Diabetes Re-admission project

Summary

N, Jaganath

2024

# Data Cleaning and Preprocessing for Diabetes Dataset

## Objective

The project aims to predict the likelihood of re-admission for diabetic patients using various machine learning techniques.

## Data Loading

* **Dataset:** diabetic\_data.csv
* **Entries:** 101,766
* **Columns:** 50
* **Tool:** Pandas DataFrame (df\_diabetes)

## Data cleaning

### Identifying Missing Values

We examined the dataset to identify missing values. Specifically, we focused on the ‘max\_glu\_serum’ and ‘A1Cresult’ columns. These columns contained some missing data, which needed to be addressed.

### Calculating Missing Value Percentage

To understand the impact of missing values, we calculated the percentage of missing data for each column. This information helped us decide whether to drop the columns entirely or impute the missing values.

### Recommendations:

* **max\_glu\_serum**: Approximately 94.75% of values were missing. We suggested either dropping this column or imputing the missing values if it was clinically relevant.
* **A1Cresult**: About 83.28% of values were missing. Similar to ‘max\_glu\_serum,’ we recommended either dropping the column or imputing the missing values based on clinical context.

### ID Mappings:

* Mapped admission\_type\_id, discharge\_disposition\_id, and admission\_source\_id using IDS\_mapping.csv.
* Merged these mappings with the main dataset and dropped the original ID columns.

#### Handling Null Values

* Columns with Null Values:
  + admission\_source: 6.66%
  + admission\_type: 5.2%
  + discharge\_disposition: 3.63%
* Imputation Strategy:
  + admission\_source: Impute with “Not Available”.
  + admission\_type: Impute with “Not Available”.
  + discharge\_disposition: Impute with “Not Mapped”.

## Handling Invalid Values

### Identifying Columns with Invalid Values

We discovered that several columns contained invalid values represented by ‘?’. These columns included:

* + race: 2.23% invalid (?)
  + gender: 0.0% invalid (Unknown/Invalid)
  + weight: 96.86% invalid (?)
  + payer\_code: 39.56% invalid (?)
  + medical\_specialty: 49.08% invalid (?)
  + diag\_1: 0.02% invalid (?)
  + diag\_2: 0.35% invalid (?)
  + diag\_3: 1.4% invalid (?)

### Imputation Strategies

For each of these columns, we proposed the following strategies:

* **race**: We replaced ‘?’ with the most frequent value, ‘Caucasian,’ assuming it is a reasonable imputation.
* **payer\_code**: We replaced ‘?’ with ‘MC’ (the most frequent value).
* **medical\_specialty**: We replaced ‘?’ with ‘InternalMedicine’ (again, the most frequent value).

## Data Imputation

### Dropping Rows Based on Conditions

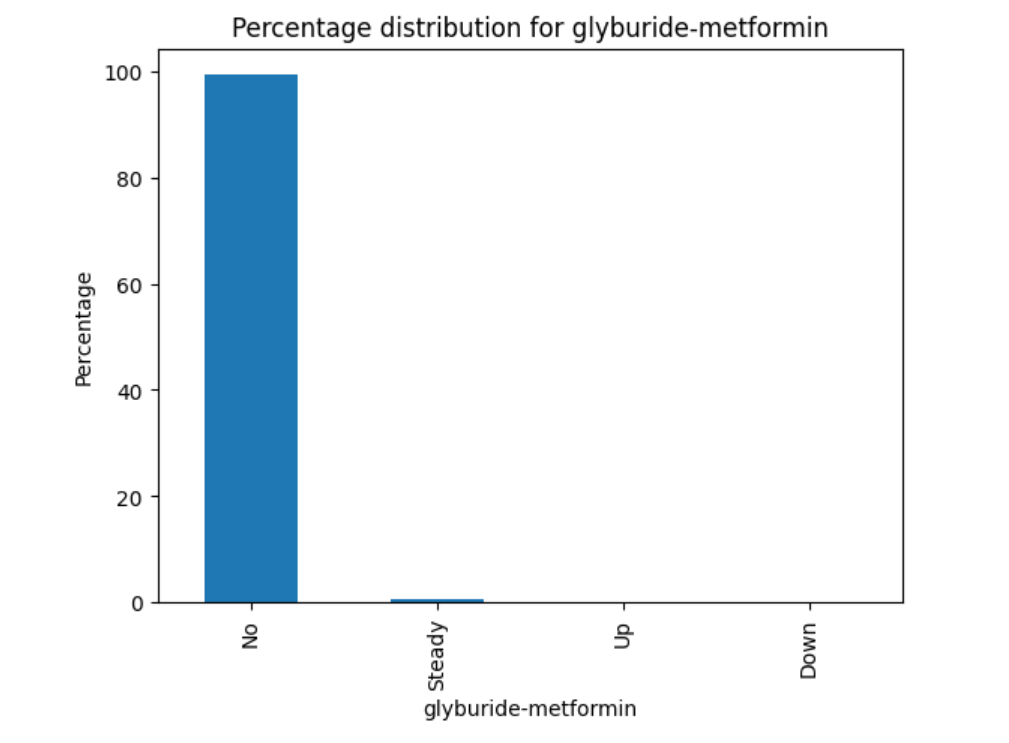
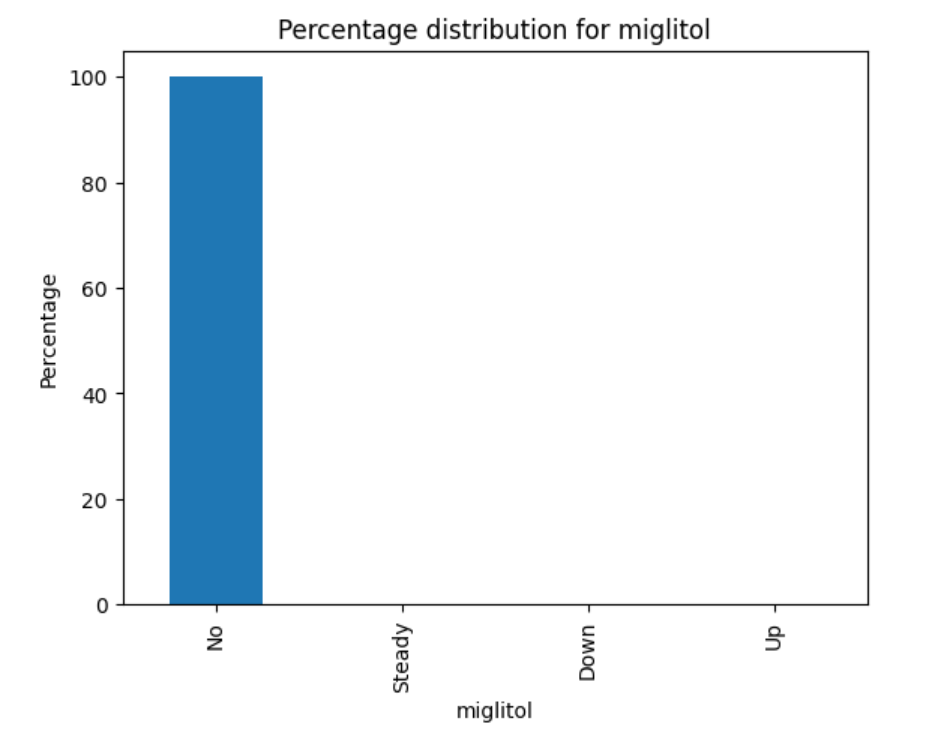
We dropped rows based on specific conditions:

* Rows with ‘Unknown/Invalid’ in the ‘gender’ column.
* Rows with missing values in the ‘diag\_1’, ‘diag\_2’, or ‘diag\_3’ columns.

## Data Validation

We verified that there were no remaining ‘?’ values or null values in the cleaned dataset. This step ensured the integrity of the data for subsequent analysis.

### Analyzing the categorical features for higher concentration of values

* **Columns with >99% Concentration in One Category:**
  + nateglinide, chlorpropamide, acetohexamide, tolbutamide, acarbose, miglitol, troglitazone, tolazamide, examide, citoglipton, glyburide-metformin, glipizide-metformin, glimepiride-pioglitazone, metformin-rosiglitazone, metformin-pioglitazone

Since the >99% concentration in one category, these columns will not help the model, hence can be dropped.

## Encoding Strategies

### Categorical Column Encoding

We considered encoding strategies for categorical columns:

* **One-Hot Encoding**: Applied to ‘race’ and ‘payer\_code’ columns.
* **Binary Encoding**: Used for ‘gender’, ‘change’, ‘diabetesMed’, and specific medications.
* **Label/Ordinal Encoding**: Applied to the ‘age’ column.

### Grouping and Encoding

* **Columns:** medical\_specialty, diag\_1, diag\_2, diag\_3, admission\_type\_id, discharge\_disposition\_id, admission\_source\_id
* **Strategy:** Group values and apply one-hot encoding or target encoding.

## Data grouping

### **ICD-9 Chapter Mapping**

* + 001-139: Infectious And Parasitic Diseases
  + 140-239: Neoplasms
  + 240-279: Endocrine, Nutritional And Metabolic Diseases, And Immunity Disorders
  + 280-289: Diseases Of The Blood And Blood-Forming Organs
  + 290-319: Mental Disorders
  + 320-389: Diseases Of The Nervous System And Sense Organs
  + 390-459: Diseases Of The Circulatory System
  + 460-519: Diseases Of The Respiratory System
  + 520-579: Diseases Of The Digestive System
  + 580-629: Diseases Of The Genitourinary System
  + 630-679: Complications Of Pregnancy, Childbirth, And The Puerperium
  + 680-709: Diseases Of The Skin And Subcutaneous Tissue
  + 710-739: Diseases Of The Musculoskeletal System And Connective Tissue
  + 740-759: Congenital Anomalies
  + 760-779: Certain Conditions Originating In The Perinatal Period
  + 780-799: Symptoms, Signs, And Ill-Defined Conditions
  + 800-999: Injury And Poisoning
  + V01-V91: Supplementary Classification Of Factors Influencing Health Status And Contact With Health Services
  + E000-E999: Supplementary Classification Of External Causes Of Injury And Poisoning

### **Medical Specialty Grouping**

* **Primary Care:** InternalMedicine, Family/GeneralPractice, Pediatrics, etc.
* **Surgery:** Surgery-General, Orthopedics, Surgery-Cardiovascular/Thoracic, etc.
* **Specialized Medicine:** Cardiology, Gastroenterology, Nephrology, etc.
* **Other Services:** Anesthesiology, Dentistry, Surgeon, etc.

### **Admission Sources Grouping**

* **Referrals:** Clinic Referral, Physician Referral, HMO Referral
* **Transfers:** Transfer from a hospital, Transfer from a Skilled Nursing Facility (SNF), etc.
* **Emergency and Law Enforcement:** Emergency Room, Court/Law Enforcement
* **Special Circumstances:** Extramural Birth, Normal Delivery, Sick Baby
* **Uncategorized/Other:** Not Mapped, Not Available, nan

### **Discharge Categories Grouping**

* **Home Discharge:** Discharged to home, Discharged/transferred to home with home health service, etc.
* **Specialized Care Facilities:** Discharged/transferred to SNF, Discharged/transferred to ICF, etc.
* **Hospitals & Rehab:** Discharged/transferred to another short-term hospital, etc.
* **Within Institution:** Discharged/transferred within this institution to Medicare approved swing bed, etc.
* **Between Institutions:** Discharged/transferred/referred to another institution for outpatient services
* **Hospice Care:** Hospice / medical facility, Expired in a medical facility. Medicaid only, hospice.
* **Expired:** Expired
* **Admissions & AMA:** Admitted as an inpatient to this hospital, Left AMA (Against Medical Advice)
* **Uncategorized:** Not Mapped, nan (Not Available/Not Applicable)

## Duplication Check

We ensured that there were no duplicated rows in the cleaned dataset.

# Cleaning and Preprocessing Summary Table

| **Column Name** | **Missing/Invalid Text** | **Count** | **% Missing** | **Action** | **Imputation** | **Reason** |
| --- | --- | --- | --- | --- | --- | --- |
| **max\_glu\_serum** | Missing | 96420 | 94.75% | Drop Column | N/A | Due to a high percentage (94.75%) of missing values, the column is dropped. If it’s a clinically significant feature, we may need SME input for possible imputation with the mode “Norm”. |
| **A1Cresult** | Missing | 84748 | 83.28% | Drop Column | N/A | With 83.28% missing, the column is dropped. Should it be clinically significant, SME input is required for potential imputation with the mode “>8”. |
| **race** | “?” | 2273 | 2.23% | Keep | “Caucasian” | The most frequent value “Caucasian” is used for imputation. |
| **gender** | “Unknown/Invalid” | 3 | 0.00% | Drop rows | N/A | Rows are dropped due to the negligible number of invalid entries (only 3). |
| **weight** | “?” | 98569 | 96.86% | Drop column | N/A | The column is dropped because of the overwhelming majority of invalid values (96%). |
| **payer\_code** | “?” | 40256 | 39.56% | Keep | “MC” | “MC” is used for imputation as the most frequent value, need SME review due to a significant amount of missing data and importance of the feature. |
| **medical\_specialty** | “?” | 49949 | 49.08% | Keep | “Internal Medicine” | “Internal Medicine” is used for imputation as the most frequent value, need SME review given the substantial missing data and feature importance. |
| **diag\_1** | “?” | 21 | 0.02% | Drop rows | N/A | Rows are dropped as diagnosis data should not be imputed to avoid misleading information. The low percentage of invalid values supports this decision. SME advice is crucial. |
| **diag\_2** | “?” | 358 | 0.35% | Drop rows | N/A | Similar to diag\_1, rows are dropped to maintain data integrity. |
| **diag\_3** | “?” | 1423 | 1.40% | Drop rows | N/A | Following the same rationale as diag\_1 and diag\_2, rows are dropped. |
| **admission\_source** | Missing | 6781 | 6.66% | Keep | “Not Available” | Considering not to bilindly impute with most frequent value, as it may false interpret the data. Advise needed on this. |
| **admission\_type** | Missing | 5291 | 5.20% | Keep | “Not Available” | Considering not to bilindly impute with most frequent value, as it may false interpret the data. Advise needed on this. |
| **discharge\_disposition** | Missing | 3691 | 3.63% | Keep | “Not Mapped” | Considering not to bilindly impute with most frequent value, as it may false interpret the data. Advise needed on this. |
| **nateglinide, chlorpropamide, acetohexamide, tolbutamide, acarbose, miglitol, troglitazone, tolazamide, examide, citoglipton, glyburide-metformin, glipizide-metformin, glimepiride-pioglitazone, metformin-rosiglitazone, metformin-pioglitazone** | Having high concentration in one category | N/A | > 99% | Drop column | N/A | Since the >99% concentration in one category, these columns will not help the model, hence can be dropped. Again SME advice is crucial. |

# Model training and evaluation

**Datasets Considered**

1. **Cleaned Dataset:** All cleaning, preprocessing, and grouping applied. Missing and invalid values handled manually.
2. **Uncleaned Dataset:** Only a few columns removed (max\_glu\_serum, A1Cresult, weight, encounter\_id, patient\_nbr). Missing and invalid values replaced with np.nan, handled by XGBoost.

**Model Considered:** XGBoost

## **Prediction with Uncleaned Dataset (Not Manually Handling Missing Values)**

* All categorical columns are one-hot encoded.
* Parameter **missing= np.nan** provided to handle null values.
* **Target Classification:** <30: **0**, >30: **1**, NO: **2**
  + Model Report (Test):
    - **Accuracy:** 0.5915
    - **Confusion Matrix:**

[[ 100 1141 1598]

[ 91 3501 5295]

[ 76 2192 11448]]

* + - **Precision:** 0.5572
    - **Recall:** 0.5915
    - **F1 Score:** 0.5478
* **Target Classification:** <30: **0**, >30: **0**, NO: **1**
  + Model Report (Test):
    - **Accuracy:** 0.6477
    - **Confusion Matrix:**

[[ 6373 5353]

[ 3611 10105]]

* + - **Precision:** 0.6466
    - **Recall:** 0.6477
    - **F1 Score:** 0.6441

## Prediction with cleaned dataset

* All categorical columns are one-hot encoded.
* In cleaning and preprocessing, may columns were removed and all the missing and invalid values are handled manually.
* **Target classification: '<30':0, '>30':0, 'NO':1**
  + Model Report (Test)
  + **Accuracy**: 0.6414
  + **Confusion Matrix**:

[[6272 5334]

[3652 9803]]

* + **Precision**: 0.6403
  + **Recall**: 0.6414
  + **F1 Score**: 0.6379

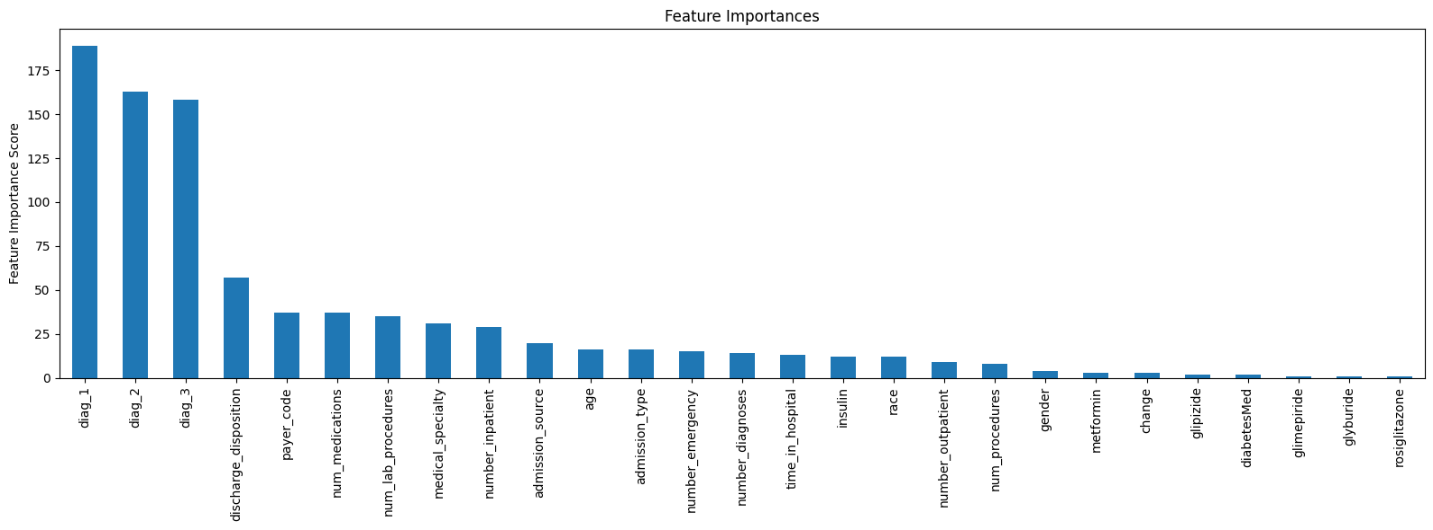
## Prediction without One-hot-encoding - Dataset not cleaned

* Applying categorical features as it is, missing and invalid values retained as NaN.
* Parameter **missing= np.nan** provided to handle null values.
* Parameter **enable\_categorical=True** provided to handle categorical values.
* **Target classification: '<30':0, '>30':0, 'NO':1**
  + Model Report (Test)
  + **Accuracy**: 0.6339
  + **Confusion Matrix**:

[[ 5693 6033]

[ 3281 10435]]

* + **Precision**: 0.6339
  + **Recall**: 0.6339
  + **F1 Score**: 0.6262



## Prediction with cleaned dataset, but without One hot encoding

* **Target classification: '<30':0, '>30':0, 'NO':1**

Model Report (Test)

**Accuracy**: 0.6427

**Confusion Matrix**:

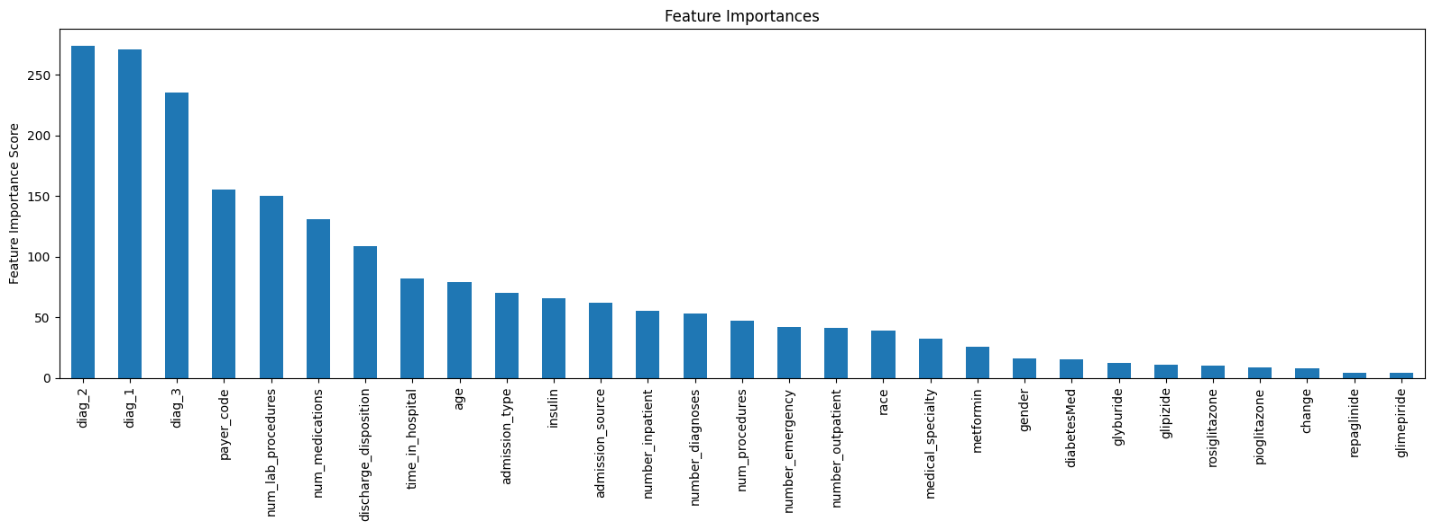
[[6298 5308]

[3646 9809]]

**Precision**: 0.6416

**Recall**: 0.6427

**F1 Score**: 0.6393



## Prediction with Cleaned dataset with Cross Validation

* **Target classification: '<30':0, '>30':0, 'NO':1**

Model Report (Train)

**Accuracy**: 0.6766

**Confusion Matrix:**

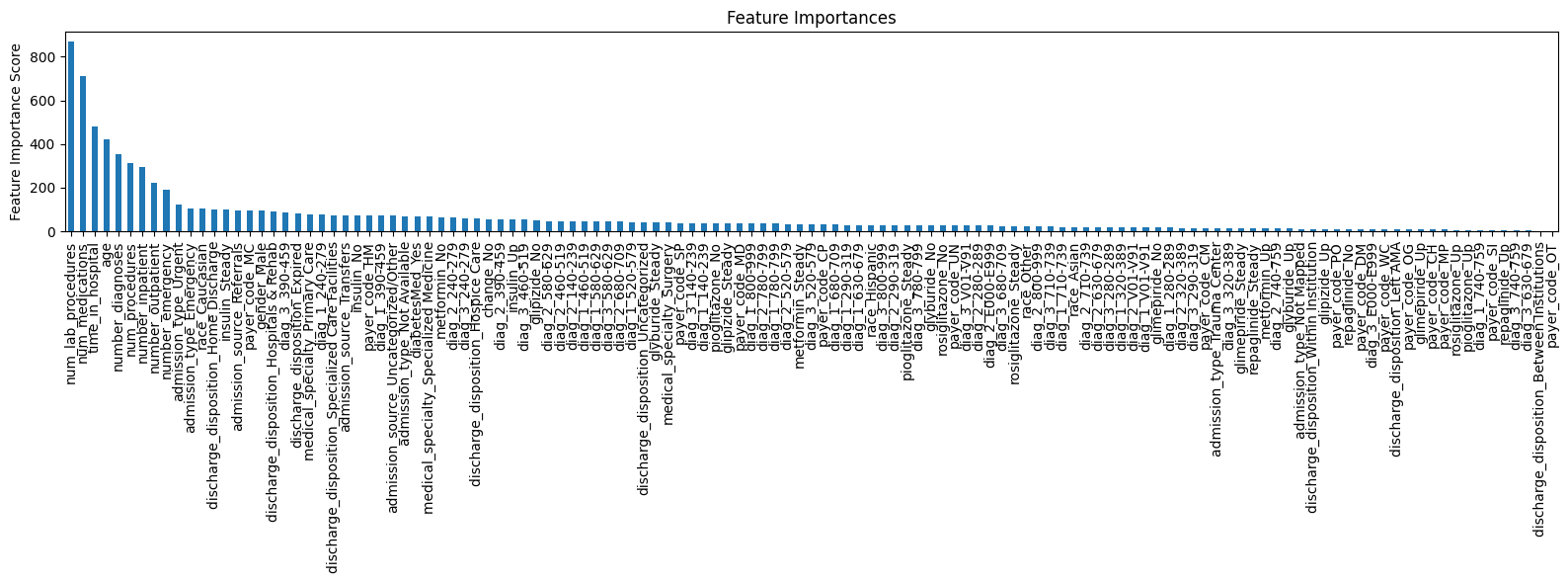
[[27193 19230]

[13182 40636]]

**Precision**: 0.6763

**Recall**: 0.6766

**F1 Score**: 0.6739



## Prediction with cross validation, but with dataset not cleaned

* **Target classification: '<30':0, '>30':0, 'NO':1**

Model Report (Train)

**Accuracy**: 0.6884

**Confusion Matrix**:

[[27916 18986]

[12724 42140]]

**Precision**: 0.6882

**Recall**: 0.6884

**F1 Score**: 0.6856

# Prediction Summary Table

| **Aspect** | **Uncleaned Dataset (One-hot Encoded)** | **Uncleaned Dataset (No One-hot Encoding)** | **Cleaned Dataset (One-hot Encoded)** | **Cleaned Dataset (No One-hot Encoding)** | **Cleaned Dataset (Cross Validation)** | **Uncleaned Dataset (Cross Validation)** |
| --- | --- | --- | --- | --- | --- | --- |
| **Categorical Handling** | One-hot encoded | Categorical features as is | One-hot encoded | Categorical features as is | One-hot encoded | Categorical features as is |
| **Missing Values Handling** | np.nan | np.nan | Manually handled | Manually handled | Manually handled | np.nan |
| **Target Classification** | **<30: 0, >30: 1, NO: 2**  **<30: 0, >30: 0, NO: 1** | <30: 0, >30: 0, NO: 1 | <30: 0, >30: 0, NO: 1 | <30: 0, >30: 0, NO: 1 | <30: 0, >30: 0, NO: 1 | <30: 0, >30: 0, NO: 1 |
| **Accuracy** | 0.5915  0.6477 | 0.6339 | 0.6414 | 0.6427 | 0.6767 | 0.6884 |
| **Confusion Matrix** | [[ 100 1141 1598] [ 91 3501 5295] [ 76 2192 11448]]  [[ 6373 5353] [ 3611 10105]] | [[ 5693 6033] [ 3281 10435]] | [[6272 5334] [3652 9803]] | [[6298 5308] [3646 9809]] | [[27193 19230] [13182 40636]] | [[27916 18986] [12724 42140]] |
| **Precision** | 0.5572  0.6466 | 0.6340 | 0.6403 | 0.6417 | 0.6763 | 0.6882 |
| **Recall** | 0.5915  0.6477 | 0.6339 | 0.6414 | 0.6427 | 0.6767 | 0.6884 |
| **F1 Score** | 0.5478  0.6441 | 0.6263 | 0.6379 | 0.6393 | 0.6740 | 0.6857 |