# Forecasting Hospital Stay Durations for Patients Using Electronic Health Records

- Tushar Jayendra Mhatre (113647237)

- Rami Reddy Yekkanti (113582522)

Instructor: Dr. Talayeh Razzaghi

Term: Spring 2024

# Table of Contents

## 1.0 Introduction and Problem Statement:

The prediction of hospital length of stay (LOS) is crucial for effective hospital resource management and patient care planning. Leveraging patient treatment data, machine learning models can provide accurate forecasts, aiding in the optimization of hospital operations and enhancing patient satisfaction. Accurate predictions of LOS help hospitals manage their resources more effectively. It enables better planning for bed occupancy, staffing, equipment, and other resources, reducing the chances of overutilization or underutilization. Hospitals can also improve their financial planning by predicting LOS. Accurate predictions can help in estimating the costs associated with patient care and in optimizing revenue management. With the advent of electronic health records (EHRs) and advances in machine learning, there is a growing interest in developing predictive models based on patient treatment data to forecast LOS.

## 2.0 Previous Related Work:

There were numerous studies conducted to predict patient's Length of stay. One of the papers we went through was 'A systematic review of the prediction of hospital length of stay: Towards a unified framework', published in PLOS digital Health Journal. One gap in most of these studies was that a lot of them didn't include the nursing admission and interaction data. Nurses, who spend significantly more time with patients compared to physicians, play a crucial role in patient care and monitoring. The dataset incorporates the data regarding the number of interactions of the patient with the nursing staff as well.

'Hospital Length of Stay Prediction Methods: A Systematic Review', another paper we went through, but this research used data of merely 5040 patients. utilizing a dataset of over 58,000 individuals, significantly expanding the scale compared to previous studies, where datasets typically comprised only a few thousand patients. This substantial increase in data volume enhances the robustness and validity of our predictive models.

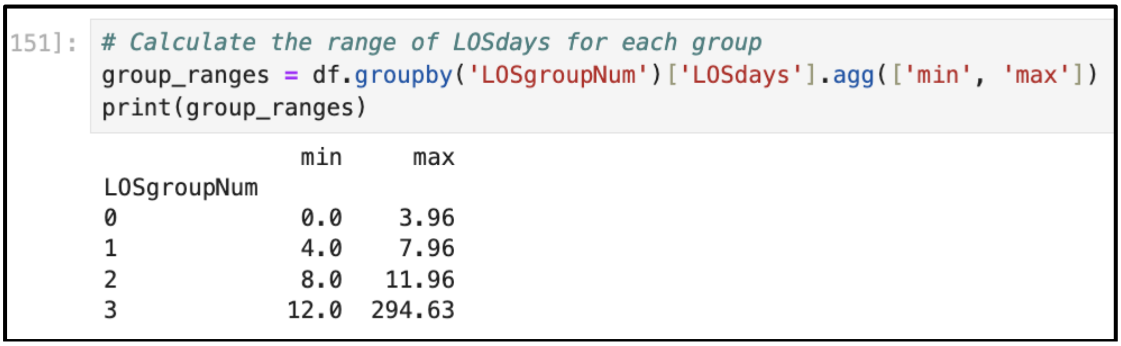
## 3.0 Dataset Description:

The dataset comes from the Multiparameter Intelligent Monitoring in Intensive Care (MIMIC-III) database, which contains comprehensive patient information" The dataset includes a wide array of features related to treatment data for intensive care patients, such as: Demographics: Age, gender, and other personal information. Lab Results: Blood tests, imaging results, and other diagnostics. Treatments: Medications, procedures, and interventions. Vital Signs: Heart rate, blood pressure, temperature, etc.

Our target variable is the Length of Stay (LOS) group number, which categorizes patients into different stay durations.

| Variables | Description |
|---|---|
| hadm_id | Unique identifier for each hospital admission. |
| gender | Patient's gender (e.g., Male/Female). |
| age | Patient's age at the time of admission. |
| LOSdays | Length of stay in days. |
| admit type | Type of admission (e.g., emergency, elective). |
| admit location | Location from which the patient was admitted (e.g., clinic referral). |
| admit diagnosis | Primary diagnosis at admission. |
| insurance | Type of insurance coverage (e.g., Private, Medicare). |
| religion | Patient's religious affiliation. |
| marital status | Patient's marital status. |
| ethnicity | Patient's ethnicity or population group. |
| num callouts | Number of medical consultations requested. |
| num diagnosis | Number of diagnoses recorded. |
| num procs | Number of procedures performed. |
| admit procedure | Primary procedure at admission. |
| num cpt events | Number of CPT-coded events (Common Procedural Terminology). |
| num input | Number of inputs administered (fluids, nutrition). |
| num labs | Number of lab tests performed. |
| num micro-labs | Number of microbiology lab tests conducted. |
| num notes | Number of clinical notes recorded. |
| num output | Number of outputs recorded (urine, etc.). |
| num rx | Number of medication prescriptions. |
| num proc events | Number of procedure events. |
| num transfers | Number of transfers to different hospital units. |
| num chart events | Number of chart events documented. |
| expired hospital | Whether the patient died in the hospital (yes/no). |
| total num interact | Total number of interactions documented. |
| LOSgroup Num | Length of stay grouped into specific categories. |

For LOSGroup NUM, the screenshot below indicates how many days of stay each group in the target variable correspond to.



```
151]: # Calculate the range of LOSdays for each group
      group_ranges = df.groupby('LOSgroupNum')['LOSdays'].agg(['min', 'max'])
      print(group_ranges)
```

```
              min     max
LOSgroupNum
0             0.0    3.96
1             4.0    7.96
2             8.0   11.96
3            12.0  294.63
```

## 4.0 Project Workflow:

I. Data Preprocessing: We began by ensuring the data is clean and ready for analysis. We handled missing values, removed outliers, and standardized features for consistency across all variables.

II. Data Splitting Strategy: Next, we divided the dataset into training and testing sets, with 80% allocated for training and 20% reserved for testing. This division allowed us to train the models effectively while ensuring accurate and unbiased evaluation.

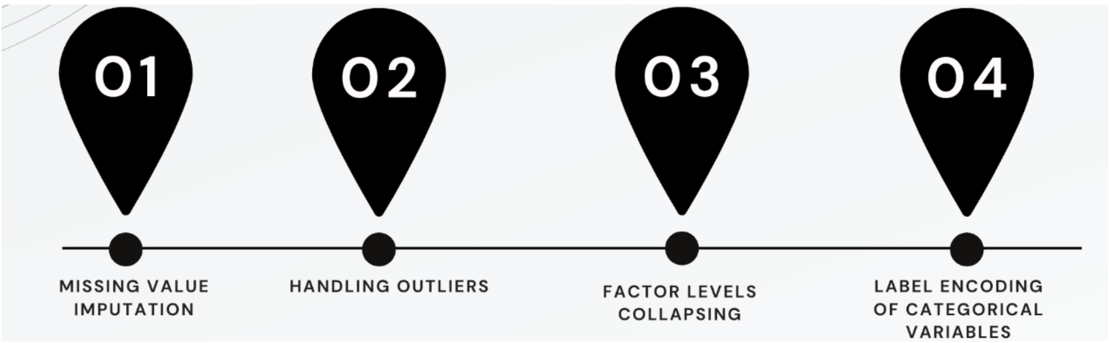
III. Feature Importance Analysis: We then analyzed the data to identify which features had the most significant impact on predicting the length of hospital stay. Finding these key variables helped us focus on what matters most during model development.

IV. Model Building: After cleaning the data and identifying important features, we built several machines learning models, including logistic regression, random forest, and support vector machines. We trained these models using the training data to prepare them for accurate predictions.

V. Performance Metrics: To assess model performance, we used several metrics such as Accuracy, Precision, Recall, and F1 Score.

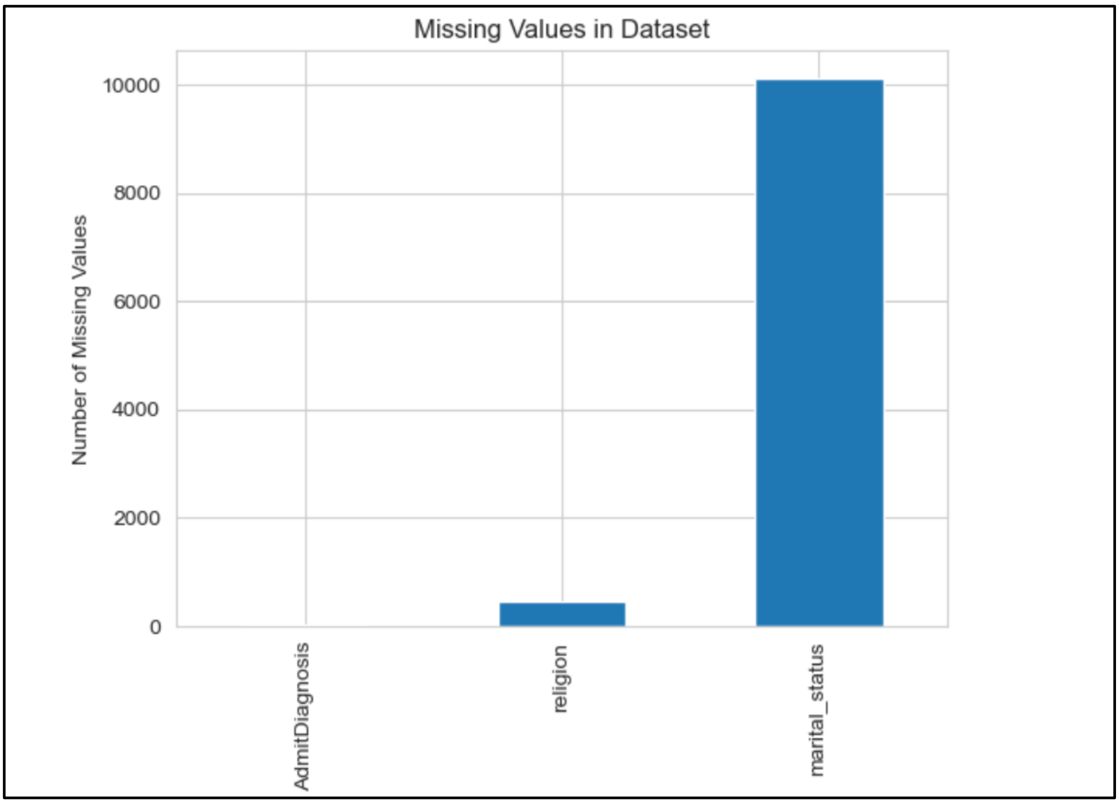
Model Comparison: Finally, we compared the performance of different models using these metrics and visualization. This comparison allowed us to identify which model was most accurate.

## 5.0 Data Preprocessing Steps:

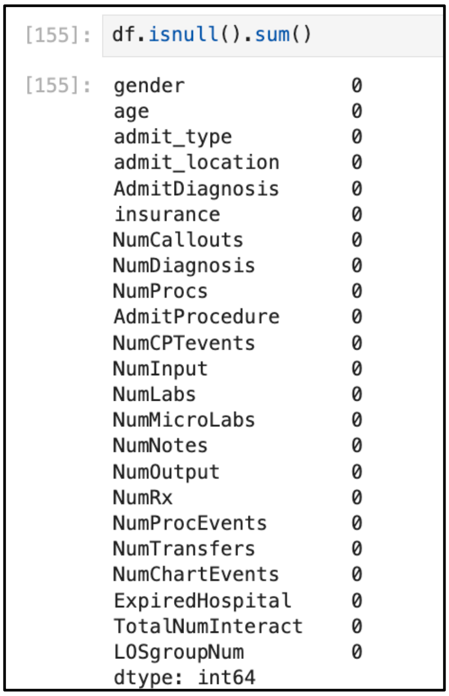


## 5.1 Handling Missing Data:

We found that only 3 variables among the 27 presents in this data set had missing values. Below bar plot showcases the number of missing values for each feature.



To handle the missing values, we decided to drop the columns religion and marital_status as we believed these demographic variables are not important to the analysis of the data. The missing values in Variable AdmitDiagnosis were removed by performing **'mode imputation'**. By taking these steps we manage to eliminate the missing values.

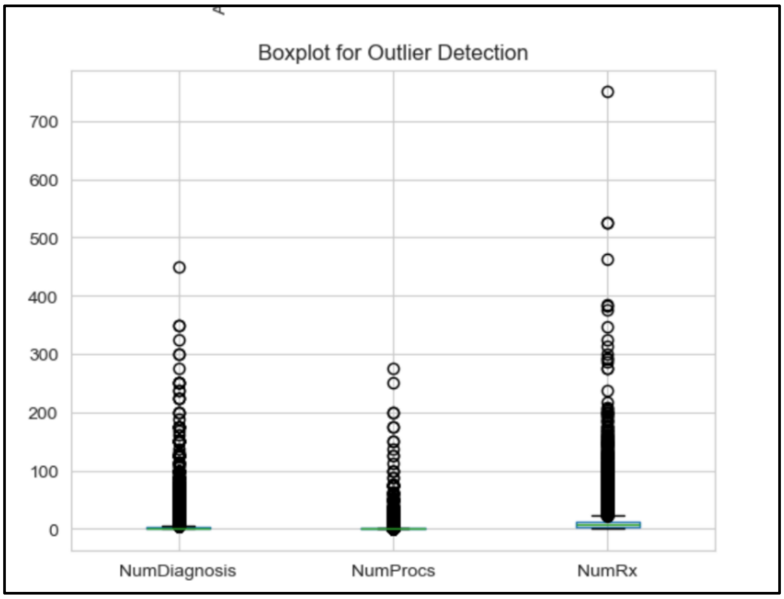


```
[155]:  df.isnull().sum()

[155]:  gender              0
        age                 0
        admit_type          0
        admit_location      0
        AdmitDiagnosis      0
        insurance           0
        NumCallouts         0
        NumDiagnosis        0
        NumProcs            0
        AdmitProcedure      0
        NumCPTevents        0
        NumInput            0
        NumLabs             0
        NumMicroLabs        0
        NumNotes            0
        NumOutput           0
        NumRx               0
        NumProcEvents       0
        NumTransfers        0
        NumChartEvents      0
        ExpiredHospital     0
        TotalNumInteract    0
        LOSgroupNum         0
        dtype: int64
```

## 5.2 Handling outliers:

The below boxplot showcases the present of outliers with extreme values for variables **‘NumDiagnosis’, ’NumProcs’, ’NumRx’.**

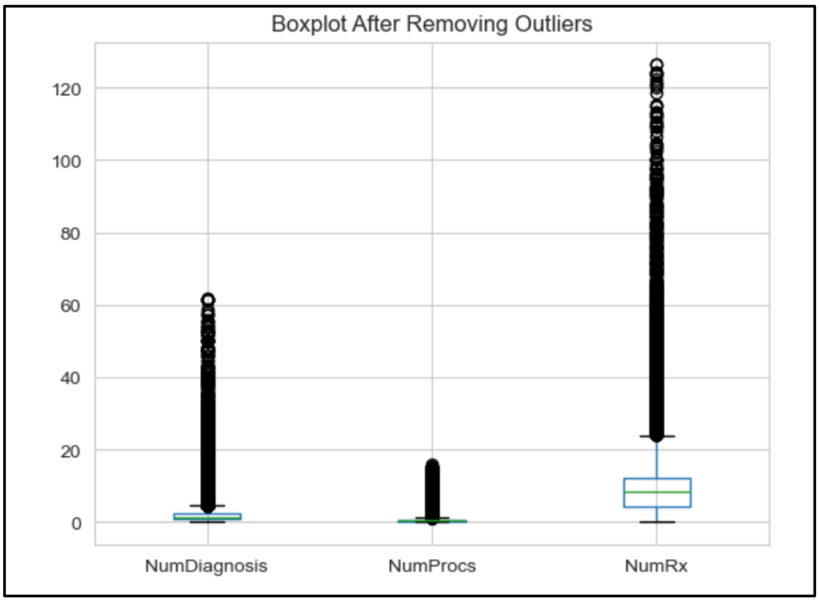


We calculated the 1st and 99th percentiles (Q1 and Q3, respectively) for each column. This choice of percentiles was made as we aimed to remove the most extreme 2% of the data while retaining the bulk of the dataset. The IQR is the difference between Q3 and Q1 and it is used to measure the statistical dispersion in the data. After the removal of Outliers, we are left with 58648 records in the dataset.

```
df=no_outliers_df
len(df)
```

```
58648
```

Here’s the boxplot after removal of outlier.



The Y-axis scale is now reduced to 0-120 from 0-700 as the extreme values lying beyond this range are now removed.

## 5.3 Factor Collapsing:

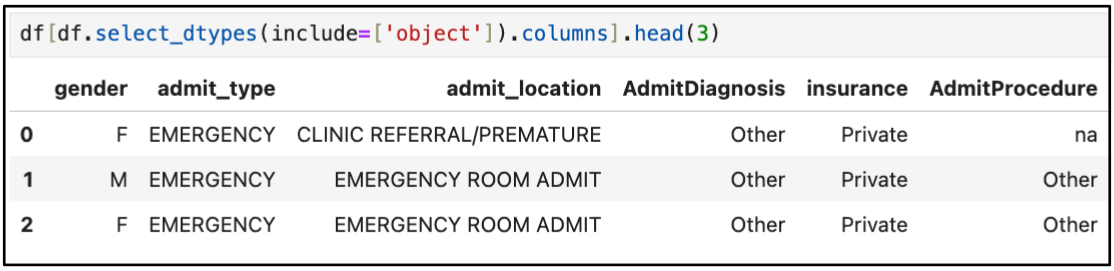
'AdmitDiagnosis' and 'AdmitProcedure' are two categorical columns which contain many levels, 15,644 and 1,276 respectively. We dropped down the number of levels to 6 for both variables by focusing on the most prevalent categories and grouping the rest as 'Other'. This approach helped to streamline the analysis and enhance the performance of machine learning models by reducing complexity of the data.

## 5.4 Label Encoding of categorical variables:

We have used **Label Encoder** from **sklearn.preprocessing** to perform this step. The encoding is applied to all identified categorical columns using the **apply()** function. This function iterates over each categorical column, applying the **fit_transform()** method of label_encode.

The following columns underwent the label Encoding **('gender', 'admit_type', 'admit_location', 'AdmitDiagnosis', 'insurance', and 'AdmitProcedure').**

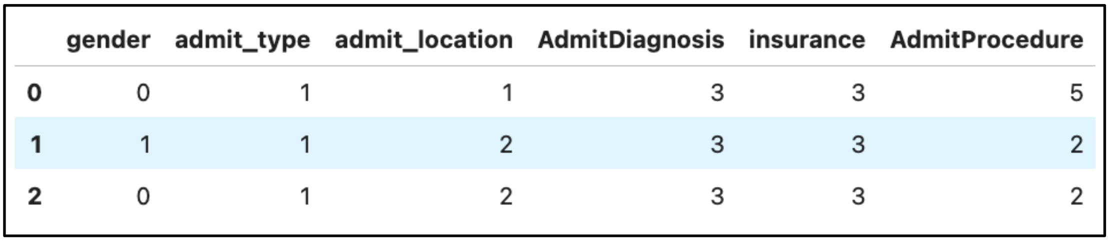
Before Label Encoding:



```
df[df.select_dtypes(include=['object']).columns].head(3)
```

|   | gender | admit_type | admit_location | AdmitDiagnosis | insurance | AdmitProcedure |
|---|---|---|---|---|---|---|
| **0** | F | EMERGENCY | CLINIC REFERRAL/PREMATURE | Other | Private | na |
| **1** | M | EMERGENCY | EMERGENCY ROOM ADMIT | Other | Private | Other |
| **2** | F | EMERGENCY | EMERGENCY ROOM ADMIT | Other | Private | Other |

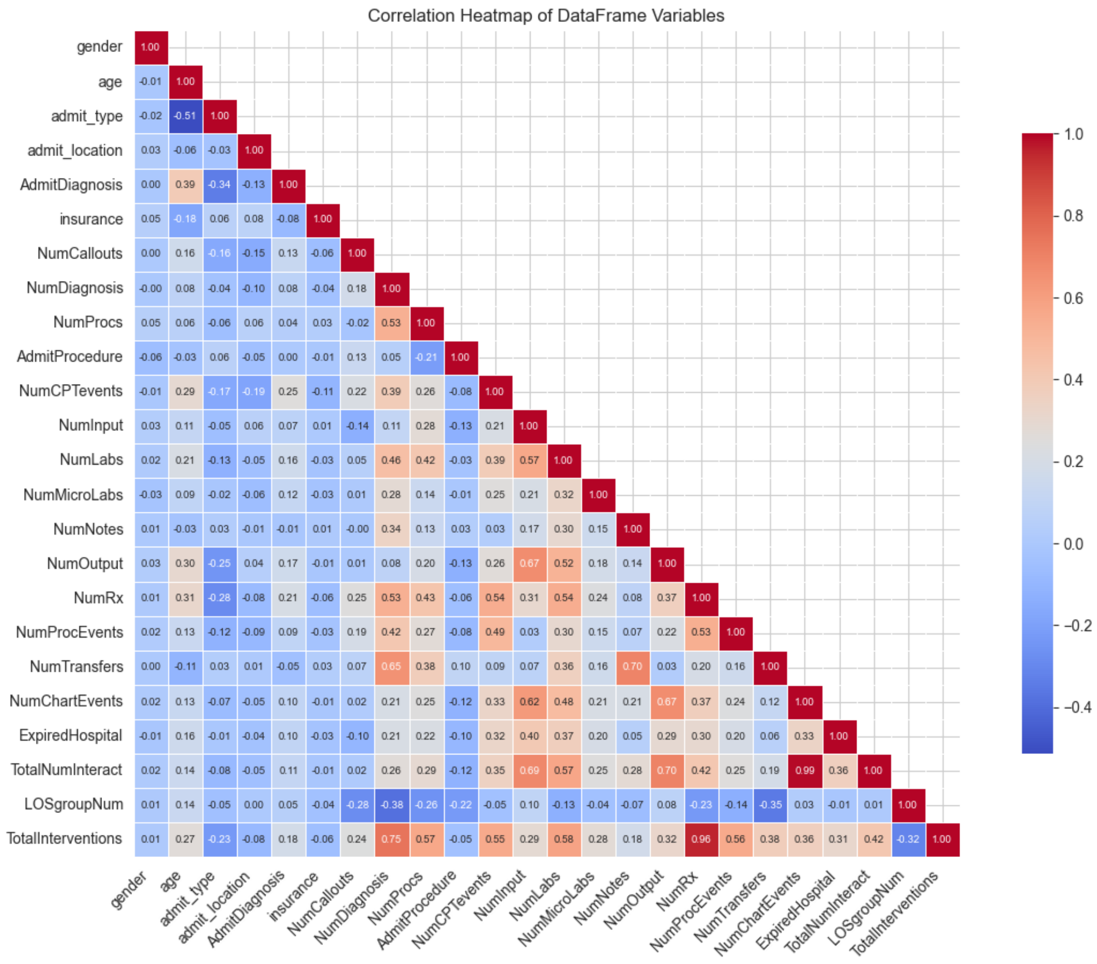
After Label Encoding:



|   | gender | admit_type | admit_location | AdmitDiagnosis | insurance | AdmitProcedure |
|---|---|---|---|---|---|---|
| **0** | 0 | 1 | 1 | 3 | 3 | 5 |
| **1** | 1 | 1 | 2 | 3 | 3 | 2 |
| **2** | 0 | 1 | 2 | 3 | 3 | 2 |

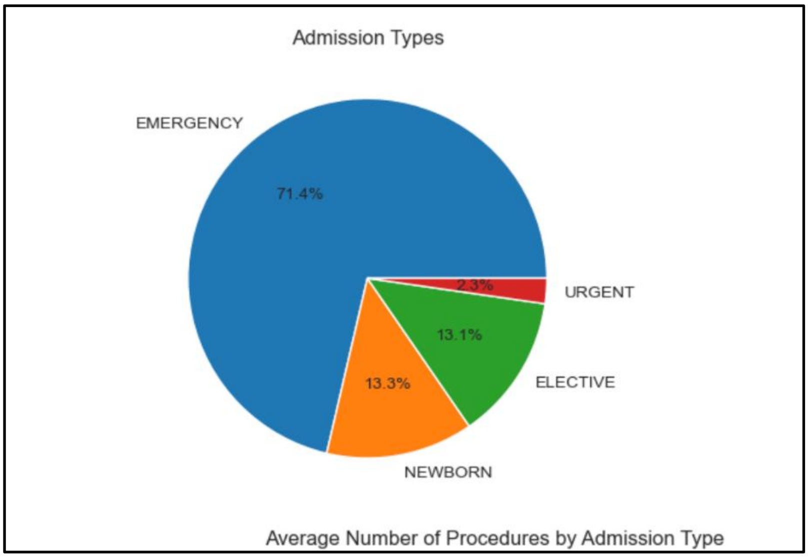
# 6.0 Exploratory Data Analysis:

## 6.1 Correlation Matrix:



First, we have created a Correlation heatmap, which visually represents the correlation coefficients between pairs of variables in a hospital dataset. This matrix quantifies the degree of relationship between all pairs of variables in the dataset.

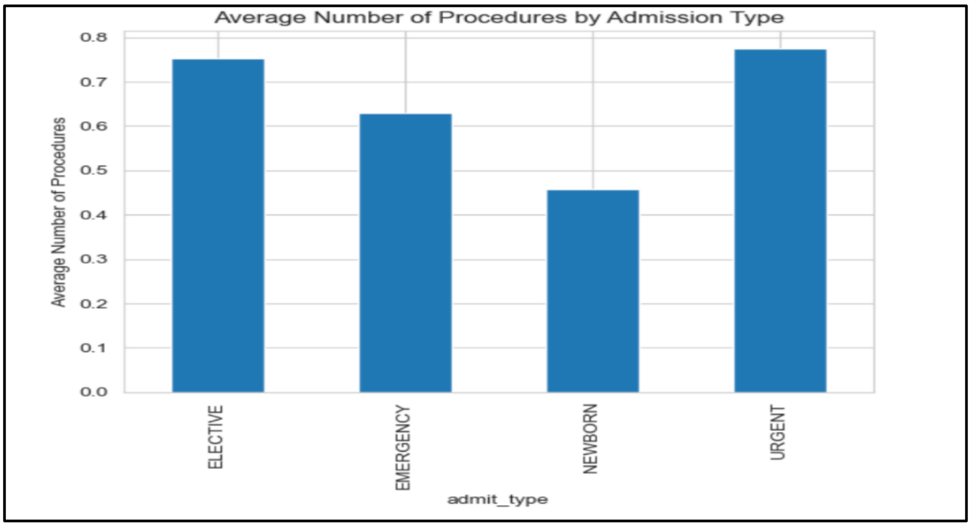
## 6.2 Pie Chart for Admission Type Counts:



This chart offers a straightforward representation of the proportion of each admission type — Emergency, Urgent, Elective, and Newborn.

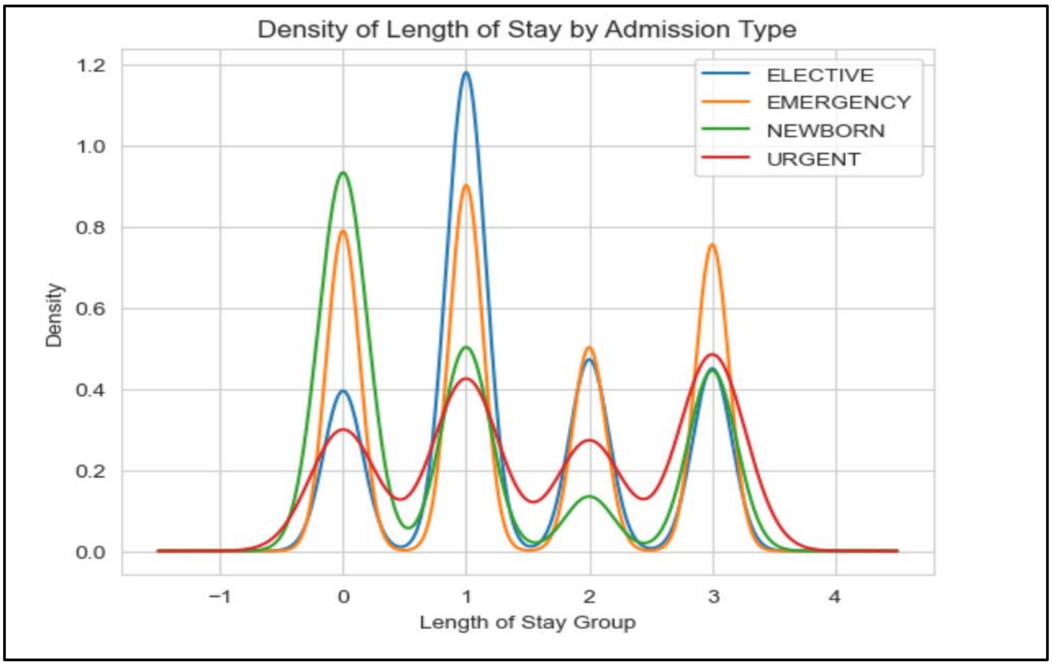
- **Emergency (71.4%)**: Represents most admissions, indicating a high volume of emergency cases relative to other types.
- **Urgent (13.3%)** and **Elective (13.1%)** are the next two categories representing significant but smaller portions of the admissions.
- **Newborn (2.3%)**: is the smallest category among the types.

## 6.3 Average Number of Procedures by Admission Type:



The above figure represents a bar chart to visualize the average number of medical procedures performed per admission type in a hospital dataset. This analysis is crucial for understanding how resource utilization varies with different types of admissions and can assist in optimizing hospital operations and improving patient care. As we can see from the graph, Elective and Urgent procedures are the most resource intensive operation followed by emergency and then finally Newborn.

## 6.4 Density plot for LOS across different Admission Types:

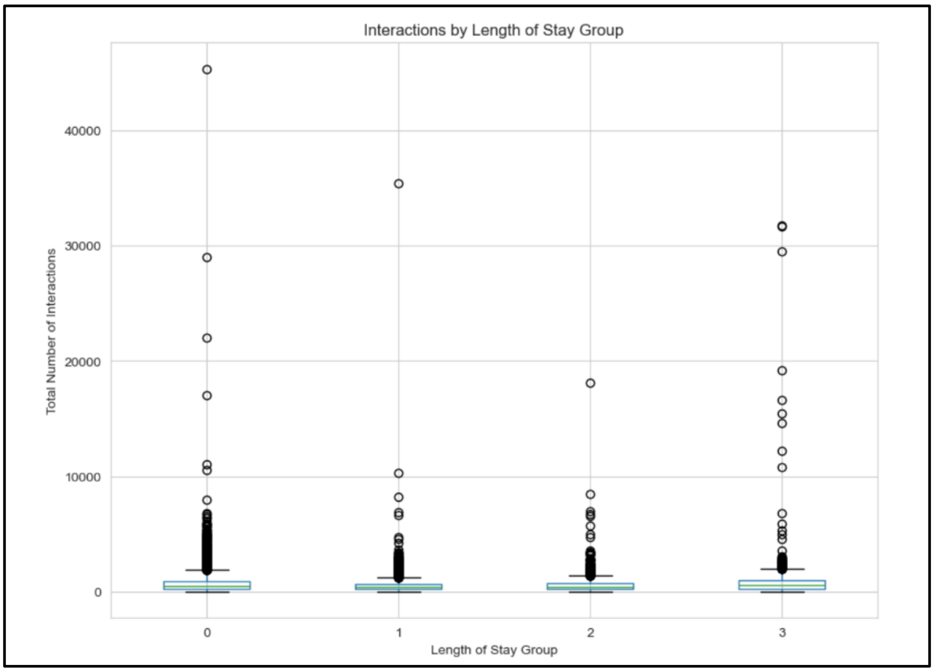


We have plotted a density plot which visualizes the distribution of hospital length of stay (LOS) across different types of admissions: Elective, Emergency, Newborn, and Urgent. This plot is useful for understanding how the length of stay varies depending on the admission type.

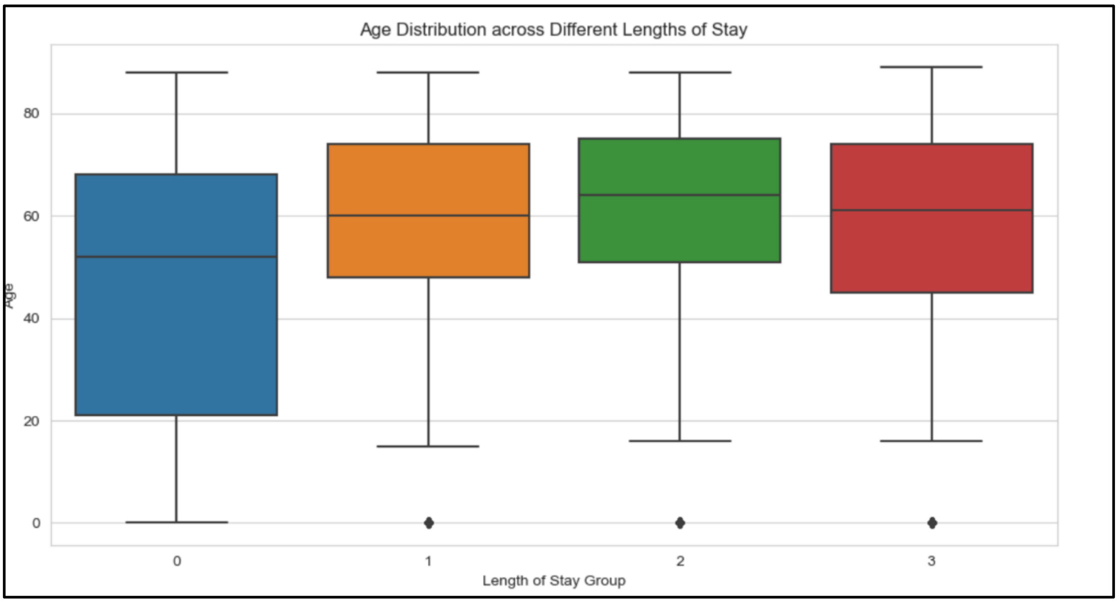
- For Elective admissions the density peaks around shorter length of stay groups, suggesting that elective admissions typically have shorter stays.
- For Emergency admissions there is a broader spread across multiple groups, indicating variability in the length of stay, but with significant peaks at groups 1 and 3 which reflects the unpredictable nature of emergency admissions.
- The peak for newborns is notably sharp and high at group 1, indicating a highly consistent and relatively short hospital stay for newborn admissions.
- Like emergency admissions, urgent admissions show peaks at groups 1 and 3, with a spread indicating variability but a generally longer stay than elective admissions.

## 6.5 Box plot for Interactions by Length of Stay Group:

Below plot visualizes the distribution of total interactions across different length of stay (LOS) groups. This plot helps us to understand how patient engagement varied throughout the period of stay.



## 6.6 Age Distribution across Different Lengths of Stay:



The box plot showcases age distribution with respect to the length of stay (LOS) it indicates that younger patients generally have shorter hospital stays (Group 0). There is a clear trend of increasing median age as the length of stay increases. Groups

1, 2, and particularly 3 show progressively higher median ages and broader age ranges. This Indicates that older adults frequently require longer hospital stays.

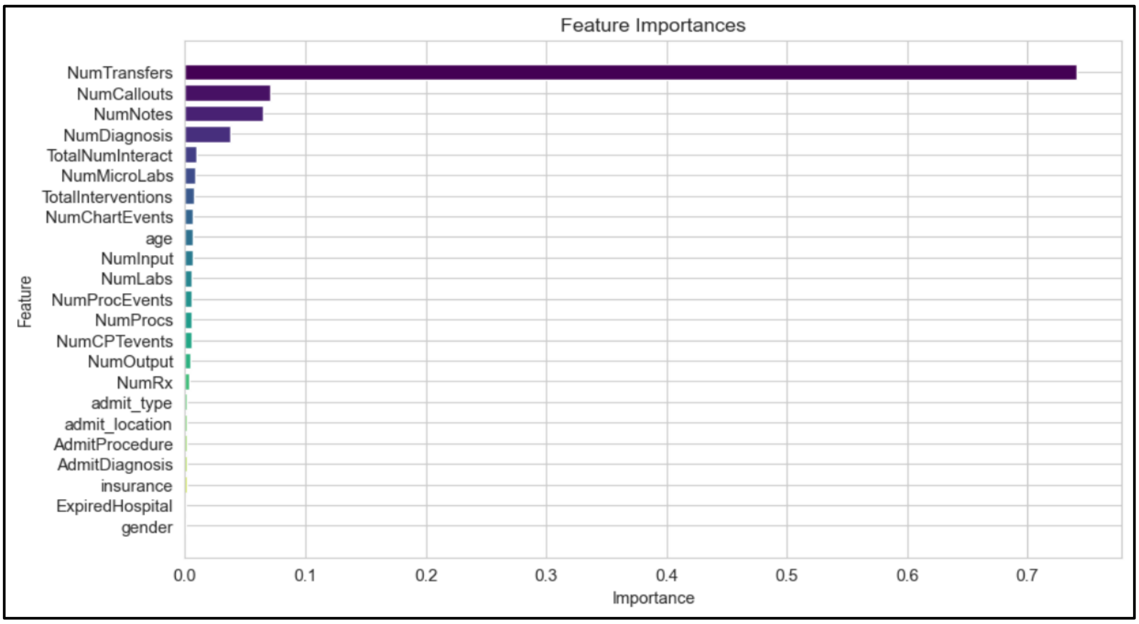
## 7. Modeling Approach:

After preprocessing the dataset to ensure its quality and suitability for analysis, we embarked on the modeling phase with a structured approach aimed at deriving meaningful insights and predictive accuracy. The methodology employed can be broken down into several key steps:

### 7.1 Data Preprocessing:

- The initial step involved was the removal of outliers from the dataset to enhance the robustness of our analysis and prevent skewed results.
- We examined the dataset for missing values, strategizing their treatment based on the percentage of missingness in each column. Columns with insignificant missing values and columns which are deemed irrelevant to the study were dropped to streamline the analysis.
- Categorical variables were encoded to numerical format using label encoding, facilitating their incorporation into machine learning algorithms.

### 7.2 Feature Selection:



- The dataset was partitioned into two subsets, the predictors, and the target variable, with LOSgroupNum serving as the focal point for prediction, while other features acted as predictors.
- Leveraging the Random Forest algorithm, we conducted a comprehensive assessment of feature importance.
- Then using the feature importance, we have selected the top 16 predictors that exhibited a substantial impact on predicting the target variable and stored them as the final predictors for the analysis.
- This step was pivotal in our analysis to improve the predictive accuracy of the target variable.
- The below plot showcases the most important features plotted along with their scores.

## 7.3 Data Splitting:

The preprocessed dataset was split into training and testing subsets in an 80/20 ratio, ensuring a balance between model training and evaluation on unseen data.

## 7.4 Model Selection:

To explore a spectrum of predictive capabilities and finding the most suitable algorithm for our dataset, we employed five distinct machine learning models:

- XGBoost
- Random Forest
- Decision Tree
- K-Nearest Neighbors (KNN)
- Support Vector Machine (SVM)

## 7.5 Evaluation:

Each model underwent rigorous evaluation using appropriate performance metrics such as accuracy, precision, recall, f1-score, and many more to gauge its efficacy in predicting the target variable.

## 7.6 XGBoost:

XGBoost is a powerful algorithm known for its efficiency in handling complex datasets and delivering high predictive performance. Its ability to handle missing values, feature importance analysis, and inherent regularization techniques make it a favored choice in various data science applications. Its multi-class classification objective aligns well with our target variable, enabling precise prediction across different LOSgroupNum categories.

## 7.7 Random Forest:

Random Forest is an ensemble learning technique that combines multiple decision trees to improve predictive accuracy and mitigate overfitting. It stands out for its versatility in handling high-dimensional data, providing insights into feature importance, which makes it well-suited for identifying significant predictors in the dataset.

The utilization of Random Forest complements our modeling strategy by harnessing ensemble learning techniques to aggregate predictions from multiple decision trees. By doing so, we gain valuable insights into feature importance, aiding in the identification of significant predictors for LOSgroupNum classification. The inherent robustness of Random Forest against overfitting aligns with our goal of constructing reliable predictive models.

## 7.8 Decision Tree:

Decision Tree serves as a fundamental model for its interpretability and simplicity, offering insights into the underlying decision-making process.

Despite its simplicity, Decision Tree provides a baseline for comparison and facilitates a clear understanding of the relationships between predictor variables and the target variable. By incorporating Decision Tree into our model selection, we ensure transparency in model interpretation, enabling everyone to grasp the factors driving LOSgroupNum classification decisions.

## 7.9 K-Nearest Neighbors (KNN):

KNN is valued for its intuitive approach to classification based on similarity, making it suitable for scenarios where instances exhibit local proximity in feature space.

Inclusion of KNN diversifies our modeling by leveraging the principle of proximity-based classification. By considering the neighbors characteristics, KNN offers a straightforward yet effective approach to predicting LOSgroupNum, particularly in scenarios where instances share similar feature patterns. The choice of K=8 for the number of neighbors balances between capturing local patterns and maintaining model generalizability.

## 7.10 Support Vector Machine (SVM):

SVM emerges as a powerful model capable of handling both linear and nonlinear classification tasks through margin maximization.

By incorporating SVM into our modeling, we tap into its ability to identify the optimal hyperplane that maximizes the margin between different classes, thereby enhancing classification accuracy. SVM's versatility in handling diverse data distributions aligns with our objective of constructing a robust predictive model for LOSgroupNum classification, accommodating both linearly separable and complex decision boundaries.

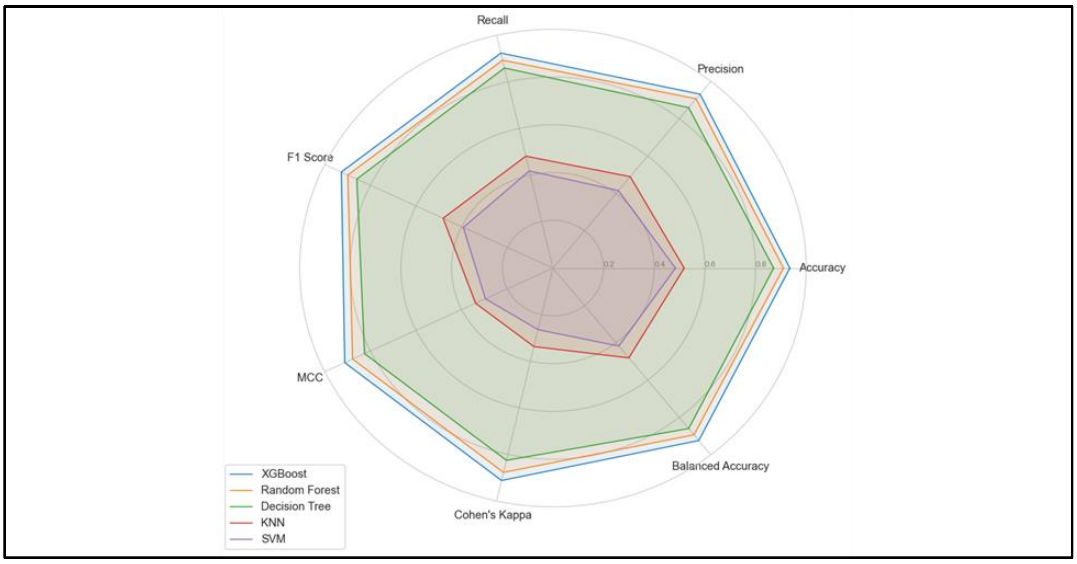
## 8.0 Results:

| Model | Accuracy | Precision | Recall | F1 Score | MCC | Kappa | Balanced Accuracy |
|---|---|---|---|---|---|---|---|
| XGBoost | 0.93 | 0.93 | 0.92 | 0.92 | 0.91 | 0.91 | 0.92 |
| Random Forest | 0.91 | 0.90 | 0.89 | 0.89 | 0.87 | 0.87 | 0.89 |
| Decision Tree | 0.87 | 0.85 | 0.85 | 0.85 | 0.82 | 0.82 | 0.85 |
| KNN | 0.51 | 0.48 | 0.48 | 0.48 | 0.33 | 0.33 | 0.48 |
| SVM | 0.48 | 0.41 | 0.41 | 0.39 | 0.29 | 0.26 | 0.41 |

Based on the evaluation metrics, the XGBoost model emerges as the best-performing model for predicting the LOSgroupNum variable. Several factors contribute to this selection:

- XGBoost achieves the highest accuracy score of 93.55%, indicating its ability to correctly classify instances into the respective LOSgroupNum categories with precision.
- The model demonstrates robust performance across multiple metrics, including precision, recall, F1-score, Matthews Correlation Coefficient (MCC), and Cohen's Kappa. This balanced performance suggests that XGBoost maintains consistency in predictive accuracy across different evaluation dimensions.
- XGBoost exhibits high values for MCC (91.22%) and Cohen's Kappa (91.21%), indicating substantial agreement between predicted and actual LOSgroupNum classifications. This underscores the model's reliability and efficacy in capturing underlying patterns within the dataset.
- With a balanced accuracy score of 92.36%, XGBoost effectively accounts for class imbalances in the dataset, ensuring accurate predictions across all LOSgroupNum categories.
- Leveraging the power of ensemble learning, XGBoost combines the strengths of multiple decision trees to enhance predictive accuracy while mitigating overfitting. This ensemble approach contributes to the model's robustness and generalizability.

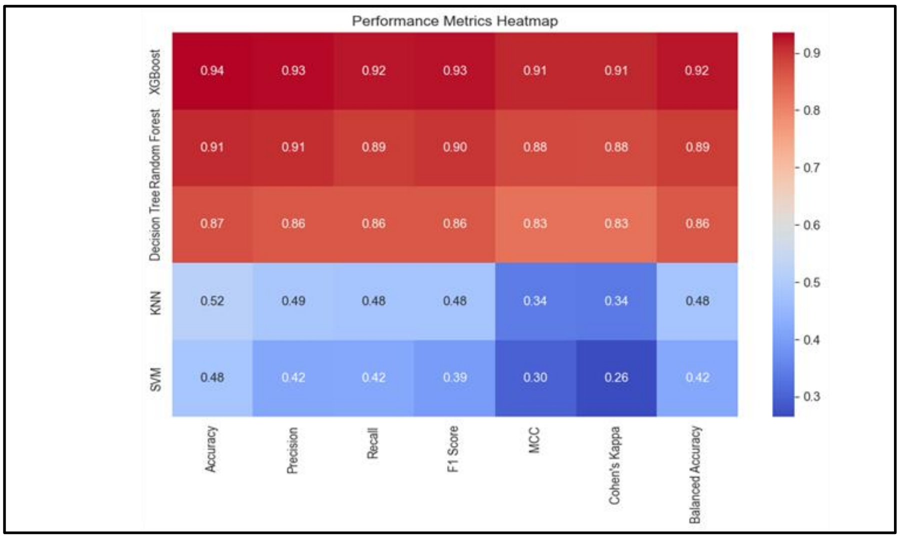
## 8.1 Radar Plot Comparing Performance of the ML Models:



This radar plot, also known as a spider chart, compares the performance of five machine learning models across various evaluation metrics. Here is a breakdown of what the plot reveals:

- XGBoost (blue line) clearly dominates across most of the metrics, indicating its superior predictive performance in this classification problem.
- Random Forest (orange line) closely follows XGBoost, performing well in terms of Accuracy, Precision, Recall, and F1 Score.
- Decision Tree (green line) maintains respectable performance but falls short of the top two ensemble methods.
- KNN (red line) and SVM (purple line) show relatively poor performance across almost all metrics.

## 8.2 Performance Metrics Heatmap:



The heatmap visualization provides a clear comparison, enabling easy identification of the models that excel across different evaluation metrics.

- The heatmap clearly demonstrates that XGBoost is the best-performing model across all the evaluation criteria, followed closely by Random Forest.
- Decision Tree provides respectable performance but doesn't reach the level of the ensemble models.
- KNN and SVM struggle with accuracy and precision, indicating that these models may not be suitable for predicting LOSgroupNum in this dataset.

## 9.0 Discussion:

The results obtained from our analysis offer valuable insights into the predictive capabilities of various machine learning models in forecasting LOSgroupNum.

The original question aimed to explore the effectiveness of different machine learning algorithms in predicting LOSgroupNum, thereby enhancing operational efficiency and decision-making in healthcare settings. Our results provide a clear answer to this question by demonstrating the varying performance of each model in accurately classifying LOSgroupNum categories.

Specifically, the superior performance of XGBoost aligns with our hypothesis that the robust performance of XGBoost and Random Forest underscores the importance of ensemble learning and feature importance analysis in accurately predicting LOSgroupNum, thereby facilitating better resource allocation, patient management, and discharge planning in healthcare facilities.

Moreover, the balanced accuracy achieved by these models highlights their effectiveness in accounting for class imbalances, a critical consideration in medical data analysis.

Despite the promising results, our analysis has a limitation which is the reliance on structured data and predefined features. Future research endeavors could explore the integration of unstructured data sources such as clinical notes or imaging reports to enrich predictive models and capture additional insights into patient outcomes and treatment trajectories. Additionally, the analysis primarily focuses on predictive accuracy and may benefit from incorporating measures of model interpretability and explainability, particularly in healthcare settings where transparency and accountability are paramount.

Furthermore, the generalizability of our findings may be limited by the specific dataset and population characteristics. Future research could involve validation studies across diverse healthcare settings and patient populations to assess the robustness and applicability of the developed models in different contexts.

## 10.0 Contribution:

The project's workload was collaboratively shared between Rami Reddy and Tushar Jayendra Mhatre. Tushar led data preprocessing, visualization, and modeling. On the other hand, Rami focused on creating the Readme documentation, visualization, and contributed to building models. The final report was a joint effort, with both Rami and Tushar equally involved in its preparation.

## 11.0 Future Work and Interpretation:

- A feedback loop can be implemented where the model's predictions are continually compared against actual outcomes.
- As "NumTransfers" is a critical feature, hospitals might reconsider their internal transfer policies to streamline patient movement.
- Anticipate and plan for bed availability in real-time, reducing waiting times for patients needing transfer, which can directly reduce LoS.

## 12.0 References:

[1] Kieran Stone, Reyer Zwiggelaar, Phil Jones, Neil Mac Parthaláin, "A systematic review of the prediction of hospital length of stay: Towards a unified framework - https://journals.plos.org/digitalhealth/article?id=10.1371/journal.pdig.0000017"

[2] Vincent Lequertier, Tao Wang, Julien Fondrevelle, Vincent Augusto, Antoine Duclos, "Hospital Length of Stay Prediction Methods: A Systematic Review - https://pubmed.ncbi.nlm.nih.gov/34310455/"