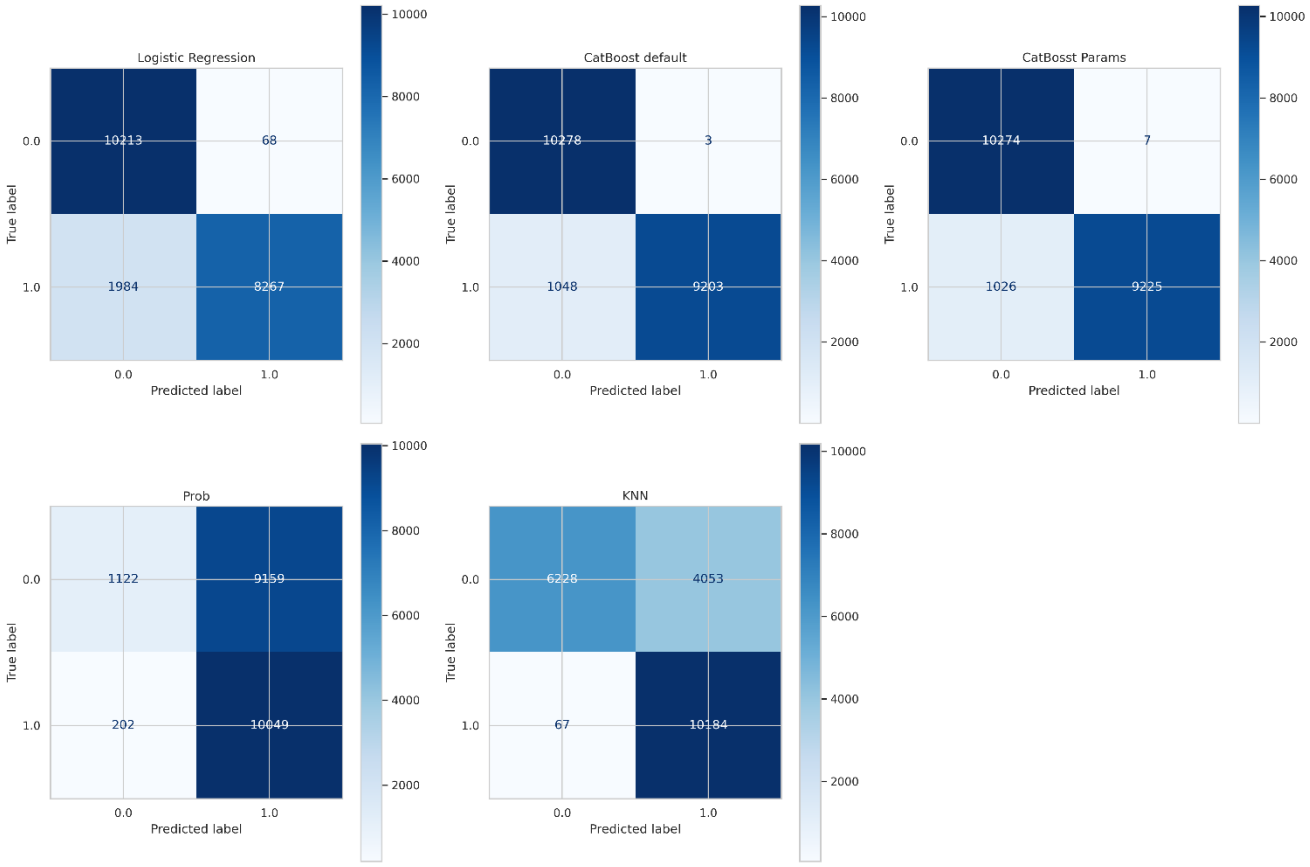
The ROC-AUC helps us compare the models based on the measure of separability. Meaning how well the model is able to distinguish between classes. A ROC-AUC closer to 1 is best. Based on figure 11, for the CatBoost default and CatBoost with hyper-parameter optimization we got a ROC-AUC of .9487 and .9496 respectively. KNN had a higher ROC-AUC of .9587 while probability based learning model had an ROC-AUC of .5447 which was the lowest one. Finally, the logistic regression model had a ROC-AUC of .9487 which means the model with the highest ROC-AUC score was the nearest neighbors model.

Figure 11.



In figure 12, we compared the confusion matrices for all the models. Overall, we can see that CatBoost default and Catboost with hyper-parameter optimization has the overall lowest amount of false positives and false negatives. This would be reflected in the f1-score of the models. Which we did see earlier was that the CatBoost models had the best f1-score out of the rest of the models.

Figure 12.



Based on the f1-score that minimizes the levels of false positives and false negatives we choose the CatBoost Classifier as our chosen model. CatBoost classifier had the second highest ROC-AUC score so we know that it can distinguish between readmitted and not readmitted well and we decided to minimize the false positives and false negatives.

3.3.3 Hyper-parameter Optimization

To maximize the performance of the **CatBoost Classifier**, we conducted hyperparameter optimization using the **Optuna** library, aiming to maximize the F1-score. This ensured a balance between precision and recall, which is critical for effectively predicting patient readmissions. A total of 20 trials were conducted using the TPE Sampler, systematically exploring key hyperparameters. These included iterations (the number of boosting iterations), learning_rate (step size for weight updates), depth (tree depth to control model complexity), boosting_type (strategy for boosting, such as Plain or Ordered), and bootstrap_type (sampling strategy like Bayesian or MVS to improve robustness). Additional hyperparameters, such as l2_leaf_reg (regularization to prevent overfitting), random_strength (noise

added to improve generalization), bagging_temperature (randomness in bagging), and early stopping criteria (od_type and od_wait), were also tuned to refine model performance.

The best combination of hyperparameters was updated to: iterations=774, learning_rate=0.0397, depth=8, boosting_type='Plain', bootstrap_type='MVS', l2_leaf_reg=1.19e-8, random_strength=7.61e-7, bagging_temperature=3.27, od_type='Iter', od_wait=16, and min_data_in_leaf=60. These optimized settings reflect a shift toward a more nuanced configuration, particularly with a lower learning rate and the MVS bootstrap strategy, emphasizing stability and generalization.

The impact of hyperparameter optimization is evident in the model's performance metrics. Before optimization, the model achieved an accuracy of **94.88%**, with a recall of **1.00** for class 0 and **0.90** for class 1, and a corresponding precision of **0.91** for class 0 and **1.00** for class 1. The ROC-AUC score stood at **0.9487**, indicating good overall discriminative ability. After optimization, the accuracy increased marginally to **94.97%**, reflecting an improvement in the balance between the classes. While the recall for class 1 remained at **0.90**, a reduction in false positives for class 0 led to a slightly higher ROC-AUC score of **0.9496**. Precision for class 0 stayed constant at **0.91**, while precision for class 1 remained at **1.00**, maintaining the model's ability to correctly identify true positives. These results demonstrate that hyperparameter tuning helped refine the model, achieving slightly better overall consistency and stability in its predictions.

4 Results and Conclusion

Out of all the four models that were tested in our analysis, the information based learning approach using the CatBoost Classifier showed the highest overall performance. With an accuracy of *94.97%* , a ROC-AUC score of *0.9496*, and balanced precision, recall, and F1 scores for both classes, it was the most effective model for predicting diabetic patient readmissions within 30 days.

The confusion matrix shows that the model correctly identified most readmissions (9,225 out of 10,251) and non readmissions (10,274 out of 10,281) with a very few misclassification. Additionally, the weighted averages in the classification report further highlight its reliability in both of the classes.

From the feature importance analysis, the top most impacting factors in predicting readmission were:

- **Time in hospital:** The number of days that patients stayed in the hospital showed a stronger correlation with readmissions.
- **Number of diagnoses:** The number of diagnoses that patients went through contributed to the likelihood of readmission
- **Number of procedures** and **service utilization:** Frequency of the procedures and services used also impacted on the likelihood of determining the readmission
- **Gender and medication related features:** During the analysis, we also observed that gender can also play an influencial role in predicting the target. Additionally, some medications such as Metamorfin and Insulin were also associated with readmissions.

These findings offer actionable insights for healthcare providers. For example:

- Patients with extended hospital stays and multiple diagnoses should be closely monitored and provided with additional post discharge care.
- Service uitlization and medication history could serve as flags to prioritize patients for follow-up.

Recommendations

To integrate these insights into healthcare system:

- We could develop a patient risk dashboard that utilizes the CatBoost model to flag high-risk patient in real time.
- We could implement structured targeted plans, such as follow-up calls or telephone consultations for flagged patients
- Since the medical science is ever evolving, we could regularly retrain and update the model with new patient data to maintain its accuracy and relevance.

References

Strack, B. *et al.* (2014) 'Impact of HbA1c Measurement on Hospital Readmission Rates: Analysis of 70,000 Clinical Database Patient Records', *BioMed Research International*, 2014, pp. 1–11. Available at: https://doi.org/10.1155/2014/781670.