

```
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
```

```
data=pd.read_csv('/content/indian_liver_patient.csv')
data.head()
```

	Age	Gender	Total_Bilirubin	Direct_Bilirubin	Alkaline_Phosphotase	Alamine_Ami
0	65	Female	0.7	0.1	187	
1	62	Male	10.9	5.5	699	
2	60	Male	7.0	1.1	100	
3	60	Male	7.0	1.1	100	
4	72	Male	3.9	2.0	195	

Automatic saving failed. This file was updated remotely or in another tab. [Show diff](#)

```
data.info()
```

```
<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: 50.2+ KB
```

```
data.isnull().any()
```

```
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
```

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

```
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
```

Automatic saving failed. This file was updated remotely or in another tab. [Show diff](#)

```
data[ 'Albumin_and_Globulin_Ratio' ]=data.fillna(data[ 'Albumin_and_Globulin_Ratio' ])
data.isnull().sum
```

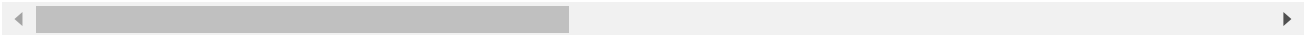
task-3

Double-click (or enter) to edit

task-3

```
data.describe()
```

	Age	Total_Bilirubin	Direct_Bilirubin	Alkaline_Phosphotase	Alamine_
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	



```
sns.distplot(data[ 'Age' ])
plt.title('Age Distribution Graph')
```

```
plt.show()
```

```
<ipython-input-84-a9533a3b6a8d>:1: UserWarning:
```

```
`distplot` is a deprecated function and will be removed in seaborn v0.14.0.
```

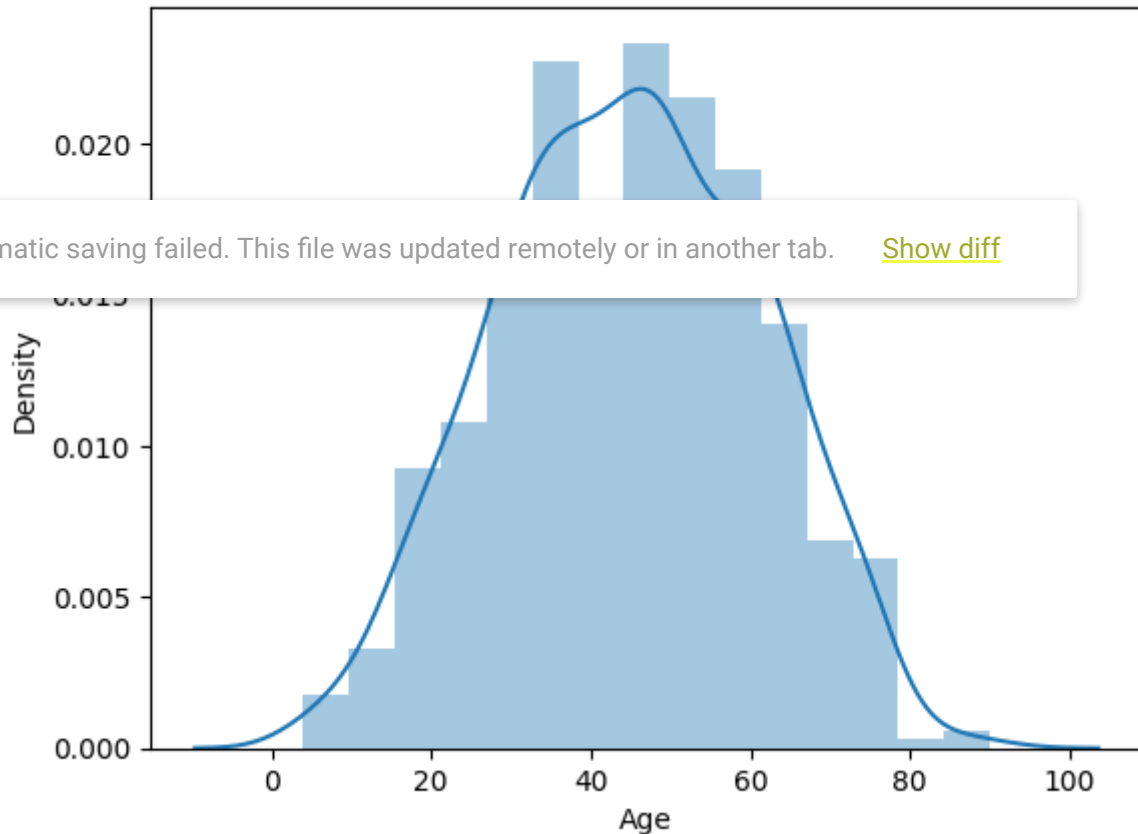
Please adapt your code to use either `displot` (a figure-level function with similar flexibility) or `histplot` (an axes-level function for histograms).

For a guide to updating your code to use the new functions, please see

<https://gist.github.com/mwaskom/de44147ed2974457ad6372750bbe5751>

```
sns.distplot(data['Age'])
```

Age Distribution Graph



```
sns.countplot
```

```
<function seaborn.categorical.countplot(data=None, *, x=None, y=None, hue=None,
order=None, hue_order=None, orient=None, color=None, palette=None, saturation=0.75,
width=0.8, dodge=True, ax=None, **kwargs)>
```

```
plt.figure(figsize=(10,7))
```

```
<Figure size 1000x700 with 0 Axes>
```

```
<Figure size 1000x700 with 0 Axes>
```

```
from sklearn.preprocessing import scale
X=data.iloc[:, :-1]
y=data.outcome
X_scaled=pd.DataFrame(scale(X), columns=X.columns)
```

```
X_scaled.head()
from sklearn.model_selection import train_test_split
X_train,X_test,y_train,y_test=train_test_split((X_scaled,y,test_size=0.2,random_state=42)
```

```
from imblearn.over_sampling import SMOTE
smote=SMOTE()
X_train_smote,y_train_smote=smote.fit_resample(X)
y_train=pd.value_counts(X_train,y_train)
y_train.value_counts
```

Automatic saving failed. This file was updated remotely or in another tab. [Show diff](#)

! 0s completed at 12:17 PM

● ×