

# LIVER DISEASE ANALYSIS CODING OUTPUT

data.head()

The screenshot shows a Google Colab notebook with the following details:

- Files:** A file named `indian_liver_patient.csv` is located in the `sample_data` directory.
- Code Cell [4]:** Imports `pandas`, `numpy`, `seaborn`, `matplotlib.pyplot`, `rcParams`, and `stats`. It then reads the CSV file into a DataFrame and displays the first five rows.
- Output:** A table with 11 columns: `Age`, `Gender`, `Total_Bilirubin`, `Direct_Bilirubin`, `Alkaline_Phosphotase`, `Alamine_Aminotransferase`, `Aspartate_Aminotransferase`, `Total_Protiens`, `Albumin`, and `Al`. The first five rows of data are shown.

	Age	Gender	Total_Bilirubin	Direct_Bilirubin	Alkaline_Phosphotase	Alamine_Aminotransferase	Aspartate_Aminotransferase	Total_Protiens	Albumin	Al
0	65	Female	0.7	0.1	187		16	18	6.8	3.3
1	62	Male	10.9	5.5	699		64	100	7.5	3.2
2	62	Male	7.3	4.1	490		60	68	7.0	3.3
3	58	Male	1.0	0.4	182		14	20	6.8	3.4
4	72	Male	3.9	2.0	195		27	59	7.3	2.4

data.info()

The screenshot shows a Google Colab notebook with the following details:

- Files:** A file named `indian_liver_patient.csv` is located in the `sample_data` directory.
- Code Cell [3]:** Executes `data.info()` to display the structure of the DataFrame.
- Output:** A text summary of the DataFrame structure, including the number of entries, columns, and data types.

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 583 entries, 0 to 582
Data columns (total 11 columns):
#   Column                                Non-Null Count  Dtype  
---  -
0   Age                                   583 non-null   int64  
1   Gender                               583 non-null   object  
2   Total_Bilirubin                       583 non-null   float64 
3   Direct_Bilirubin                      583 non-null   float64 
4   Alkaline_Phosphotase                  583 non-null   int64  
5   Alamine_Aminotransferase              583 non-null   int64  
6   Aspartate_Aminotransferase            583 non-null   int64  
7   Total_Protiens                        583 non-null   float64 
8   Albumin                              583 non-null   float64 
9   Albumin_and_Globulin_Ratio            579 non-null   float64 
10  Dataset                              583 non-null   int64  
dtypes: float64(5), int64(5), object(1)
memory usage: 58.2+ KB
```

data.isnull().any()

The screenshot shows a Google Colab notebook with the following content:

```
[5] 2 Total_Bilirubin      583 non-null    float64
     3 Direct_Bilirubin   583 non-null    float64
     4 Alkaline_Phosphotase 583 non-null    int64
     5 Alamine_Aminotransferase 583 non-null    int64
     6 Aspartate_Aminotransferase 583 non-null    int64
     7 Total_Protiens     583 non-null    float64
     8 Albumin            583 non-null    float64
     9 Albumin_and_Globulin_Ratio 579 non-null    float64
    10 Dataset            583 non-null    int64
dtypes: float64(5), int64(5), object(1)
memory usage: 50.2+ KB
```

The output of `data.isnull().any()` is displayed as a table:

Age	False
Gender	False
Total_Bilirubin	False
Direct_Bilirubin	False
Alkaline_Phosphotase	False
Alamine_Aminotransferase	False
Aspartate_Aminotransferase	False
Total_Protiens	False
Albumin	False
Albumin_and_Globulin_Ratio	True
Dataset	False
dtype:	bool

Below the output, the following code is visible:

```
[4] import pandas as pd
import numpy as np
import seaborn as sns
import matplotlib.pyplot as plt
from matplotlib import rcParams
from scipy import stats
```

The notebook interface shows the file explorer on the left with 'sample\_data' and 'indian\_liver\_patient.csv'. The bottom status bar indicates 'completed at 11:29 AM'.

data.isnull().sum()

The screenshot shows a Google Colab notebook with the following content:

```
data.isnull().sum()
```

The output is displayed as a table:

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

The notebook interface shows the file explorer on the left with 'sample\_data' and 'indian\_liver\_patient.csv'. The bottom status bar indicates 'completed at 11:30 AM'.

## data.describe()

Copy of Untitled2.ipynb - Colab: x Untitled3.ipynb - Colaboratory x +

colab.research.google.com/drive/1gckir6YbbjhDa-aU2vP1Ucblxc66Mzyy#scrollTo=vGrcIC1GS3BN

Gmail YouTube Maps Definition of learning WhatsApp Scimago Journal &...

Untitled3.ipynb ☆

File Edit View Insert Runtime Tools Help All changes saved

Comment Share

Files

sample\_data

Indian\_liver\_patient.csv

+ Code + Text

data.describe()

	Age	Total_Bilirubin	Direct_Bilirubin	Alkaline_Phosphatase	Alamine_Aminotransferase	Aspartate_Aminotransferase	Total_Protiens	Albumin
count	583.000000	583.000000	583.000000	583.000000	583.000000	583.000000	583.000000	583.000000
mean	44.746141	3.298799	1.486106	290.576329	80.713551	109.910805	6.483190	3.141800
std	16.189833	6.209522	2.808498	242.937989	182.620356	288.918529	1.085451	0.795500
min	4.000000	0.400000	0.100000	63.000000	10.000000	10.000000	2.700000	0.900000
25%	33.000000	0.800000	0.200000	175.500000	23.000000	25.000000	5.800000	2.600000
50%	45.000000	1.000000	0.300000	208.000000	35.000000	42.000000	6.600000	3.100000
75%	58.000000	2.600000	1.300000	298.000000	60.500000	87.000000	7.200000	3.800000
max	90.000000	75.000000	19.700000	2110.000000	2000.000000	4929.000000	9.600000	5.500000

84.49 GB available

completed at 11:31 AM

31°C Mostly cloudy 11:31 11-04-2023

Copy of Untitled2.ipynb - Colab: x Untitled3.ipynb - Colaboratory x +

colab.research.google.com/drive/1XPfryfx-UVTkagEcdGmHpsSICtDQLMa#scrollTo=3VwNuc6jmgYT

Gmail YouTube Maps Definition of learning WhatsApp Scimago Journal &...

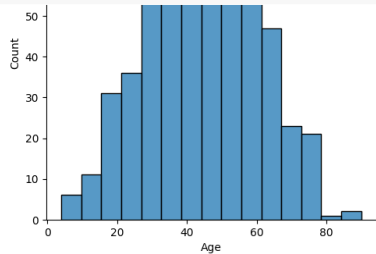
Copy of Untitled2.ipynb ☆

File Edit View Insert Runtime Tools Help All changes saved

Comment Share

+ Code + Text

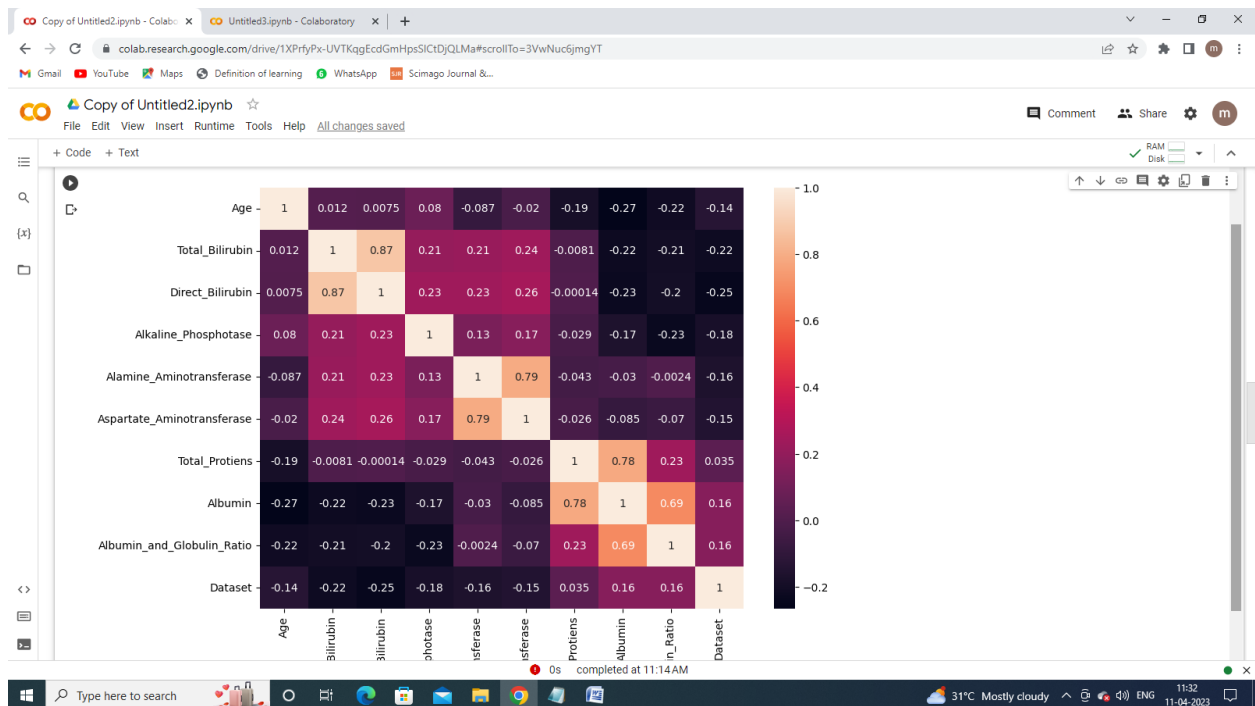
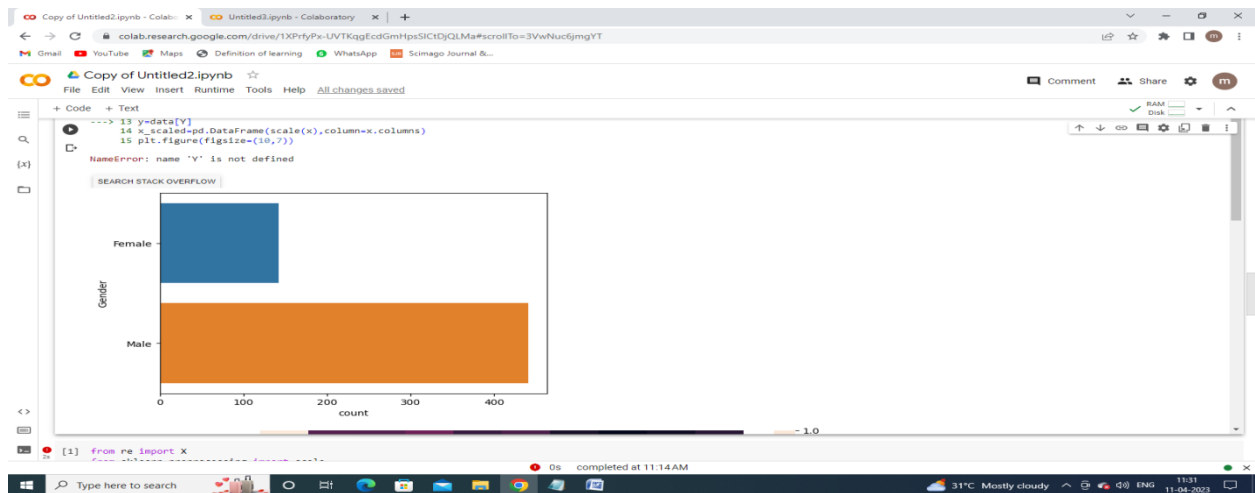
```
sns.heatmap(df.corr(),annot=True)
from sklearn.preprocessing import scale
x=data.iloc[:, :-1]
y=data[Y]
x_scaled=pd.DataFrame(scale(x),column=x.columns)
plt.figure(figsize=(10,7))
sns.heatmap(df.corr(),annot=True)
```



```
-----
NameError                                Traceback (most recent call last)
<ipython-input-47-c2a82aa14360> in <cell line: 13>()
     11 from sklearn.preprocessing import scale
     12 x=data.iloc[:, :-1]
--> 13 y=data[Y]
     14 x_scaled=pd.DataFrame(scale(x),column=x.columns)
```

completed at 11:14 AM

31°C Mostly cloudy 11:32 11-04-2023



```
1 from sklearn.preprocessing import scale
2 X_scaled=pd.DataFrame (scale(x), columns=x.columns)
```

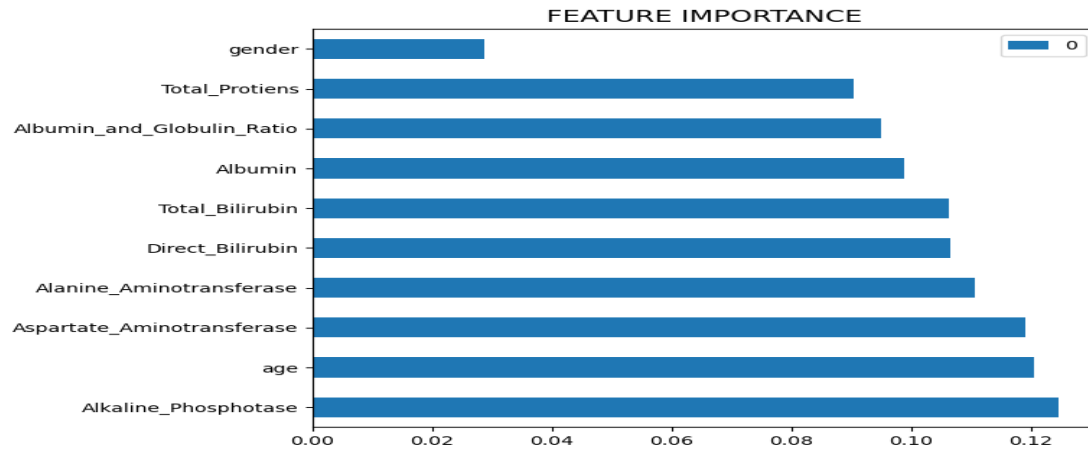
```
1 X_scaled.head()
```

	age	gender	Total_Bilirubin	Direct_Bilirubin	Alkaline_Phosphotase
0	1.252098	-1.762281	-0.418878	-0.493964	-0.426715
1	1.066637	0.567446	1.225171	1.430423	1.682629
2	1.066637	0.567446	0.644919	0.931508	0.821588
3	0.819356	0.567446	-0.370523	-0.387054	-0.447314
4	1.684839	0.567446	0.096902	0.183135	-0.393756

```

1 dd.plot(kind='barh', figsize=(7,6))
2 plt.title("FEATURE IMPORTANCE",fontsize=14)
Text(0.5, 1.0, 'FEATURE IMPORTANCE')

```



**Direct\_Bilirubin & Total\_Bilirubin are the most important features to predict the outcome**

```

1 from sklearn.ensemble import ExtraTreesClassifier
2 model=ExtraTreesClassifier()
3 model.fit(X,y)

```

ExtraTreesClassifier()

```

1 model.feature_importances_

```

```

array([0.1205029 , 0.02863187, 0.10625368, 0.10648548, 0.1245292 ,
        0.11049943, 0.118963 , 0.09033392, 0.09882431, 0.09497621])

```

```

1 dd=pd.DataFrame(model.feature_importances_,index=X.columns).sort_values(0,ascending=False)
2 dd

```

	0
Alkaline_Phosphotase	0.124529
age	0.120503
Aspartate_Aminotransferase	0.118963
Alanine_Aminotransferase	0.110499
Direct_Bilirubin	0.106485
Total_Bilirubin	0.106254
Albumin	0.098824
Albumin_and_Globulin_Ratio	0.094976
Total_Protiens	0.090334
gender	0.028632

```
1 plt.figure(figsize=(7,5))
2 plt.xticks(rotation=90)
3 plt.title('Classification models & accuracy scores after SMOTE',fontsize=18)
4 sns.barplot(x="classification models", y="accuracy_score", data=Liverpatient_pred,palette = "Set2")
```

<AxesSubplot:title={'center':'Classification models & accuracy scores after SMOTE'}, xlabel='classification  
l='accuracy\_score'>

## Classification models & accuracy scores after SMOTE

