SPRINT 2

TEAMID:PNT2022TMID36376

[1]:
```
%matplotlibinline
```

[2]:
```
#IMPORTREQUIREDLIBRARIES
importpandasaspdimportn
umpyasnp
importmatplotlib.pyplotaspltimportse
abornassns
importwarnings

warnings.filterwarnings('ignore')
```

[3]:
```
 #importdatasetandloadindataframedf=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	rchtn	dm	cad	appet	pe	ane	classification
0	…	44	7800	5.2yes	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	3.9yes	no	no	poor	yes	yes	ckd
4	…	35	7300	4.6	no	no	no	good	no	no	ckd

[5rowsx26columns]

[4]:
```
#datasetadjustment
df['classification']=df['classification'].replace(['ckd\t'],['notckd'])
```

[5]:
```
df['classification'].value_counts()
```

[5]:
```
ckd        248
notckd     152
Name:classification,dtype:int64
```

[6]:
```
#checkingthedescriptionandgatheringtheinformationaboutthedataset
df.describe().T
```

[6]:

	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

[7]:
```
df.info()
```

```
<class
'pandas.core.frame.DataFrame'>RangeIn
dex: 400 entries, 0 to
399Datacolumns(total26columns):
 #   Column          Non-NullCountDtype
---  ------          ----------------  ---------
 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
10bgr                356 non-null      float64
11bu                 381 non-null      float64
12sc                 383 non-null      float64
13sod                313 non-null      float64
14pot                312 non-null      float64
15hemo               348 non-null      float64
16pcv                330 non-null      object
17wc                 295 non-null      object
18rc                 270 non-null      object
```

```
 19htn                398non-null      object
 20dm                 398non-null      object
 21cad                398non-null      object
 22appet              399non-null      object
 23pe                 399non-null      object
 24ane                399non-null      object
 25classification     400non-null      object
dtypes:float64(11),int64(1),object(14)mem
oryusage:81.4+KB
```

[8]:
```
#countingforthenullvalues
df.isna().sum()
```

[8]:
```
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
```

[9]:
```
#replacingthe null values with median and mode
oc=[]#objectdata type columns
ic=[]#inttype columns

foriin 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']
```

[10]:
```
#replacingthenullwithmedian
foriinic:
    if(df[i].isna().any()==True):
        df[i]=df[i].fillna(df[i].median())
    #checking
    print("Attribute"+i+"\t",df[i].isna().sum())
```

```
Attributeid     0
Attributeage    0
Attributebp     0
Attributesg     0
Attributeal     0
Attributesu     0
Attributebgr    0
Attributebu     0
Attributesc     0
Attributesod    0
Attributepot    0
Attributehemo   0
```

[11]:
```
#replacingthenullwithmode
foriinoc:
    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
```

[12]:
```
df.isna().sum().sum()
```

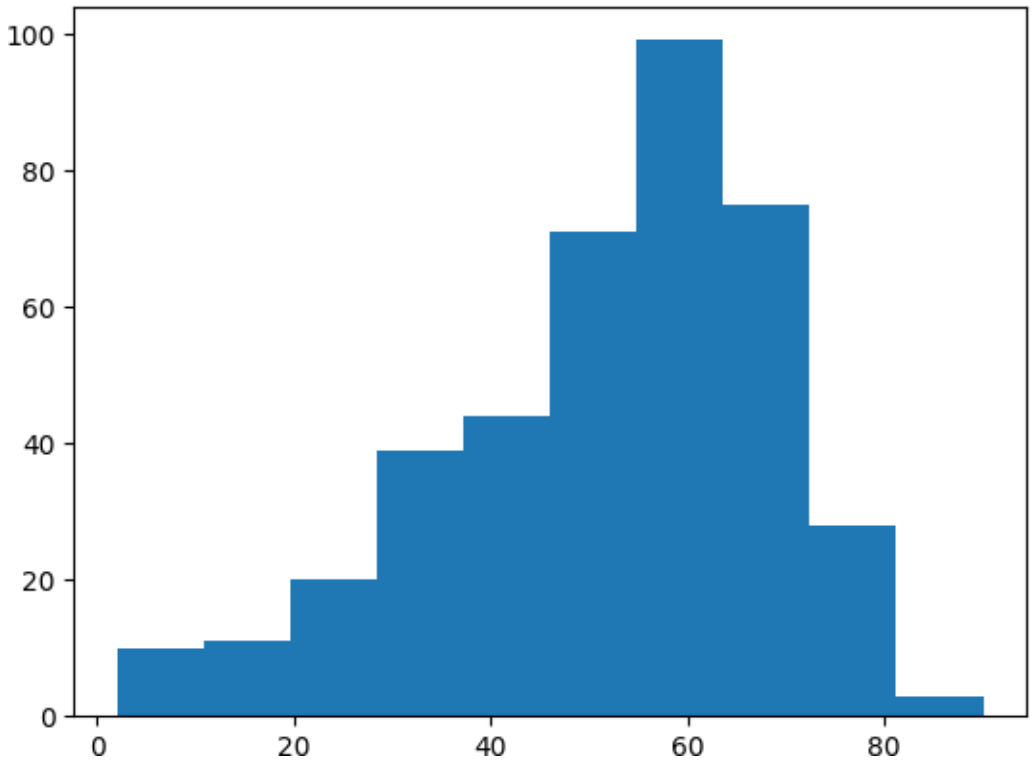
[12]:0

[13]:
```
#encodinglabels
fromsklearn.preprocessingimportLabelEncoderle
=LabelEncoder()#labelencoderobject
foriinoc:
    df[i]=le.fit_transform(df[i])#labelencodingalltheobjectdtypes

df.head(3)
```

[13]:
```
   id   age    bp     sg    al  surbc    pcpcc      ba …pcv    wc  rchtn\
0   0  48.0  80.0  1.02   1.0 0.0     1    1     0     0 …   32   72  34     1
1   1   7.0  50.0  1.02   4.0 0.0     1    1     0     0 …    26  56  34     0
2   2  62.0  80.0  1.01   2.0 3.0     1    1     0     0 …    19  70  34     0

   dmcadappetpeaneclassification0          4
    1     0        0  0     0
1   3     1        0  0     0                  0
2   4     1        1  0     1                  0

[3rowsx26columns]
```

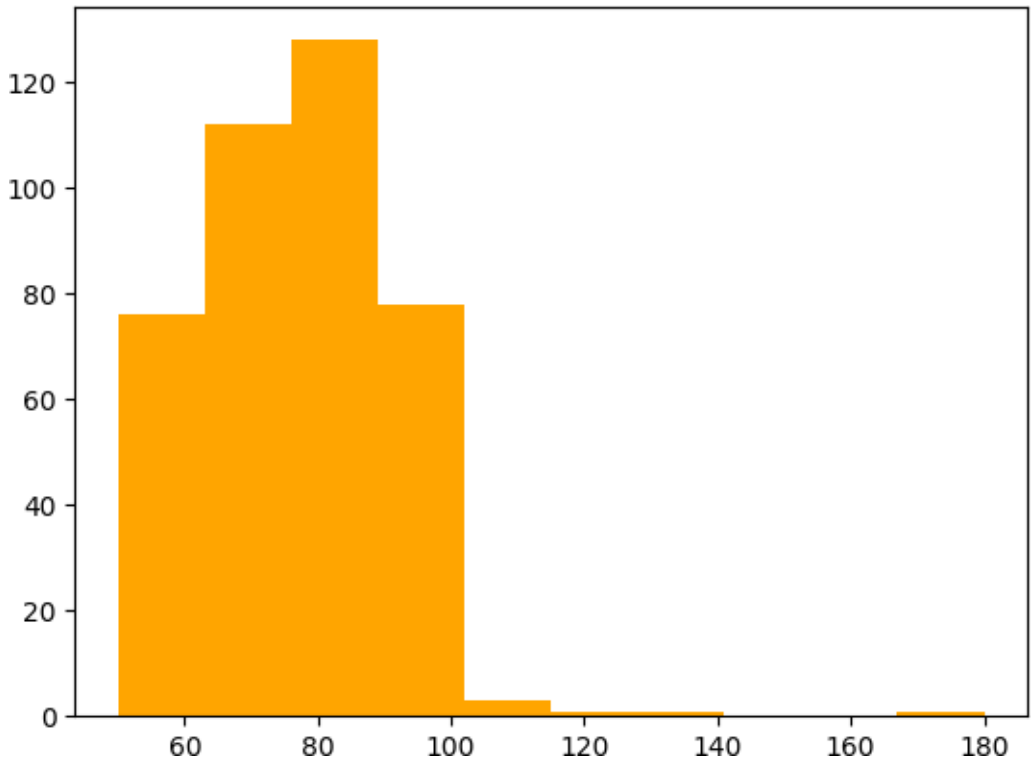
[14]:
```
plt.hist(df['age'])
```

[14]:
```
(array([10., 11., 20., 39., 44., 71., 99., 75., 28., 3.]),
 array([2., 10.8, 19.6, 28.4, 37.2, 46., 54.8, 63.6, 72.4, 81.2, 90.]),
 <BarContainerobjectof10artists>)
```



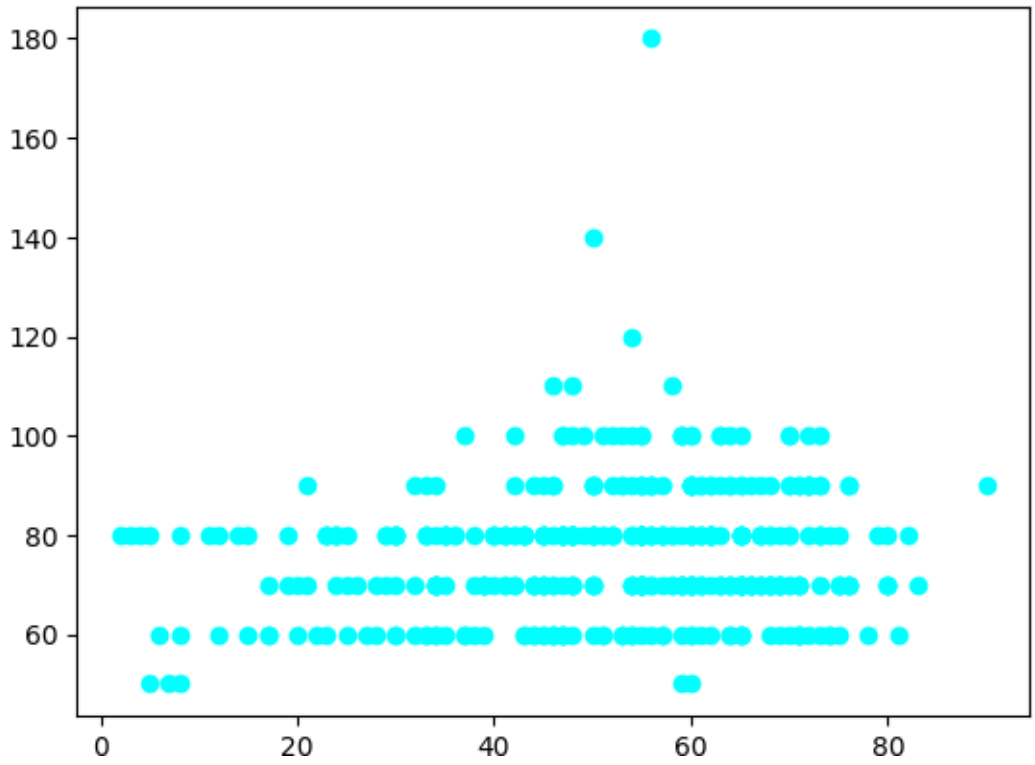
[15]:
```
plt.hist(df['bp'],color="orange")
```

[15]:(array([76.,112.,128.,78., 3., 1., 1., 0., 0., 1.]),
array([50.,63.,76.,89.,102.,115.,128.,141.,154.,167.,180.]),
<BarContainerobjectof10artists>)



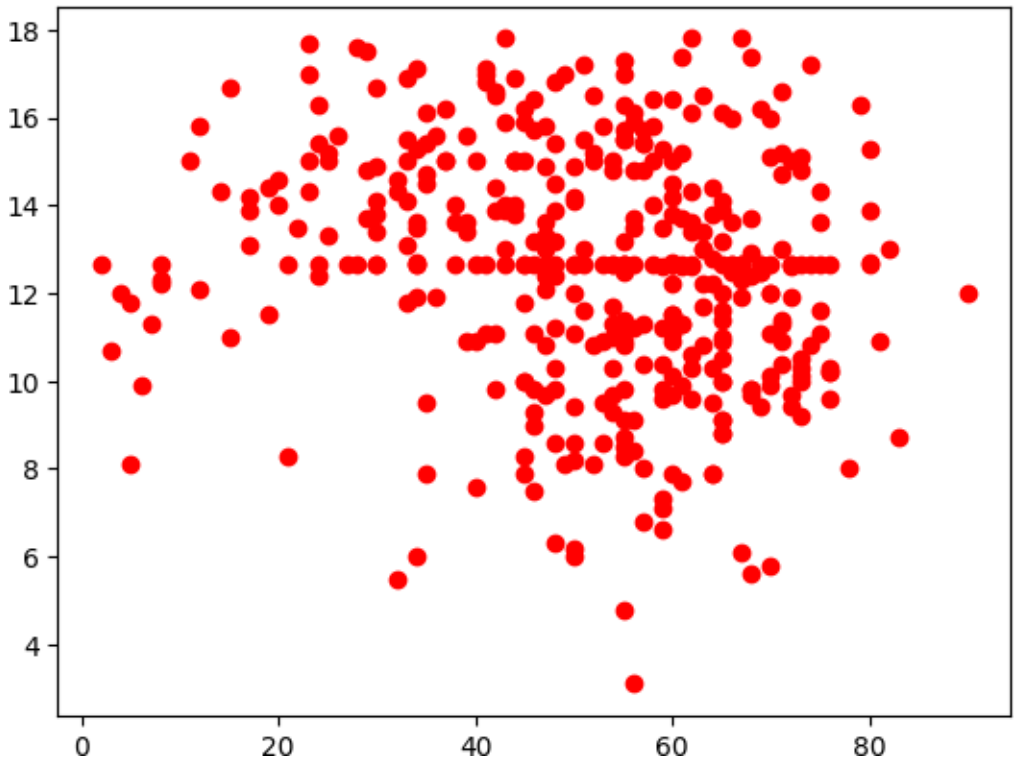
[16] :
```
plt.scatter(df['age'],df['bp'],color="cyan")
```

[16] :<matplotlib.collections.PathCollectionat0x7fbe95433a00>



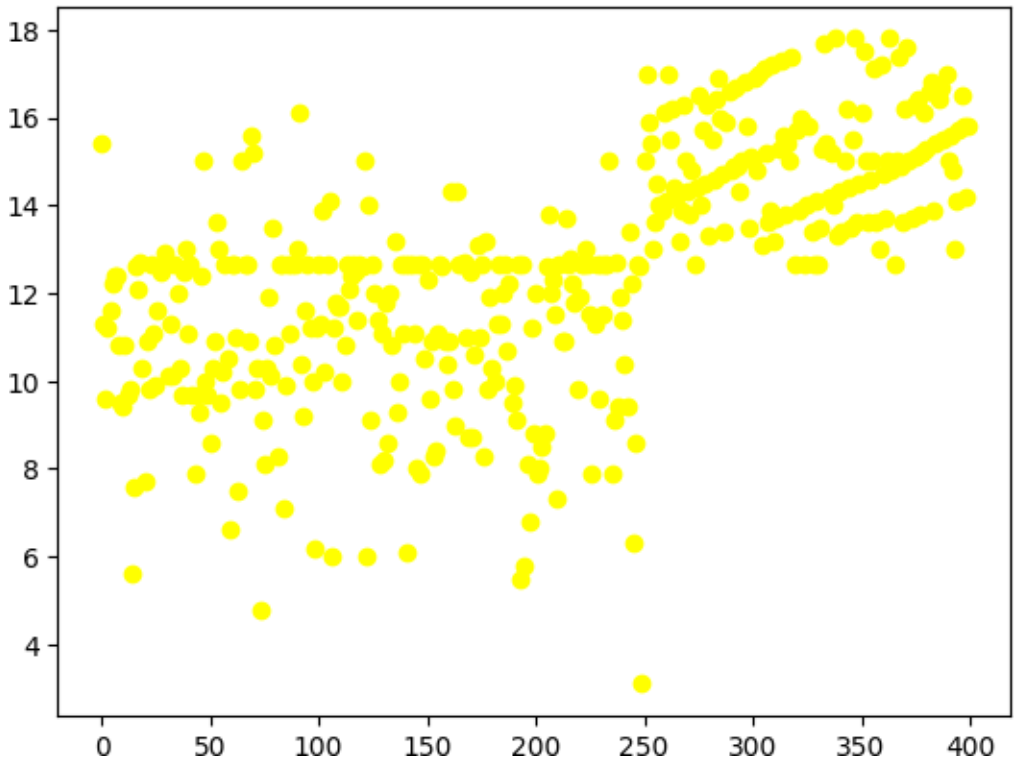
[17] :
```
plt.scatter(df['age'],df['hemo'],color='red')
```

[17] :<matplotlib.collections.PathCollectionat0x7fbe95269810>



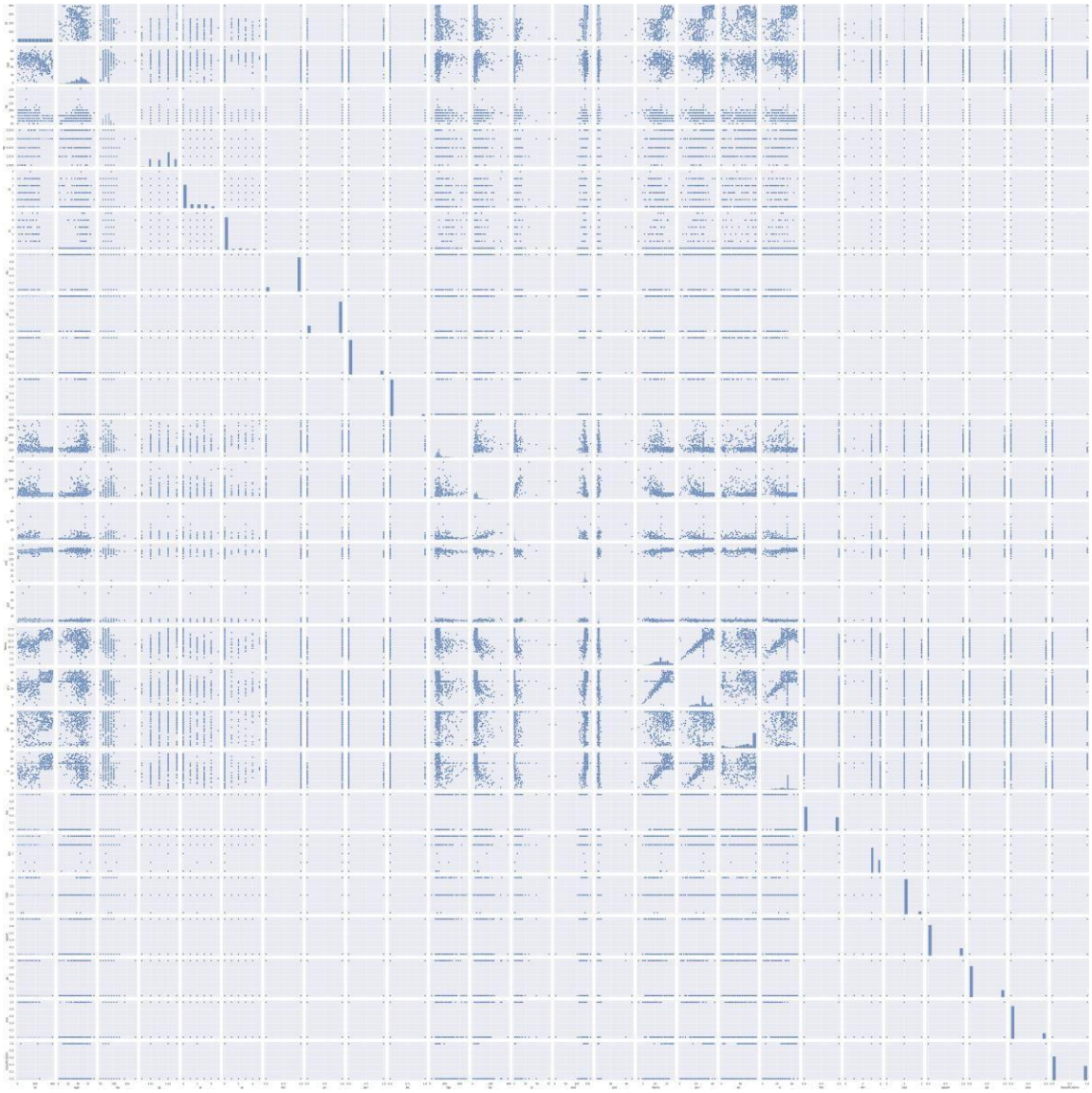
[18] : `plt.scatter(df['id'],df['hemo'],color="yellow")`

[18] :<matplotlib.collections.PathCollectionat0x7fbe9532a950>



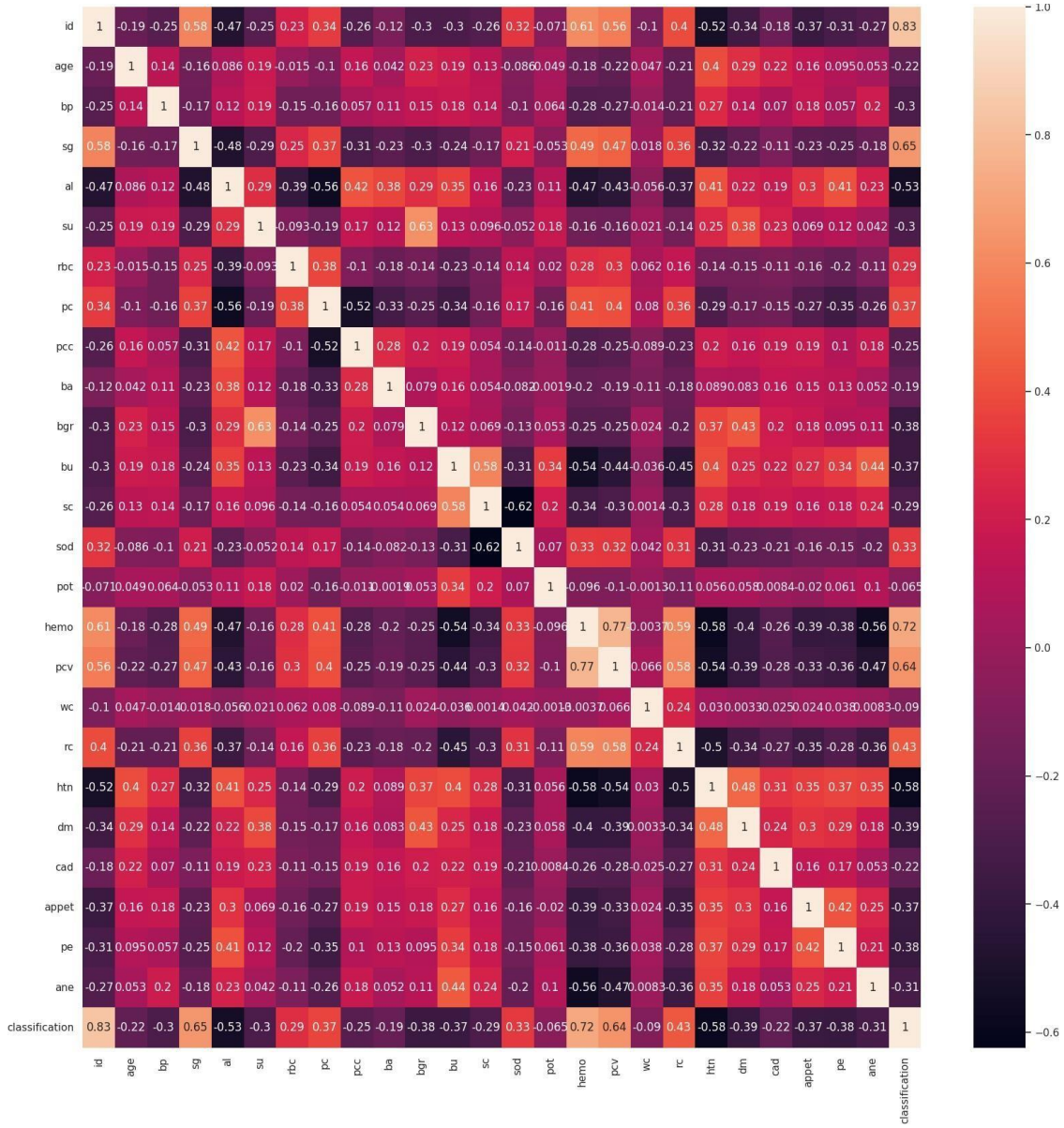
[19]:
```
sns.set(rc={'figure.figsize':(13,2)})
sns.pairplot(df)
```

[19]: <seaborn.axisgrid.PairGridat0x7fbe952ef2e0>



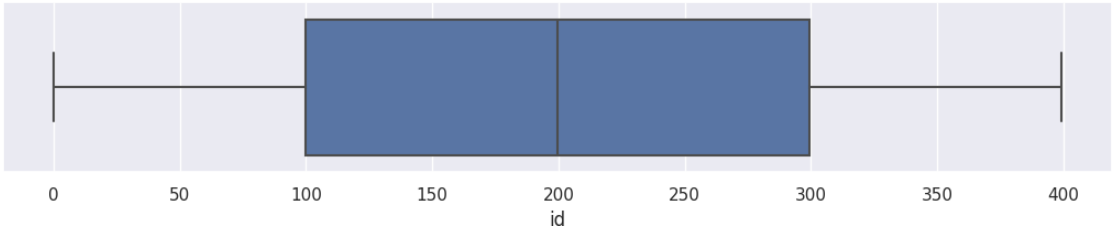
[20]:
```
df.corr()fig=plt.figure(figsize
=(20,20))

sns.heatmap(data=df.corr(),annot=True)
```

[20]:<AxesSubplot:>



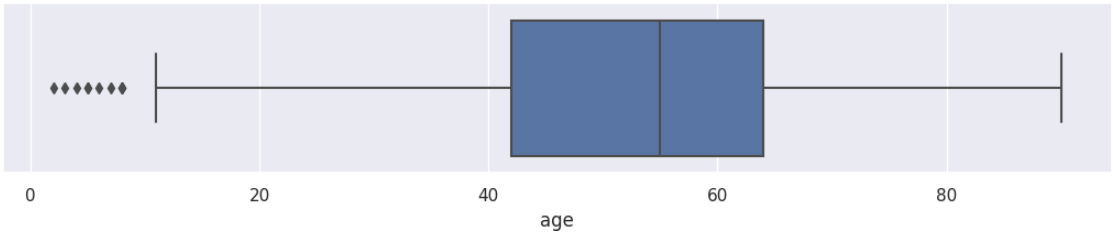
[21] :
```
#seeingoutliers
sns.boxplot(df['id'])
```

[21] :<AxesSubplot:xlabel='id'>



[22]:
```
sns.boxplot(df['age'])
```

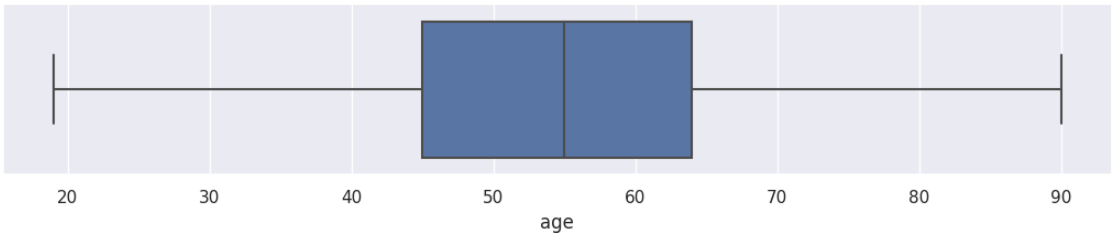
[22]:<AxesSubplot:xlabel='age'>



[23]:
```
#replacingtheoutliersmedian=df['age'].
median()print(median)
df['age']=df['age'].mask(df['age']<19,median)
sns.boxplot(df['age'])
```

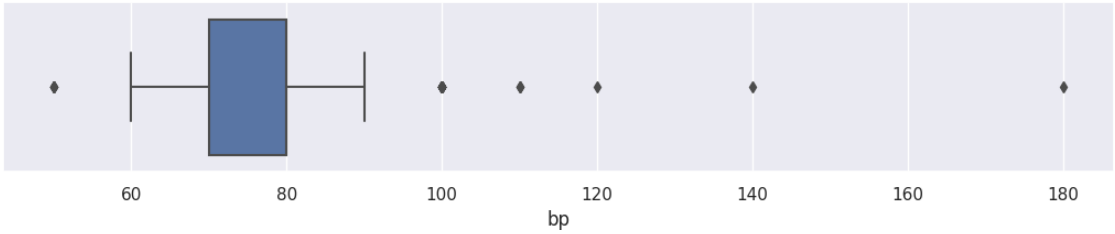
55.0

[23]:<AxesSubplot:xlabel='age'>



[24]:
```
sns.boxplot(df['bp'])
```

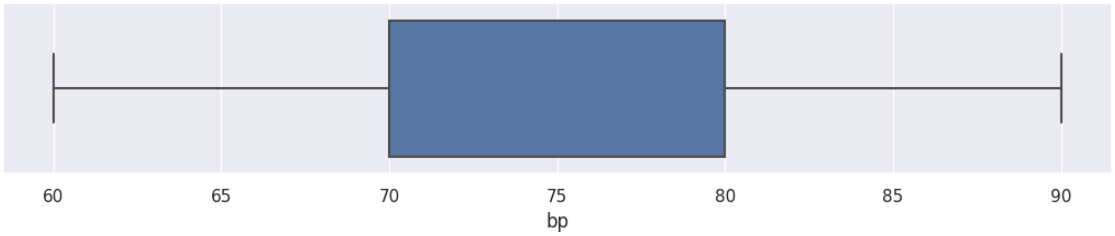
[24]:<AxesSubplot:xlabel='bp'>



[25]:
```
#replacingoutliersmedian=df['bp'].medi
an()print(median)
df['bp']=df['bp'].mask(df['bp']<60,median)
df['bp']=df['bp'].mask(df['bp']>90,median)
sns.boxplot(df['bp'])
```

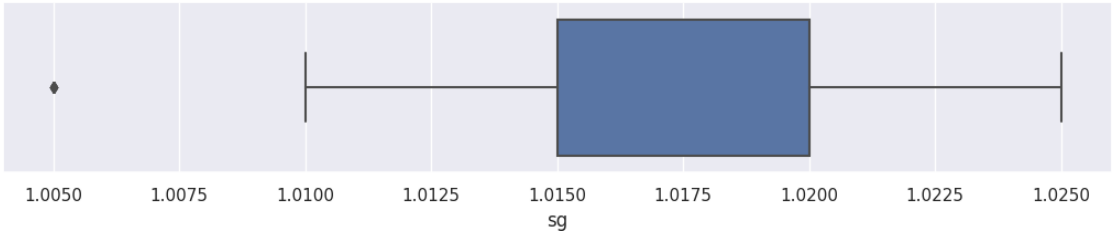
80.0

[25]:<AxesSubplot:xlabel='bp'>



[26]:
```
sns.boxplot(df['sg'])
```

[26]:<AxesSubplot:xlabel='sg'>

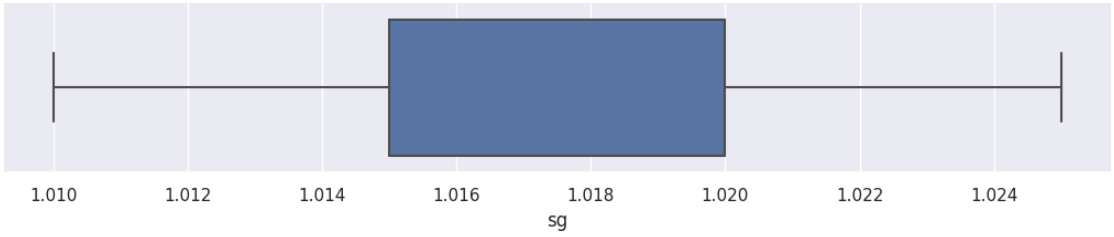


[27]:
```
#replacingoutliersmedian=df['sg'].medi
an()print(median)
```

```
df['sg']=df['sg'].mask(df['sg']<1.0100,median)
sns.boxplot(df['sg'])
```

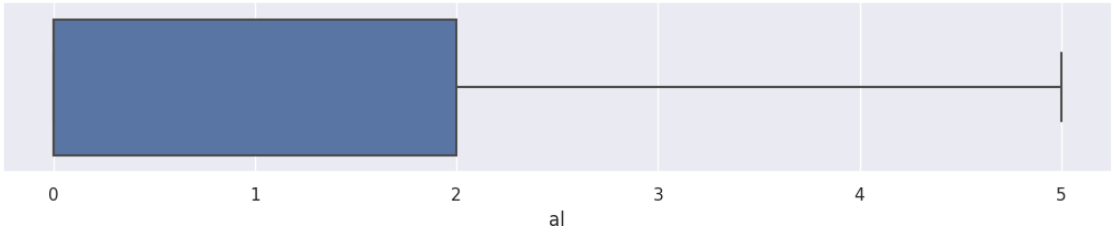
1.02

[27] :<AxesSubplot:xlabel='sg'>



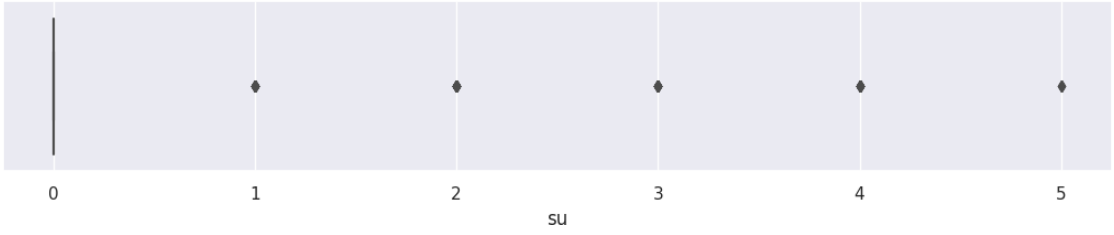
[28] :
```
sns.boxplot(df['al'])
```

[28] :<AxesSubplot:xlabel='al'>



[29] :
```
sns.boxplot(df['su'])
```

[29] :<AxesSubplot:xlabel='su'>



[30] :
```
#replacingoutliersmedian=df['su'].medi
an()print(median)
```

```
df['su']=df['su'].mask(df['su']>0,median)
sns.boxplot(df['su'])
```

0.0

[30] :<AxesSubplot:xlabel='su'>

[31] :
```
idv=df.iloc[:,:-1]#independentvariables
dv=df.iloc[:,-1]#dependentvariables
idv
```

[31]:

	id	age	bp	sg	al	su	rbc	pc	pcc	ba	…	hemo	pcv	wc	\
0	0	48.0	80.0	1.020	1.0	0.0	1	1	0	0	…	15.4	32	72	
1	1	55.0	80.0	1.020	4.0	0.0	1	1	0	0	…	11.3	26	56	
2	2	62.0	80.0	1.010	2.0	0.0	1	1	0	0	…	9.6	19	70	
3	3	48.0	70.0	1.020	4.0	0.0	1	0	1	0	…	11.2	20	62	
4	4	51.0	80.0	1.010	2.0	0.0	1	1	0	0	…	11.6	23	68	
..	…	…	…	…	.....	......	...	...	...	…	…	...			
395	395	55.0	80.0	1.020	0.0	0.0	1	1	0	0	…	15.7	35	62	
396	396	42.0	70.0	1.025	0.0	0.0	1	1	0	0	…	16.5	42	72	
397	397	55.0	80.0	1.020	0.0	0.0	1	1	0	0	…	15.8	37	61	
398	398	55.0	60.0	1.025	0.0	0.0	1	1	0	0	…	14.2	39	67	
399	399	58.0	80.0	1.025	0.0	0.0	1	1	0	0	…	15.8	41	63	

	rc	htndmcadappetpe				ane
0	34	1	4	1	0	0	0
1	34	0	3	1	0	0	0
2	34	0	4	1	1	0	1
3	19	1	3	1	1	1	1
4	27	0	3	1	0	0	0
..	..	...	…	...	…		
395	30	0	3	1	0	0	0
396	44	0	3	1	0	0	0
397	36	0	3	1	0	0	0
398	41	0	3	1	0	0	0
399	43	0	3	1	0	0	0

[400rowsx25columns]

[32]:
```
#splittingdatasets
fromsklearn.model_selectionimporttrain_test_splitx_train,x_test,y_train,y_test=train_
test_split(idv,dv,test_size=0.
↪2,shuffle=True)
```

[33]:
```
x_train.shape
```

[33]:(320,25)

[34]:
```
#creatingmodels
fromsklearn.linear_modelimportLogisticRegressionmodel=Logisti
cRegression()
```

[35]:
```
model.fit(x_train,y_train)
```

[35]:LogisticRegression()

[36]:
```
#acccuracypred=model.predict(x
_test)pred
```

[36]:array([0,0,0,0,0,1,1,1,0,0,0,1,0,0,1,0,0,1,0,1,1,1,
1,1,1,0,0,0,0,1,1,1,1,0,1,1,0,0,0,1,0,1,1,1,
0,1,0,0,0,0,1,0,1,0,0,1,0,1,0,1,1,0,0,0,0,1,
0,0,0,0,1,1,0,1,0,1,0,0,0,0])

[37]:
```
#for checking.....
fromsklearn.svmimportSVCsvmmodel=SVC
()
```

[38]:
```
svmmodel.fit(x_train,y_train)
```

[38]:SVC()

[39]:
```
#acccuracysvc_pred=model.predict
(x_test)svc_pred
```

[39]:array([0,0,0,0,0,1,1,1,0,0,0,1,0,0,1,0,0,1,0,1,1,1,
1,1,1,0,0,0,0,1,1,1,1,0,1,1,0,0,0,1,0,1,1,1,
0,1,0,0,0,0,1,0,1,0,0,1,0,1,0,1,1,0,0,0,0,1,
0,0,0,0,1,1,0,1,0,1,0,0,0,0])

[40]:
```
fromsklearn.metricsimportaccuracy_score,confusion_matrixac
curacy_score(y_test,pred)
```

[40]:0.9875

[41]:
```
confusion_matrix(y_test,pred)
```

[41]:array([[46,1],
[0,33]])

[42]:
```
y_train.value_counts()
```

[42]:0 201
1 119
Name:classification,dtype:int64

[43]:
```
#svmaccuracy & confusion matrix
accuracy_score(y_test,svc_pred)
```

[43]:0.9875

[44]:
```
confusion_matrix(y_test,svc_pred)
```

[44]:array([[46,1],
[0,33]])

[45]:
```
#creatingmodel
importpickle
```

[46]:
```
pickle.dump(model,open('ckdmodel.pkl','wb'))pr
int("modelsavedsuccessfully")
```

modelsavedsuccessfully

[]:

[]: