ICU SURVIVAL ANALYSIS

ACHIMUGU A.S

26-062022

Outline

Introduction

BUSINESS USE CASE

In clinical practice, prediction of ICU mortality risk can be useful in

- Triage and resource allocation
- To determine appropriate levels of care
- To prepare discussions with patients and their families around expected outcomes
- Help policymakers identify useful policies

OBJECTIVES

Create a model that uses data from the early hours of intensive care to predict patient survival with

- Better prediction probability than apache
- Minimize apache features
- Transparent (easy to explain)
- Generalizability Less complexity

METHODOLGY

- Data Collection
- Data Cleaning
- Data Visualisation
- Data Pre Processing

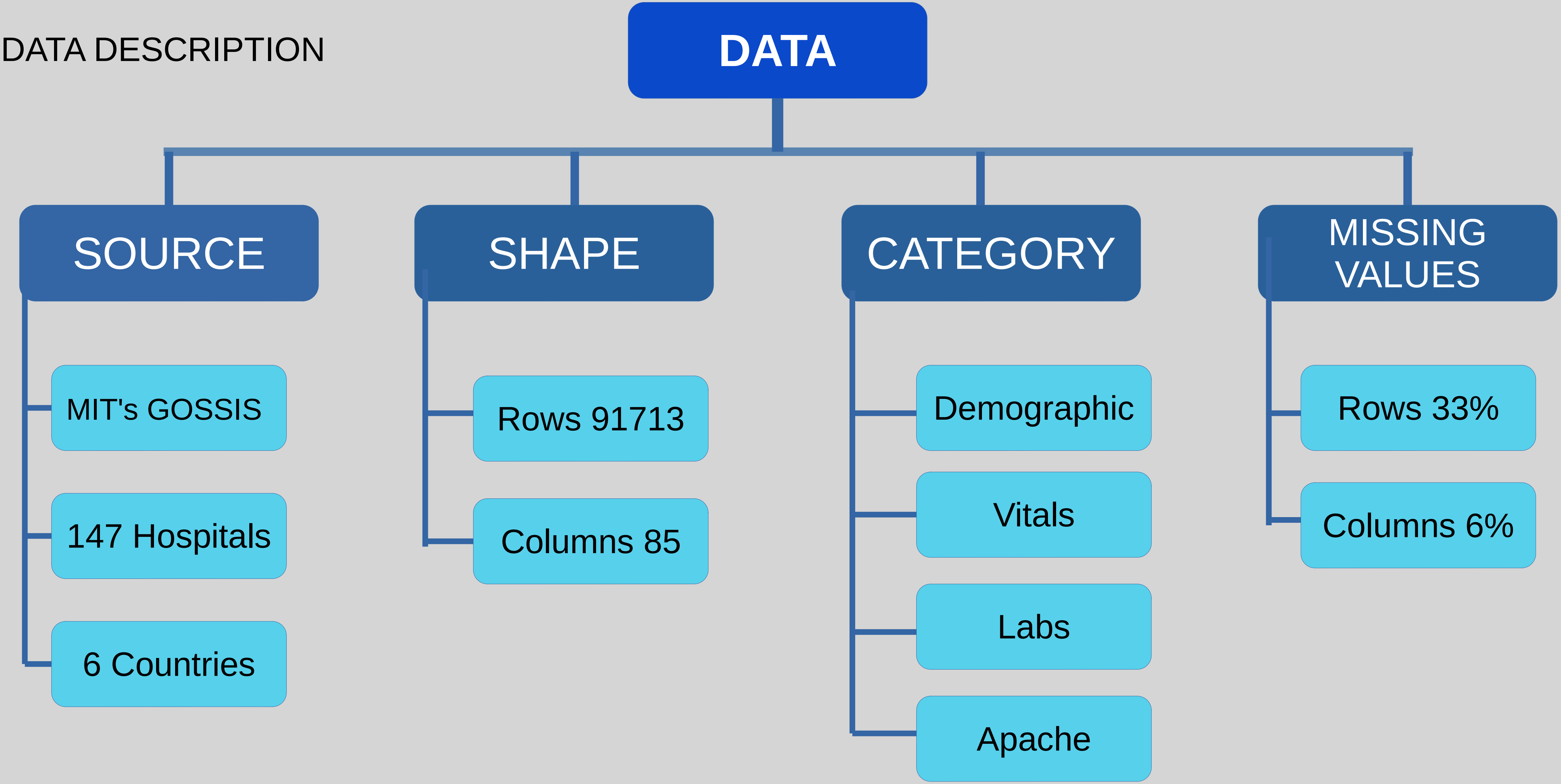
Methodology

DATA

MIT's GOSSIS community initiative, with privacy certification from the Harvard Privacy Lab, provided the dataset of more than 90000 hospital Intensive Care Unit (ICU) visits from patients, spanning a one-year timeframe. The data is part of a growing global effort and consortium spanning Argentina, Australia, New Zealand, Sri Lanka, Brazil, and more than 200 hospitals in the United States. Dataset can be found here

Methodology

DATA DESCRIPTION



Methodology

DATA CLEANING

Dropped Features:

- That are irrelevants to the model or analysis e.g 'encounter_id', 'patient_id'.
- With many missing rows that can’t be handled without affecting data authenticiy.
- That are hihly corrollecte with each other e.g APACHE II and APACHE III, d1_potassium_min and h1_potassium_min.

Impute:

Impute:

- Fill the Nan of normal distributed columns with column mean
- Replaced all CCU-CTICU CSICU CTICU with Cardiac ICU
- Replaced all negative pre_icu_los_days with zero

Add features:

- bmi_cat by grouping bmi into underweight, normal, overweight, obesed.
- gcs_cat by adding gcs_eyes, gcs_motor, gcs_verbal and grouping into normal mild moderate and server.
- Other created features include age_cat, h1_pluse_P, heart_rate_cat map_cat.

Methodology

DATA CLEANING

Dropped Features:

- That are irrelevants to the model or analysis e.g 'encounter_id', 'patient_id'.
- With many missing rows that can’t be handled without affecting data authenticiy.
- That are hihly corrollecte with each other e.g APACHE II and APACHE III, d1_potassium_min and h1_potassium_min.

Impute:

Impute:

- Fill the Nan of normal distributed columns with column mean
- Replaced all CCU-CTICU CSICU CTICU with Cardiac ICU
- Replaced all negative pre_icu_los_days with zero

Add features:

- bmi_cat by grouping bmi into underweight, normal, overweight, obesed.
- gcs_cat by adding gcs_eyes, gcs_motor, gcs_verbal and grouping into normal mild moderate and server.
- Other created features include age_cat, h1_pluse_P, heart_rate_cat map_cat.

Methodology

DATA VISUALISATION

Visualisation was done Seaborn Scatter plot, Bar plot, Piechart, Histogram and Boxplot.

Barplot:

- Used to compare gender, ethnicity apache_3j_bodysystem, apache_2_bodysystem, bmi_cat, gcs_cat h1_pluse_P, map_cat, and heart_rate_cat.

Piechart:

- Used to show percentage distribution of categories in hospital_death, icu_type.

Histtogram& Boxplot:

- Used to show the distribution and five point summary of age, bmi, height and weight.

Scatterplot:

- Used to show the relationship betweenh1_mbp_max and d1_mbp_max.

Methodology

DATA PREPROCESSING

Data Encoding

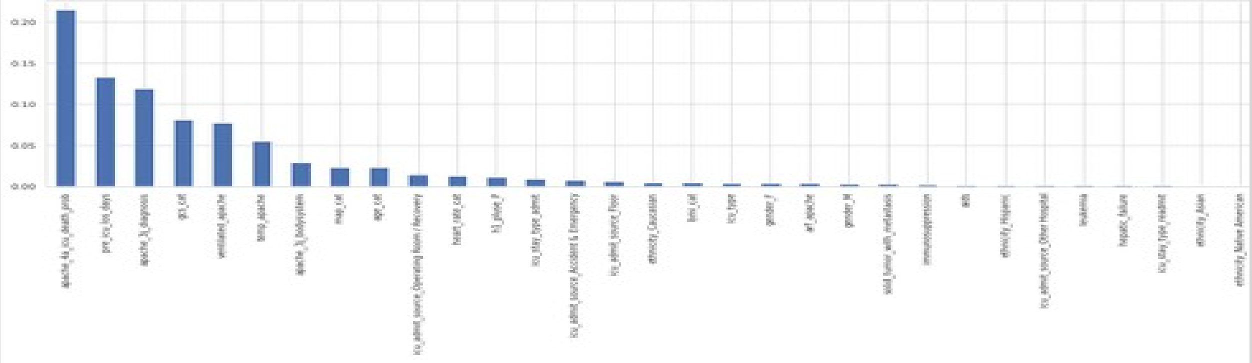
- Ordinal categories such as gcs score, heart_cat, bmi_cat were label ecode while other categories were one hpt encoded.

Data Oversampling

- Imbalance in dataset was handle by random Oversampling of training data.

Feature Selection

- Feature selection was done using mutual infomation gain inother to get top features that contributed most to the target feature.



```
apache_4a_icu_death_prob                        0.214602
pre_icu_los_days                                0.132541
apache_3j_diagnosis                             0.118689
gcs_cat                                         0.081284
ventilated_apache                               0.077031
temp_apache                                     0.054355
apache_3j_bodysystem                            0.029404
map_cat                                         0.022980
age_cat                                         0.022761
icu_admit_source_Operating Room / Recovery      0.014421
heart_rate_cat                                  0.012944
h1_pluse_P                                      0.011205
icu_stay_type_admit                             0.009292
icu_admit_source_Accident & Emergency           0.007484
icu_admit_source_Floor                          0.005785
dtype: float64
```

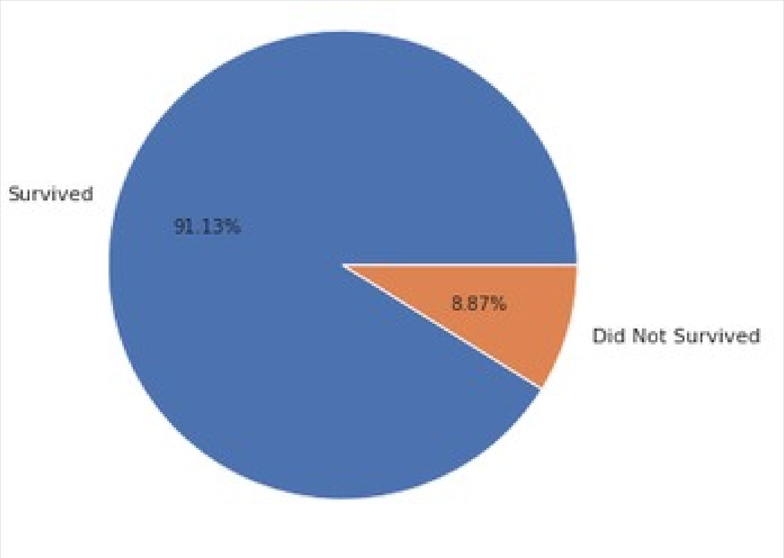
IBM Developer
SKILLS NETWORK

EXPLORATORY DATA ANALYSIS

<Name>
<Date>

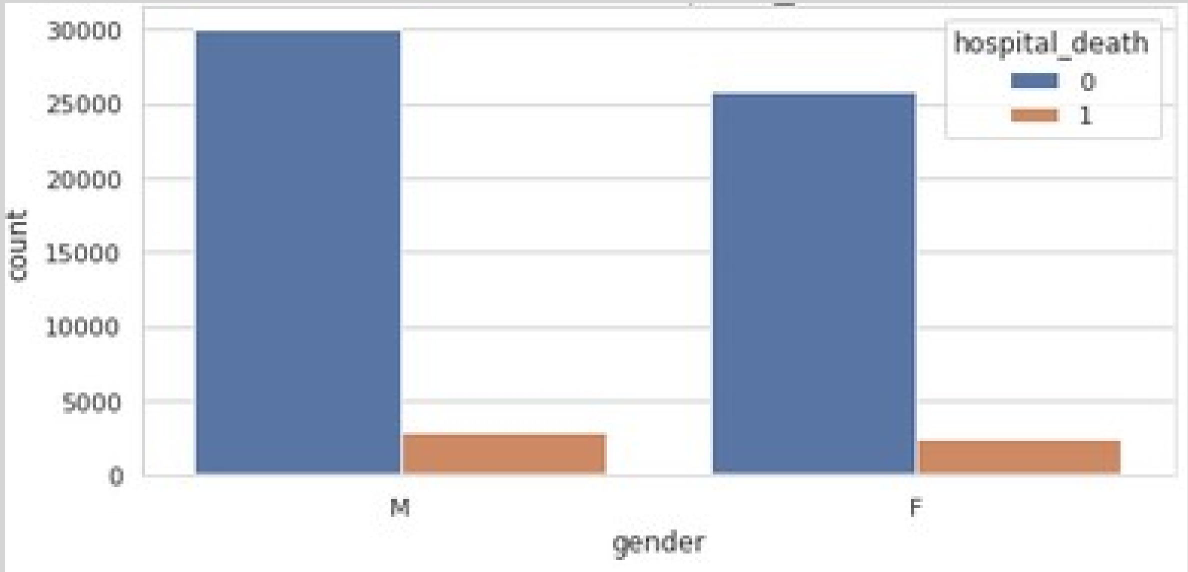
EDA with Data Visualization

Percentage Representation of hospital_death



- The target feature (hospital_death) is highly imbalance with with a 91% survival rate and 9% non survival

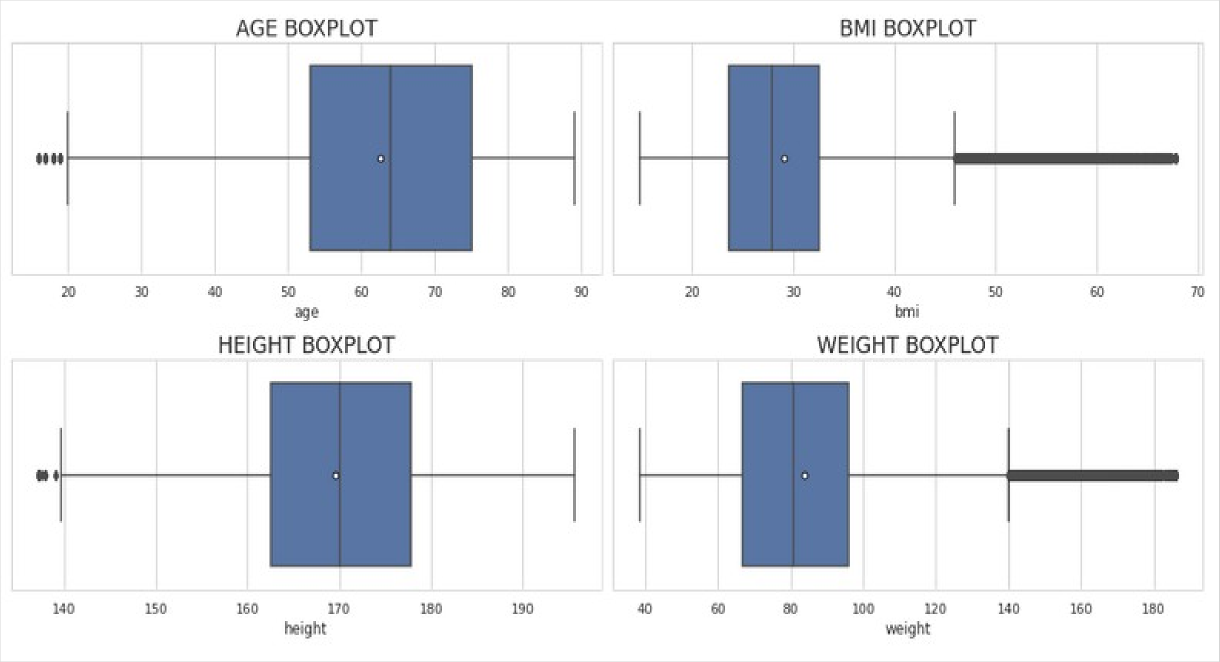
Gender

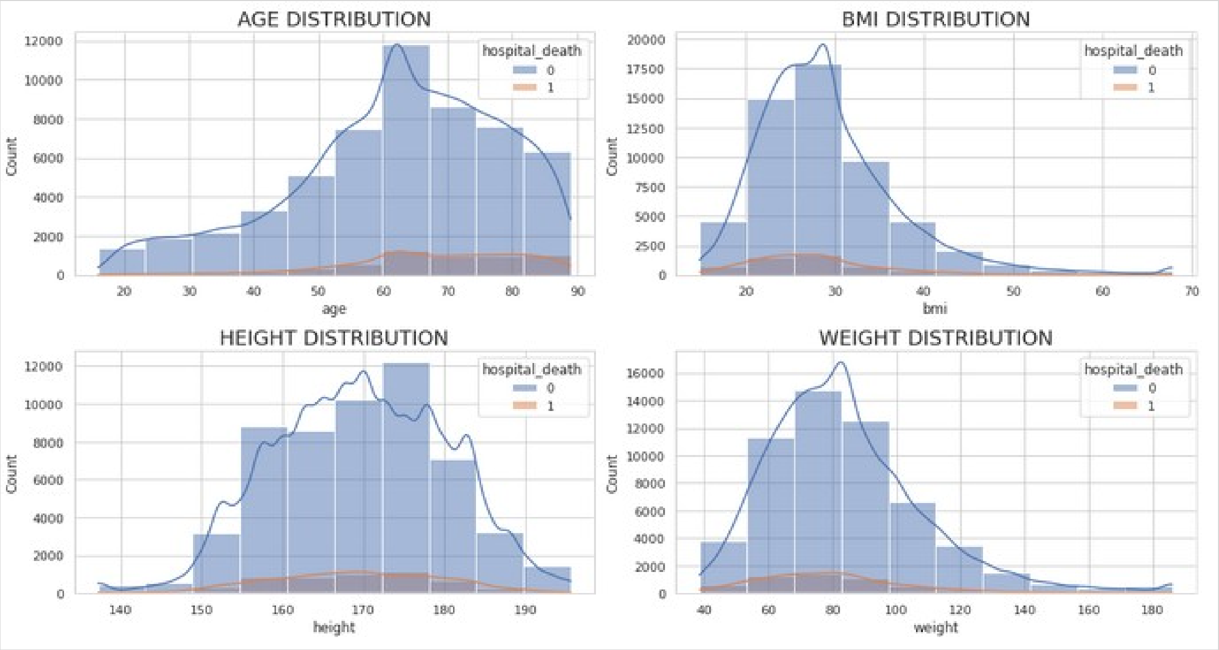


- They are 41898 male and 35746 female
- They are 32817 female and 38423 male who suvrived while 3129 female and 3475 male did not.

EDA with Data Visualization

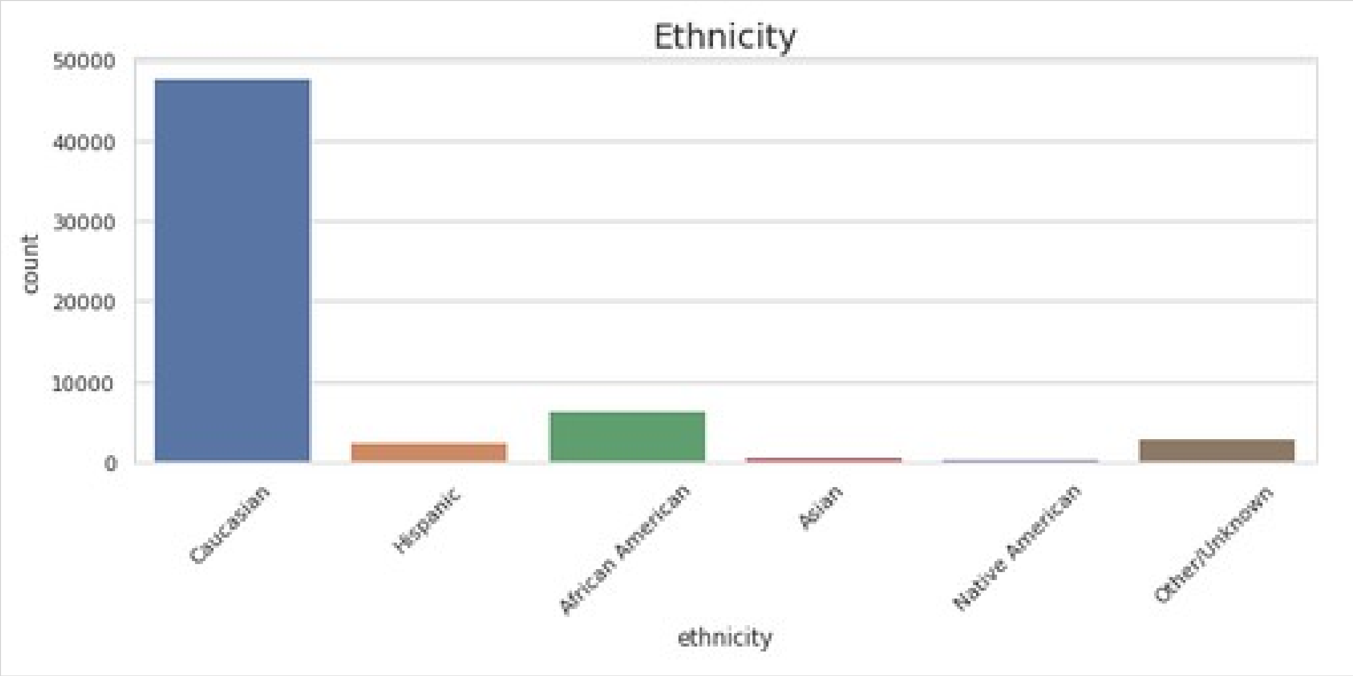
Age, BMI, Weight and Height Distribution,Boxplot

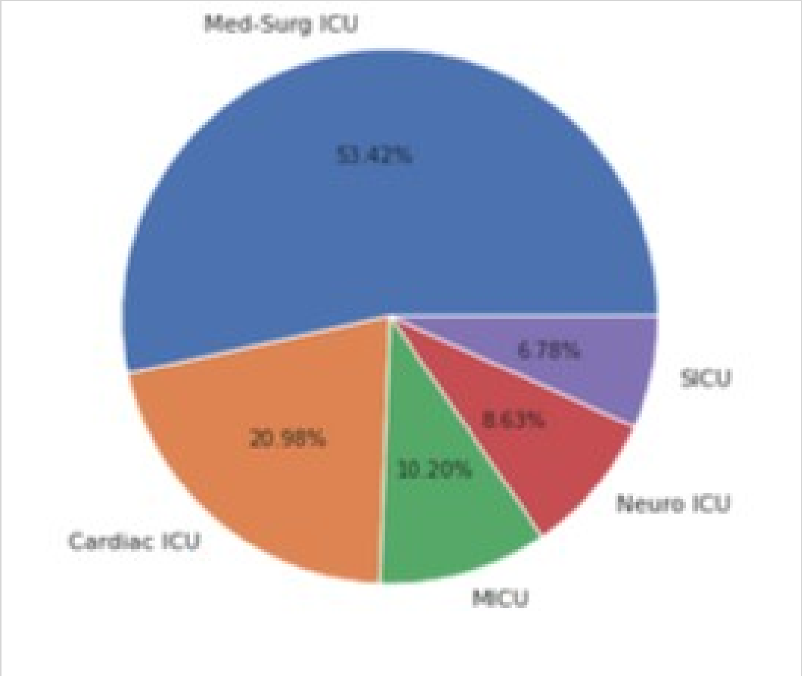




- Age seems to a little left skewed with few outlier
- MI (body max index) and weight are right skewed with alot of outliers
- Height has normally distribution.
- The mean age, bmi, height and weight are 62.48, 29.15, 169.66cm and 83.98kg respectively.

EDA with Data Visualization



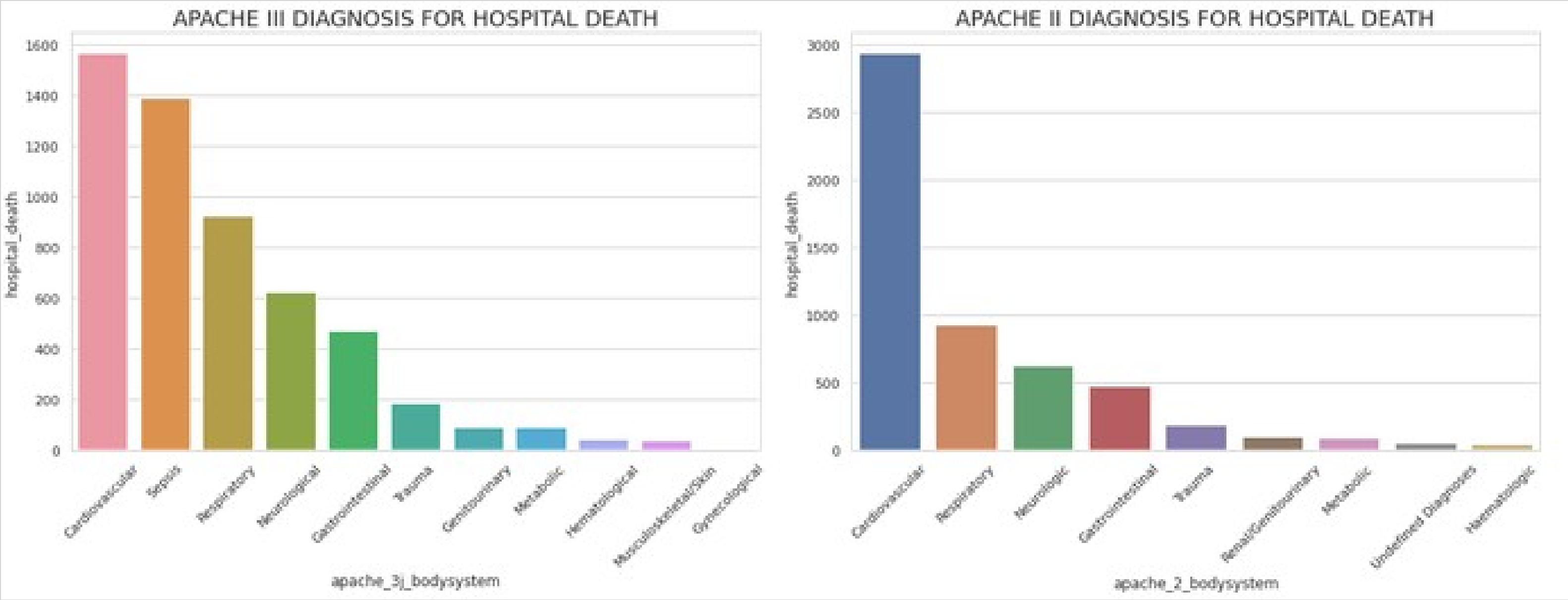


- Caucasian > African American > Hispanic > Asian > Native American
- Caucasian make 78% of the observation among total Ethnicity.

- From the observation Med-Surg ICU provided the most care accounting for more than 50% of all ICU type.
- Cardiac ICU accounts for 21% of all ICU type.

EDA with Data Visualization

```
#median is used here as the data is very tailed to the right  and the mean will be greatly affect by outliers
df['pre_icu_los_days'] = df['pre_icu_los_days']*24 # converts to hours
stay=round(df['pre_icu_los_days'].median(),2)
print(f'The average length of stay of the patient between hospital ward admission and unit admission = {stay} hours')
```

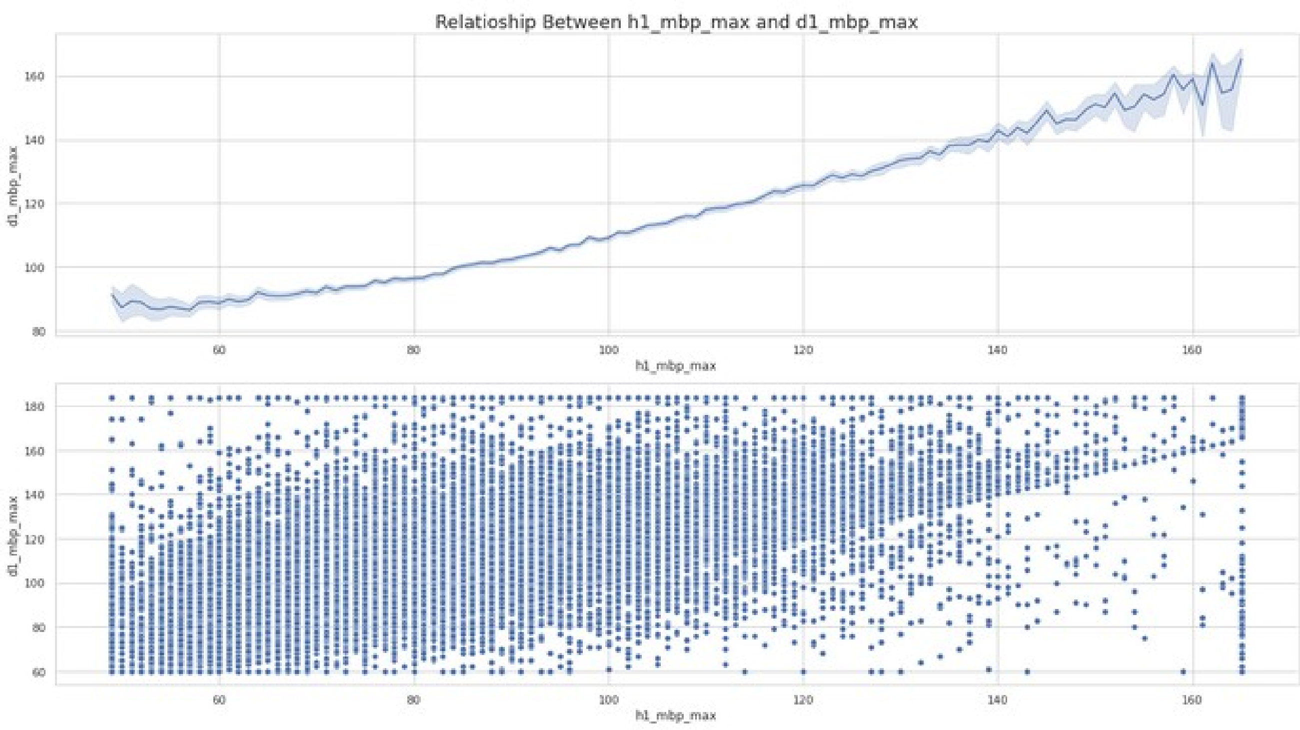


The average length of stay of the patient between hospital ward admission and unit admission = 3.28 hours

- In admission diagnosis for both APACHE III and APACHE II, Cardivascular disorder have the highest frequency for those who didnt survive
- Respiratory conditons comes third in APACHE III after Sepsis and second in APACHE II
- Gynecology and Heamatologic condictions account for least death for both the APACHE III AND APACHE II respectively.

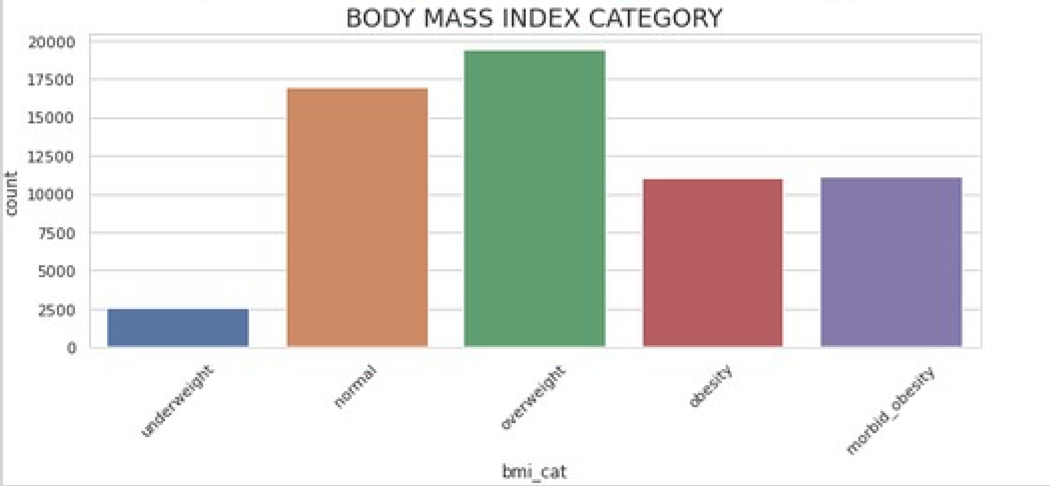
EDA with Data Visualization

Relatioship Between h1_mbp_max and d1_mbp_max



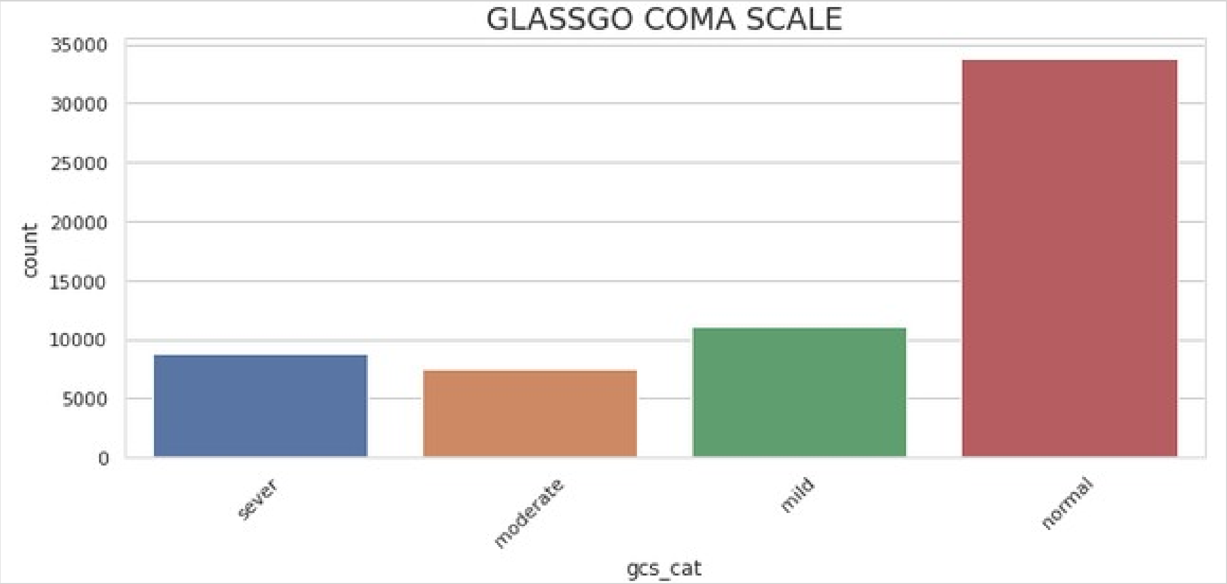
- They seem to be correlection between first hour mean blood pressure and 24 hour mean blood pressure

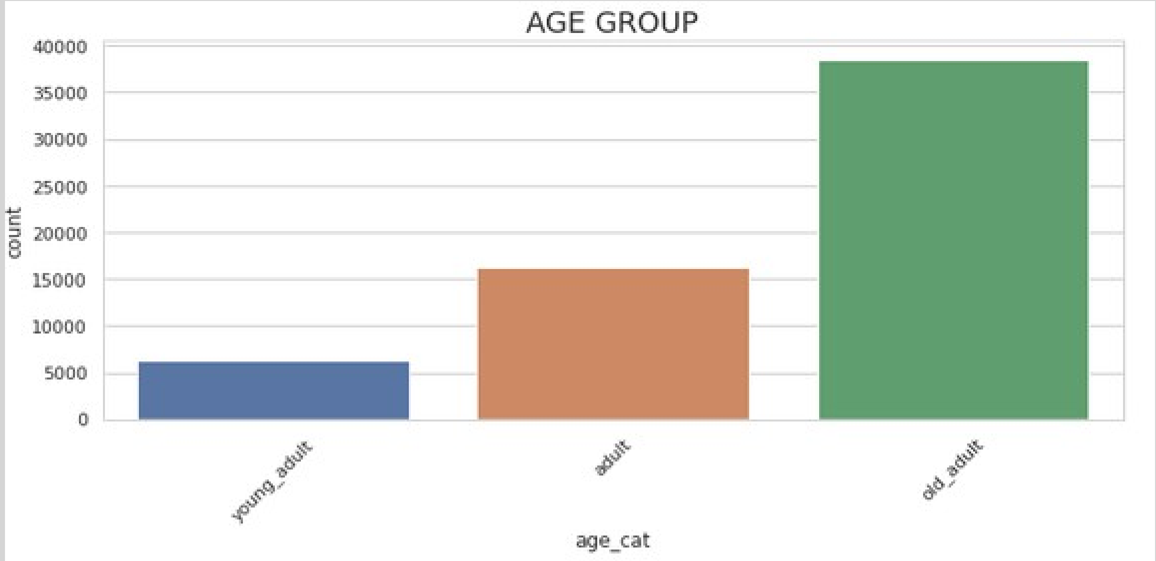
BODY MASS INDEX CATEGORY



- Most of the patient are overweight and above
- 27% of patient are normal BMI while 4% are underweight

EDA with Data Visualization





- 63% patient from the observation are aged above 60 years old adults
- 55% were conscious while 45% has some level of impaired consciousness

IBM Developer
SKILLS NETWORK

PREDICTIVE ANALYSIS

- Models Training
- Models Assement Criteria
- Model Tuning
- Model Evaluation

<Name>
<Date>

Predictive Analysis

MODELS TRAINING

Classification algorithms used on data sets are

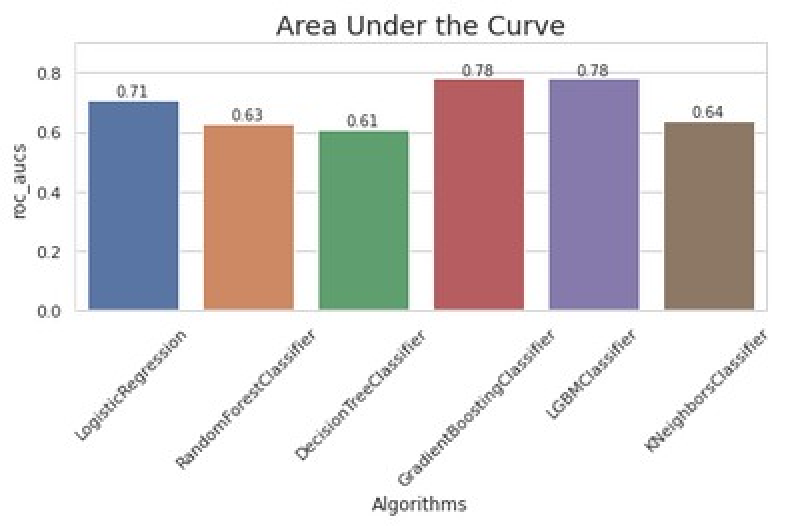
- Logistic Regression Classifier
- Decsion Tree Classifier
- Random Forest Classifier
- K Nearest Neighbour
- Gradient Boosting Classifier
- Light Gradient Boosting Classifier

Predictive Analysis

MODELS ASSESSMENT CRITERIA

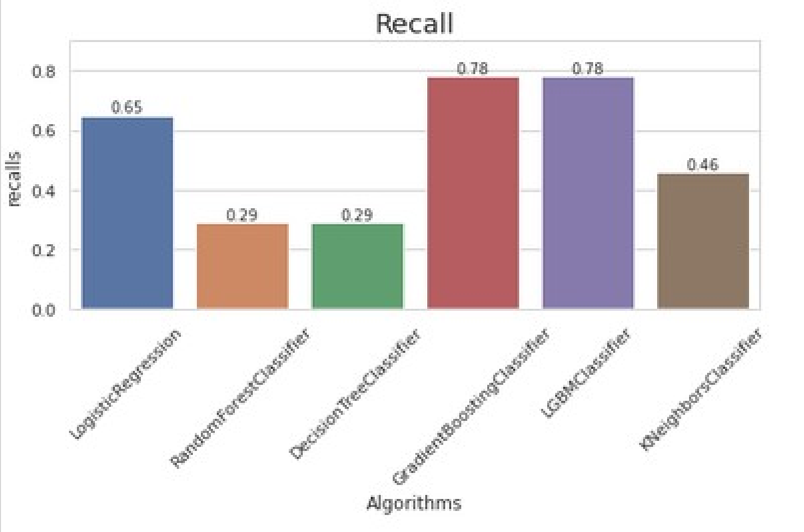
AUC Score:

In imbalanced dataset False positive rate and true positive rate are more important than accuracy



Recall:

Recall measures the ability of a model to detect positve samples.



- From the graphs above, Gradient Booster classifer and Light Gradeint Booster has the highest area under the curve of 0.78 on test data and a recall of 0.78with default parameter. Both algorimthms will be tunned with different parameter for best scores

Predictive Analysis

4.1 Gradient Boosting Parameters Tuning

```
]: parameters ={# Setting training with parameters
        "learning_rate":[0.01, 0.1],
        'n_estimators':[100, 500, 1000],
        'subsample':[0.1, 0.3, 0.6, 0.9, 1],
        'max_depth':[3, 6, 9],}
```

```
]: gb=GradientBoostingClassifier()
gb_rscv = RandomizedSearchCV(gb, parameters,scoring='roc_auc', cv=3, n_iter=3)
gb_tuned = gb_rscv.fit(X_train,y_train)
print("train_auc :",gb_tuned.best_score_)
```

```
train_auc : 0.8710871336152417
```

```
]: gb_tuned_predictions=gb_tuned.predict(X_test)
auc = roc_auc_score(y_test, gb_tuned_predictions)
print("test_auc: ",auc)
```

```
test_auc:  0.7785844667155245
```

```
]: print("tuned hpyerparameters :(best parameters) ",gb_tuned.best_params_)
```

```
tuned hpyerparameters :(best parameters)  {'subsample': 1, 'n_estimators': 100, 'max_depth': 3, 'learning_rate': 0.
1}
```

Predictive Analysis

4.2 Light Gradient Booster Parameters Tuning

```
parameters ={# Setting training with parameters
    "learning_rate":[0.01, 0.001],# ,
    'num_iterations':[500, 700, 1000],
    'num_leaves':[5, 10, 15],
    'max_depth':[3, 6, 9]}
```

```
lgb = Lgb.LGBMClassifier()
lgb_rscv = RandomizedSearchCV(lgb, parameters,scoring='roc_auc', cv=3, n_iter=3)
lgb_tuned = lgb_rscv.fit(X_train,y_train)
print("train_auc :",lgb_tuned.best_score_)
```

```
train_auc : 0.8690252700532076
```

```
lgb_tuned_predictions=lgb_tuned.predict(X_test)
auc = roc_auc_score(y_test, lgb_tuned_predictions)
print("test_auc: ",auc)
```

```
test_auc:  0.7782795950561334
```

```
print("tuned hpyerparameters :(best parameters) ",lgb_tuned.best_params_)
```

```
tuned hpyerparameters :(best parameters)  {'num_leaves': 10, 'num_iterations': 500, 'max_depth': 6, 'learning_rate': 0.01}
```

Predictive Analysis

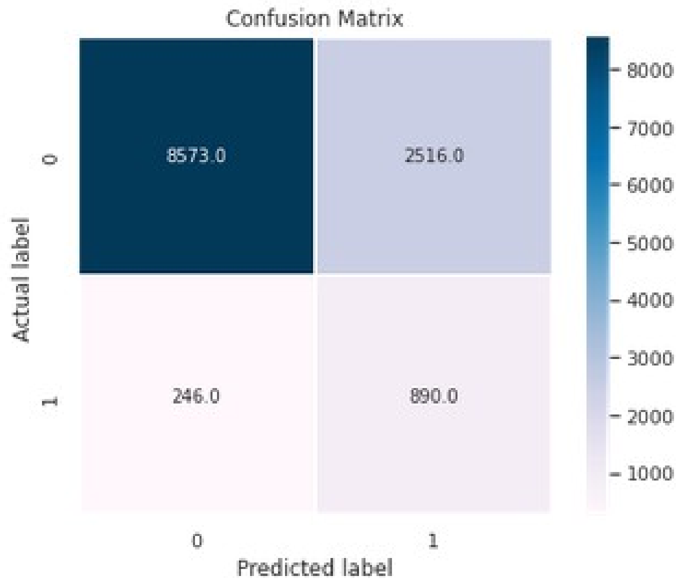
Best model metrics

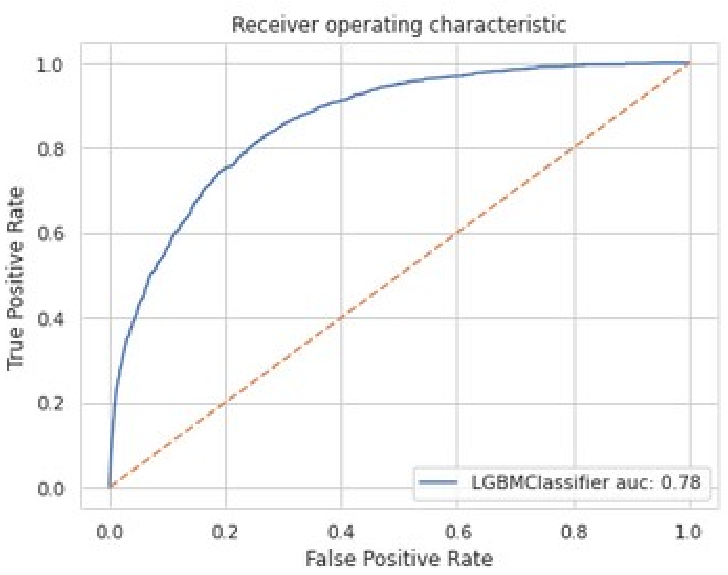
AUC

Train: 0.86

Test: 0.78

RECALL: 0.78






```
print('classification_report for Light Gradient Booster, \n')
print(classification_report(y_test, lgb_pred))
```

```
classification_report for Light Gradient Booster,

              precision    recall  f1-score   support

           0       0.97      0.77      0.86     11089
           1       0.26      0.78      0.39      1136

    accuracy                           0.77     12225
   macro avg       0.62      0.78      0.63     12225
weighted avg       0.91      0.77      0.82     12225
```

Advantages of Light Gradient boosting

- Faster traning time
- Lower memory usage
- Support parrallisation on distributed systems

Comment: No sign of overfitting.

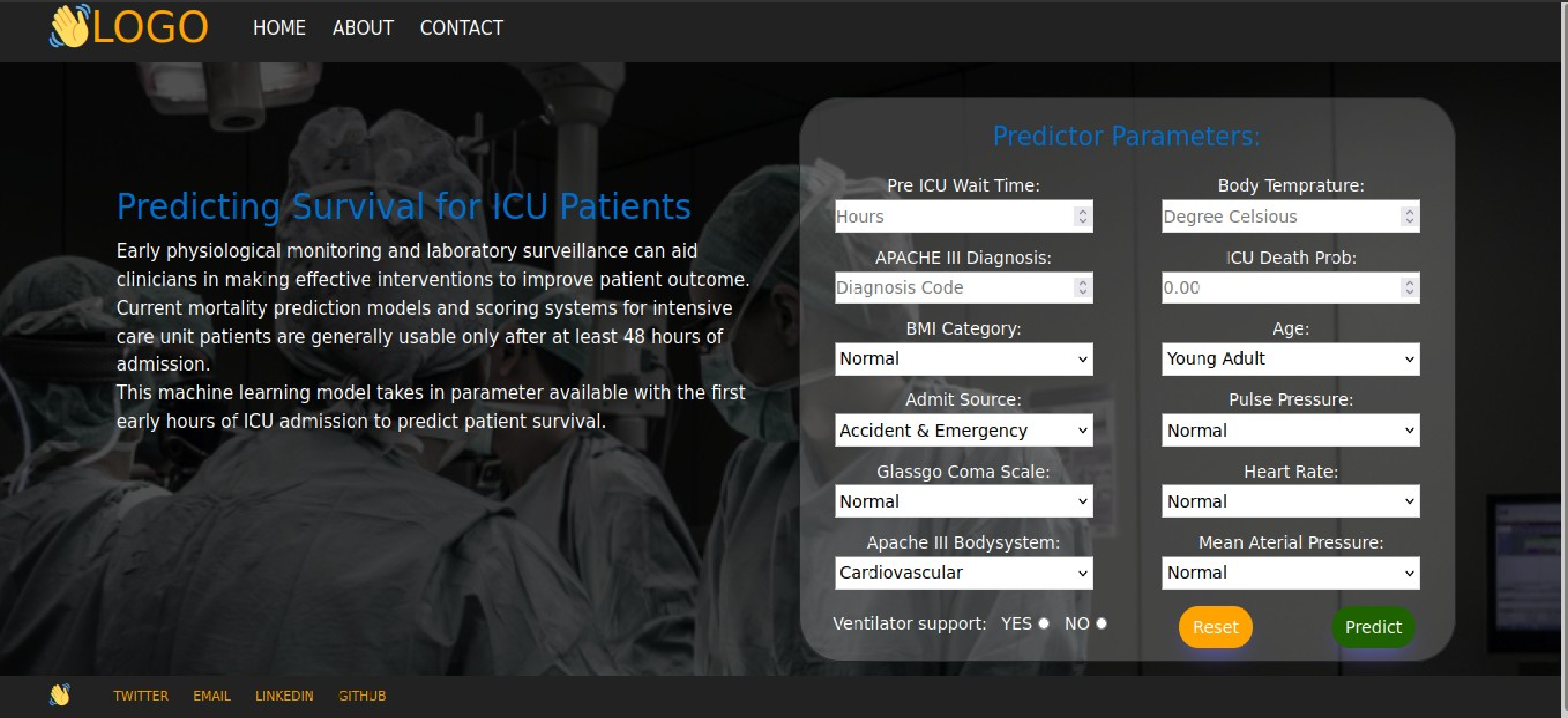
Introduction

DEPLOYMENT & CONCLUSION

Deployment & Conculsion

After data proccessing and fitting with different classifiers, Both light Gradient Boosting Classifier Gradient Boosting Classifier peformed best among other classifer with a roc_auc of 0.78 and recall of 0.78. - - - Light Gradient Boosting model was be safed as lgb_model.pkl and deployed on the with Django and free Heroku

Web Deployment



Introduction

IBM Developer
SKILLS NETWORK

APPENDIX

<Name>
<Date>

DICTIONARY

AUC:	Area U nder The Curve
ICU:	Intensive Care Unit
EDA:	Exploratory Data Analysis
APACHE	Acute physiological And Chronic Health Evaluation
BMI	Body Mass Index
h1_mbp_max:	One Hour Maximum Mean Blood Pressure
Med-Surg ICU:	Medical and Surgical Intensive Care Unit
map_cat:	Mean Aterial Pressure
CCU	Critical Care Unit
CSICU	Cradiac surgey intensive care unit
CTICU	Cardiothrocic Intensive Care Unit

THANK YOU

<Name>
<Date>