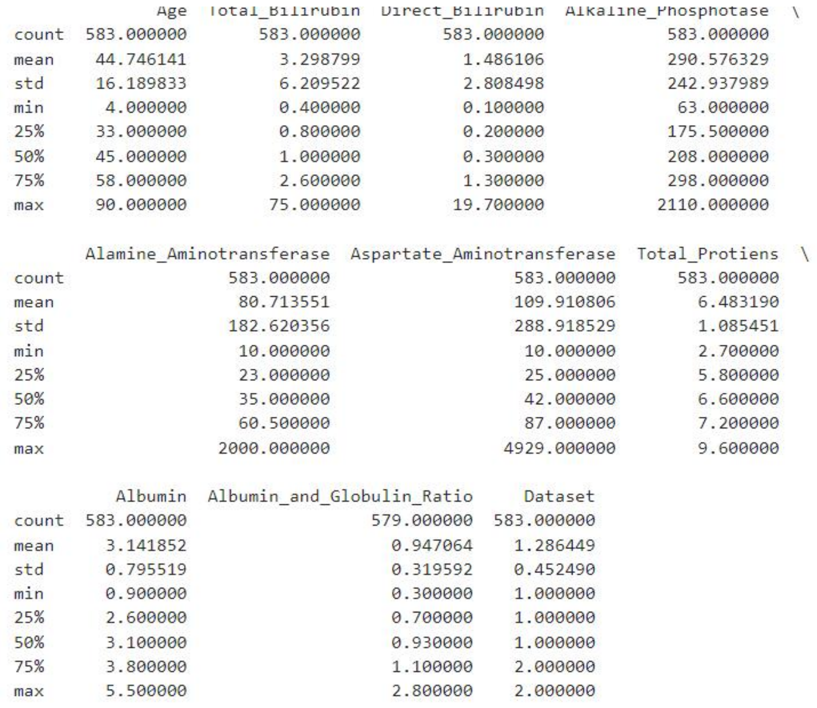
Data Collection and Pre processing Phase

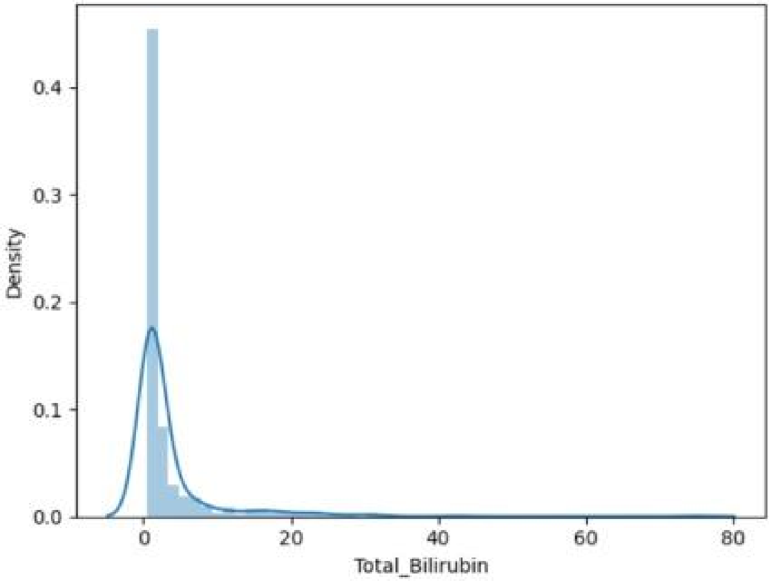
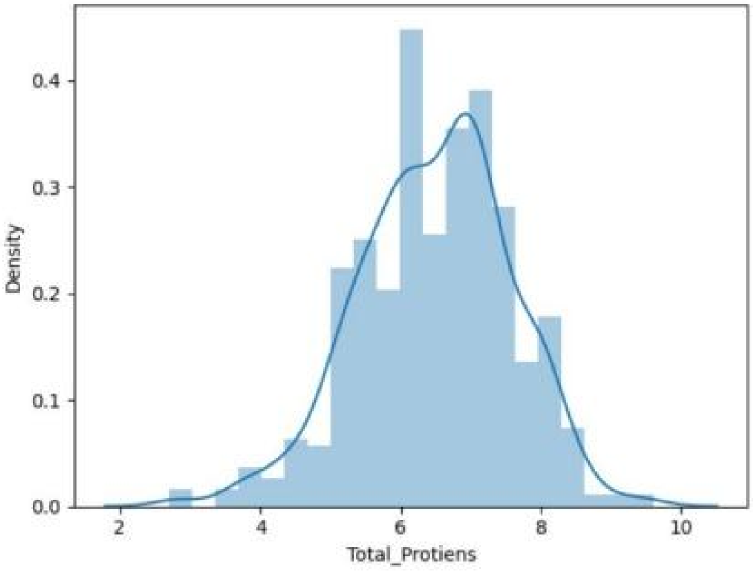
Date	09 July 2024
Team ID	SWTID1720023141
Project Title	Prediction and Analysis of Liver Patient Data Using Machine Learning
Maximum Marks	6 Marks

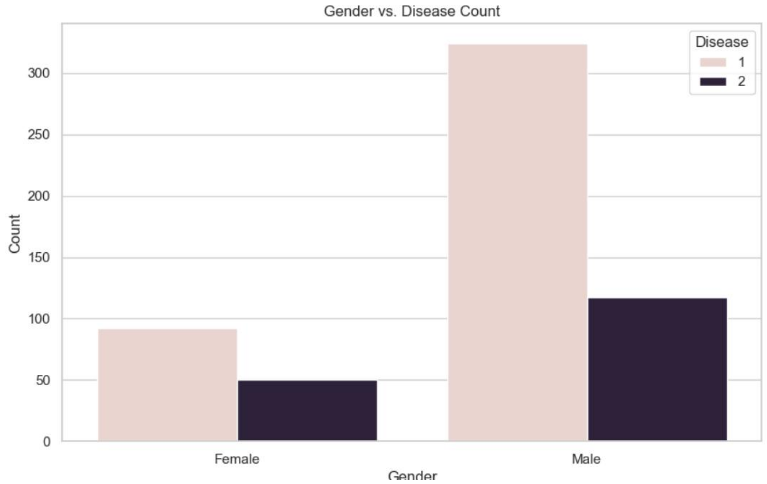
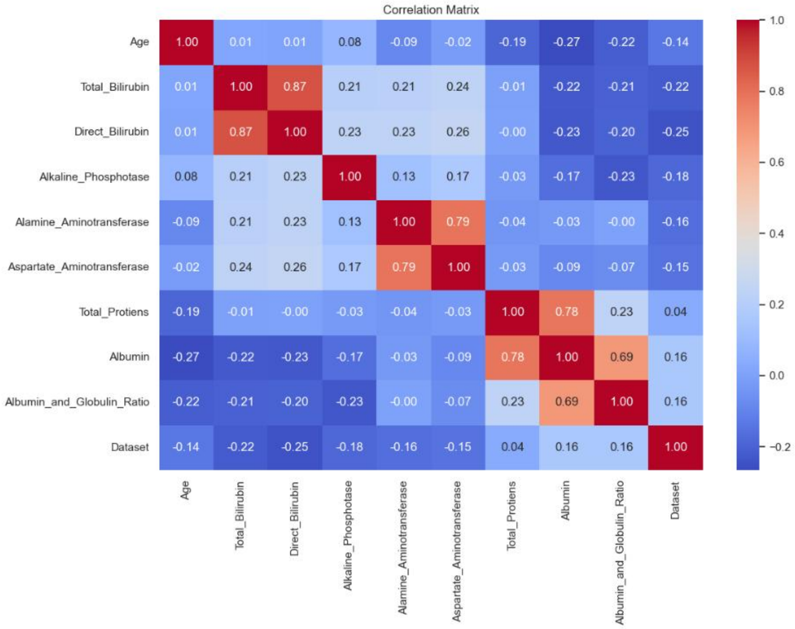
Data Exploration and Preprocessing Template

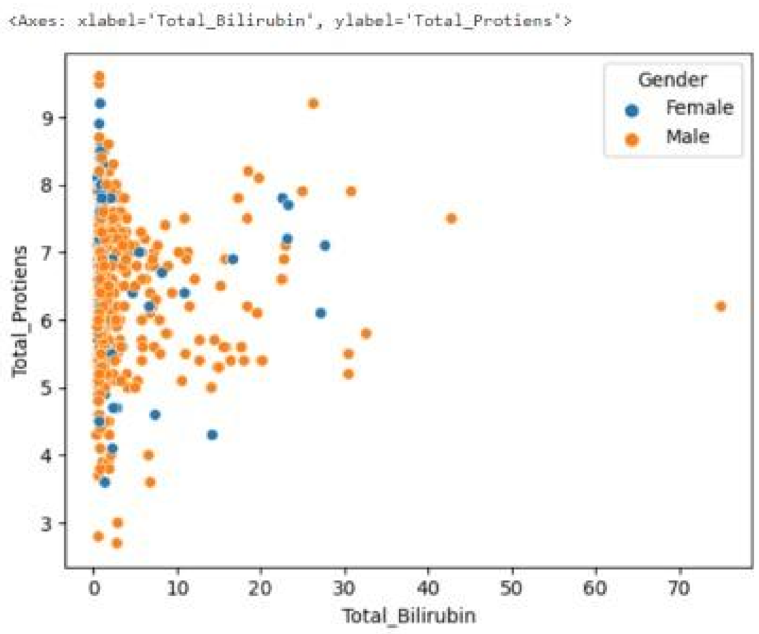
Identifies data sources, assesses quality issues like missing values and duplicates, and implements resolution plans to ensure accurate and reliable analysis.

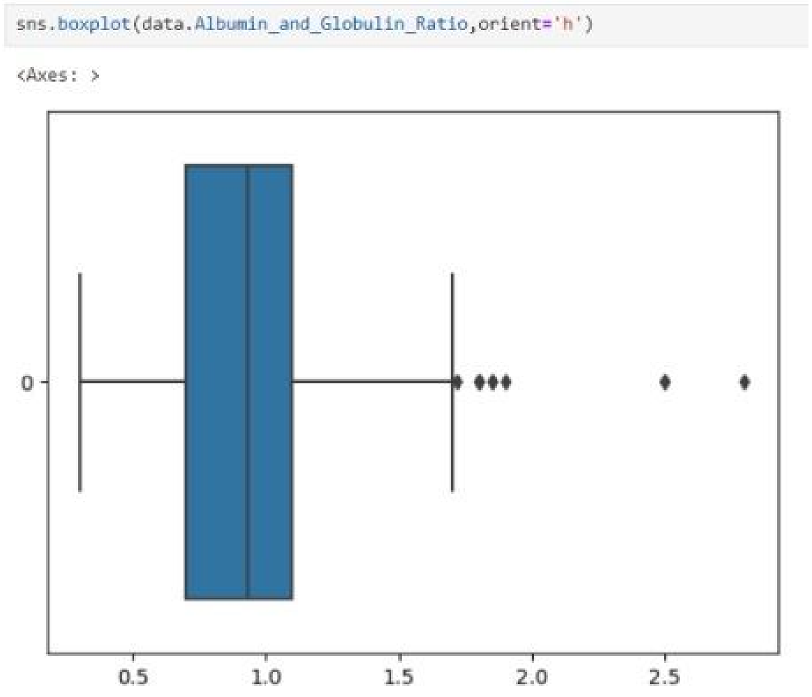
Section	Description
Data Overview	 .

```
              Age  Total_Bilirubin  Direct_Bilirubin  Alkaline_Phosphotase  \
count  583.000000       583.000000        583.000000            583.000000   
mean    44.746141         3.298799          1.486106            290.576329   
std     16.189833         6.209522          2.808498            242.937989   
min      4.000000         0.400000          0.100000             63.000000   
25%     33.000000         0.800000          0.200000            175.500000   
50%     45.000000         1.000000          0.300000            208.000000   
75%     58.000000         2.600000          1.300000            298.000000   
max     90.000000        75.000000         19.700000           2110.000000   

       Alamine_Aminotransferase  Aspartate_Aminotransferase  Total_Protiens  \
count                583.000000                  583.000000      583.000000   
mean                  80.713551                  109.910806        6.483190   
std                  182.620356                  288.918529        1.085451   
min                   10.000000                   10.000000        2.700000   
25%                   23.000000                   25.000000        5.800000   
50%                   35.000000                   42.000000        6.600000   
75%                   60.500000                   87.000000        7.200000   
max                 2000.000000                 4929.000000        9.600000   

          Albumin  Albumin_and_Globulin_Ratio     Dataset  
count  583.000000                  579.000000  583.000000  
mean     3.141852                    0.947064    1.286449  
std      0.795519                    0.319592    0.452490  
min      0.900000                    0.300000    1.000000  
25%      2.600000                    0.700000    1.000000  
50%      3.100000                    0.930000    1.000000  
75%      3.800000                    1.100000    2.000000  
max      5.500000                    2.800000    2.000000  
```

Univariate Analysis	`<Axes: xlabel='Total_Bilirubin', ylabel='Density'>`
	
	
	`<Axes: xlabel='Total_Protiens', ylabel='Density'>`
	

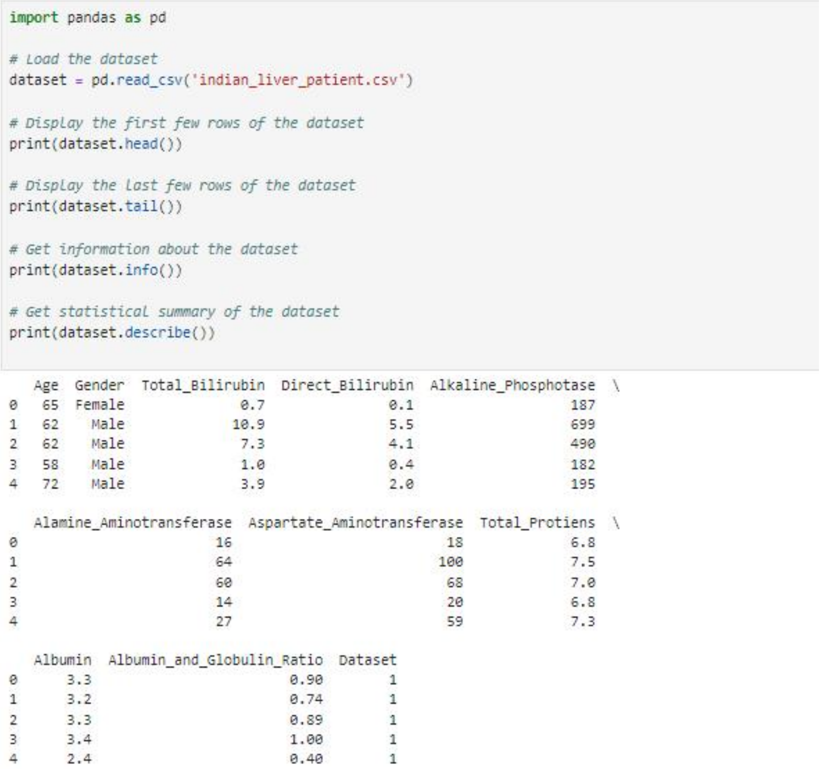
Bivariate Analysis	 .
Multivariate Analysis	

	 `<Axes: xlabel='Total_Bilirubin', ylabel='Total_Protiens'>`

Outliers and Anomalies	
	`sns.boxplot(data.Albumin_and_Globulin_Ratio,orient='h')`
	`<Axes: >`

Data Preprocessing Code Screenshots	

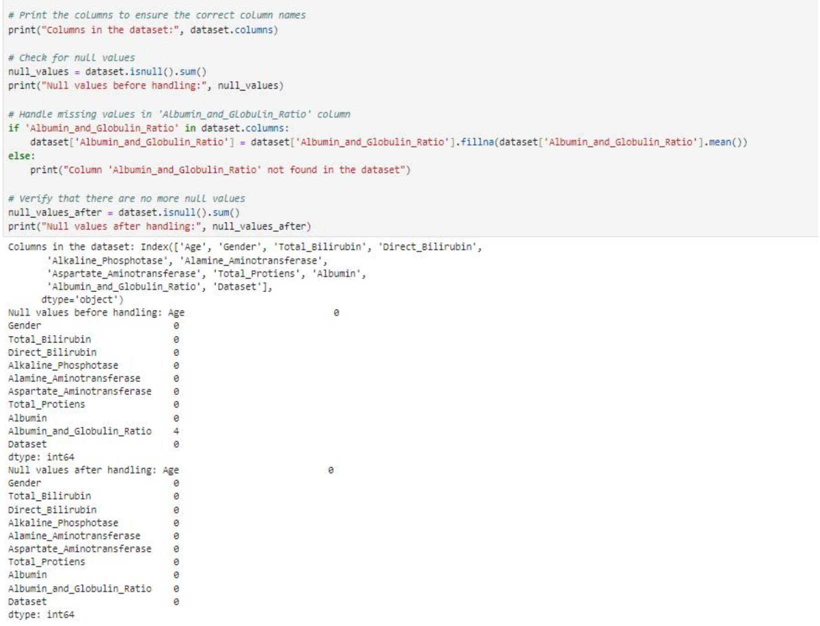
<table>
<tr>
<td>Loading Data</td>
<td>



```
import pandas as pd

# Load the dataset
dataset = pd.read_csv('indian_liver_patient.csv')

# Display the first few rows of the dataset
print(dataset.head())

# Display the last few rows of the dataset
print(dataset.tail())

# Get information about the dataset
print(dataset.info())

# Get statistical summary of the dataset
print(dataset.describe())
```

```
   Age  Gender  Total_Bilirubin  Direct_Bilirubin  Alkaline_Phosphotase  \
0   65  Female              0.7               0.1                   187
1   62    Male             10.9               5.5                   699
2   62    Male              7.3               4.1                   490
3   58    Male              1.0               0.4                   182
4   72    Male              3.9               2.0                   195

   Alamine_Aminotransferase  Aspartate_Aminotransferase  Total_Protiens  \
0                        16                          18             6.8
1                        64                         100             7.5
2                        60                          68             7.0
3                        14                          20             6.8
4                        27                          59             7.3

   Albumin  Albumin_and_Globulin_Ratio  Dataset
0      3.3                        0.90        1
1      3.2                        0.74        1
2      3.3                        0.89        1
3      3.4                        1.00        1
4      2.4                        0.40        1
```

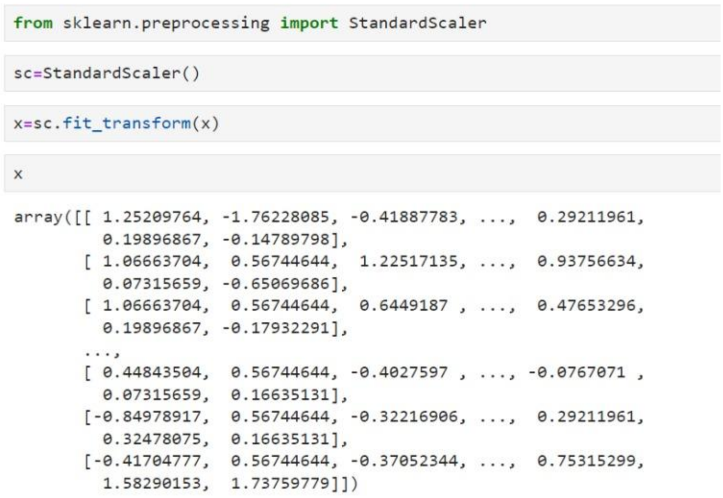
</td>
</tr>
<tr>
<td>Handling Missing Data</td>
<td>



```
# Print the columns to ensure the correct column names
print("Columns in the dataset:", dataset.columns)

# Check for null values
null_values = dataset.isnull().sum()
print("Null values before handling:", null_values)

# Handle missing values in 'Albumin_and_Globulin_Ratio' column
if 'Albumin_and_Globulin_Ratio' in dataset.columns:
    dataset['Albumin_and_Globulin_Ratio'] = dataset['Albumin_and_Globulin_Ratio'].fillna(dataset['Albumin_and_Globulin_Ratio'].mean())
else:
    print("Column 'Albumin_and_Globulin_Ratio' not found in the dataset")

# Verify that there are no more null values
null_values_after = dataset.isnull().sum()
print("Null values after handling:", null_values_after)
```

```
Columns in the dataset: Index(['Age', 'Gender', 'Total_Bilirubin', 'Direct_Bilirubin',
       'Alkaline_Phosphotase', 'Alamine_Aminotransferase',
       'Aspartate_Aminotransferase', 'Total_Protiens', 'Albumin',
       'Albumin_and_Globulin_Ratio', 'Dataset'],
      dtype='object')
Null values before handling: Age                           0
Gender                        0
Total_Bilirubin               0
Direct_Bilirubin              0
Alkaline_Phosphotase          0
Alamine_Aminotransferase      0
Aspartate_Aminotransferase    0
Total_Protiens                0
Albumin                       0
Albumin_and_Globulin_Ratio    4
Dataset                       0
dtype: int64
Null values after handling: Age                           0
Gender                        0
Total_Bilirubin               0
Direct_Bilirubin              0
Alkaline_Phosphotase          0
Alamine_Aminotransferase      0
Aspartate_Aminotransferase    0
Total_Protiens                0
Albumin                       0
Albumin_and_Globulin_Ratio    0
Dataset                       0
dtype: int64
```

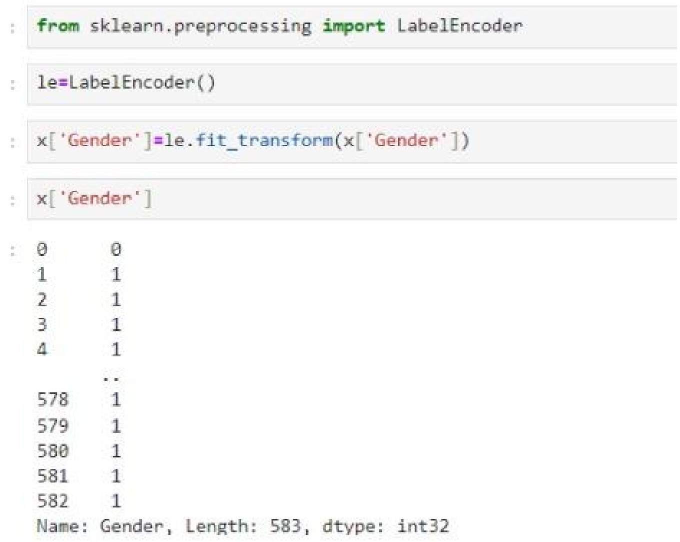
</td>
</tr>
</table>

<table>
<tr><td>Data Transformation</td><td>
<pre>
from sklearn.preprocessing import StandardScaler

sc=StandardScaler()

x=sc.fit_transform(x)

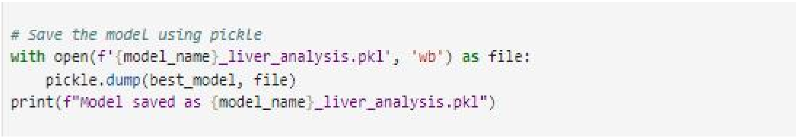
x

array([[1.25209764, -1.76228085, -0.41887783, ..., 0.29211961,
         0.19896867, -0.14789798],
       [1.06663704, 0.56744644, 1.22517135, ..., 0.93756634,
         0.07315659, -0.65069686],
       [1.06663704, 0.56744644, 0.6449187 , ..., 0.47653296,
         0.19896867, -0.17932291],
       ...,
       [0.44843504, 0.56744644, -0.4027597 , ..., -0.0767071 ,
         0.07315659, 0.16635131],
       [-0.84978917, 0.56744644, -0.32216906, ..., 0.29211961,
         0.32478075, 0.16635131],
       [-0.41704777, 0.56744644, -0.37052344, ..., 0.75315299,
         1.58290153, 1.73759779]])
</pre></td></tr>
<tr><td>Feature Engineering</td><td>
<pre>
from sklearn.preprocessing import LabelEncoder

le=LabelEncoder()

x['Gender']=le.fit_transform(x['Gender'])

x['Gender']

0   0
1   1
2   1
3   1
4   1
      ..
578   1
579   1
580   1
581   1
582   1
Name: Gender, Length: 583, dtype: int32
</pre></td></tr>
<tr><td>Save Processed Data</td><td>
<pre>
Save the model using pickle
with open(f'{model_name}_liver_analysis.pkl', 'wb') as file:
    pickle.dump(best_model, file)
print(f"Model saved as {model_name}_liver_analysis.pkl")
</pre></td></tr>
</table>