

Data Collection and Preprocessing Phase

Date	15 July 2024
Team ID	739706
Project Title	One Year Life Expectancy post on Thoracic Surgery using Machine Learning
Maximum Marks	6 Marks

Data Exploration and Preprocessing Report

Dataset variables will be statistically analyzed to identify patterns and outliers, with Python employed for pre-processing tasks like normalization and feature engineering. Data cleaning will address missing values and outliers, ensuring quality for subsequent analysis and modeling, and forming a strong foundation for insights and predictions.

Section

Data Overview

Description

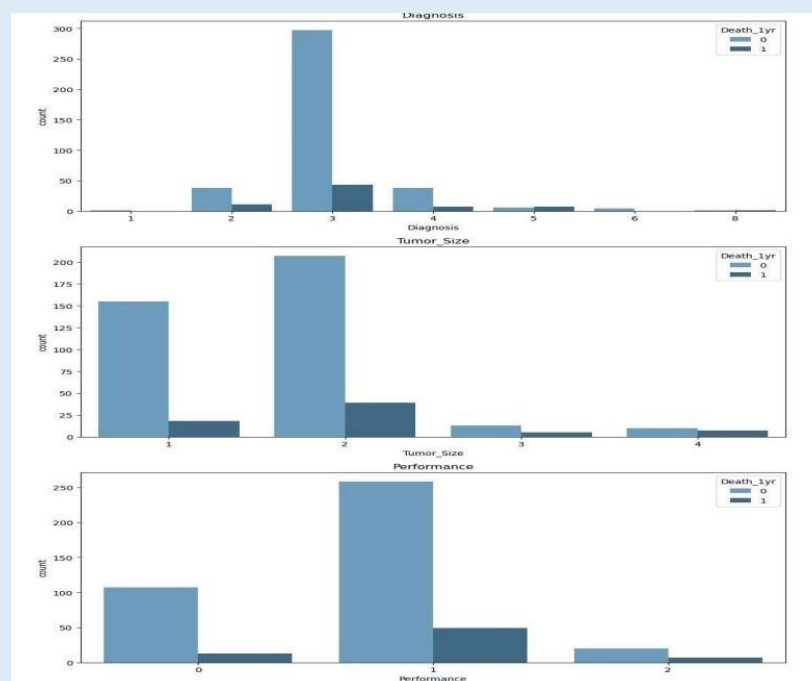
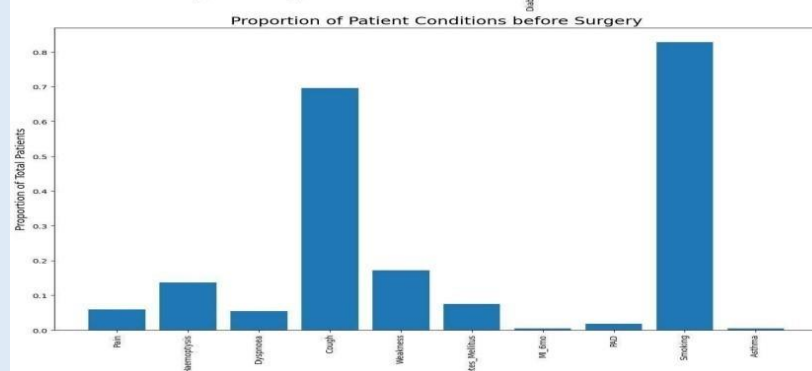
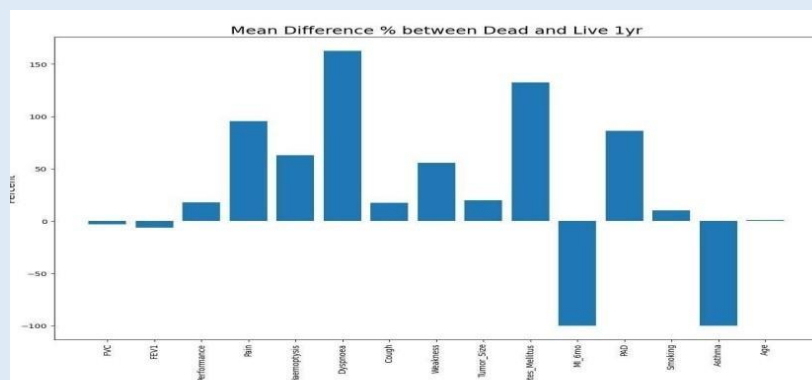
Dimension:

454 rows × 17 columns

Descriptive statistics:

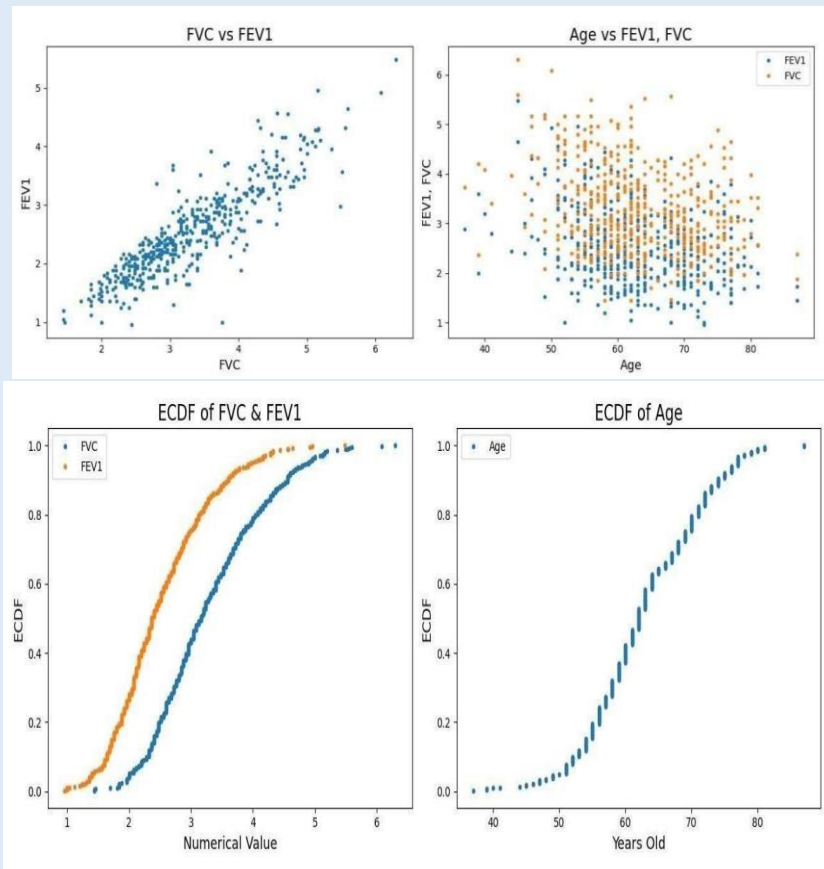
	Diagnosis	FVC	FEV1	Performance	Pain	Haemoptysis	Dyspnoea	Cough	Weakness	Tumor_Size	Diabetes_Mellitus	MI_6m
count	454.000000	454.000000	454.000000	454.000000	454.000000	454.000000	454.000000	454.000000	454.000000	454.000000	454.000000	454.000000
mean	3.092511	3.287952	2.51685	0.795154	0.059471	0.136564	0.055066	0.696035	0.171806	1.733480	0.074890	0.004400
std	0.715817	0.872347	0.77189	0.531459	0.236766	0.343765	0.228361	0.460475	0.377628	0.707499	0.263504	0.066200
min	1.000000	1.440000	0.960000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	1.000000	0.000000	0.000000
25%	3.000000	2.600000	1.960000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	1.000000	0.000000	0.000000
50%	3.000000	3.160000	2.360000	1.000000	0.000000	0.000000	0.000000	1.000000	0.000000	2.000000	0.000000	0.000000
75%	3.000000	3.840000	2.977500	1.000000	0.000000	0.000000	0.000000	1.000000	0.000000	2.000000	0.000000	0.000000
max	8.000000	6.300000	5.480000	2.000000	1.000000	1.000000	1.000000	1.000000	1.000000	4.000000	1.000000	1.000000

Exploratory Data analysis



--	--

Correlation Coefficient



Data Preprocessing Code Screenshots

Loading Data

```
In [29]: # import necessary libraries
import pandas as pd
import numpy as np
from sklearn.model_selection import train_test_split
from sklearn.linear_model import LogisticRegression
from sklearn.metrics import accuracy_score, f1_score, classification_report, confusion_matrix
import matplotlib.pyplot as plt
from sklearn.ensemble import RandomForestClassifier, GradientBoostingClassifier
from sklearn.tree import DecisionTreeClassifier
from sklearn.neighbors import KNeighborsClassifier
import itertools

# ignore warnings
warnings.filterwarnings('ignore')

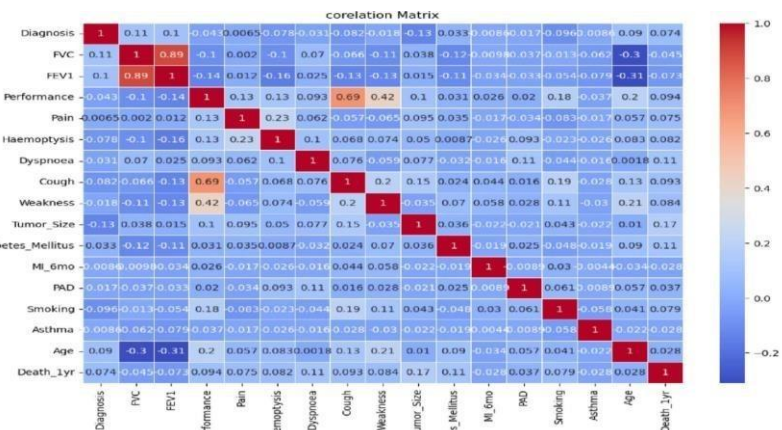
# Load your dataset
df = pd.read_csv('ThoracicSurgery.csv')

# Feature selection
# select features relevant for prediction
features = ['FVC', 'FEV1', 'Performance', 'Pain', 'Haemoptysis', 'Dyspnoea',
            'Cough', 'Weakness', 'Tumor_Size', 'Diabetes_Mellitus', 'MI_emo',
            'PAD', 'Smoking', 'Asthma', 'Age']
target = 'Death_Yr'

# Prepare the data
X = df[features]
y = df[target]

# Split the data into training and testing sets
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)
```

Correlation Matrix



Data Transformation

```
In [24]: x=df.iloc[:,0:15].values
         y=df.iloc[:,15:16].values

In [25]: x_train,x_test,y_train,y_test=train_test_split(x,y,test_size=0.2,random_state=0)

In [26]: print('Shape of x_train {}'.format(x_train.shape))
         print('Shape of y_train {}'.format(y_train.shape))
         print('Shape of x_test {}'.format(x_test.shape))
         print('Shape of y_test {}'.format(y_test.shape))

         Shape of x_train (363, 15)
         Shape of y_train (363, 1)
         Shape of x_test (91, 15)
         Shape of y_test (91, 1)

In [27]: from sklearn.preprocessing import StandardScaler

         # Standard scaling
         sc = StandardScaler()
         x_train = sc.fit_transform(x_train)
         x_test = sc.transform(x_test)
```

Feature Engineering

Attached the codes in final submission.

Save Processed Data	Data saved in the form of model .pkl file
---------------------	---