Data Wrangling Class2024 : Analysis of Heart Diseases Data and Summary of Discovery From Olakunle Makanjuola

In [1]:
```
#  Library Import
import pandas as pd
import os
import matplotlib.pyplot as plt
import numpy as np
```

In [ ]:
```
#DATA SUMMARY
# The data provided is a randomly collected data with 918 random samples and 12 featur
# Data have no null or missing value
# Data is a mix of numerical and categorical features
# Data contain 5 categorical features and 7 numerical features
```

In [6]:
```
os.listdir('data')
```

Out[6]:
```
['heart.csv', 'kidney_disease_d.csv', 'loans.csv']
```

In [9]:
```
# load the Heart Diseses data
fPath = 'data'
fileName = 'heart.csv'


dk = pd.read_csv(fPath + '/' + fileName)
```

In [11]:
```
fPath + '/' + fileName
```

Out[11]:
```
'data/heart.csv'
```

In [12]:
```
 #Data Loading
#Loading of the dataset using Pandas Library
dk=pd.read_csv('data/heart.csv')
```

In [13]:
```
#Previewing Dataset:This is our dataset after importing using pandas library

dk.head()
```

Out[13]:

	Age	Sex	ChestPainType	RestingBP	Cholesterol	FastingBS	RestingECG	MaxHR	ExerciseAngina
0	40	M	ATA	140	289	0	Normal	172	N
1	49	F	NAP	160	180	0	Normal	156	N
2	37	M	ATA	130	283	0	ST	98	N
3	48	F	ASY	138	214	0	Normal	108	Y
4	54	M	NAP	150	195	0	Normal	122	N

In [14]:
```
dk.tail()
```

Out[14]:

	Age	Sex	ChestPainType	RestingBP	Cholesterol	FastingBS	RestingECG	MaxHR	ExerciseAngin
913	45	M	TA	110	264	0	Normal	132	I
914	68	M	ASY	144	193	1	Normal	141	I
915	57	M	ASY	130	131	0	Normal	115	
916	57	F	ATA	130	236	0	LVH	174	I
917	38	M	NAP	138	175	0	Normal	173	I

In [15]:
```
# Peak at the top 10 samples or bottom 10 samples
dk.head(10)
```

Out[15]:

	Age	Sex	ChestPainType	RestingBP	Cholesterol	FastingBS	RestingECG	MaxHR	ExerciseAngina
0	40	M	ATA	140	289	0	Normal	172	N
1	49	F	NAP	160	180	0	Normal	156	N
2	37	M	ATA	130	283	0	ST	98	N
3	48	F	ASY	138	214	0	Normal	108	Y
4	54	M	NAP	150	195	0	Normal	122	N
5	39	M	NAP	120	339	0	Normal	170	N
6	45	F	ATA	130	237	0	Normal	170	N
7	54	M	ATA	110	208	0	Normal	142	N
8	37	M	ASY	140	207	0	Normal	130	Y
9	48	F	ATA	120	284	0	Normal	120	N

In [16]:
```
# Bottom 10 samples
dk.tail(10)
```

Out[16]:

	Age	Sex	ChestPainType	RestingBP	Cholesterol	FastingBS	RestingECG	MaxHR	ExerciseAngin
908	63	M	ASY	140	187	0	LVH	144	
909	63	F	ASY	124	197	0	Normal	136	
910	41	M	ATA	120	157	0	Normal	182	I
911	59	M	ASY	164	176	1	LVH	90	I
912	57	F	ASY	140	241	0	Normal	123	
913	45	M	TA	110	264	0	Normal	132	I
914	68	M	ASY	144	193	1	Normal	141	I
915	57	M	ASY	130	131	0	Normal	115	
916	57	F	ATA	130	236	0	LVH	174	I
917	38	M	NAP	138	175	0	Normal	173	I

In [17]:
```
# Random peeking at our data ----> Means randomly grab samples from my data and show w
dk.sample(10)
```

Out[17]:

	Age	Sex	ChestPainType	RestingBP	Cholesterol	FastingBS	RestingECG	MaxHR	ExerciseAngin
311	60	M	ASY	125	0	1	Normal	110	I
33	41	M	ASY	130	172	0	ST	130	I
231	40	M	NAP	130	281	0	Normal	167	I
32	54	M	ASY	125	224	0	Normal	122	I
247	48	M	ASY	122	275	1	ST	150	
605	51	F	ASY	114	258	1	LVH	96	I
496	58	M	ASY	132	458	1	Normal	69	I
626	53	M	ASY	142	226	0	LVH	111	
529	72	M	ASY	143	211	0	Normal	109	
221	51	F	ASY	160	303	0	Normal	150	

In [18]:
```
#To check the data size (Give us total number of sample,and features in a data)
dk.shape
```

Out[18]:
```
(918, 12)
```

In [19]:
```
#This shows the info about the diff columns,the nullcount and the  data types we have
dk.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 918 entries, 0 to 917
Data columns (total 12 columns):
 #   Column          Non-Null Count  Dtype  
---  ------          --------------  -----  
 0   Age             918 non-null    int64  
 1   Sex             918 non-null    object 
 2   ChestPainType   918 non-null    object 
 3   RestingBP       918 non-null    int64  
 4   Cholesterol     918 non-null    int64  
 5   FastingBS       918 non-null    int64  
 6   RestingECG      918 non-null    object 
 7   MaxHR           918 non-null    int64  
 8   ExerciseAngina  918 non-null    object 
 9   Oldpeak         918 non-null    float64
 10  ST_Slope        918 non-null    object 
 11  HeartDisease    918 non-null    int64  
dtypes: float64(1), int64(6), object(5)
memory usage: 86.2+ KB
```

In [20]:
```
# To determine the number of Not Null in a data
dk.isnull().sum()
```

Out[20]:
```
Age               0
Sex               0
ChestPainType     0
RestingBP         0
Cholesterol       0
FastingBS         0
RestingECG        0
MaxHR             0
ExerciseAngina    0
Oldpeak           0
ST_Slope          0
HeartDisease      0
dtype: int64
```

In [21]:
```
# To determine how many numericals and norminal features are in the data
# To get column data types
dk.dtypes
```

Out[21]:
```
Age                int64
Sex               object
ChestPainType     object
RestingBP          int64
Cholesterol        int64
FastingBS          int64
RestingECG        object
MaxHR              int64
ExerciseAngina    object
Oldpeak          float64
ST_Slope          object
HeartDisease       int64
dtype: object
```

In [22]:
```
#To get the names of columns in a data
dk.columns
```

Out[22]:
```
Index(['Age', 'Sex', 'ChestPainType', 'RestingBP', 'Cholesterol', 'FastingBS',
       'RestingECG', 'MaxHR', 'ExerciseAngina', 'Oldpeak', 'ST_Slope',
       'HeartDisease'],
      dtype='object')
```

In [24]:
```
list(dk.columns)
```

Out[24]:
```
['Age',
 'Sex',
 'ChestPainType',
 'RestingBP',
 'Cholesterol',
 'FastingBS',
 'RestingECG',
 'MaxHR',
 'ExerciseAngina',
 'Oldpeak',
 'ST_Slope',
 'HeartDisease']
```

In [25]:
```
#To get the data types of one feature
dk['Age'].dtype
```

Out[25]:
```
dtype('int64')
```

In [29]:
```
#To get the data types of one feature
dk['Oldpeak'].dtype
```

Out[29]:
```
dtype('float64')
```

In [32]:
```
#Return unique values of Series object.
dk['ST_Slope'].unique()
```

Out[32]:
```
array(['Up', 'Flat', 'Down'], dtype=object)
```

In [33]:
```
#Return unique values of Series object.
dk['Age'].unique()
```

Out[33]:
```
array([40, 49, 37, 48, 54, 39, 45, 58, 42, 38, 43, 60, 36, 44, 53, 52, 51,
       56, 41, 32, 65, 35, 59, 50, 47, 31, 46, 57, 55, 63, 66, 34, 33, 61,
       29, 62, 28, 30, 74, 68, 72, 64, 69, 67, 73, 70, 77, 75, 76, 71],
      dtype=int64)
```

In [35]:
```
#fetch specific feature or column from your data
dk['Age']
```

Out[35]:
```
0      40
1      49
2      37
3      48
4      54
       ..
913    45
914    68
915    57
916    57
917    38
Name: Age, Length: 918, dtype: int64
```

In [36]:
```
# To able to determine features that is categoricals or numericals
cols = list(dk.columns)  #We assign it to a variable so that we can look through it

numericalCols =[]
categoricalCols = []

for col in cols:
    dataType = str(dk[col].dtype)   #Must be pass as a string else you won't able to u
    if 'object' in dataType:
        categoricalCols.append(col)
    else:
        numericalCols.append(col)
```

In [37]:
```
numericalCols
```

Out[37]:
```
['Age',
 'RestingBP',
 'Cholesterol',
 'FastingBS',
 'MaxHR',
 'Oldpeak',
 'HeartDisease']
```

In [38]:
```
categoricalCols
```

Out[38]:
```
['Sex', 'ChestPainType', 'RestingECG', 'ExerciseAngina', 'ST_Slope']
```

In [39]:
```
# To count the content of a list
len(categoricalCols)
```

Out[39]:
```
5
```

In [40]:
```
len(numericalCols)
```

Out[40]:
```
7
```

In [41]:
```
dk.head()
```

Out[41]:

	Age	Sex	ChestPainType	RestingBP	Cholesterol	FastingBS	RestingECG	MaxHR	ExerciseAngina
0	40	M	ATA	140	289	0	Normal	172	N
1	49	F	NAP	160	180	0	Normal	156	N
2	37	M	ATA	130	283	0	ST	98	N
3	48	F	ASY	138	214	0	Normal	108	Y
4	54	M	NAP	150	195	0	Normal	122	N

In [42]:
```
# To extract Age,RestingBP,Cholesterol,MaxHR into a new data
subData= dk[['Age', 'RestingBP', 'Cholesterol', 'MaxHR']]
```

In [43]:
```
# This is what is called subsetting or data slicing
subData
```

Out[43]:

	Age	RestingBP	Cholesterol	MaxHR
0	40	140	289	172
1	49	160	180	156
2	37	130	283	98
3	48	138	214	108
4	54	150	195	122
...	...	...	...	...
913	45	110	264	132
914	68	144	193	141
915	57	130	131	115
916	57	130	236	174
917	38	138	175	173

918 rows × 4 columns

In [44]:
```
# Extract numerical features into a different dataset and also do same for categorical
catDM = dk[categoricalCols]

catDM
```

Out[44]:

	Sex	ChestPainType	RestingECG	ExerciseAngina	ST_Slope
0	M	ATA	Normal	N	Up
1	F	NAP	Normal	N	Flat
2	M	ATA	ST	N	Up
3	F	ASY	Normal	Y	Flat
4	M	NAP	Normal	N	Up
...	...	...	...	...	...
913	M	TA	Normal	N	Flat
914	M	ASY	Normal	N	Flat
915	M	ASY	Normal	Y	Flat
916	F	ATA	LVH	N	Flat
917	M	NAP	Normal	N	Up

918 rows × 5 columns

In [51]:
```
catDM2 = dk[['Sex', 'ChestPainType', 'RestingECG', 'ExerciseAngina', 'ST_Slope']]

catDM2
```

Out[51]:

	Sex	ChestPainType	RestingECG	ExerciseAngina	ST_Slope
0	M	ATA	Normal	N	Up
1	F	NAP	Normal	N	Flat
2	M	ATA	ST	N	Up
3	F	ASY	Normal	Y	Flat
4	M	NAP	Normal	N	Up
...	...	...	...	...	...
913	M	TA	Normal	N	Flat
914	M	ASY	Normal	N	Flat
915	M	ASY	Normal	Y	Flat
916	F	ATA	LVH	N	Flat
917	M	NAP	Normal	N	Up

918 rows × 5 columns

In [52]:
```
numDM = dk[numericalCols]

numDM
```

Out[52]:

	Age	RestingBP	Cholesterol	FastingBS	MaxHR	Oldpeak	HeartDisease
0	40	140	289	0	172	0.0	0
1	49	160	180	0	156	1.0	1
2	37	130	283	0	98	0.0	0
3	48	138	214	0	108	1.5	1
4	54	150	195	0	122	0.0	0
...	...	...	...	...	...	...	...
913	45	110	264	0	132	1.2	1
914	68	144	193	1	141	3.4	1
915	57	130	131	0	115	1.2	1
916	57	130	236	0	174	0.0	1
917	38	138	175	0	173	0.0	0

918 rows × 7 columns

In [54]:
```
numDM2 = dk[['Age', 'RestingBP', 'Cholesterol', 'FastingBS', 'MaxHR', 'Oldpeak', 'Hear
numDM2
```

Out[54]:

	Age	RestingBP	Cholesterol	FastingBS	MaxHR	Oldpeak	HeartDisease
0	40	140	289	0	172	0.0	0
1	49	160	180	0	156	1.0	1
2	37	130	283	0	98	0.0	0
3	48	138	214	0	108	1.5	1
4	54	150	195	0	122	0.0	0
...	...	...	...	...	...	...	...
913	45	110	264	0	132	1.2	1
914	68	144	193	1	141	3.4	1
915	57	130	131	0	115	1.2	1
916	57	130	236	0	174	0.0	1
917	38	138	175	0	173	0.0	0

918 rows × 7 columns

In [ ]:
```
#DATA SUMMARY
# The data provided is a randomly collected data with 918 random samples and 12 featur
# Data have no null or missing value
# Data is a mix of numerical and categorical features
# Data contain 5 categorical features and 7 numerical features
```