Advanced Predictive Analytics for High-Risk COVID-19 Patients Using Machine Learning

Srilakshmi Harimitha Madikuntala, Yashwanth Mannava Chowdary, Navya Sree Madagoni, Dhana Lakshmi Prasanna Manthena

This proposal outlines a project dedicated to advancing predictive analytics through machine learning to identify high-risk COVID-19 patients by analyzing Cerner's Real-World Data (CRWD). Although the immediate crisis phase of COVID-19 has passed, the importance of preparedness for future health crises remains paramount. Our focus is to develop robust predictive models with Cerner's extensive dataset, aiming to predict patient outcomes and identify key risk factors accurately. By offering healthcare professionals actionable insights, we facilitate informed decision-making and strategic planning. This project not only aims to enhance individual patient care but also strives to bolster the healthcare system's responsiveness to potential future pandemics through data-driven innovation. In this project, we will leverage CRWD's rich repository of patient data, including demographics, comorbidities, clinical tests, and treatment outcomes, to build and validate our predictive models. By utilizing advanced machine learning techniques, we aim to identify patterns and correlations that can predict which patients are at higher risk of severe outcomes from COVID-19. This will enable healthcare providers to prioritize resources, improve patient management, and potentially save lives. Furthermore, the insights gained from this project will have broader implications for managing other infectious diseases and health emergencies. By strengthening our predictive capabilities, we can better anticipate and respond to future pandemics, ensuring a more resilient healthcare system.

Predictive analytics | Machine learning | COVID-19 | Real-World Data | Healthcare

Introduction

The COVID-19 pandemic has profoundly impacted global health systems, economies, and daily lives, demonstrating the critical need for preparedness and resilience in the face of emerging health crises. Despite the progress made in managing the pandemic through vaccinations, treatments, and public health measures, the unpredictable nature of infectious diseases necessitates ongoing vigilance and innovation. One of the most pressing challenges has been identifying individuals at high risk of severe outcomes, such as hospitalization, ICU admission, or mortality, to optimize healthcare delivery and resource allocation. Cerner's Real-World Data (CRWD) offers a valuable resource in this endeavor, encompassing comprehensive patient data from diverse healthcare settings. This dataset includes demographic information, comorbidities, clinical test results, treatment histories, and outcomes, providing a robust foundation for predictive analytics. By harnessing the power of machine learning, we can analyze this extensive dataset to develop predictive models that accurately identify high-risk COVID-19 patients. The significance of such predictive models extends beyond the immediate context of COVID-19. They offer a framework for managing future pandemics and health crises, enhancing the healthcare system's ability to respond swiftly and effectively. By predicting patient outcomes and identifying key risk factors, these models can inform clinical decision-making, guide resource allocation, and improve patient care. As of June 2024, approximately 1 million deaths have been reported due to COVID-19 in the U.S. alone, with over 50% of hospitalized patients aged 65 and older. Patients with underlying health conditions such as cardiovascular diseases, diabetes, and respiratory disorders are at significantly higher risk of severe outcomes. The economic impact of the pandemic in the U.S. exceeds $16 trillion, highlighting the extensive cost of healthcare and economic disruption (source: CDC, JAMA Network). These staggering figures underscore the urgent need for innovative solutions to manage and mitigate the effects of infectious diseases. This project proposes leveraging CRWD and machine learning to develop a predictive model that addresses this critical need. Our approach involves data preprocessing, feature selection, model development, and validation. By identifying high-risk patients early, healthcare providers can implement proactive measures and personalized care plans, improving patient outcomes and optimizing healthcare resources. The ultimate goal of this project is not only to enhance individual patient care but also to strengthen the healthcare system's preparedness and responsiveness to future health emergencies. Through data-driven innovation, we aim to contribute to a more resilient and adaptive healthcare infrastructure, capable of effectively managing both current and future challenges.

Problem Statement

The COVID-19 pandemic has highlighted the critical need for healthcare systems to quickly and accurately identify high-risk patients to optimize resource allocation and treatment strategies. Despite the wealth of data available, predicting which patients will develop severe complications remains a challenge due to the complex interplay of various risk factors. As of June 2024, approximately 1 million deaths have been reported due to COVID-19 in the U.S. (source: CDC). Over 50% of hospitalized COVID-19 patients are aged 65 and older. Patients with comorbidities such as cardiovascular diseases, diabetes, and respiratory conditions are at higher risk of hospitalization. The estimated economic cost of COVID-19 in the U.S. exceeds $16 trillion, including healthcare costs and economic disruption (source: JAMA Network). To address this challenge, we propose developing a machine learning-based predictive model using Cerner's Real-World Data (CRWD). The goal is to predict high-risk COVID-19 patients based on a comprehensive set of features, including demographic information, underlying health conditions, clinical test results, and treatment history.

Specifically, our problem statement is:

How can we accurately predict high-risk COVID-19 patients using machine learning models trained on Cerner's Real-World Data?

The model should be able to: - Identify patients at higher risk of severe outcomes such as hospitalization, ICU admission, or mortality. - Highlight key risk factors contributing to these outcomes, providing actionable insights for healthcare providers. - Support healthcare professionals in making data-driven decisions to improve patient outcomes and optimize resource utilization.

By achieving these objectives, we aim to not only enhance the immediate management of COVID-19 but also contribute to the long-term preparedness and resilience of healthcare systems against future health crises.

Dataset

The unprocessed dataset comprises 11,580 rows and 88 columns, encompassing a wide range of patient information relevant to COVID-19 risk prediction at the time of admission. This dataset includes demographic details, clinical test results, medical history, and other critical indicators that could influence a patient's risk of severe COVID-19 outcomes. Given the objective of predicting patient risk at the time of admission, it was imperative to filter out post-admission features to ensure the model's relevance and accuracy.

Feature Selection. Feature selection is a critical step in the data preprocessing phase of machine learning projects. It involves identifying and selecting a subset of relevant features that contribute the most to the predictive modeling task. This process helps in reducing the dimensionality of the data, eliminating noise, and improving model performance by focusing on the most informative variables. In our case, we meticulously reviewed the dataset and removed features recorded after the patient's admission to ensure that the predictive model is based solely on data available at the time of admission.

Removed Features. We removed several features that are not relevant to the time of admission:

- Post-admission features: Columns such as *dischargedate*, *dischargedisposition*, *enc_cvd_lab_recs*, *pat_cvd_lab*, *enc_dx_recs*, *enc_result_recs*, *enc_med_recs*, and *enc_px_recs* were removed because they contain information recorded after admission.

- Length of Stay (LoS): The *LoS* feature, which indicates the length of a patient's hospital stay, was excluded since this information is not known at the time of admission.

- Identifier columns: All *id* columns were removed as they do not hold predictive value and can lead to overfitting.

- Date columns: Various *date* columns were also removed to avoid temporal biases and complexities not pertinent to initial risk assessment.

Why Feature Selection?. Feature selection is paramount for several reasons. Firstly, it helps in improving the model's performance by eliminating irrelevant or redundant data, which can reduce overfitting and enhance generalization to new, unseen data. Secondly, it speeds up the training process by reducing the computational load, as fewer features mean less complexity and faster processing times. Thirdly, it improves interpretability, making it easier for healthcare professionals to understand the model's decision-making process and identify key risk factors. Lastly, it helps in focusing the model on the most significant variables, ensuring that the insights derived are both accurate and actionable.

By applying rigorous feature selection criteria, we ensure that our predictive model is not only efficient but also robust and reliable. This process of refining the dataset to include only the most relevant features lays a strong foundation for developing a model that can accurately identify high-risk COVID-19 patients, thereby aiding healthcare providers in making timely and informed decisions. The meticulous curation of features enhances the model's capability to deliver precise predictions, ultimately contributing to better patient management and resource allocation in the healthcare system.

Exploratory Data Analysis (EDA)

Target Variable. The target variable for our analysis is 'deceased', which indicates whether a patient has died due to COVID-19 or not. This binary variable is crucial for our predictive modeling as it directly reflects the outcome we aim to predict. However, an initial exploration of the dataset revealed a significant imbalance in this target variable. Specifically, the positive class (representing deceased patients) is markedly underrepresented, accounting for only 10% of the dataset, while the negative class (representing non-deceased patients) constitutes the remaining 90%. This imbalance poses a challenge as it can lead to biased models that are heavily skewed towards predicting the majority class, thereby failing to accurately identify high-risk patients who are at greater risk of mortality. To address this issue, we implemented techniques to balance the target variable. Balancing the dataset is essential to ensure that the model can learn to recognize patterns associated with both classes effectively. We employed a combination of undersampling the majority class and oversampling the minority class. Undersampling involves reducing the number of instances in the majority class (non-deceased) to match the minority class (deceased), while oversampling involves increasing the instances of the minority class by duplicating existing records or generating new synthetic samples. These methods help in creating a more balanced dataset, which can improve the model's performance in predicting the minority class accurately. In our approach, we utilized the RandomUndersampler and SMOTE (Synthetic Minority Over-sampling Technique) packages. The RandomUndersampler reduces the majority class randomly, ensuring that the dataset remains diverse and representative of various scenarios. On the other hand, SMOTE generates synthetic samples by interpolating between existing minority class instances, thereby enriching the dataset with new, plausible examples of deceased patients. This dual approach of undersampling and oversampling ensures that the model is exposed to a balanced view of both classes during training. Balancing the target variable is a critical preprocessing step as it directly impacts the model's ability to make unbiased predictions. An imbalanced dataset would result in a model that predominantly predicts the majority class, thereby failing to identify patients at high risk of severe outcomes. By addressing the imbalance, we enhance the model's capability to discern between deceased and non-deceased patients accurately. This balanced approach not only improves the model's predictive performance but also ensures that the insights derived from the model are reliable and actionable for healthcare professionals. Consequently, balanced data helps in making informed decisions, optimizing resource allocation, and ultimately improving patient care and outcomes in the context of COVID-19.

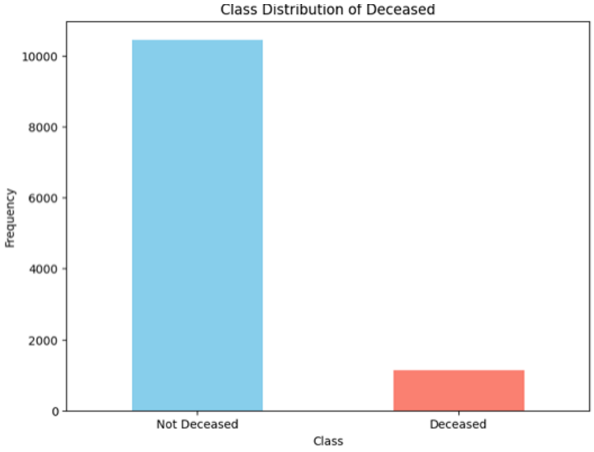

Fig. 1. Imbalanced target variable

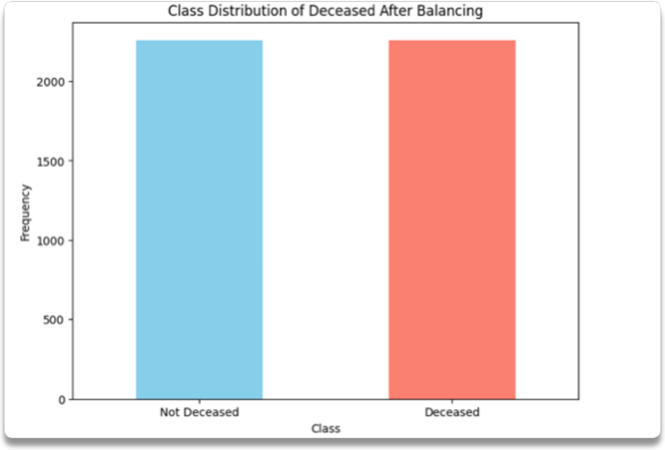

Fig. 2. Balanced target variable after a combination of oversampling and undersampling

Age Distribution. The age distribution analysis of the dataset reveals a bell-shaped histogram, with a prominent peak in the 50-60 years age range, indicating this is the most common age range for hospital encounters in the dataset. There is a noticeable decline in the frequency of encounters for individuals below 30 and above 80 years of age. This pattern suggests that middle-aged individuals, particularly those in their 50s and 60s, are more frequently encountered in the dataset, while younger and older age groups have fewer encounters. The overlaid Kernel Density Estimate (KDE) follows the histogram's contour closely, reinforcing the observation of a normal distribution trend in the age variable. This indicates that the data is symmetrically distributed around the mean age, with fewer occurrences at the extreme ends of the age spectrum. Understanding the age distribution is crucial as it helps identify the age groups that are most affected and can guide targeted healthcare interventions and resource allocation.

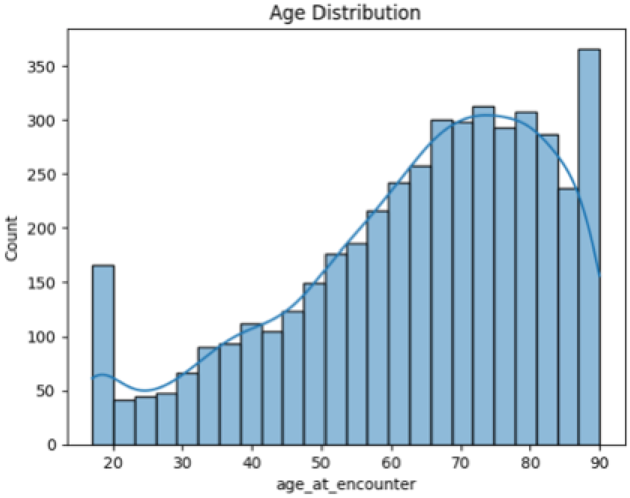

Fig. 3. Age Distribution

Data Pre-Processing

In preparation for a comprehensive analysis, we have implemented several key pre-processing steps to ensure the integrity and usability of our dataset:

- **Removal of Highly Sparse Columns**: Columns exhibiting more than 70% missing values were deemed unreliable for providing valuable insights and were subsequently removed from the dataset. This step helps maintain the robustness of our analytical model by focusing on more complete data. Sparse columns often introduce noise and bias, negatively impacting the performance of machine learning algorithms. By eliminating these columns, we ensure that the remaining data is more representative of the underlying patterns and relationships within the dataset.

- **Treatment of Missing Values**: To understand the pattern of the missing values, we performed Little's MCAR (Missing Completely at Random) test. This statistical test helps determine whether the missing values in our dataset are randomly distributed or if there is a systematic pattern to the missingness. Since the Little's MCAR test rejected the null hypothesis, indicating that the missing values are not completely random, we employed a KNN (K-Nearest Neighbors) imputer for the numerical columns in the final dataset. The KNN imputer fills in missing values by finding the 5 nearest neighbors (other rows in the dataset with similar values for the other columns) and using these neighbors to estimate the missing values. This approach ensures that the imputed values are statistically coherent with the existing data, thereby enhancing the overall quality of the dataset.

- **Mapping Categorical Data**: The 'route' column, which contains categorical data, was standardized by mapping its entries to a predefined dictionary. This transformation facilitates more efficient analysis by converting raw text data into a structured format that is easier to manipulate and interpret. For instance, categorical values such as 'Route A', 'Route B', etc., were mapped to numerical codes or standardized labels. This step not only reduces the complexity of the data but also helps in maintaining consistency and uniformity across the dataset, enabling more accurate and reliable analyses.
- **Mapping Disease Codes**: Many columns in the dataset contain codes for diseases. To ensure that these codes are accurately represented and interpreted, we used the International Classification of Diseases (ICD) codes to match each letter or code in the dataset. By mapping the disease codes to their corresponding ICD descriptions, we make the dataset more comprehensible and accessible. This step is crucial for accurate disease classification and analysis, as it aligns our dataset with widely recognized medical standards, thereby enhancing its utility for predictive modeling and other analytical tasks.
- **Imputation Strategy**: After determining that the missing values were not completely random (as indicated by the Little's MCAR test), we employed a KNN imputer to fill in the missing values in the numerical columns. The KNN imputer works by identifying the 5 nearest neighbors for each missing value based on the other columns in the dataset. The missing value is then estimated using the values from these nearest neighbors. This method leverages the similarity between different data points to provide more accurate imputations, thereby preserving the integrity and continuity of the dataset. By using this sophisticated imputation technique, we ensure that the dataset remains robust and reliable for subsequent analysis and modeling.

SHAP Analysis

Introduction to SHAP Analysis. SHAP (SHapley Additive exPlanations) is a method derived from cooperative game theory to explain the output of machine learning models. The primary goal of SHAP analysis is to attribute the contribution of each feature to the final prediction, providing a detailed understanding of the model's behavior.

Why Use SHAP?.

1. **Interpretability**: SHAP values offer a unified measure of feature importance that is consistent across different models. This interpretability helps in understanding how features influence individual predictions.
2. **Consistency**: SHAP ensures that the feature importances are consistent. If a model is adjusted to make a feature more influential, its SHAP value will increase, maintaining an intuitive understanding of feature contributions.
3. **Local and Global Explanation**: SHAP provides explanations at both the global level (overall feature importance) and the local level (individual predictions), offering a comprehensive view of model behavior.

How BORUTASHAP Enhances Feature Selection. BORUTASHAP is an advanced feature selection algorithm that leverages the interpretability of SHAP values. It identifies and retains only those features that have a significant impact on the model's predictions, enhancing model performance by reducing noise and improving interpretability.

BORUTASHAP Workflow.

1. **Calculate SHAP Values**: SHAP values are computed for each feature in the dataset, indicating their importance in the model's predictions.
2. **Feature Ranking**: Features are ranked based on their SHAP values. Features with higher SHAP values are considered more important.
3. **Threshold Selection**: A threshold is set to retain features with significant SHAP values, removing less important features to improve model performance and reduce complexity.

For our analysis, we selected all features with an average importance greater than 0.01. This threshold was chosen because it balances the need to retain important features while discarding those with minimal impact, thus optimizing the model's accuracy and performance.

Detailed Explanation of the SHAP Summary Plot. The SHAP summary plot visualizes the impact of each feature on the model's predictions. Each point in the plot represents a SHAP value for a specific feature and instance in the dataset.

- **X-axis**: Represents the SHAP value (impact on model output). Positive SHAP values indicate a positive contribution to the prediction, while negative values indicate a negative contribution.
- **Y-axis**: Lists the features ranked by importance. Features at the top are more influential in the model's predictions.

- **Color**: Indicates the feature value (red for high and blue for low). This coloring helps identify how different values of the feature affect the prediction.

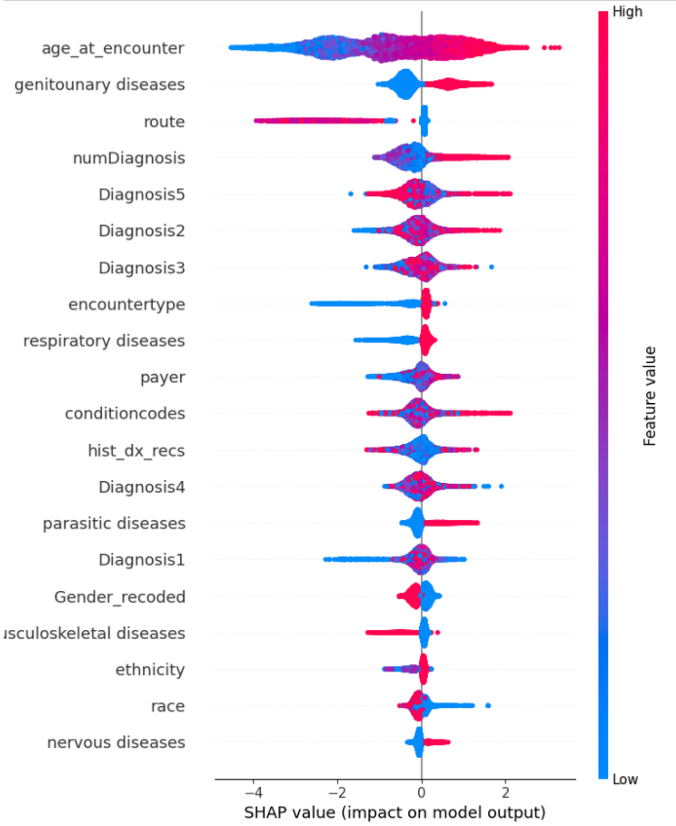

Fig. 4. SHAP Summary Plot

For instance, in the provided SHAP summary plot:

- `age_at_encounter` has a high impact, with higher values increasing the prediction.

- `genitourinary diseases` also significantly influence the model, with specific values contributing more to the predictions.

Detailed Explanation of the Average Feature Importance Plot. The average feature importance plot aggregates SHAP values across all splits, providing a holistic view of feature importance.

- **X-axis**: Lists the features.

- **Y-axis**: Represents the average SHAP value across all splits, indicating the overall importance of each feature.

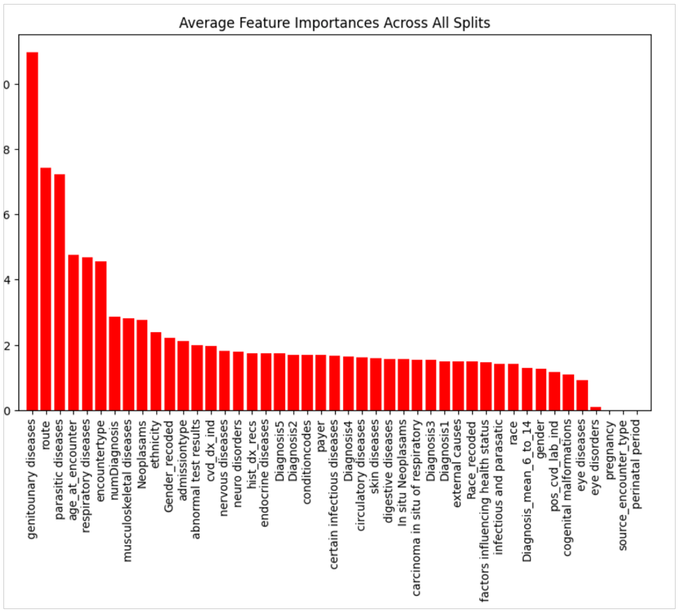

Fig. 5. Average Feature Importance Plot

In the plot, features like `genitourinary diseases`, `route`, and `numDiagnosis` show high average importance, highlighting their significant roles in the model's predictive performance.

Top 15 Features by Gain

The bar chart below delineates the top 15 features based on their gain values. The feature labeled `genitourinary diseases` emerges as the most critical, providing the highest gain and thus exerting the greatest influence when the data is split during the model training process. Following closely is the `parasitic diseases` feature, which holds significant predictive power as well. The importance of features gradually diminishes for others such as `encountertype` and `age_at_encounter`, as indicated by their lower gain scores in the chart.

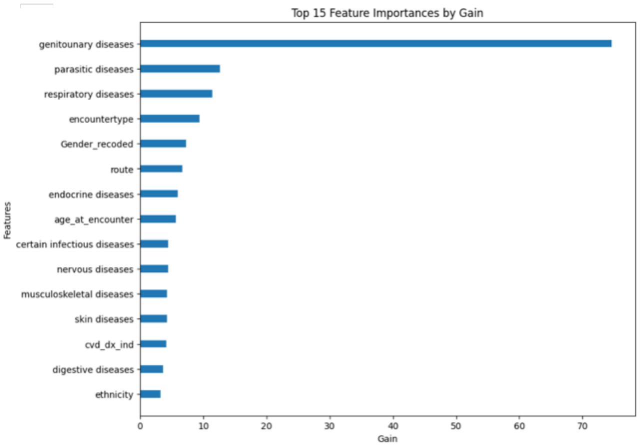

Fig. 6. Top 15 features by gain

Final Features

In this section, we detail the final set of 31 features that were selected for our predictive model based on their average importance, which was greater than 0.01. These features were identified using BORUTASHAP and are critical in predicting the target variable 'deceased'. Each feature is described in detail to provide a comprehensive understanding of its relevance and contribution to the model.

Feature Descriptions.

1. **Genitourinary Diseases**: Diseases of the genitourinary system, including urinary and reproductive organs.
2. **Route**: Indicates the route of encounter (e.g., emergency, outpatient).
3. **Parasitic Diseases**: Diseases caused by parasites (e.g., malaria).
4. **Age at Encounter**: Age of the patient at the time of the medical encounter.
5. **Respiratory Diseases**: Diseases affecting the respiratory system (e.g., asthma).
6. **Encounter Type**: Type of medical encounter (e.g., inpatient, outpatient).
7. **NumDiagnosis**: Number of diagnoses received by the patient.
8. **Musculoskeletal Diseases**: Diseases of the musculoskeletal system (e.g., arthritis).
9. **Neoplasms**: Presence of tumors (benign or malignant).
10. **Ethnicity**: Ethnic background of the patient.
11. **Gender Recoded**: Recoded gender of the patient for consistency.
12. **Admission Type**: Type of admission (e.g., elective, emergency).
13. **Abnormal Test Results**: Presence of abnormal test results.
14. **CVD Dx Ind**: Indicator for cardiovascular disease diagnosis.
15. **Nervous Diseases**: Diseases of the nervous system (e.g., epilepsy).
16. **Neuro Disorders**: Neurological disorders (e.g., migraines).
17. **Hist Dx Recs**: Historical diagnosis records.
18. **Endocrine Diseases**: Diseases of the endocrine system (e.g., diabetes).
19. **Diagnosis5**: Fifth recorded diagnosis.
20. **Diagnosis2**: Second recorded diagnosis.
21. **Condition Codes**: Medical condition codes.
22. **Payer**: Information about the patient's insurance provider.
23. **Certain Infectious Diseases**: Specific infectious diseases.
24. **Diagnosis4**: Fourth recorded diagnosis.
25. **Circulatory Diseases**: Diseases of the circulatory system (e.g., hypertension).
26. **Skin Diseases**: Diseases affecting the skin (e.g., eczema).
27. **Digestive Diseases**: Diseases of the digestive system (e.g., Crohn's disease).
28. **In Situ Neoplasms**: Localized tumors that have not spread.
29. **Carcinoma in Situ of Respiratory**: Localized carcinoma in the respiratory system.
30. **Diagnosis3**: Third recorded diagnosis.
31. **Deceased**: Indicator if the patient has died due to COVID-19.

Algorithm

In this analysis, we employ a variety of machine learning algorithms to address the predictive modeling tasks. Each algorithm has distinct characteristics and is chosen based on its ability to handle large datasets, feature interactions, and non-linear relationships. The algorithms used include Logistic Regression, Random Forest, CatBoost, XGBoost, LightGBM described as follows:

Logistic Regression. Logistic Regression is a statistical model that estimates the probability of a binary outcome based on one or more predictor variables. It is simple yet efficient and provides a good baseline for binary classification problems. The model outputs probabilities, and by applying a threshold, we can classify observations into one of two classes. This model is particularly useful for its interpretability and the ease of measuring the effect of each variable.

Random Forest. Random Forest is an ensemble learning method that operates by constructing a multitude of decision trees during training and outputting the class that is the mode of the classes predicted by individual trees. This method is effective for classification tasks because it can handle high dimensional spaces as well as large numbers of training examples. It can also model complex interactions between features and is less prone to overfitting compared to individual decision trees.

CatBoost. CatBoost (Category Boosting) is an algorithm that builds on gradient boosting frameworks by incorporating categorical variables with minimal preprocessing. It is designed to handle categorical features automatically converting them to numerical types in an optimal way. CatBoost is robust and versatile, providing powerful built-in handling of categorical variables, thus reducing the need for extensive data preprocessing.

XGBoost. XGBoost (Extreme Gradient Boosting) is an implementation of gradient boosted decision trees designed for speed and performance. XGBoost is renowned for its efficiency and flexibility. It includes regularization parameters that help in reducing overfitting, which is a serious issue in many tree-based algorithms. XGBoost can handle sparse data and has been the algorithm behind many winning models in machine learning competitions.

LightGBM. LightGBM (Light Gradient Boosting Machine) is an efficient and high-performance gradient boosting framework designed for speed and scalability. It uses a novel technique called histogram-based learning to handle large datasets and high-dimensional data quickly, making it particularly suitable for tasks like ranking, classification, and regression. LightGBM is known for its ability to handle large-scale data and its effectiveness in producing highly accurate models with minimal tuning.
Each of these models has been chosen to leverage their unique strengths in dealing with diverse aspects of the predictive modeling process. The performance of these models will be evaluated based on metrics such as accuracy, precision, recall, F1-score, and ROC-AUC, to determine which model performs best under different scenarios.

Implementation

In this section, we detail the implementation of five machine learning models for predicting high-risk COVID-19 patients. The models utilized are Logistic Regression, Random Forest, CatBoost, XGBoost, and LightGBM. The implementation includes data preprocessing, model training, cross-validation, and evaluation.

Data Preprocessing. We began by defining our target variable, 'deceased', and separating the features and the target from our final dataset. The features (X) are all the columns except the target, and the target (y) is the 'deceased' column. Next, we performed a train-test split, allocating 20% of the data for testing while maintaining the class distribution using stratified sampling. This ensures that the training and test sets are representative of the original dataset. Stratified sampling is particularly important in this context to ensure that the proportion of deceased and non-deceased patients is consistent across both training and test sets.

Model Definition. We defined five machine learning models:

- Logistic Regression
- Random Forest
- CatBoost
- XGBoost
- LightGBM

Each model was instantiated with its respective parameters to fit the nature of our dataset. Logistic Regression provides a baseline model that is easy to interpret. Random Forest, an ensemble method, helps in capturing complex interactions between features. CatBoost, XGBoost, and LightGBM are gradient boosting algorithms known for their efficiency and performance on structured data, handling categorical features effectively and providing robust predictions.

Cross-Validation Strategy. To ensure robust evaluation, we employed stratified k-fold cross-validation with 5 folds. This technique divides the data into five subsets, ensuring each fold has the same proportion of classes as the original dataset. By shuffling and splitting the data into training and validation sets multiple times, cross-validation helps mitigate overfitting and provides a more reliable estimate of model performance.

Model Evaluation Using Cross-Validation. Each model was evaluated using cross-validation on the training set. The mean scores for each metric across the 5 folds were calculated and stored. This process allowed us to assess the consistency and reliability of each model across different subsets of the data.

Training and Testing on Entire Dataset. After cross-validation, each model was trained on the entire training set and then evaluated on the test set. The performance on the test set was measured using the same scoring metrics. This step ensures that the models are not only trained effectively but also tested on unseen data to validate their predictive power.

Evaluation Metrics

In this section, we describe the evaluation metrics used to assess the performance of our machine learning models. These metrics provide a comprehensive view of how well the models perform in predicting high-risk COVID-19 patients. The metrics used are ROC AUC Score, Accuracy, F1 Score, Precision, and Recall.

ROC AUC Score. The ROC AUC Score (Receiver Operating Characteristic Area Under the Curve) is a performance measurement for classification problems at various threshold settings. The ROC curve is a plot of the true positive rate (sensitivity) against the false positive rate (1-specificity). The AUC (Area Under the Curve) represents the degree or measure of separability, indicating how well the model distinguishes between classes. A higher AUC value indicates better model performance. An AUC of 1.0 represents a perfect model, while an AUC of 0.5 represents a model with no discrimination ability, equivalent to random guessing.

Accuracy. Accuracy is the most intuitive performance measure and it is simply the ratio of correctly predicted observation to the total observations. While accuracy is a great measure, it is not suitable when the classes are imbalanced. In our context, where the positive class (deceased) is less frequent, accuracy alone can be misleading. This is because it may lead to high accuracy due to the majority class being predicted correctly most of the time, while the minority class (which is critical) may be misclassified.

F1 Score. The F1 Score is the harmonic mean of Precision and Recall, providing a single metric that balances both concerns. It is particularly useful in cases where the class distribution is imbalanced, as it considers both false positives and false negatives. The F1 Score is calculated as follows:

$$\text{F1 Score} = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}$$

This metric is especially important in our analysis because it gives a better measure of the incorrectly classified cases than the accuracy metric, particularly for the minority class.

Precision. Precision, also known as Positive Predictive Value, is the ratio of correctly predicted positive observations to the total predicted positives. High precision indicates a low false positive rate. Precision is critical when the cost of false positives is high. In the context of predicting high-risk COVID-19 patients, a high precision value ensures that the identified high-risk patients are indeed at high risk, minimizing unnecessary alerts or interventions.

$$\text{Precision} = \frac{TP}{TP + FP}$$

where: - TP = True Positives - FP = False Positives

Recall. Recall, also known as Sensitivity or True Positive Rate, is the ratio of correctly predicted positive observations to the all observations in actual class. High recall indicates a low false negative rate. Recall is crucial when the cost of false negatives is high, meaning that we want to capture as many true positive cases as possible. In the context of our study, a high recall value ensures that most high-risk patients are identified, which is critical for timely medical intervention.

$$\text{Recall} = \frac{TP}{TP + FN}$$

where: - TP = True Positives - FN = False Negatives

Summary of Evaluation Metrics. The combination of these five metrics provides a robust framework for evaluating the performance of our machine learning models.

- **ROC AUC Score** gives an overall measure of the model's ability to distinguish between classes.

- **Accuracy** provides a general measure of correct classifications.

- **F1 Score** balances precision and recall, offering a single metric that accounts for both false positives and false negatives.

- **Precision** ensures that the positive predictions are accurate.

- **Recall** ensures that the actual positive cases are captured.

By evaluating our models using these metrics, we ensure a comprehensive assessment that considers various aspects of model performance.

Evaluation Metrics

The following table presents the performance metrics of five machine learning models evaluated on the test set. The metrics include Accuracy, F1 Score, ROC AUC Score, Precision, and Recall. These metrics provide a comprehensive understanding of how each model performs in predicting high-risk COVID-19 patients.

Model	**Accuracy**	**F1 Score**	**ROC AUC**	**Precision**	**Recall**
Logistic Regression	0.81	0.82	0.81	0.77	0.88
Random Forest	0.83	0.83	0.83	0.84	0.83
CatBoost	0.86	0.86	0.86	0.84	0.88
XGBoost	0.84	0.84	0.84	0.83	0.85
LightGBM	0.84	0.84	0.84	0.84	0.85

Table 1. Performance Metrics of Models on the Test Set

Performance Metrics Table.

Detailed Explanation of Results. The table above shows the performance of each model across five different evaluation metrics. Here is a detailed explanation of each metric and how the models performed:

- **Accuracy**: This metric measures the proportion of correctly classified instances among the total instances. Random Forest, XGBoost, and LightGBM have similar accuracy scores of 0.83 and 0.84, indicating that these models correctly classified around 83% to 84% of the instances. CatBoost achieved the highest accuracy of 0.86, while Logistic Regression had the lowest at 0.81.

- **F1 Score**: The F1 Score balances precision and recall, making it particularly useful for imbalanced datasets. Random Forest, XGBoost, and LightGBM all achieved an F1 Score of 0.84, while CatBoost slightly outperformed them with an F1 Score of 0.86. Logistic Regression had an F1 Score of 0.82, reflecting its performance in balancing false positives and false negatives.

- **ROC AUC Score**: The ROC AUC Score represents the model's ability to distinguish between the positive and negative classes. Random Forest, XGBoost, and LightGBM all had an ROC AUC Score of 0.84, showing good discrimination ability. CatBoost again led with an ROC AUC Score of 0.86, while Logistic Regression was slightly lower at 0.81.

- **Precision**: Precision measures the proportion of true positive predictions among all positive predictions. It is crucial when the cost of false positives is high. Random Forest had a precision of 0.84, which is slightly higher than XGBoost and LightGBM, both at 0.84. CatBoost achieved a precision of 0.84, and Logistic Regression had a lower precision of 0.77, indicating more false positives compared to other models.

- **Recall**: Recall measures the proportion of actual positives that were correctly identified. This metric is critical in healthcare scenarios where identifying all positive cases is important. Logistic Regression had the highest recall of 0.88, meaning it identified 88% of the actual positive cases. CatBoost also had a high recall of 0.88. Random Forest, XGBoost, and LightGBM had slightly lower recall values of 0.83 and 0.85, respectively.

By examining these metrics, we can better understand the strengths and weaknesses of each model in predicting high-risk COVID-19 patients.

Conclusion

The CatBoost model emerged as the best performer in our evaluation of predictive models for identifying high-risk COVID-19 patients. This conclusion is based on several key performance metrics where CatBoost excelled:

- **Highest F1 Score**: With an F1 Score of 0.86, CatBoost effectively balances precision and recall. This balance is crucial in medical predictions where both false positives and false negatives have significant implications. A high F1 Score indicates that the model has a strong ability to identify true positives while maintaining a low rate of false positives and false negatives.

- **High Accuracy**: CatBoost achieved an accuracy of 0.86, demonstrating its strong ability to correctly classify both high-risk and low-risk patients. High accuracy reflects the overall correctness of the model and indicates its reliability in distinguishing between different classes.

- **Balanced Precision and Recall**: CatBoost maintained a precision of 0.84 and a recall of 0.88. Precision measures the proportion of true positive predictions among all positive predictions, while recall measures the proportion of actual positives that were correctly identified. The high recall value ensures that the majority of high-risk patients are identified, while the high precision value ensures that the identified high-risk patients are indeed at high risk.

The superior performance of CatBoost in these metrics means that our predictive model is both robust and reliable. Accurate identification of high-risk patients is vital for timely medical intervention and optimal resource allocation. In the context of a pandemic, where healthcare resources are often stretched thin, the ability to accurately predict which patients are at the highest risk can significantly impact patient outcomes and healthcare system efficiency.

Additionally, the balance between precision and recall achieved by CatBoost means it is well-suited for real-world applications where both false positives and false negatives need to be minimized. In healthcare, false positives can lead to unnecessary stress and treatment for patients, while false negatives can result in critical cases being overlooked. CatBoost's ability to minimize these errors ensures better patient care and more efficient use of medical resources.

In summary, the CatBoost model provides a highly effective tool for predicting high-risk COVID-19 patients, offering healthcare providers actionable insights to enhance patient care and optimize healthcare resources. The model's performance underscores its value in supporting critical decision-making processes during health crises, ensuring that the right patients receive the right level of care at the right time.