```
1 # Importing the libraries
2 import numpy as np
3 import matplotlib.pyplot as plt
4 import pandas as pd
```

```
1
2 # Importing the dataset
3 df = pd.read_csv('Data.csv')
```

```
1 df['Salary'].fillna((df['Salary'].mean()), inplace= True)
```

```
1 df['Age'].fillna((df['Age'].mean()), inplace= True)
```

```
1 df['Purchased'] = df['Purchased'].apply(lambda x: 0 if x=='No' else 1)
```

```
1 df=pd.get_dummies(data=df, columns=['Country'])
```

```
1 df
```

	Age	Salary	Purchased	Country_France	Country_Germany	Country_Spain
0	44.000000	72000.000000	0	1	0	0
1	27.000000	48000.000000	1	0	0	1

```
1 from sklearn.preprocessing import MinMaxScaler
2 scaler = MinMaxScaler()
```

	Age	Salary	Purchased	Country_France	Country_Germany	Country_Spain
3	38.000000	61000.000000	0	0	0	1

```
1 scaler.fit(df)
```

MinMaxScaler(copy=True, feature_range=(0, 1))

	Age	Salary	Purchased	Country_France	Country_Germany	Country_Spain
6	38.777778	52000.000000	0	0	0	1

```
1 MinMaxScaler(copy=True, feature_range=(0, 1))
```

MinMaxScaler(copy=True, feature_range=(0, 1))

	Age	Salary	Purchased	Country_France	Country_Germany	Country_Spain
8	37.000000	67000.000000	1	1	0	0

```
1 scaled_features = scaler.transform(df)
```

```
1 scaled_features
```

```
array([[0.73913043, 0.68571429, 0.        , 1.        , 0.        ,
        0.        ],
       [0.        , 0.        , 1.        , 0.        , 0.        ,
        1.        ],
       [0.13043478, 0.17142857, 0.        , 0.        , 1.        ,
        0.        ],
       [0.47826087, 0.37142857, 0.        , 0.        , 0.        ,
        1.        ],
       [0.56521739, 0.45079365, 1.        , 0.        , 1.        ,
        0.        ],
       [0.34782609, 0.28571429, 1.        , 1.        , 0.        ,
        0.        ],
       [0.51207729, 0.11428571, 0.        , 0.        , 0.        ,
        1.        ],
       [0.91304348, 0.88571429, 1.        , 1.        , 0.        ,
        0.        ],
       [1.        , 1.        , 0.        , 0.        , 1.        ,
        0.        ],
```

```
        [0.43478261, 0.54285714, 1.          , 1.          , 0.          ,
         0.          ]])
```

```
1 df_MinMax = pd.DataFrame(data=scaled_features, columns=["Age", "Salary","Purchased","
2 df_MinMax
```

|	Age	Salary	Purchased	Country_France	Country_Germany	Country_spain
0	0.739130	0.685714	0.0	1.0	0.0	0.0
1	0.000000	0.000000	1.0	0.0	0.0	1.0
2	0.130435	0.171429	0.0	0.0	1.0	0.0
3	0.478261	0.371429	0.0	0.0	0.0	1.0
4	0.565217	0.450794	1.0	0.0	1.0	0.0
5	0.347826	0.285714	1.0	1.0	0.0	0.0
6	0.512077	0.114286	0.0	0.0	0.0	1.0
7	0.913043	0.885714	1.0	1.0	0.0	0.0
8	1.000000	1.000000	0.0	0.0	1.0	0.0
9	0.434783	0.542857	1.0	1.0	0.0	0.0

```
1 import seaborn as sns
2 plt.figure(figsize=(12,8))
3 sns.distplot(df_MinMax['Salary'])
4 import statistics
5 print("Standard Deviation of df_MinMax_Salary is % s "
6                 % (statistics.stdev(df_MinMax['Salary'])))
```

```
/usr/local/lib/python3.7/dist-packages/seaborn/distributions.py:2557: FutureWarning: `distplot` is a deprecated functi
  warnings.warn(msg, FutureWarning)
Standard Deviation of df_MinMax_Salary is 0.33040284015892535
```
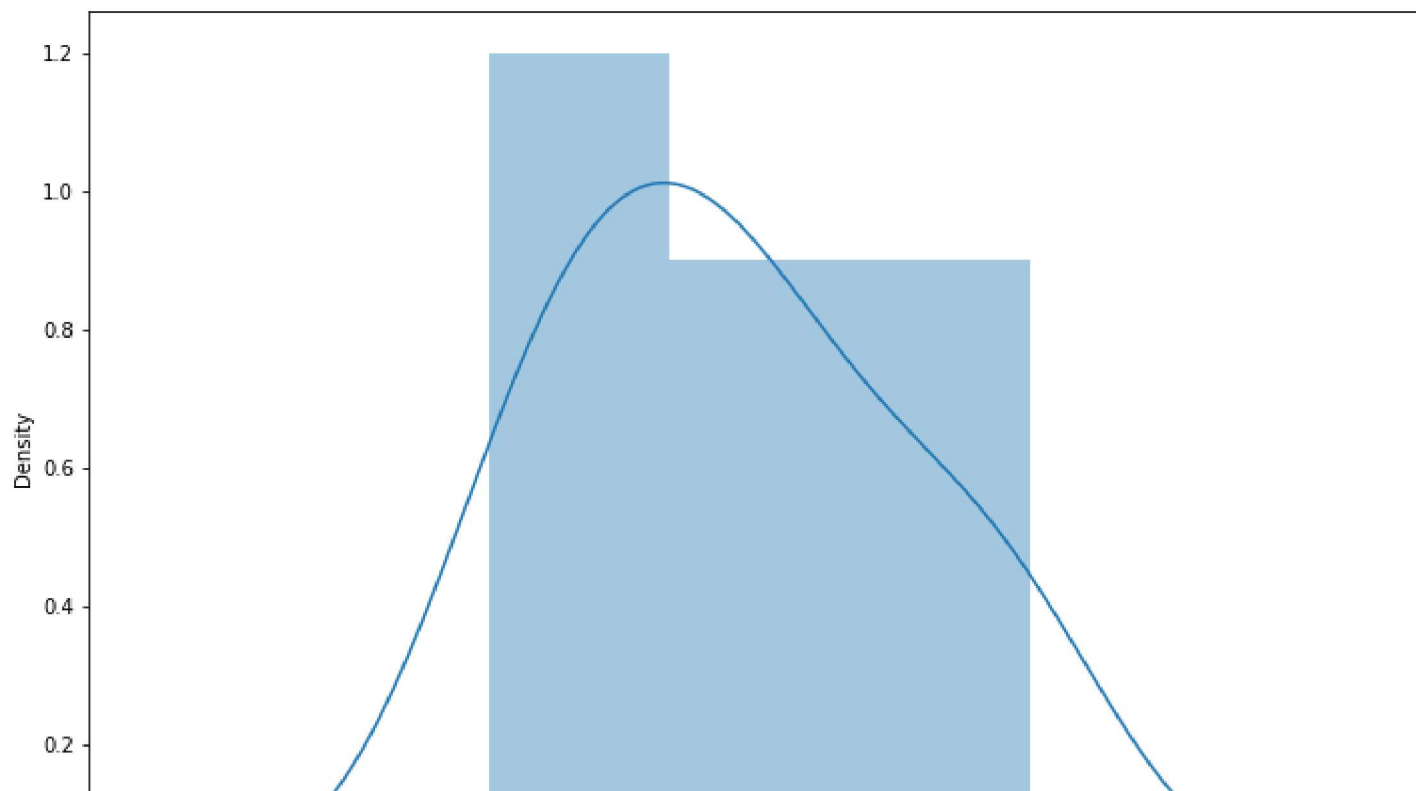


```
1 plt.figure(figsize=(12,8))
2 sns.distplot(df_MinMax['Age'])
3 print("Standard Deviation of df_MinMax_Age is % s "
4                % (statistics.stdev(df_MinMax['Age'])))
```

```
/usr/local/lib/python3.7/dist-packages/seaborn/distributions.py:2557: FutureWarning: `distplot` is a deprecated functi
  warnings.warn(msg, FutureWarning)
Standard Deviation of df_MinMax_Age is 0.3153816182405694
```

1
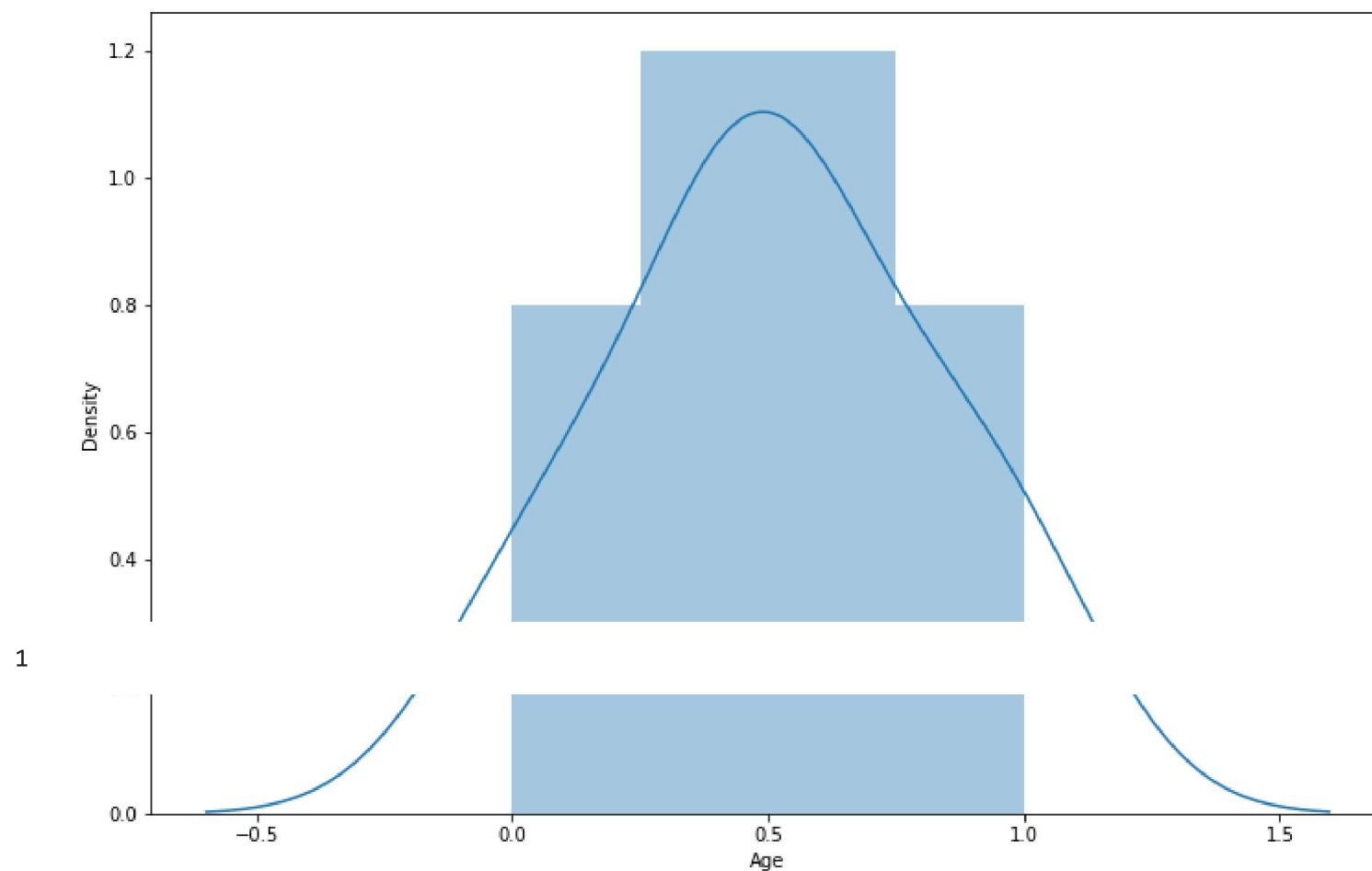
✓ 0s   completed at 9:45 AM   ● ✕

Could not connect to the reCAPTCHA service. Please check your internet connection and reload to get a reCAPTCHA challenge.