

|  |
| --- |
|  |
| covid-19 icu admissions |
| predictions using ml |
| BRITTANY BROWN | UTA: DATA SCIENCE & ANALYTICS BOOTCAMP |

|  |  |
| --- | --- |
|  | Decorative |
| Introduction |
| THE PROBLEM The COVID-19 pandemic has significantly strained healthcare systems worldwide, with Brazil being one of the hardest-hit countries. As of May 26, 2021, Brazil reported over 16 million confirmed COVID-19 cases and 454,429 confirmed deaths. The rapid surge in cases overwhelmed hospital capacities, particularly Intensive Care Units (ICUs). To address this issue, a data science team at Hospital Sírio-Libanês, São Paulo and Brasilia has released a dataset on Kaggle, aiming to develop machine learning (ML) models that can predict whether a COVID-19 patient will require ICU admission. The objective of this research is to leverage ML to anticipate ICU needs, thereby optimizing resource allocation and potentially saving lives. TASK #1 **Predict admission to the ICU of confirmed COVID-19 cases.** Based on the data available, is it feasible to predict which patients will need intensive care unit support? The aim is to provide tertiary and quaternary hospitals with the most accurate answer, so ICU resources can be arranged, or patient transfer can be scheduled.   TASK #2 **Predict NOT admission to the ICU of confirmed COVID-19 cases.** Based on the subsample of widely available data, is it feasible to predict which patients will need intensive care unit support? The aim is to provide local and temporary hospitals a good enough answer, so frontline physicians can safely discharge and remotely follow up with these patients.[[1]](#endnote-1)  Figure 1 |
|  |
|  |

|  |  |  |
| --- | --- | --- |
|  |  | Decorative |
| Bar graph with upward trend |  | DATA DESCRIPTION |
| The dataset provided by the hospital includes four categories of clinical and demographic features collected from confirmed COVID-19 patients. All personally identifying data has been anonymized. Data points have been cleaned and scaled by column using a Min-Max Scaler to fit between -1 and 1. The data contains patient demographic information, patient previous grouped diseases, blood results, and vital signs. Patient demographics(03) [AGE\_ABOVE65] – Binary variable: ≤ 65 or > 65  [AGE\_PERCENTIL] – 10th to Above 90th (does not necessarily indicate age range)  [GENDER] – Binary variable: Male/Female disease groupings (09) [DISEASE\_GROUPING\_1] – Unidentified for confidentiality  [DISEASE\_GROUPING\_2] – Unidentified for confidentiality  [DISEASE\_GROUPING\_3] – Unidentified for confidentiality  [DISEASE\_GROUPING\_4] – Unidentified for confidentiality  [DISEASE\_GROUPING\_5] – Unidentified for confidentiality  [DISEASE\_GROUPING\_6] – Unidentified for confidentiality  [HTN] - Hypertension  [IMMUNOCOMPROMISED] – Unidentified for confidentiality  [OTHER] – Unidentified for confidentiality   blood test results (36) Blood results and Vital Signs features are expanded to include measurements for the Min, Max, Median, Mean, Diff, and Relative Diff. Thus, 36 bloodwork features are expanded to 216 columns.   |  |  |  |  | | --- | --- | --- | --- | | [ALBUMIN] | [FFA] | [NEUTROPHILES] | [POTASSIUM] | | [BE\_ARTERIAL] | [GGT] | [P02\_ARTERIAL] | [SAT02\_ARTERIAL] | | [BE\_VENOUS] | [GLUCOSE] | [P02\_VENOUS] | [SAT02\_VENOUS] | | [BIC\_ARTERIAL] | [HEMATOCRITE] | [PC02\_ARTERIAL] | [SODIUM] | | [BIC\_VENOUS] | [HEMOGLOBIN] | [PC02\_VENOUS] | [TGO] | | [BILLIRUBIN] | [INR] | [PCR] | [TGP] | | [BLAST] | [LACTATE] | [PH\_ARTERIAL] | [TTPA] | | [CALCIUM] | [LEUKOCYTES] | [PH\_VENOUS] | [UREA] | | [CREATININ] | [LINFOCITOS] | [PLATELETS] | [DIMER] | |

|  |  |
| --- | --- |
| Research with solid fill | VITAL SIGNS(06) There are nine Vital Signs measured which are also expanded to include Min, Max, Median, Mean, Diff, and Relative Diff. Thus, six vital sign features are expanded to 36 columns.  [BLOOD PRESSURE\_DIASTOLIC]  [BLOOD PRESSURE\_SISTOLIC]  [HEART\_RATE]  [RESPIRATORY\_RATE]  [TEMPERATURE]  [OXYGEN SATURATION] |
| DecorativeMETHODS The following methodological steps were undertaken:   1. **Exploratory Data Analysis (EDA)**: Initial data analysis to understand the distribution, correlations, and potential anomalies within the data. The original dataset was loaded into a Microsoft SQL Server database where multiple queries were written into new tables. I connected the database to PowerBI to create multiple visuals analyzing relationships between variables. 2. **Data Preprocessing**: Cleaning the data by filling or removing missing values, encoding categorical variables, scaling numerical features, and splitting the data into training and testing sets. I used Python in Microsoft Visual Studio Code to import and clean the data, which was then exported to a new spreadsheet to analyze model data. Missing blood results and vital signs were filled using k-Nearest Neighbor Imputation.   The two categorical columns ([AGE\_PERCENTIL] and [WINDOW]) were encoded using Label Encoder. The dataset was already scaled using Min Max Scaler. Because the Target Variable is ICU = 1 (patient admitted), we could not use any data where the target variable is present. Therefore, we dropped the rows where the windows included the target variable and set the window immediately before as the new target.   1. **Model Development**: Developing multiple ML models including Logistic Regression, Decision Tree, Random Forest, Support Vector Machines, and Naïve Bayes. The data was split into a 20/80 test/train set and run through these models in VS Code. 2. **Model Evaluation**: Evaluating models using metrics such as accuracy, precision, recall, F1-score to determine the best-performing model. 3. **Hyperparameter Tuning**: Optimizing the selected model to enhance performance and reduce overfitting. Synthetic Minority Over-sampling Technique (SMOTE) was imported from IMBLearn oversample the minority class to assess the model performance. |
|  |

A diagram with numbers and circles

Description automatically generated with medium confidence

A diagram of a medical procedure

Description automatically generated

|  |  |
| --- | --- |
| Decorative | |
| MACHINE LEARNING MODEL DEVELOPMENT | |
| Since the goal is to predict ICU admissions, I used five different machine learning models to address this binary classification issues. | |
| logistic regression A simple linear model for binary classification that estimates the probability that a given input belongs to a particular class. | |
| from sklearn.linear\_model import LogisticRegression  from sklearn.model\_selection import train\_test\_split  from sklearn.metrics import accuracy\_score, classification\_report  # Split the data  X = df\_cleaned.drop(columns=['ICU']) #input  y = df\_cleaned['ICU'] #output  X\_train, X\_test, y\_train, y\_test = train\_test\_split(X, y, test\_size=0.2, random\_state=42)  # Logistic Regression  log\_reg = LogisticRegression(max\_iter=1000)  log\_reg.fit(X\_train, y\_train)  y\_pred\_log = log\_reg.predict(X\_test) | |
| decision tree classifier Non-linear model that splits the data based on feature values, easy to interpret but can overfit. | |
| from sklearn.tree import DecisionTreeClassifier  # Decision Tree  dt = DecisionTreeClassifier(random\_state=42)  dt.fit(X\_train, y\_train)  y\_pred\_dec = dt.predict(X\_test) | |
| random forest An ensemble of decision trees, reducing overfitting by averaging multiple trees. | |
| from sklearn.ensemble import RandomForestClassifier  # Random Forest  rf = RandomForestClassifier(n\_estimators=100, random\_state=42)  rf.fit(X\_train, y\_train)  y\_pred\_rf = rf.predict(X\_test) | |
| support vector machines (svm) Finds the optimal hyperplane that maximizes the margin between classes. | |
| from sklearn.svm import SVC  # Support Vector Machine  svc = SVC(kernel='rbf', probability=True)  svc.fit(X\_train, y\_train)  y\_pred\_svm = svc.predict(X\_test) | |
| naïve bayes Based on Bayes’ theorem with the assumption of feature independence. | |
| from sklearn.naive\_bayes import GaussianNB  # Naive Bayes  nb = GaussianNB()  nb.fit(X\_train, y\_train)  y\_pred\_nb = nb.predict(X\_test) | |

|  |  |
| --- | --- |
| Decorative | |
| PERFORMANCE ANALYSIS | |
| |  | | --- | | Logistic Regression Accuracy: 0.972318339100346  Logistic Regression Report:  precision recall f1-score support  0.0 0.98 0.99 0.98 251  1.0 0.94 0.84 0.89 38  ***Accuracy****: Proportion of correctly predicted measures*  ***Precision:*** *Proportion of positive predictions that are correct*  ***Recall:*** *Proportion of actual positives that are correctly identified*  ***F1 Score:*** *Harmonic mean of precision and recall*  accuracy 0.97 289  macro avg 0.96 0.92 0.94 289  weighted avg 0.97 0.97 0.97 289 | | Decision Tree Accuracy: 0.9411764705882353  Decision Tree Report:  precision recall f1-score support  0.0 0.97 0.96 0.97 251  1.0 0.76 0.82 0.78 38  accuracy 0.94 289  macro avg 0.86 0.89 0.88 289  weighted avg 0.94 0.94 0.94 289 | | Random Forest Accuracy: 0.9653979238754326  Random Forest Report:  precision recall f1-score support  0.0 0.98 0.98 0.98 251  1.0 0.89 0.84 0.86 38  accuracy 0.97 289  macro avg 0.93 0.91 0.92 289  weighted avg 0.96 0.97 0.96 289 | | SVM Accuracy: 0.8685121107266436  SVM Report:  precision recall f1-score support  0.0 0.87 1.00 0.93 251  1.0 0.00 0.00 0.00 38  accuracy 0.87 289  macro avg 0.43 0.50 0.46 289  weighted avg 0.75 0.87 0.81 289 | | Naive Bayes Accuracy: 0.9411764705882353  Naive Bayes Report:  precision recall f1-score support  0.0 0.95 0.98 0.97 251  1.0 0.86 0.66 0.75 38  accuracy 0.94 289  macro avg 0.91 0.82 0.86 289  weighted avg 0.94 0.94 0.94 289 | | |
| List  *Vital signs are sampled more frequently than blood labs.* | Decorative |
| CONCLUSION |
| The main question for each task asked whether it was feasible to predict which patients will need ICU support and which patients can be safely discharged and followed up with remotely.  Preliminary analysis of the data – using Logistic Regression and Random Forest machine learning models, specifically – yield evidence of highly predictive capabilities.  All 230 input columns were analyzed using these models, still producing 97.2% and 96.5% accuracy scores, respectively. However, future analyses should run the four distinct categories of data separately through each model if one is to assume categorical independence. The disparities in the frequency of data collection also warrant this separation.  Because hospital staff need to be able to identify patients of concern as early possible, future analyses should also limit the range of ICU admissions windows to the 0–2-hour window, increasing clinical relevance.  This model was trained specifically with data collected from the São Paulo area. Wider implementations would need to use and compare data collected local to their respective geographic areas. |

1. <https://www.kaggle.com/datasets/S%C3%ADrio-Libanes/covid19> [↑](#endnote-ref-1)