**Project: Predicting COVID-19 ICU Admissions**

**Shimaa Hamadelnel**

**07/26/2024**

**Introduction**

Brazil has been one of the countries most affected by the COVID-19 pandemic, with more than 16 million confirmed cases and 454,429 confirmed deaths as of May 26, 2021. The country was unprepared for the pandemic and was unable to respond adequately due to the strain on hospital capacity.

A data science team at a top-tier hospital in Brazil has released a dataset on the Kaggle platform which seeks interesting solutions and findings from the public. The team is using ML to help reduce the strain on hospital ICU beds, where the objective is to develop an ML model to predict if a patient of a confirmed COVID-19 case will require admission to the ICU.

**Objective:**

This report presents an analysis of Intensive Care Unit (ICU) admissions, focusing on predictive modeling and machine learning techniques to understand and forecast patient admissions. The goal was to analyze data related to ICU admissions to identify trends, predict future admissions, and provide actionable insights for improving healthcare outcomes.

**Data Collection:**

The dataset collected for the research is publicly available on the data science platform Kaggle. A full description of the dataset can be found on the platform:  [Kaggle COVID-19 Dataset](https://www.kaggle.com/datasets/S%C3%ADrio-Libanes/covid19) including patient demographics, admission reasons, length of stay, and outcomes. The dataset included variables such as age, gender, medical history, and ICU admission.

**Exploratory Data Analysis (EDA)**

**Predictive Analytics:** Predictive models were developed to forecast ICU admissions based on historical data. Key variables included patient demographics, prior medical conditions, and historical admission patterns.

**Steps:**

1. **Understand the Data:**
   * **Load the Data:** Use pd.read\_excel Panda library to upload the data frame.
   * **Inspect Data Types and Summary Statistics:** Use df.info() and df.describe() to get a sense of data types and summary statistics.
   * **Check for Missing Values:** Identify which columns have missing values and the extent of missingness using df.isnull().sum().
2. **Identify Trends and Patterns:**
   * **Distribution of Target Variable:** Analyze the distribution of ICU admissions (ICU) to understand class balance.
   * **Correlation Analysis:** Use df.corr() to find correlations between numerical features and the target variable. Visualize this using a heatmap.
   * **Feature Analysis:** Explore the distribution of individual features to identify potential outliers or anomalies.
3. **Visualize Key Insights:**
   * **Histograms:** Visualize the distribution of continuous variables.
   * **Box Plots:** Identify outliers in continuous features.
   * **Bar Charts:** Show the frequency of categorical features.
   * **Pie chart:** Show the distribution of numerical features.

**Analysis and Findings (Descriptive Analysis)**

**1. Patient Visit Identifier**

In the dataset, the PATIENT\_VISIT\_IDENTIFIER column contains 385 unique values. This number is exactly one-fifth of the total row count of 1925, indicating that each of the 385 patients is represented by exactly 5 rows. These 5 rows for each patient correspond to different time windows, as indicated by the WINDOW column.

**2. Age Above 65**

* **Mean:** 0.47
* **Standard Deviation:** 0.50

Approximately 47% of the dataset includes patients above the age of 65. The standard deviation indicates variability in the age distribution, highlighting that age is a considerable factor in this dataset.

**3. Disease Grouping**

* **Disease Grouping 1:** Mean = 0.11
* **Disease Grouping 2:** Mean = 0.03
* **Disease Grouping 3:** Mean = 0.10
* **Disease Grouping 4:** Mean = 0.02
* **Disease Grouping 5:** Mean = 0.13
* **Disease Grouping 6:** Mean = 0.05

These metrics reveal that different disease groupings have varying levels of presence within the dataset. Disease Grouping 5 is the most prevalent, while Disease Grouping 2 is the least represented.

**4. Health Conditions**

* **HTN (Hypertension):** Mean = 0.21
* **Immunocompromised:** Mean = 0.16

Around 21% of patients have hypertension, while 16% are immunocompromised. Both conditions show moderate presence, suggesting they are notable factors in patient health.

**5. Laboratory Metrics**

* **Albumin (Median, Mean, Min, Max):** All values are 0.53 on average with a standard deviation of 0.15. This uniformity across metrics suggests consistent albumin levels in patients.
* **Albumin Difference:** Mean = -1.00 (constant value)

The albumin difference indicates a potential issue with measurement or data consistency, as the value is consistently -1.00.

**6. Blood Gas and Chemistry Metrics**

* **BE (Base Excess), BIC (Bicarbonate), BILLIRUBIN, Calcium, Creatinin, FFA, GGT, Glucose:** Most metrics have mean values around -0.9 to 0.3 with standard deviations indicating moderate variability.

These metrics generally show low variability in measurements, with some values consistently being -1.00 for differences, suggesting potential issues with data consistency or reporting.

**7. Blood Cell Counts**

* **Leukocytes, Lymphocytes, Neutrophiles:** All metrics show median values around -0.74 to -0.71 with high consistency.

Blood cell counts have low variability and appear to be uniformly distributed across the dataset.

**8. Gas Exchange Metrics**

* **P02 (Partial Pressure of Oxygen), PC02 (Partial Pressure of Carbon Dioxide):** Mean values for arterial and venous measurements are consistently around -0.78 with minor deviations.

These values reflect stable measurements of gas exchange in the blood, indicating consistent patient conditions.

**9. Lactate**

* **Median, Mean, Min, Max:** Values are consistently around 0.27 to 1.00 with a high standard deviation.

Lactate levels show considerable variation, indicating differing metabolic states among patients.

**Summary**

The dataset reveals several key insights:

* **Total Admissions:** The dataset included the number of patients who did not require ICU admission is 1410 and the number of patients who required ICU admission is 515.
* **Admission Reasons:** The most common reasons for ICU admission included respiratory failure, sepsis, and postoperative complications.
* **Demographic Distribution:** Nearly half the patients are over 65, with a balanced gender distribution.
* **Disease and Health Conditions:** Certain disease groupings and health conditions are more prevalent.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**Preprocess the Data**

**Objective:**

**Machine Learning Techniques:** Various machine learning algorithms, including Random Forest, decision trees, and SVC algorithm, were applied to analyze the ICU admission data. These techniques were used to identify trends and predict future admissions.

**Steps:**

1. **Handle Missing Values:**
   * **Imputation:** Fill missing values with appropriate imputation methods (mean, median).
2. **Encode Categorical Variables:**
   * **One-Hot Encoding:** For categorical variables like GENDER and window columns.
   * **Label Encoding:** If needed, for ordinal categorical features.
3. **Normalize/Scale Numerical Variables:**
   * **Standardization/Normalization:** The Data has been cleaned and scaled by column according to MinMaxScaler to fit between -1 and 1.
4. **Split the Data:**
   * **Training and Testing Sets:** Use train\_test\_split from sklearn to split the dataset into training and testing sets. We used 80-20 split is used.

**3. Develop ML Models**

**Objective:**

Train various machine learning models and evaluate their performance.

**Steps:**

1. **Train Several Candidate ML Models:**
   * First model was **Random Forest using pipeline** method scoring 93%
   * Then after preprocessing the data, we used **Random Forest** 86% & **Support Vector** Machine (SVM) 72% and **Logistic Regression** 85%
2. **Evaluate Models:**
   * **Metrics:** Use metrics like accuracy, precision, recall, and F1-score to evaluate model performance.
3. **Analyze Model Performance:**
   * **Overfitting:** Check for overfitting by comparing training and testing performance.
   * **Feature Importance:** Analyze feature importance for better interpretability.
   * **Hyperparameter Tuning:** Optimize model performance using techniques like GridSearchCV or RandomizedSearchCV.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**Machine Learning Insights:**

* **Trend Analysis:** Our analysis revealed distinct patterns in ICU admissions based on patient demographics and admission reasons. For example, older patients with chronic conditions were more likely to experience prolonged ICU stays.
* **Key Factors:** The Radom Forest model achieved an accuracy of 93% on the test set. This high level of accuracy indicates that the model can reliably predict the need for ICU admission for COVID-19 patients based on the provided clinical data.

**Conclusion**

* The Random Forest model achieved an accuracy of 93% on the test set, which is the highest among the three models we trained. This indicates that the Random Forest classifier can reliably predict the need for ICU admission for COVID-19 patients based on the provided clinical data.
* The Random Forest model is more suitable for this classification task as it can handle a large number of features and is less prone to overfitting compared to the other models. It also provides feature importances, which can help us understand the importance of different clinical features in predicting ICU admission.

**Recommendations for Future Projects:**

1. **Evaluate Preprocessing Methods**: Systematically test and compare various imputation and encoding techniques to identify those that most effectively handle the specific characteristics of your data. This approach will help ensure that the preprocessing methods selected contribute to the overall robustness and accuracy of the model.
2. **Implement Cross-Validation**: Employ cross-validation to rigorously evaluate the impact of different preprocessing strategies on model performance. This will provide insights into how well each method generalizes across different subsets of the data, leading to more reliable and consistent results.
3. **Optimize Hyperparameters**: Incorporate hyperparameter tuning into your model development process. By systematically exploring and optimizing hyperparameters, you can enhance model performance and ensure that the chosen configurations are well-suited to your data and objectives.