Assignment 1 and 2

January 29, 2025

[102]:
```
import pandas as pd

df=pd.read_csv("heart_disease.csv")
```

[103]:
```
df.head()
```

[103]:

	Age	Gender	Blood Pressure	Cholesterol Level	Exercise Habits	Smoking	\
0	56.0	Male	153.0	155.0	High	Yes	
1	69.0	Female	146.0	286.0	High	No	
2	46.0	Male	126.0	216.0	Low	No	
3	32.0	Female	122.0	293.0	High	Yes	
4	60.0	Male	166.0	242.0	Low	Yes	

	Family Heart Disease	Diabetes	BMI	High Blood Pressure	…	\
0	Yes	No	24.991591	Yes	…	
1	Yes	Yes	25.221799	No	…	
2	No	No	29.855447	No	…	
3	Yes	No	24.130477	Yes	…	
4	Yes	Yes	20.486289	Yes	…	

	High LDL Cholesterol	Alcohol Consumption	Stress Level	Sleep Hours	\
0	No	High	Medium	7.633228	
1	No	Medium	High	8.744034	
2	Yes	Low	Low	4.440440	
3	Yes	Low	High	5.249405	
4	No	Low	High	7.030971	

	Sugar Consumption	Triglyceride Level	Fasting Blood Sugar	CRP Level	\
0	Medium	342.0	NaN	12.969246	
1	Medium	133.0	157.0	9.355389	
2	Low	393.0	92.0	12.709873	
3	High	293.0	94.0	12.509046	
4	High	263.0	154.0	10.381259	

	Homocysteine Level	Heart Disease Status
0	12.387250	No
1	19.298875	No
2	11.230926	No

```
3            5.961958                    No
4            8.153887                    No

[5 rows x 21 columns]
```

[104]:
```
df.describe()
```

[104]:
```
                Age  Blood Pressure  Cholesterol Level          BMI  \
count  9971.000000     9981.000000        9970.000000  9978.000000
mean     49.296259      149.757740         225.425577    29.077269
std      18.193970       17.572969          43.575809     6.307098
min      18.000000      120.000000         150.000000    18.002837
25%      34.000000      134.000000         187.000000    23.658075
50%      49.000000      150.000000         226.000000    29.079492
75%      65.000000      165.000000         263.000000    34.520015
max      80.000000      180.000000         300.000000    39.996954

       Sleep Hours  Triglyceride Level  Fasting Blood Sugar    CRP Level  \
count  9975.000000         9974.000000          9978.000000  9974.000000
mean      6.991329          250.734409           120.142213     7.472201
std       1.753195           87.067226            23.584011     4.340248
min       4.000605          100.000000            80.000000     0.003647
25%       5.449866          176.000000            99.000000     3.674126
50%       7.003252          250.000000           120.000000     7.472164
75%       8.531577          326.000000           141.000000    11.255592
max       9.999952          400.000000           160.000000    14.997087

       Homocysteine Level
count         9980.000000
mean            12.456271
std              4.323426
min              5.000236
25%              8.723334
50%             12.409395
75%             16.140564
max             19.999037
```

[105]:
```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 10000 entries, 0 to 9999
Data columns (total 21 columns):
 #   Column                Non-Null Count  Dtype
---  ------                --------------  -----
 0   Age                   9971 non-null   float64
 1   Gender                9981 non-null   object
 2   Blood Pressure        9981 non-null   float64
 3   Cholesterol Level     9970 non-null   float64
```

```
 4   Exercise Habits       9975 non-null   object
 5   Smoking               9975 non-null   object
 6   Family Heart Disease  9979 non-null   object
 7   Diabetes              9970 non-null   object
 8   BMI                   9978 non-null   float64
 9   High Blood Pressure   9974 non-null   object
 10  Low HDL Cholesterol   9975 non-null   object
 11  High LDL Cholesterol  9974 non-null   object
 12  Alcohol Consumption   7414 non-null   object
 13  Stress Level          9978 non-null   object
 14  Sleep Hours           9975 non-null   float64
 15  Sugar Consumption     9970 non-null   object
 16  Triglyceride Level    9974 non-null   float64
 17  Fasting Blood Sugar   9978 non-null   float64
 18  CRP Level             9974 non-null   float64
 19  Homocysteine Level    9980 non-null   float64
 20  Heart Disease Status  10000 non-null  object
dtypes: float64(9), object(12)
memory usage: 1.6+ MB
```

[106]:
```
print(df.loc[10:13,["Age","Gender","Blood Pressure"]])
```

```
     Age  Gender  Blood Pressure
10  36.0  Female           179.0
11  40.0  Female           134.0
12  28.0  Female           143.0
13  28.0  Female           134.0
```

[107]:
```
df.groupby('Exercise Habits')['Blood Pressure'].mean()
```

[107]:
```
Exercise Habits
High      149.858712
Low       149.588001
Medium    149.829819
Name: Blood Pressure, dtype: float64
```

[108]:
```
df.groupby(['Exercise Habits','Gender'])[['Blood Pressure','Sleep Hours']].
  ↪mean()
```

[108]:

Exercise Habits	Gender	Blood Pressure	Sleep Hours
High	Female	150.014109	7.004038
	Male	149.713770	6.958927
Low	Female	149.463145	7.001250
	Male	149.710172	7.016555
Medium	Female	149.957565	6.964543
	Male	149.755187	6.999762

[109]:
```
import numpy as np
np.mean(df['Blood Pressure'])
```

[109]:
```
149.75773970544034
```

[110]:
```
X=df.drop("Heart Disease Status",axis=1)
y=df[["Heart Disease Status"]]
```

[111]:
```
X.head(2)
```

[111]:
```
    Age  Gender  Blood Pressure  Cholesterol Level Exercise Habits Smoking  \
0  56.0    Male           153.0              155.0            High     Yes
1  69.0  Female           146.0              286.0            High      No

  Family Heart Disease Diabetes        BMI High Blood Pressure  \
0                  Yes       No  24.991591                 Yes
1                  Yes      Yes  25.221799                  No

  Low HDL Cholesterol High LDL Cholesterol Alcohol Consumption Stress Level  \
0                 Yes                   No                High       Medium
1                 Yes                   No              Medium         High

   Sleep Hours Sugar Consumption  Triglyceride Level  Fasting Blood Sugar  \
0     7.633228            Medium               342.0                  NaN
1     8.744034            Medium               133.0                157.0

   CRP Level  Homocysteine Level
0  12.969246           12.387250
1   9.355389           19.298875
```

[112]:
```
y.head(2)
```

[112]:
```
  Heart Disease Status
0                   No
1                   No
```

[113]:
```
#y.nunique()
y.value_counts()
```

[113]:
```
Heart Disease Status
No                     8000
Yes                    2000
Name: count, dtype: int64
```

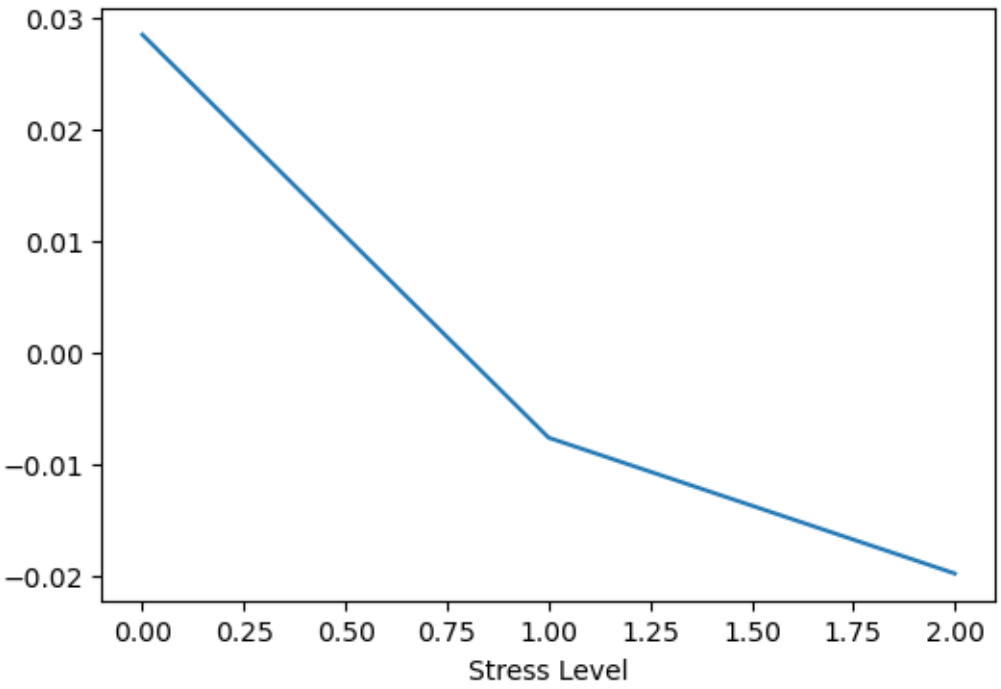
[114]:
```
X.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 10000 entries, 0 to 9999
```

```
Data columns (total 20 columns):
 #   Column                Non-Null Count  Dtype
---  ------                --------------  -----
 0   Age                   9971 non-null   float64
 1   Gender                9981 non-null   object
 2   Blood Pressure        9981 non-null   float64
 3   Cholesterol Level     9970 non-null   float64
 4   Exercise Habits       9975 non-null   object
 5   Smoking               9975 non-null   object
 6   Family Heart Disease  9979 non-null   object
 7   Diabetes              9970 non-null   object
 8   BMI                   9978 non-null   float64
 9   High Blood Pressure   9974 non-null   object
 10  Low HDL Cholesterol   9975 non-null   object
 11  High LDL Cholesterol  9974 non-null   object
 12  Alcohol Consumption   7414 non-null   object
 13  Stress Level          9978 non-null   object
 14  Sleep Hours           9975 non-null   float64
 15  Sugar Consumption     9970 non-null   object
 16  Triglyceride Level    9974 non-null   float64
 17  Fasting Blood Sugar   9978 non-null   float64
 18  CRP Level             9974 non-null   float64
 19  Homocysteine Level    9980 non-null   float64
dtypes: float64(9), object(11)
memory usage: 1.5+ MB
```

[149]:
```
plt.figure(figsize=(6,4))
df.groupby('Stress Level')['Sleep Hours'].mean().plot()
```

[149]:
```
<Axes: xlabel='Stress Level'>
```



[116]:
```
from scipy import stats
stats.mode(X['Age'])
```

[116]:
```
ModeResult(mode=71.0, count=187)
```

[117]:
```
a=np.array([1,2,3,4,5])
p=np.percentile(a,50)
print(p)
```

```
3.0
```

[118]:
```
for i in range(10,20,3): print(i)
```

```
10
13
16
19
```

[119]:
```
df.isnull().sum()
```

[119]:
```
Age                    29
Gender                 19
Blood Pressure         19
Cholesterol Level      30
Exercise Habits        25
```

```
Smoking                    25
Family Heart Disease       21
Diabetes                   30
BMI                        22
High Blood Pressure        26
Low HDL Cholesterol        25
High LDL Cholesterol       26
Alcohol Consumption      2586
Stress Level               22
Sleep Hours                25
Sugar Consumption          30
Triglyceride Level         26
Fasting Blood Sugar        22
CRP Level                  26
Homocysteine Level         20
Heart Disease Status        0
dtype: int64
```

[120]:
```
df_num=df.copy()
for col in df_num.columns:
    if df_num[col].dtype==object:
        df_num=df_num.drop(col,axis=1)
```

[121]:
```
from sklearn.impute import SimpleImputer

imputer=SimpleImputer(strategy="mean")
imputer.fit(df_num)
```

[121]:
```
SimpleImputer()
```

[122]:
```
imputer.statistics_
```

[122]:
```
array([ 49.29625915, 149.75773971, 225.42557673,  29.07726893,
         6.99132945, 250.73440946, 120.14221287,   7.47220059,
        12.45627088])
```

[123]:
```
X=imputer.transform(df_num)
df_transformed=pd.DataFrame(X,columns=df_num.columns,index=df_num.index)
```

[124]:
```
df_transformed.isnull().sum()
```

[124]:
```
Age                    0
Blood Pressure         0
Cholesterol Level      0
BMI                    0
Sleep Hours            0
Triglyceride Level     0
Fasting Blood Sugar    0
```

```
CRP Level              0
Homocysteine Level     0
dtype: int64
```

[125]:
```
df_cat=df.copy()
for col in df_cat.columns:
    if df_cat[col].dtype!=object:
        df_cat=df_cat.drop(col,axis=1)
```

[126]:
```
imputer=SimpleImputer(strategy="most_frequent")
imputer.fit(df_cat)
imputer.statistics_
```

[126]:
```
array(['Male', 'High', 'Yes', 'No', 'No', 'Yes', 'Yes', 'No', 'Medium',
       'Medium', 'Low', 'No'], dtype=object)
```

[127]:
```
X=imputer.transform(df_cat)
df_cat_transformed=pd.DataFrame(X,columns=df_cat.columns,index=df_cat.index)
```

[128]:
```
df_cat_transformed.isnull().sum()
```

[128]:
```
Gender                 0
Exercise Habits        0
Smoking                0
Family Heart Disease   0
Diabetes               0
High Blood Pressure    0
Low HDL Cholesterol    0
High LDL Cholesterol   0
Alcohol Consumption    0
Stress Level           0
Sugar Consumption      0
Heart Disease Status   0
dtype: int64
```

[129]:
```
df_trans=pd.concat([df_transformed,df_cat_transformed],axis=1)
df_trans
```

[129]:

	Age	Blood Pressure	Cholesterol Level	BMI	Sleep Hours	\
0	56.0	153.0	155.0	24.991591	7.633228	
1	69.0	146.0	286.0	25.221799	8.744034	
2	46.0	126.0	216.0	29.855447	4.440440	
3	32.0	122.0	293.0	24.130477	5.249405	
4	60.0	166.0	242.0	20.486289	7.030971	
…	…	…	…	…	…	
9995	25.0	136.0	243.0	18.788791	6.834954	
9996	38.0	172.0	154.0	31.856801	8.247784	
9997	73.0	152.0	201.0	26.899911	4.436762	

```
9998  23.0           142.0              299.0  34.964026     8.526329
9999  38.0           128.0              193.0  25.111295     5.659394

      Triglyceride Level  Fasting Blood Sugar  CRP Level  Homocysteine Level  \
0                  342.0           120.142213  12.969246           12.387250
1                  133.0           157.000000   9.355389           19.298875
2                  393.0            92.000000  12.709873           11.230926
3                  293.0            94.000000  12.509046            5.961958
4                  263.0           154.000000  10.381259            8.153887
…                  …                  …        …                  …
9995               343.0           133.000000   3.588814           19.132004
9996               377.0            83.000000   2.658267            9.715709
9997               248.0            88.000000   4.408867            9.492429
9998               113.0           153.000000   7.215634           11.873486
9999               121.0           149.000000  14.387810            6.208531

      Gender  … Smoking Family Heart Disease Diabetes High Blood Pressure  \
0       Male  …     Yes                  Yes       No                  Yes
1     Female  …      No                  Yes      Yes                   No
2       Male  …      No                   No       No                   No
3     Female  …     Yes                  Yes       No                  Yes
4       Male  …     Yes                  Yes      Yes                  Yes
…      …  …     …                  …      …                  …
9995  Female  …     Yes                   No       No                  Yes
9996    Male  …      No                   No       No                  Yes
9997    Male  …     Yes                   No      Yes                   No
9998    Male  …     Yes                   No      Yes                  Yes
9999  Female  …     Yes                  Yes      Yes                   No

     Low HDL Cholesterol High LDL Cholesterol Alcohol Consumption  \
0                    Yes                   No                High
1                    Yes                   No              Medium
2                    Yes                  Yes                 Low
3                     No                  Yes                 Low
4                     No                   No                 Low
…                    …                  …                  …
9995                  No                  Yes              Medium
9996                  No                  Yes              Medium
9997                 Yes                  Yes              Medium
9998                  No                  Yes              Medium
9999                 Yes                  Yes                High

     Stress Level Sugar Consumption Heart Disease Status
0          Medium            Medium                   No
1            High            Medium                   No
2             Low               Low                   No
3            High              High                   No
```

```
4              High              High                    No
…              …                 …                   …
9995           High            Medium                   Yes
9996           High               Low                   Yes
9997            Low               Low                   Yes
9998           High            Medium                   Yes
9999         Medium              High                   Yes

[10000 rows x 21 columns]
```

0.1 displaying mean median ….

[130]:
```
df_trans.describe()
```

[130]:
```
                Age  Blood Pressure  Cholesterol Level           BMI  \
count  10000.000000    10000.000000       10000.000000  10000.000000
mean      49.296259      149.757740         225.425577     29.077269
std       18.167567       17.556265          43.510390      6.300156
min       18.000000      120.000000         150.000000     18.002837
25%       34.000000      134.000000         187.000000     23.668887
50%       49.000000      150.000000         225.425577     29.077269
75%       65.000000      165.000000         263.000000     34.509009
max       80.000000      180.000000         300.000000     39.996954

        Sleep Hours  Triglyceride Level  Fasting Blood Sugar     CRP Level  \
count  10000.000000        10000.000000         10000.000000  10000.000000
mean       6.991329          250.734409           120.142213      7.472201
std        1.751002           86.953954            23.558052      4.334601
min        4.000605          100.000000            80.000000      0.003647
25%        5.455288          176.000000            99.000000      3.681800
50%        6.996016          250.734409           120.000000      7.472201
75%        8.527938          326.000000           141.000000     11.244879
max        9.999952          400.000000           160.000000     14.997087

       Homocysteine Level
count        10000.000000
mean            12.456271
std              4.319100
min              5.000236
25%              8.729771
50%             12.421274
75%             16.130968
max             19.999037
```

[131]:
```
for col in df_trans.columns:
    if(df_trans[col].dtype!=object):
        print(col,"Variance",df_trans[col].var())
```

```
        print(col,"Median",df_trans[col].median())
        print(col,"Mode",df_trans[col].mode())
```

```
Age Variance 330.0604910957757
Age Median 49.0
Age Mode 0    71.0
Name: Age, dtype: float64
Blood Pressure Variance 308.2224437051496
Blood Pressure Median 150.0
Blood Pressure Mode 0    134.0
Name: Blood Pressure, dtype: float64
Cholesterol Level Variance 1893.154043197707
Cholesterol Level Median 225.42557673019058
Cholesterol Level Mode 0    292.0
Name: Cholesterol Level, dtype: float64
BMI Variance 39.69196365052222
BMI Median 29.077268927511035
BMI Mode 0    29.077269
Name: BMI, dtype: float64
Sleep Hours Variance 3.0660085806056365
Sleep Hours Median 6.996016461298234
Sleep Hours Mode 0    6.991329
Name: Sleep Hours, dtype: float64
Triglyceride Level Variance 7560.990044071526
Triglyceride Level Median 250.73440946460798
Triglyceride Level Mode 0    307.0
Name: Triglyceride Level, dtype: float64
Fasting Blood Sugar Variance 554.9818181758084
Fasting Blood Sugar Median 120.0
Fasting Blood Sugar Mode 0    119.0
Name: Fasting Blood Sugar, dtype: float64
CRP Level Variance 18.788765926151363
CRP Level Median 7.472200593944747
CRP Level Mode 0    7.472201
Name: CRP Level, dtype: float64
Homocysteine Level Variance 18.654623382292105
Homocysteine Level Median 12.421273890606692
Homocysteine Level Mode 0    12.456271
Name: Homocysteine Level, dtype: float64
```

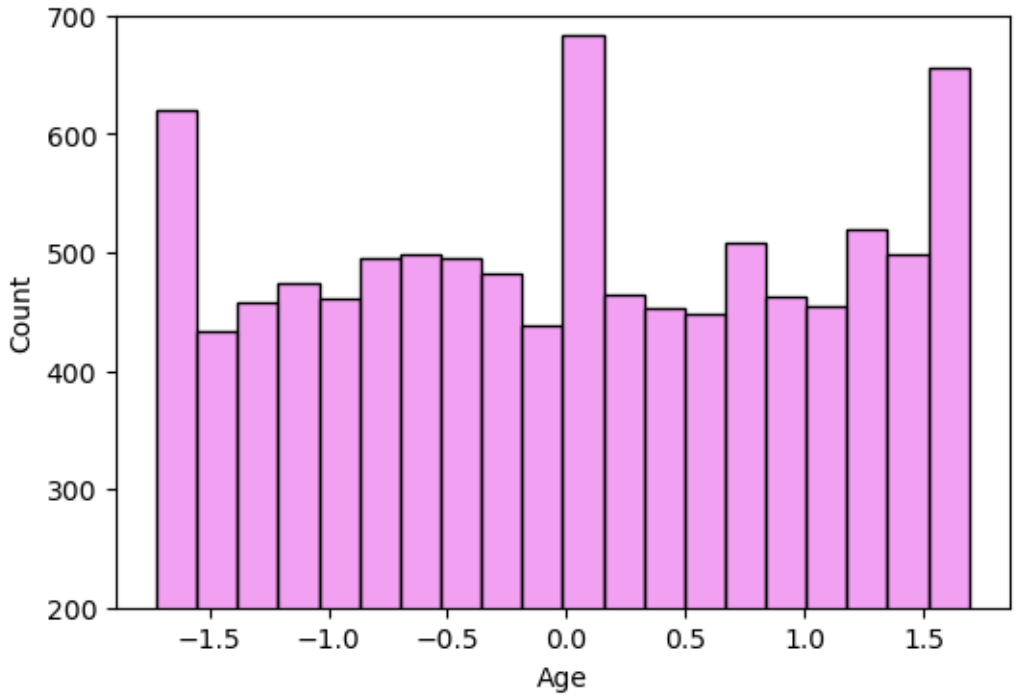
[132]:
```
df=df_trans
```

[133]:
```
def Euc_dist(point1,point2):
    return np.sqrt(np.sum((np.array(point1)-np.array(point2))**2))

def Man_dist(point1,point2):
    return sum(abs(a-b) for a,b in zip(point1,point2))
```

[150]:
```
import seaborn as sns
plt.figure(figsize=(6,4))
a=sns.histplot(df['Age'],bins=20,color='violet')
a.set_ylim(200,700)
```

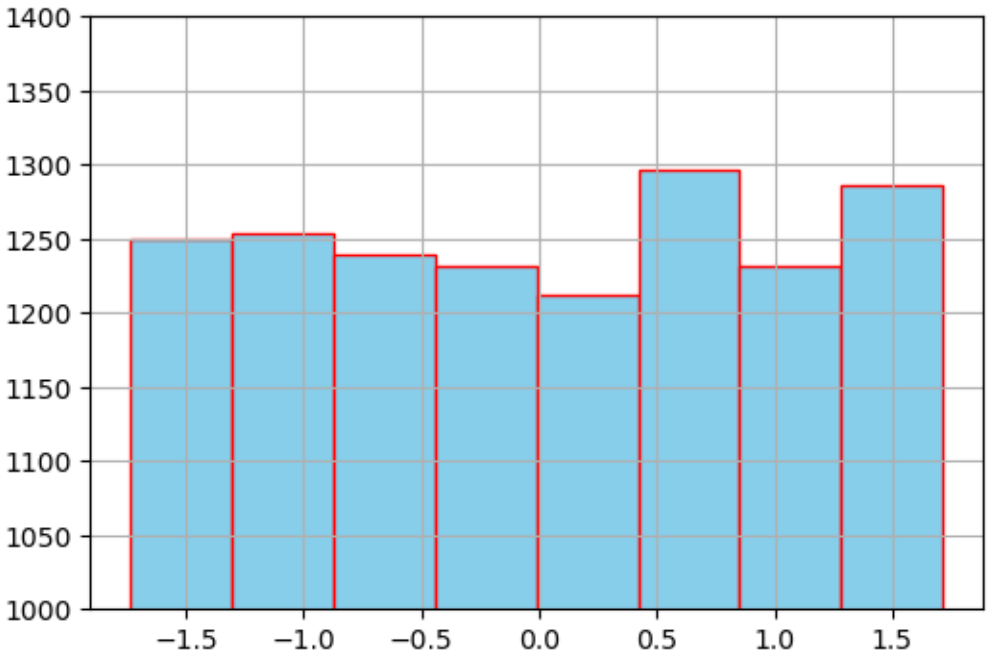
[150]: (200.0, 700.0)



there is no proper trend between age and the no. of instances, there are hikes around 20, 50 and 80 ages

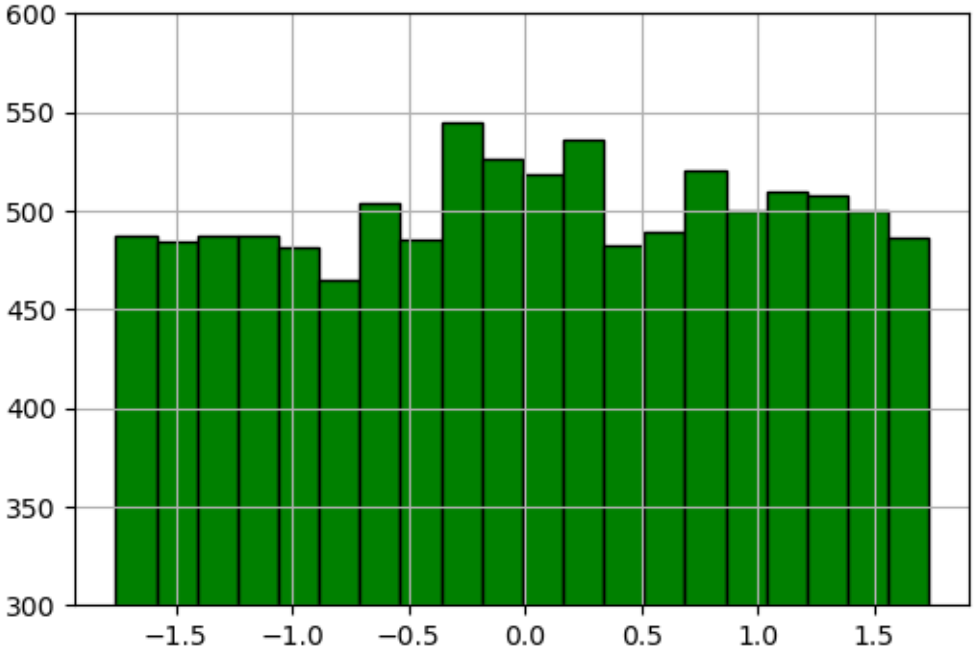
[151]:
```
plt.figure(figsize=(6,4))
a=df['Cholesterol Level'].hist(bins=8,color='skyblue',edgecolor='red')
a.set_ylim(1000,1400)
```

[151]: (1000.0, 1400.0)



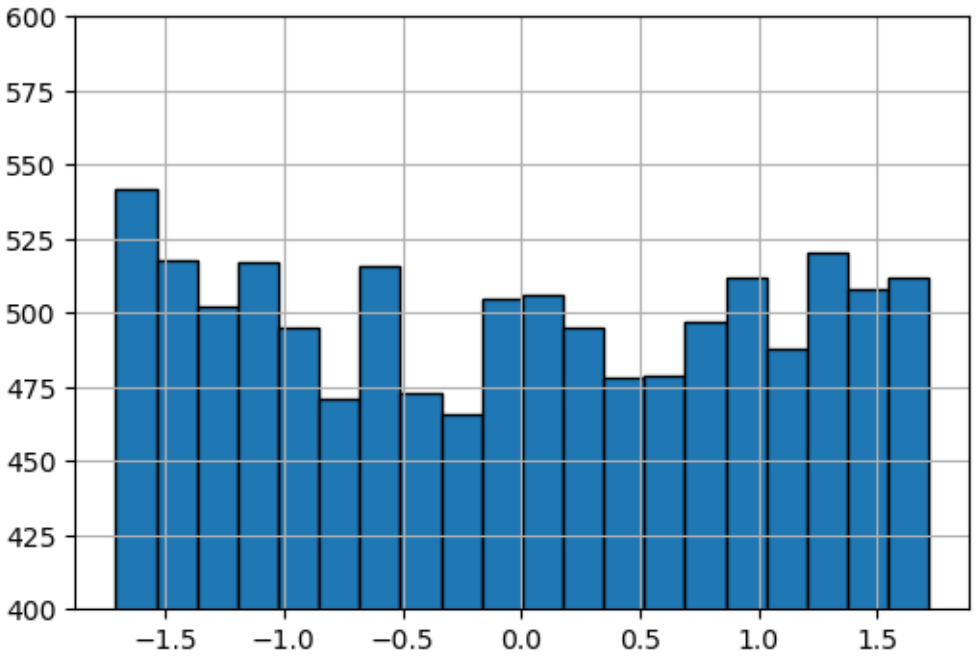
[152]:
```
plt.figure(figsize=(6,4))
df['BMI'].hist(color='green',bins=20,edgecolor='black').set_ylim(300,600)
```

[152]: (300.0, 600.0)



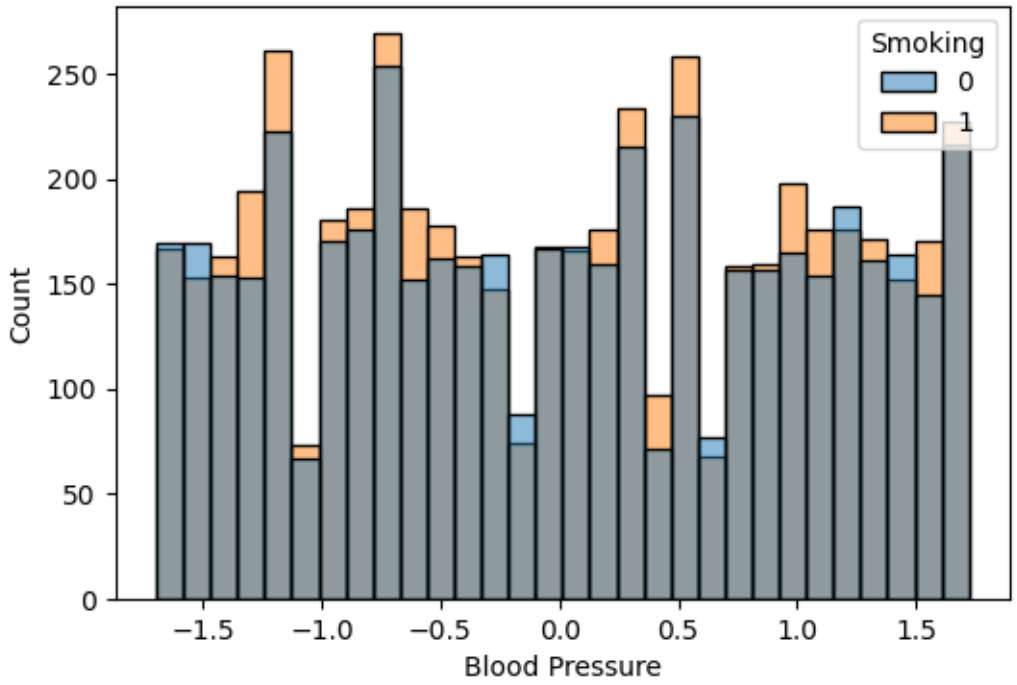
[153]:
```
plt.figure(figsize=(6,4))
df['Sleep Hours'].hist(bins=20,edgecolor='black').set_ylim(400,600)
```

[153]: (400.0, 600.0)



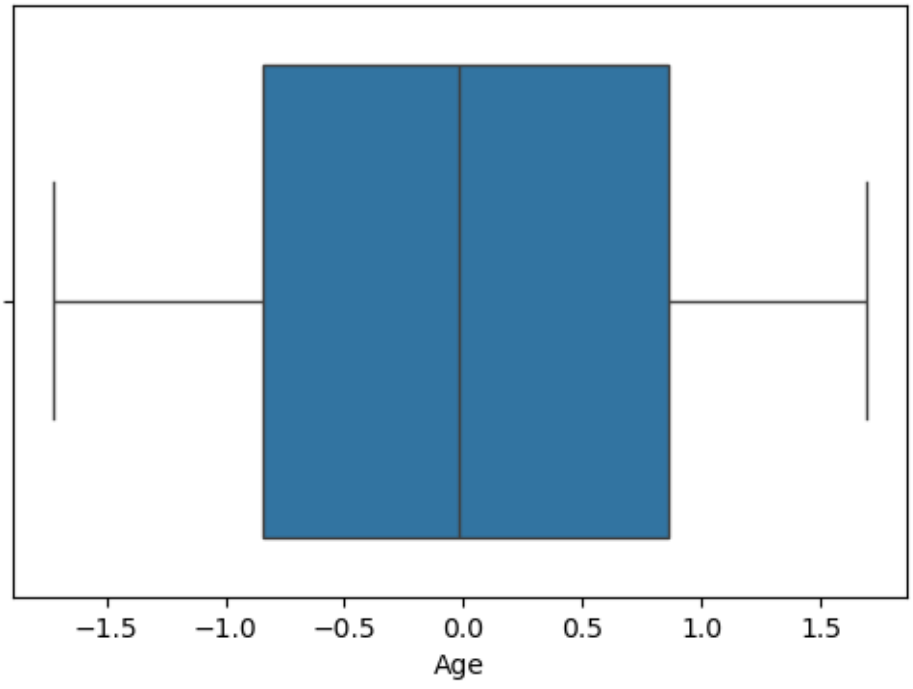
[154]:
```
import seaborn as sns
plt.figure(figsize=(6,4))
sns.histplot(df,x=df['Blood Pressure'],hue=df['Smoking'],bins=30)
```

[154]: <Axes: xlabel='Blood Pressure', ylabel='Count'>



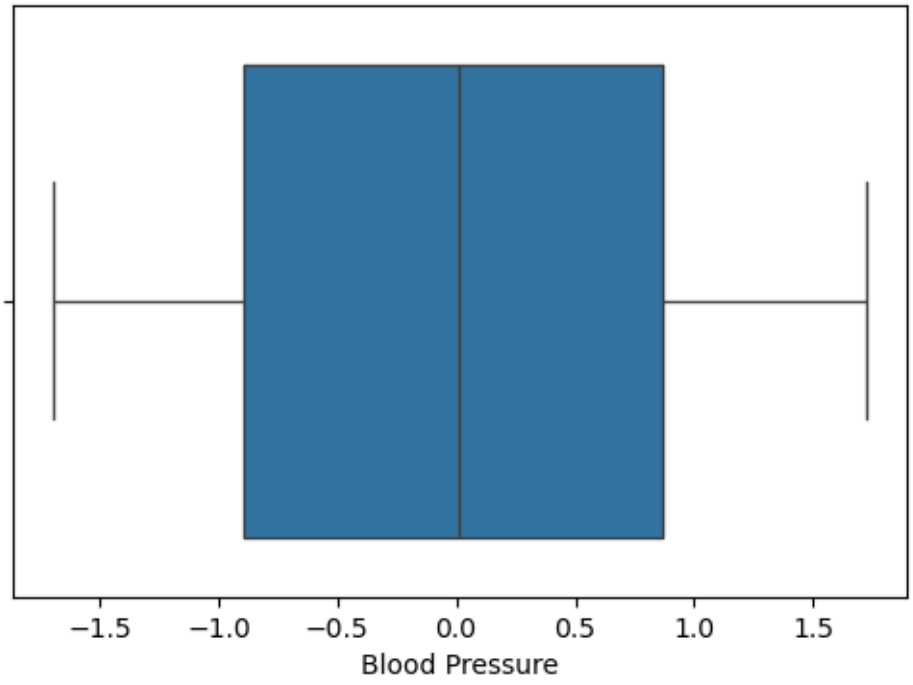
[155]:
```
plt.figure(figsize=(6,4))
sns.boxplot(x='Age',data=df)
```

[155]: <Axes: xlabel='Age'>



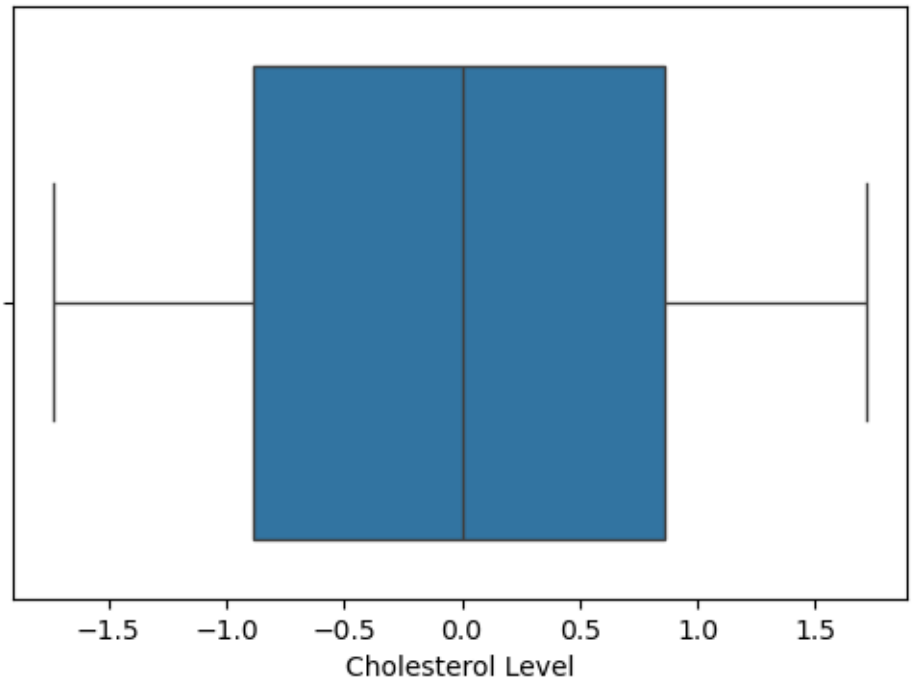
[156]:
```
plt.figure(figsize=(6,4))
sns.boxplot(x='Blood Pressure',data=df)
```

[156]: <Axes: xlabel='Blood Pressure'>



[157]:
```
plt.figure(figsize=(6,4))
sns.boxplot(x='Cholesterol Level',data=df)
```

[157]: <Axes: xlabel='Cholesterol Level'>



The data used here is for predicting heart disease status of a human. The data overall is quite natural with not much variations. It does not have any outliers. All the null values were handlers which also did not generate any outliers.

[142]:
```
sum(df.duplicated())
```

[142]: 0

[143]:
```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 10000 entries, 0 to 9999
Data columns (total 21 columns):
 #   Column                Non-Null Count  Dtype
---  ------                --------------  -----
 0   Age                   10000 non-null  float64
 1   Blood Pressure        10000 non-null  float64
 2   Cholesterol Level     10000 non-null  float64
 3   BMI                   10000 non-null  float64
 4   Sleep Hours           10000 non-null  float64
 5   Triglyceride Level    10000 non-null  float64
 6   Fasting Blood Sugar   10000 non-null  float64
 7   CRP Level             10000 non-null  float64
 8   Homocysteine Level    10000 non-null  float64
 9   Gender                10000 non-null  object
```

```
 10  Exercise Habits       10000 non-null  object
 11  Smoking               10000 non-null  object
 12  Family Heart Disease  10000 non-null  object
 13  Diabetes              10000 non-null  object
 14  High Blood Pressure   10000 non-null  object
 15  Low HDL Cholesterol   10000 non-null  object
 16  High LDL Cholesterol  10000 non-null  object
 17  Alcohol Consumption   10000 non-null  object
 18  Stress Level          10000 non-null  object
 19  Sugar Consumption     10000 non-null  object
 20  Heart Disease Status  10000 non-null  object
dtypes: float64(9), object(12)
memory usage: 1.6+ MB
```

[144]:
```
from sklearn import preprocessing
label_encoder=preprocessing.LabelEncoder()
for column in df.columns:
    if df[column].dtype=="object":
        df[column]=label_encoder.fit_transform(df[column])

df.head()
```

[144]:

	Age	Blood Pressure	Cholesterol Level	BMI	Sleep Hours	\
0	56.0	153.0	155.0	24.991591	7.633228	
1	69.0	146.0	286.0	25.221799	8.744034	
2	46.0	126.0	216.0	29.855447	4.440440	
3	32.0	122.0	293.0	24.130477	5.249405	
4	60.0	166.0	242.0	20.486289	7.030971	

	Triglyceride Level	Fasting Blood Sugar	CRP Level	Homocysteine Level	\
0	342.0	120.142213	12.969246	12.387250	
1	133.0	157.000000	9.355389	19.298875	
2	393.0	92.000000	12.709873	11.230926	
3	293.0	94.000000	12.509046	5.961958	
4	263.0	154.000000	10.381259	8.153887	

	Gender	…	Smoking	Family Heart Disease	Diabetes	High Blood Pressure	\
0	1	…	1	1	0	1	
1	0	…	0	1	1	0	
2	1	…	0	0	0	0	
3	0	…	1	1	0	1	
4	1	…	1	1	1	1	

	Low HDL Cholesterol	High LDL Cholesterol	Alcohol Consumption	\
0	1	0	0	
1	1	0	2	
2	1	1	1	

```
3                    0                    1                    1
4                    0                    0                    1

   Stress Level  Sugar Consumption  Heart Disease Status
0             2                  2                     0
1             0                  2                     0
2             1                  1                     0
3             0                  0                     0
4             0                  0                     0

[5 rows x 21 columns]
```

[145]:
```
for column in df.columns:
    if(df[column].dtype=="float64"):
        df[column]=((df[column]-df[column].mean())/df[column].std())
```

[146]:
```
df.head()
```

[146]:
```
        Age  Blood Pressure  Cholesterol Level       BMI  Sleep Hours  \
0  0.368995        0.184678          -1.618592 -0.648504     0.366589
1  1.084556       -0.214040           1.392183 -0.611964     1.000972
2 -0.181436       -1.353234          -0.216628  0.123517    -1.456817
3 -0.952040       -1.581073           1.553064 -0.785186    -0.994816
4  0.589168        0.925155           0.380930 -1.363614     0.022640

   Triglyceride Level  Fasting Blood Sugar  CRP Level  Homocysteine Level  \
0            1.049585             0.000000   1.268178           -0.015980
1           -1.353986             1.564552   0.434455            1.584266
2            1.636103            -1.194590   1.208340           -0.283704
3            0.486069            -1.109693   1.162009           -1.503626
4            0.141058             1.437207   0.671125           -0.996130

   Gender  …  Smoking  Family Heart Disease  Diabetes  High Blood Pressure  \
0       1  …        1                     1         0                    1
1       0  …        0                     1         1                    0
2       1  …        0                     0         0                    0
3       0  …        1                     1         0                    1
4       1  …        1                     1         1                    1

   Low HDL Cholesterol  High LDL Cholesterol  Alcohol Consumption  \
0                    1                     0                    0
1                    1                     0                    2
2                    1                     1                    1
3                    0                     1                    1
4                    0                     0                    1

   Stress Level  Sugar Consumption  Heart Disease Status
```

```
0              2                  2                  0
1              0                  2                  0
2              1                  1                  0
3              0                  0                  0
4              0                  0                  0

[5 rows x 21 columns]
```