Assignment No 1

```
[1] import pandas as pd

import numpy as np

[3] df=[URL_39]ad_csv("Heart.csv")

[4] [URL_39]scribe()
```

[DATE_TIME_0]

[4]:

Unnamed: 0

Age

Sex

count

303.[US_DRIVER_LICENSE_181] 303.[US_DRIVER_LICENSE_181]

mean

152.[US_DRIVER_LICENSE_181]

54.[US_DRIVER_LICENSE_181]

std

87.[US_DRIVER_LICENSE_181]

9.[US_DRIVER_LICENSE_181]

min

1.[US_DRIVER_LICENSE_181]

29.[US_DRIVER_LICENSE_181]

RestBP
303.[US_DRIVER_LICENSE_181] 303.[US_DRIVER_LICENSE_181] 303.[US_DRIVER_LICENSE_181]
0.[US_DRIVER_LICENSE_181] 131.[US_DRIVER_LICENSE_181] 246.[US_DRIVER_LICENSE_181]
[PHONE_NUMBER_47] 17.[US_DRIVER_LICENSE_181] 51.[US_DRIVER_LICENSE_181]
0.[US_DRIVER_LICENSE_181] 94.[US_DRIVER_LICENSE_181] 126.[US_DRIVER_LICENSE_181]

Chol

Fbs

■303.[US_DRIVER_LICENSE_181]

0.[US_DRIVER_LICENSE_181]

[PHONE_NUMBER_47]

0.[US_DRIVER_LICENSE_181]

25%

76.[US_DRIVER_LICENSE_181]

48.[US_DRIVER_LICENSE_181]

0.[US_DRIVER_LICENSE_181] 120.[US_DRIVER_LICENSE_181] 211.[US_DRIVER_LICENSE_181]

0.[US_DRIVER_LICENSE_181]

50%

152.[US_DRIVER_LICENSE_181]

56.[US_DRIVER_LICENSE_181]

75%

max

227.[US_DRIVER_LICENSE_181] 61.[US_DRIVER_LICENSE_181]
303.[US_DRIVER_LICENSE_181] 77.[US_DRIVER_LICENSE_181]

1.[US_DRIVER_LICENSE_181] 130.[US_DRIVER_LICENSE_181] 241.[US_DRIVER_LICENSE_181]
1.[US_DRIVER_LICENSE_181] 140.[US_DRIVER_LICENSE_181] 275.[US_DRIVER_LICENSE_181]
1.[US_DRIVER_LICENSE_181] 200.[US_DRIVER_LICENSE_181]

0.[US_DRIVER_LICENSE_181]

0.[US_DRIVER_LICENSE_181]

564.[US_DRIVER_LICENSE_181]

1.[US_DRIVER_LICENSE_181]

count

RestECG
303.[US_DRIVER_LICENSE_181]

MaxHR

■mean

std

303.[US_DRIVER_LICENSE_181]
[PHONE_NUMBER_47]
[PHONE_NUMBER_47]

ExAng
303.[US_DRIVER_LICENSE_181]

[NRP_4]
303.[US_DRIVER_LICENSE_181]

Slope

Ca

303.[US_DRIVER_LICENSE_181]

299.[US_DRIVER_LICENSE_181]

0.[US_DRIVER_LICENSE_181]

1.[US_DRIVER_LICENSE_181]

1.[US_DRIVER_LICENSE_181]

[PHONE_NUMBER_47]

0.[US_DRIVER_LICENSE_181]

1.[US_DRIVER_LICENSE_181]

[PHONE_NUMBER_47]

[PHONE_NUMBER_47]

min

0.[US_DRIVER_LICENSE_181] 71.[US_DRIVER_LICENSE_181]

0.[US_DRIVER_LICENSE_181]

0.[US_DRIVER_LICENSE_181]

1.[US_DRIVER_LICENSE_181]

0.[US_DRIVER_LICENSE_181]

25%

■0.[US_DRIVER_LICENSE_181] 133.[US_DRIVER_LICENSE_181]

0.[US_DRIVER_LICENSE_181]

0.[US_DRIVER_LICENSE_181]

1.[US_DRIVER_LICENSE_181]

0.[US_DRIVER_LICENSE_181]

50%

1.[US_DRIVER_LICENSE_181] 153.[US_DRIVER_LICENSE_181]

0.[US_DRIVER_LICENSE_181]

0.[US_DRIVER_LICENSE_181]

2.[US_DRIVER_LICENSE_181]

0.[US_DRIVER_LICENSE_181]

75%

2.[US_DRIVER_LICENSE_181] 166.[US_DRIVER_LICENSE_181]

1.[US_DRIVER_LICENSE_181]

1.[US_DRIVER_LICENSE_181]

2.[US_DRIVER_LICENSE_181]

1.[US_DRIVER_LICENSE_181]

max

2.[US_DRIVER_LICENSE_181] 202.[US_DRIVER_LICENSE_181]

1.[US_DRIVER_LICENSE_181]

6.[US_DRIVER_LICENSE_181]

3.[US_DRIVER_LICENSE_181]

3.[US_DRIVER_LICENSE_181]

:

[5] [URL_39]ape

■[5] (303, 15)

[6] [URL_39]null().sum()

[6] Unnamed: 0

Age
Sex

Chest Pain

0

0

O O O O

1

RestBP

Chol

Fbs

0

RestECG

MaxHR

0

ExAng

Oldpeak

Slope

Ca

Thal

AHD

0

0

O o O 420

```
■dtype: int64

[8]: print("Total Missing Records are:", [URL_39]null().sum().sum())

Total Missing Records are: 6

[9] df.dtypes

:

:

[9] Unnamed: 0

int64

Age
Sex

int64

int64

Chest Pain

object

RestBP

int64

Chol

int64

Fbs

int64

RestECG

int64

MaxHR

int64

ExAng

int64
```

■[NRP_4]

float64

Slope

int64

Ca

float64

Thal

object

AHD

object

dtype: object

[10] (df==0). [PERSON_10])

[10] Unnamed: 0

:

Age

0

Sex

97

Chest Pain

RestBP

0

Chol

Fbs

258

RestECG

■151

```
MaxHR
0
21
ExAng
204
Oldpeak
99
Slope
0
Ca
176
Thal
0
AHD
dtype: int64
"
[11] print("Total Zero's Are: (df==0).sum (axis=0).sum())
:
Total Zero's Are: 985
[12] mean_age=df['Age'].mean()
[13]: print("Mean of patients age is: ",mean_age)
Mean of patients age is:
54.[US_BANK_NUMBER_52]
:
■[14] [URL_39]lumns
```

```
[14]: Index(['Unnamed: 0', 'Age', 'Sex', 'ChestPain', 'RestBP', 'Chol', 'Fbs',
'RestECG', 'MaxHR', 'ExAng', 'Oldpeak', 'Slope', 'Ca', 'Thal', 'AHD'],
dtype='object')

[15]: df1=[NRP_4]', 'Sex', 'Chest Pain', 'RestBP', 'Chol'])

[16] [URL_39]scribe()

Sex

Chol

RestBP
303.[US_DRIVER_LICENSE_181] 303.[US_DRIVER_LICENSE_181] 303.[US_DRIVER_LICENSE_181]
0.[US_DRIVER_LICENSE_181] 131.[US_DRIVER_LICENSE_181] 246.[US_DRIVER_LICENSE_181]
[PHONE_NUMBER_47] 17.[US_DRIVER_LICENSE_181] 51.[US_DRIVER_LICENSE_181]
0.[US_DRIVER_LICENSE_181] 94.[US_DRIVER_LICENSE_181] 126.[US_DRIVER_LICENSE_181]
0.[US_DRIVER_LICENSE_181] 120.[US_DRIVER_LICENSE_181] 211.[US_DRIVER_LICENSE_181]
1.[US_DRIVER_LICENSE_181] 130.[US_DRIVER_LICENSE_181] 241.[US_DRIVER_LICENSE_181]
1.[US_DRIVER_LICENSE_181] 140.[US_DRIVER_LICENSE_181] 275.[US_DRIVER_LICENSE_181]
1.[US_DRIVER_LICENSE_181] 200.[US_DRIVER_LICENSE_181] 564.[US_DRIVER_LICENSE_181]

[16]:

Age

count 303.[US_DRIVER_LICENSE_181]

mean

54.[US_DRIVER_LICENSE_181]

std

9.[US_DRIVER_LICENSE_181]

min

25%

29.[US_DRIVER_LICENSE_181]
48.[US_DRIVER_LICENSE_181]

50%

56.[US_DRIVER_LICENSE_181]

75%

■61.[US_DRIVER_LICENSE_181]
```

```
max

77.[US_DRIVER_LICENSE_181]

[17]:

df1

[17]:

Age Sex

Chest Pain RestBP

Chol

0

63

1

typical

145 233

1

67

1 asymptomatic

160 286

2

67

1 asymptomatic

120 229

3

37

1

4

■41
```

nonanginal
0 nontypical

130 250

130

204

:

3

298

45

1

typical

110 264

299

68

1 asymptomatic

144 193

300

57

1

asymptomatic

130

131

301 57

0 nontypical

■130 236

302 38

```
1

nonanginal

138 175

[303 rows x 5 columns]

[18] mean=df['Ca'] .mean ()

:

[19] df ['Ca'].fillna (value=mean, inplace=True)

[23] mode=df['Thal'] .mode() .iloc[0]

:

[24] df ['Thal'].fillna(value=mode, inplace=True)

[25] [URL_39]null().sum()

:

[25] Unnamed: 0

0

Age

0

Sex

0

Chest Pain

0

RestBP

0

Chol

Fbs

■RestECG
```

```
MaxHR

ExAng

Oldpeak

Slope

0

Ca

Thal

0

AHD

dtype: int64
```

```
[28] import [URL_39]plot as plt

!pip install seaborn

import seaborn as sns
```

```
Defaulting to user installation because normal site-packages is not writeable
Collecting seaborn

Downloading seaborn-0.13.2-py3-none-any.whl (294 kB)

294.9/294.9 KB 2.2 MB/s eta 0:00:00 [31m2.9 MB/s

eta 0:00:01

Requirement already satisfied: numpy!=1.24.0,>=1.20 in

/home/administrator/.local/lib/python3.10/site-packages (from [PERSON_10]) (1.25.1)

:

Requirement already satisfied: matplotlib!=3.6.1,>=3.4 in

/home/administrator/.local/lib/python3.10/site-packages (from [PERSON_10]) (3.7.2)

Requirement already satisfied: pandas>=1.2 in

■/home/administrator/.local/lib/python3.10/site-packages (from [PERSON_10]) (2.0.3)

Requirement already satisfied: cycler>=0.10 in
/home/administrator/.local/lib/python3.10/site-packages (from
matplotlib!=3.6.1,>=3.4->seaborn) (0.11.0)

Requirement already satisfied: kiwisolver>=1.0.1 in
/home/administrator/.local/lib/python3.10/site-packages (from

matplotlib!=3.6.1,>=3.4->seaborn) (1.4.4)

Requirement already satisfied: packaging>=20.0 in
/home/administrator/.local/lib/python3.10/site-packages (from

matplotlib!=3.6.1,>=3.4->seaborn) (23.0)

Requirement already satisfied: pyparsing<3.1,>=2.3.1 in /usr/lib/python3/dist-
packages (from matplotlib!=3.6.1,>=3.4->seaborn) (2.4.7)
Requirement already satisfied: python-dateutil>=2.7 in
/home/administrator/.local/lib/python3.10/site-packages (from

matplotlib!=3.6.1,>=3.4->seaborn) (2.8.2)

Requirement already satisfied: contourpy>=1.0.1 in
/home/administrator/.local/lib/python3.10/site-packages (from

matplotlib!=3.6.1,>=3.4->seaborn) (1.1.0)

Requirement already satisfied: fonttools>=4.22.0 in
/home/administrator/.local/lib/python3.10/site-packages (from

matplotlib!=3.6.1,>=3.4->seaborn) (4.41.0)

Requirement already satisfied: pillow>=6.2.0 in /usr/lib/python3/dist-packages

(from matplotlib!=3.6.1,>=3.4->seaborn) (9.0.1)

Requirement already satisfied: tzdata>=2022.1 in

/home/administrator/.local/lib/python3.10/site-packages (from

pandas>=1.2->seaborn) (2023.3)

Requirement already satisfied: pytz>=2020.1 in /usr/lib/python3/dist-packages
(from pandas>=1.2->seaborn) (2022.1)

Requirement already satisfied: six>=1.5 in /usr/lib/python3/dist-packages (from
python-dateutil>=2.7->matplotlib!=3.6.1,>=3.4->seaborn) (1.16.0)

Installing collected packages: [PERSON_10]Successfully installed seaborn-0.13.2
```

```
[33] [URL_39]t_style("whitegrid")

[URL_39]untplot(x='AHD',data=df, hue="AHD",palette='pastel')
```

```
plt.xlabel("Acute Heart Disease")

plt.ylabel("Total count")

plt.title("Frequency of heart disease")

plt.legend (['No', 'Yes'],loc='upper right')
```

```
[33]: <matplotlib.legend.Legend at 0x734bb30a52a0>
```

5

nn

160

140

120

Total count

100

80

60

60

40

40

20

20

Frequency of heart disease

0

No

■Yes

Acute Heart Disease

No

Yes

```
[32] fig, ax
:

=

[URL_39]bplots()

[URL_39]t_size_inches (10, 6)

sns.histplot(x="Age", data=df, hue="AHD", palette="bright")
[URL_39]t_style("whitegrid")

plt.title("Effect of age on frequency of heart disease")
plt.xlabel("Age")

plt.ylabel("Frequency")

plt. legend (["Heart Disease", "No Heart Disease"])
[URL_39]ow()
```

6

30

25

Frequency

20

15

10

10

5

Effect of age on frequency of heart disease

0

■30

40

50

60

Age

Heart Disease

No Heart Disease

70

10

```
[34] df ['Chest Pain'] .unique()
```

:

```
[34] array(['typical', 'asymptomatic', 'nonanginal', 'nontypical'],
dtype=object)
```

```
[35] fig, ax = [URL_39]bplots()

[URL_39]t_size_inches (10, 6)

sns. .countplot(x="AHD", hue="ChestPain", data=df, palette="dark")
plt.title("Type of Chest Pain")

plt.xlabel("Heart Disease")

plt.ylabel("Frequency")

plt.legend (["Typical Angina", "Asymptomatic", "Non-Anginal pain",
"Atypical Angina"])

[URL_39]ow()
```

7

100

Frequency

80

■00

40

40

20

Type of Chest Pain

0

No

Yes

Heart Disease

Typical Angina
Asymptomatic
Non-Anginal pain
Atypical Angina

```
[44]: X

=

df [['Age', 'Sex', 'ChestPain', 'RestBP', 'Chol', 'RestECG', 'MaxHR']]
Y= df ['AHD']

from [URL_39]del_selection import train_test_split

7_train, 7_test, [PERSON_10], Y_test = train_test_split(X, Y, test_size=0.25)

[45] X_[URL_39]()
```

```
<class '[URL_39]ame.DataFrame'>

Index: 227 entries, 101 to 35

Data columns (total 7 columns):

#

Column

Non-Null Count Dtype

0

Age

■227 non-null

int64

1

Sex
```

```
227 non-null
int64
2
Chest Pain
227 non-null
object
3
RestBP
227 non-null
int64
4
Chol
227 non-null
int64
5
RestECG
227 non-null
int64
6 MaxHR
227 non-null
int64
■dtypes: int64(6), object (1)
memory usage: 14.2+ KB
[46] Y_[URL_39]()
<class '[URL_39]ries'>
Index: 227 entries, 101 to 35
```

```
Series name: AHD

Non-Null Count

227 non-null

Dtype

object

dtypes: object(1)

memory usage: 3.5+ KB

[47] X_[URL_39]()

<class '[URL_39]ame.DataFrame'>
Index: 76 entries, 56 to 225

Data columns (total 7 columns):

#

Column

Non-Null Count Dtype

0

Age

76 non-null

int64

1

Sex

76 non-null

■int64

2

Chest Pain

76 non-null

object
```

```
3

RestBP

76 non-null

int64

4

Chol

76 non-null

int64

5

RestECG

76 non-null

int64

6 MaxHR

76 non-null

int64

dtypes: int64(6), object (1)

memory usage: 4.8+ KB

[48] Y_[URL_39]()

:

<class '[URL_39]ries. Series'>

■Index: 76 entries, 56 to 225

Series name: AHD

Non-Null Count

76 non-null

Dtype
```

```
object

dtypes: object (1)

memory usage: 1.2+ KB

[49] from [URL_39]trics import confusion_matrix, accuracy_score, precision_score,
recall_score, f1_score

=

y_true
[1] *45+ [1] *5 +
y_pred [1] *45+ [0] *5 +

cm =

=

[0] *55 + [0] *395

[1] *55 + [0] *395

confusion matrix(y_true, y_pred)
print("Confusion Matrix:\n", cm)

acc = accuracy_score (y_true, y_pred)
print("Accuracy:", acc)

[ ]:

=

prec precision_score (y_true, y_pred)
print("Precision:", prec)

rec =

recall_score (y_true, y_pred)

■print("Recall:", rec)

[US_DRIVER_LICENSE_181] = f1_score (y_true, y_pred)
print("[US_DRIVER_LICENSE_181] Score:", [PERSON_10])

Confusion Matrix:

[[395 55]

[ 545]]
Accuracy: 0.88
```

Precision: 0.45

Recall: 0.9

[US_DRIVER_LICENSE_181] Score: 0.6

10

10

■