Sprint1

October 31, 2022

1 TEAMID: PNT2022TMID44392

[1]:
```
#IMPORT REQUIRED LIBRARIES
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
```

[3]:
```
#import dataset and load in dataframe
df=pd.read_csv('chronickidneydisease.csv')
df.head()
```

[3]:

	id	age	bp	sg	al	su	rbc	pc	pcc	ba	\
0	0	48.0	80.0	1.020	1.0	0.0	NaN	normal	notpresent	notpresent	
1	1	7.0	50.0	1.020	4.0	0.0	NaN	normal	notpresent	notpresent	
2	2	62.0	80.0	1.010	2.0	3.0	normal	normal	notpresent	notpresent	
3	3	48.0	70.0	1.005	4.0	0.0	normal	abnormal	present	notpresent	
4	4	51.0	80.0	1.010	2.0	0.0	normal	normal	notpresent	notpresent	

	…	pcv	wc	rc	htn	dm	cad	appet	pe	ane	classification
0	…	44	7800	5.2	yes	yes	no	good	no	no	ckd
1	…	38	6000	NaN	no	no	no	good	no	no	ckd
2	…	31	7500	NaN	no	yes	no	poor	no	yes	ckd
3	…	32	6700	3.9	yes	no	no	poor	yes	yes	ckd
4	…	35	7300	4.6	no	no	no	good	no	no	ckd

[5 rows x 26 columns]

[4]:
```
#checking the description and gathering the information about the dataset
df.describe().T
```

[4]:

	count	mean	std	min	25%	50%	75%	max
id	400.0	199.500000	115.614301	0.000	99.75	199.50	299.25	399.000
age	391.0	51.483376	17.169714	2.000	42.00	55.00	64.50	90.000
bp	388.0	76.469072	13.683637	50.000	70.00	80.00	80.00	180.000
sg	353.0	1.017408	0.005717	1.005	1.01	1.02	1.02	1.025
al	354.0	1.016949	1.352679	0.000	0.00	0.00	2.00	5.000

```
su     351.0    0.450142    1.099191   0.000    0.00    0.00    0.00    5.000
bgr    356.0  148.036517   79.281714  22.000   99.00  121.00  163.00  490.000
bu     381.0   57.425722   50.503006   1.500   27.00   42.00   66.00  391.000
sc     383.0    3.072454    5.741126   0.400    0.90    1.30    2.80   76.000
sod    313.0  137.528754   10.408752   4.500  135.00  138.00  142.00  163.000
pot    312.0    4.627244    3.193904   2.500    3.80    4.40    4.90   47.000
hemo   348.0   12.526437    2.912587   3.100   10.30   12.65   15.00   17.800
```

[5]:
```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 400 entries, 0 to 399
Data columns (total 26 columns):
 #   Column          Non-Null Count  Dtype
---  ------          --------------  -----
 0   id              400 non-null    int64
 1   age             391 non-null    float64
 2   bp              388 non-null    float64
 3   sg              353 non-null    float64
 4   al              354 non-null    float64
 5   su              351 non-null    float64
 6   rbc             248 non-null    object
 7   pc              335 non-null    object
 8   pcc             396 non-null    object
 9   ba              396 non-null    object
 10  bgr             356 non-null    float64
 11  bu              381 non-null    float64
 12  sc              383 non-null    float64
 13  sod             313 non-null    float64
 14  pot             312 non-null    float64
 15  hemo            348 non-null    float64
 16  pcv             330 non-null    object
 17  wc              295 non-null    object
 18  rc              270 non-null    object
 19  htn             398 non-null    object
 20  dm              398 non-null    object
 21  cad             398 non-null    object
 22  appet           399 non-null    object
 23  pe              399 non-null    object
 24  ane             399 non-null    object
 25  classification  400 non-null    object
dtypes: float64(11), int64(1), object(14)
memory usage: 81.4+ KB
```

[6]:
```
#counting for the null values
df.isna().sum()
```

[6]:
```
id                0
age               9
bp               12
sg               47
al               46
su               49
rbc             152
pc               65
pcc               4
ba                4
bgr              44
bu               19
sc               17
sod              87
pot              88
hemo             52
pcv              70
wc              105
rc              130
htn               2
dm                2
cad               2
appet             1
pe                1
ane               1
classification    0
dtype: int64
```

[11]:
```
#replacing the null values with median and mode
oc=[]#object data type columns
ic=[]#int type columns

for i in df.columns:
    if(df[i].dtype=='object'):
        oc.append(i)
    else:
        ic.append(i)
print("ic\t",ic,"\noc\t",oc)
```

```
ic      ['id', 'age', 'bp', 'sg', 'al', 'su', 'bgr', 'bu', 'sc', 'sod', 'pot',
'hemo']
oc      ['rbc', 'pc', 'pcc', 'ba', 'pcv', 'wc', 'rc', 'htn', 'dm', 'cad',
'appet', 'pe', 'ane', 'classification']
```

[40]:
```
#replacing the null with median
for i in ic:
    if(df[i].isna().any()==True):
```

```
        df[i]=df[i].fillna(df[i].median())
    #checking
    print("Attribute "+i+"\t",df[i].isna().sum())
```

```
Attribute: id    0
Attribute: age   0
Attribute: bp    0
Attribute: sg    0
Attribute: al    0
Attribute: su    0
Attribute: bgr   0
Attribute: bu    0
Attribute: sc    0
Attribute: sod   0
Attribute: pot   0
Attribute: hemo  0
```

[46]:
```
#replacing the null with mode
for i in oc:
    if(df[i].isna().any()==True):
        df[i]=df[i].fillna(df[i].mode()[0])
    #checking
    print("Attribute: "+i+"\t\t\t",df[i].isna().sum())
```

```
Attribute: rbc                  0
Attribute: pc                   0
Attribute: pcc                  0
Attribute: ba                   0
Attribute: pcv                  0
Attribute: wc                   0
Attribute: rc                   0
Attribute: htn                  0
Attribute: dm                   0
Attribute: cad                  0
Attribute: appet                        0
Attribute: pe                   0
Attribute: ane                  0
Attribute: classification                       0
```

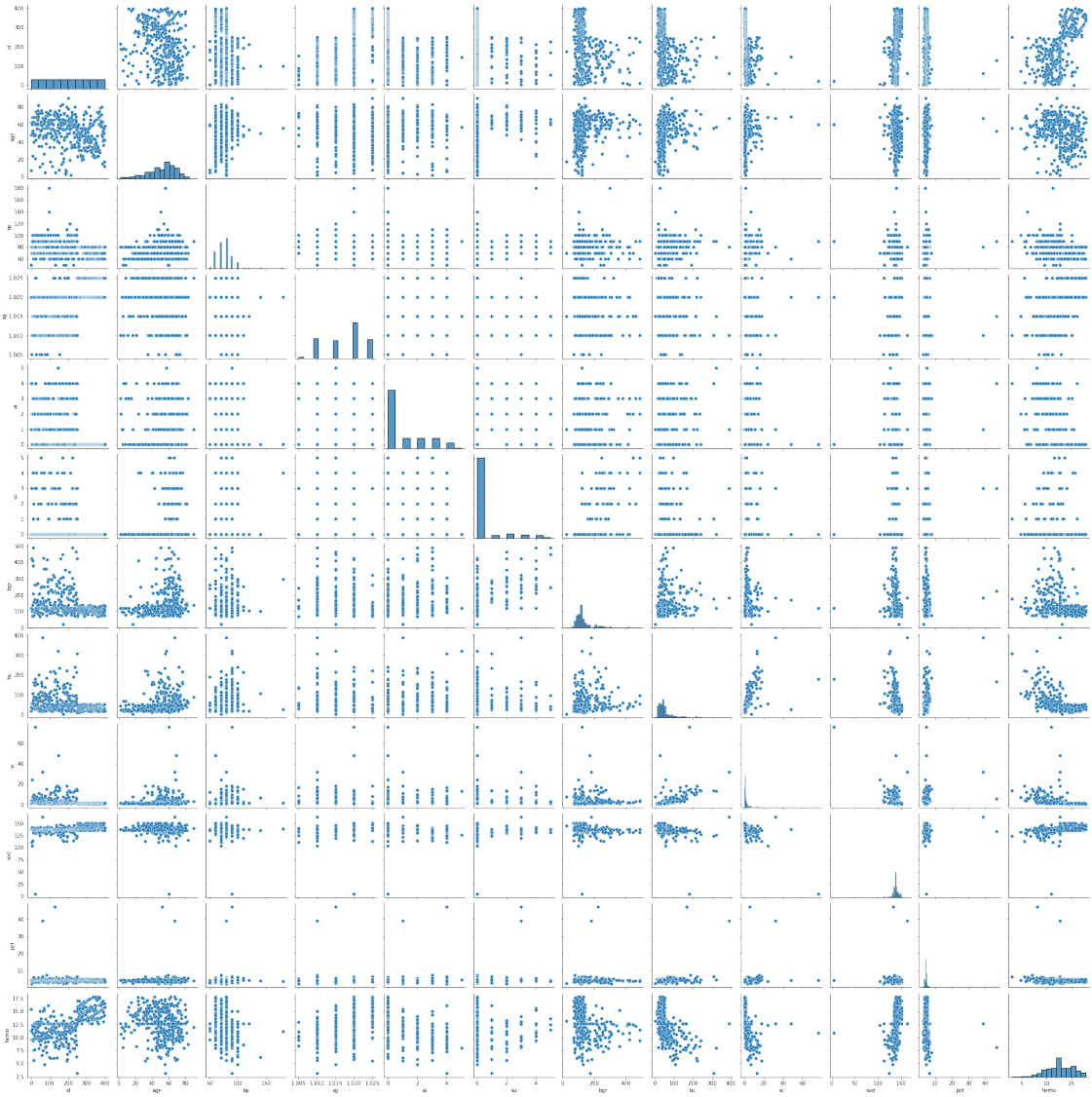
[47]:
```
df.isna().sum()
```

[47]:
```
id              0
age             0
bp              0
sg              0
al              0
su              0
```

```
rbc               0
pc                0
pcc               0
ba                0
bgr               0
bu                0
sc                0
sod               0
pot               0
hemo              0
pcv               0
wc                0
rc                0
htn               0
dm                0
cad               0
appet             0
pe                0
ane               0
classification    0
dtype: int64
```

[50]:
```
#visualizing the datasets
sns.pairplot(df)
```

[50]:
```
<seaborn.axisgrid.PairGrid at 0x7fbb94b144c0>
```


id
age
bp
sg
al
su
bgr
bu
sc
sod
pot
hemo