asasaignment-3-lokesh

September 13, 2023

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

[2]:
```
df = pd.read_csv('/content/penguins_size.csv')
df.head()
```

[2]:

	species	island	culmen_length_mm	culmen_depth_mm	flipper_length_mm	body_mass_g	sex
0	Adelie	Torgersen	39.1	18.7	181.0	3750.0	MALE
1	Adelie	Torgersen	39.5	17.4	186.0	3800.0	FEMALE
2	Adelie	Torgersen	40.3	18.0	195.0	3250.0	FEMALE
3	Adelie	Torgersen	NaN	NaN	NaN	NaN	NaN
4	Adelie	Torgersen	36.7	19.3	193.0	3450.0	FEMALE

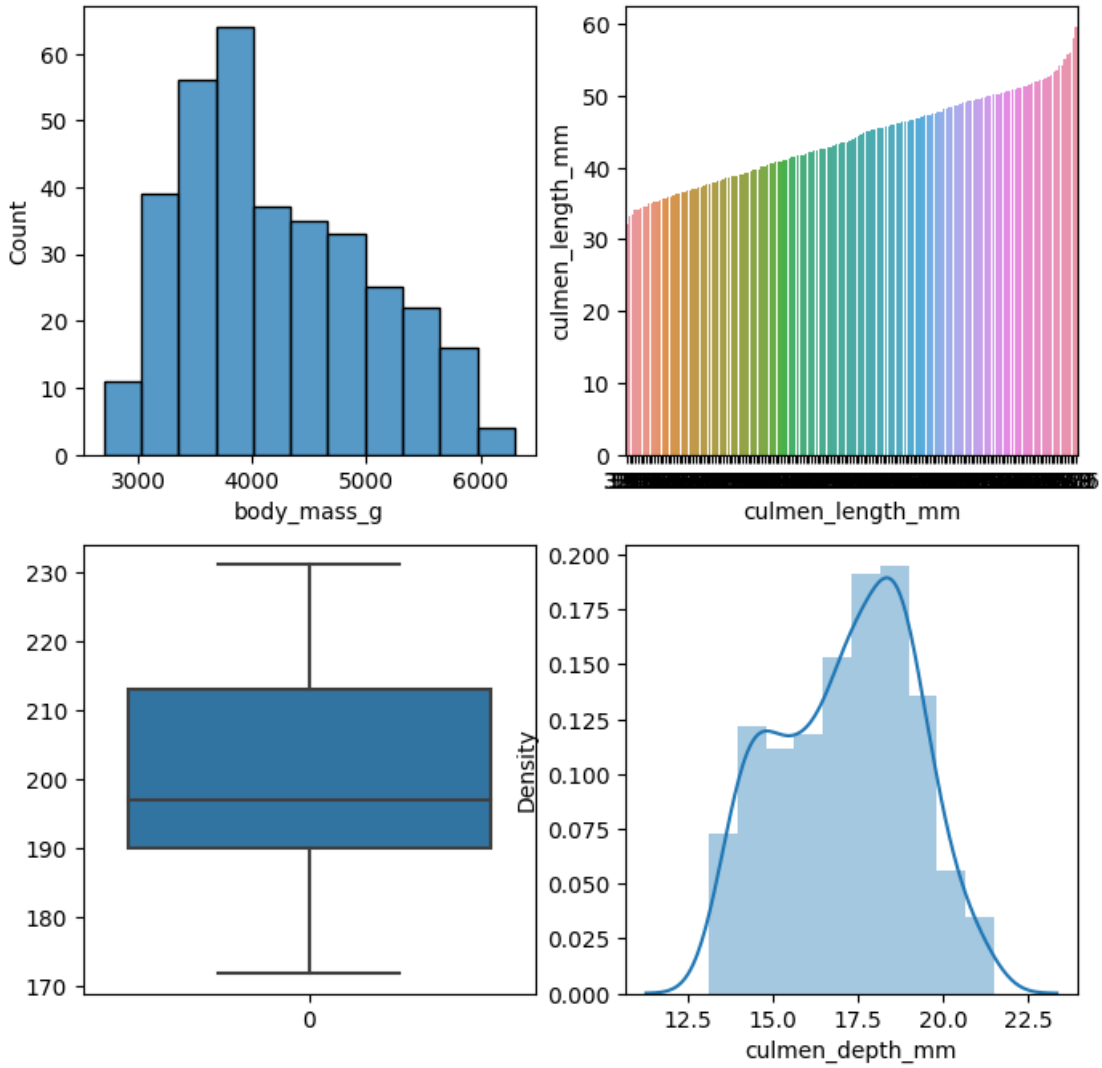
[16]:
```
from matplotlib import rcParams
rcParams['figure.figsize']=8,8
fig,axes=plt.subplots(2,2)
sns.histplot(data=df['body_mass_g'],ax=axes[0,0])
sns.distplot(df['culmen_depth_mm'],ax=axes[1,1])
sns.barplot(x=df['culmen_length_mm'],y=df['culmen_length_mm'],ax=axes[0,1])
sns.boxplot(data=df['flipper_length_mm'],ax=axes[1,0])
```

```
<ipython-input-16-5906f08a3de5>:5: UserWarning:

`distplot` is a deprecated function and will be removed in seaborn v0.14.0.

Please adapt your code to use either `displot` (a figure-level function with
similar flexibility) or `histplot` (an axes-level function for histograms).

For a guide to updating your code to use the new functions, please see
```

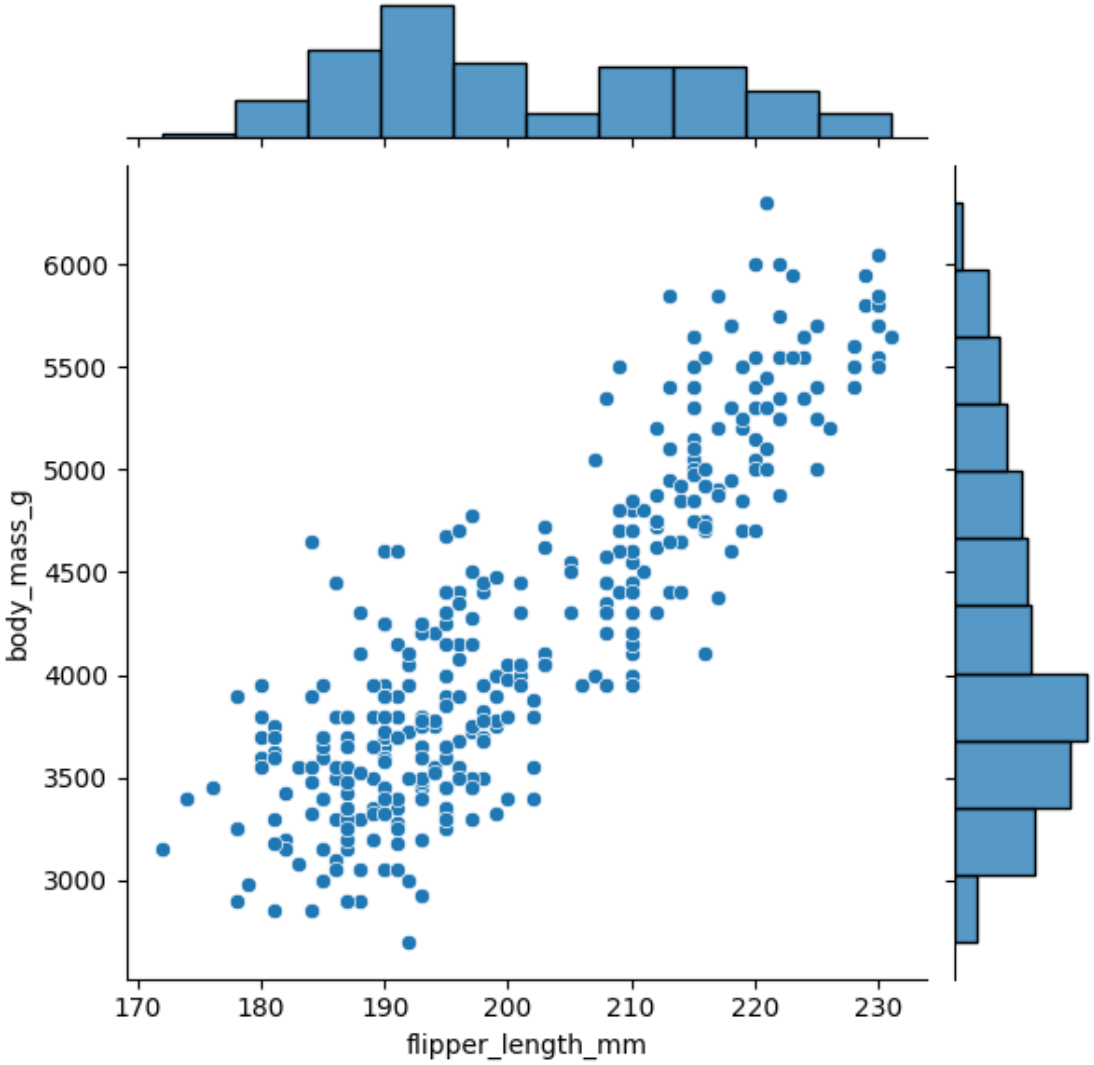
https://gist.github.com/mwaskom/de44147ed2974457ad6372750bbe5751

```
sns.distplot(df['culmen_depth_mm'],ax=axes[1,1])
```

[16]: <Axes: >



[17]:
```
sns.jointplot(x='flipper_length_mm',y='body_mass_g',data=df)
plt.show()
```



[18]:
```
df.corr()
```

```
<ipython-input-18-2f6f6606aa2c>:1: FutureWarning: The default value of
numeric_only in DataFrame.corr is deprecated. In a future version, it will
default to False. Select only valid columns or specify the value of numeric_only
to silence this warning.
  df.corr()
```

[18]:

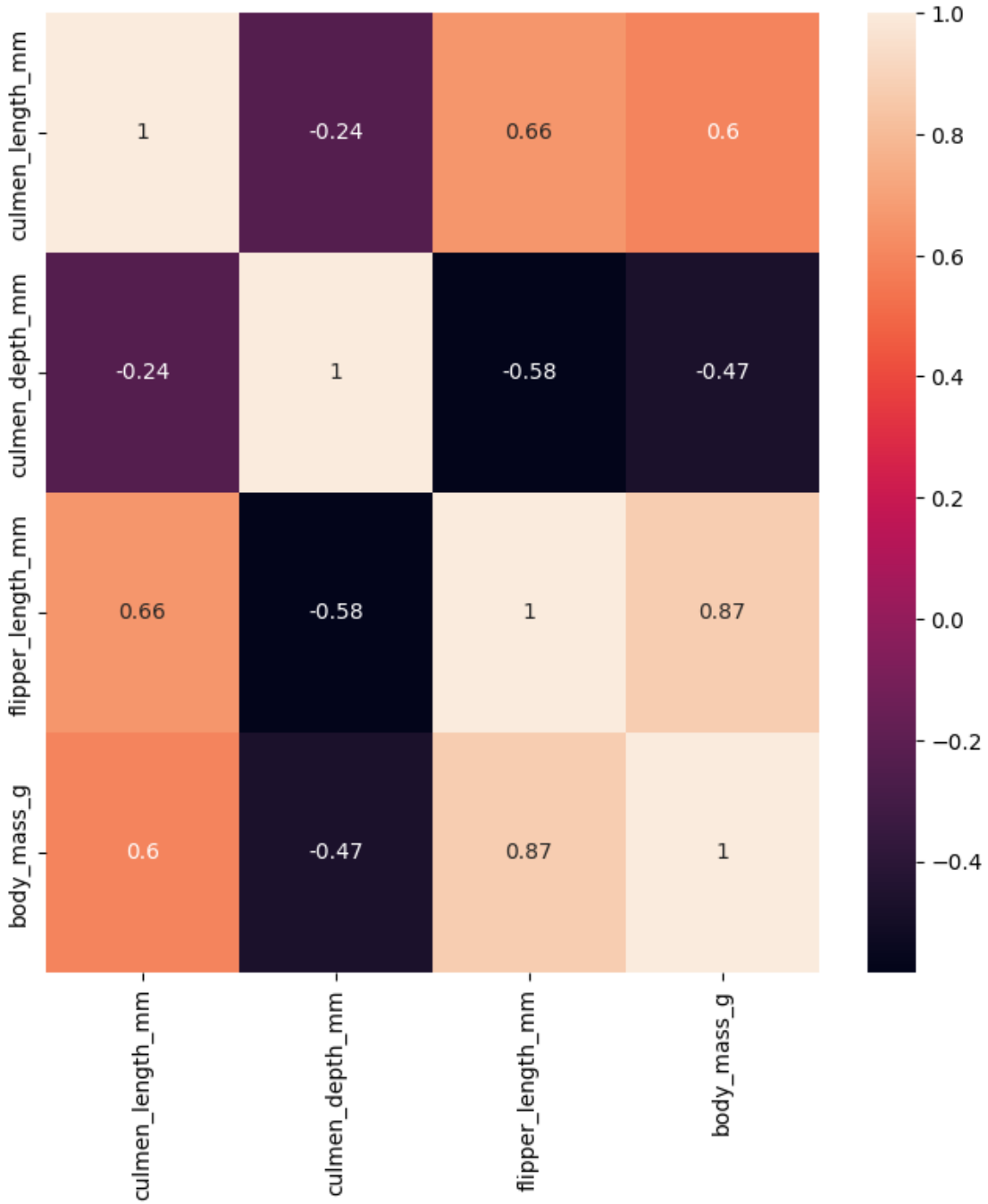
	culmen_length_mm	culmen_depth_mm	flipper_length_mm	\
culmen_length_mm	1.000000	-0.235053	0.656181	
culmen_depth_mm	-0.235053	1.000000	-0.583851	
flipper_length_mm	0.656181	-0.583851	1.000000	
body_mass_g	0.595110	-0.471916	0.871202	

```
                   body_mass_g
culmen_length_mm      0.595110
culmen_depth_mm      -0.471916
flipper_length_mm     0.871202
body_mass_g           1.000000
```

[19]:
```
sns.heatmap(df.corr(),annot=True)
```

```
<ipython-input-19-8df7bcac526d>:1: FutureWarning: The default value of
numeric_only in DataFrame.corr is deprecated. In a future version, it will
default to False. Select only valid columns or specify the value of numeric_only
to silence this warning.
  sns.heatmap(df.corr(),annot=True)
```

[19]:
```
<Axes: >
```



[20]:
```
df.describe()
```

[20]:

	culmen_length_mm	culmen_depth_mm	flipper_length_mm	body_mass_g
count	342.000000	342.000000	342.000000	342.000000
mean	43.921930	17.151170	200.915205	4201.754386
std	5.459584	1.974793	14.061714	801.954536

```
min          32.100000        13.100000         172.000000  2700.000000
25%          39.225000        15.600000         190.000000  3550.000000
50%          44.450000        17.300000         197.000000  4050.000000
75%          48.500000        18.700000         213.000000  4750.000000
max          59.600000        21.500000         231.000000  6300.000000
```

[22]:
```
df.isnull().sum()
```

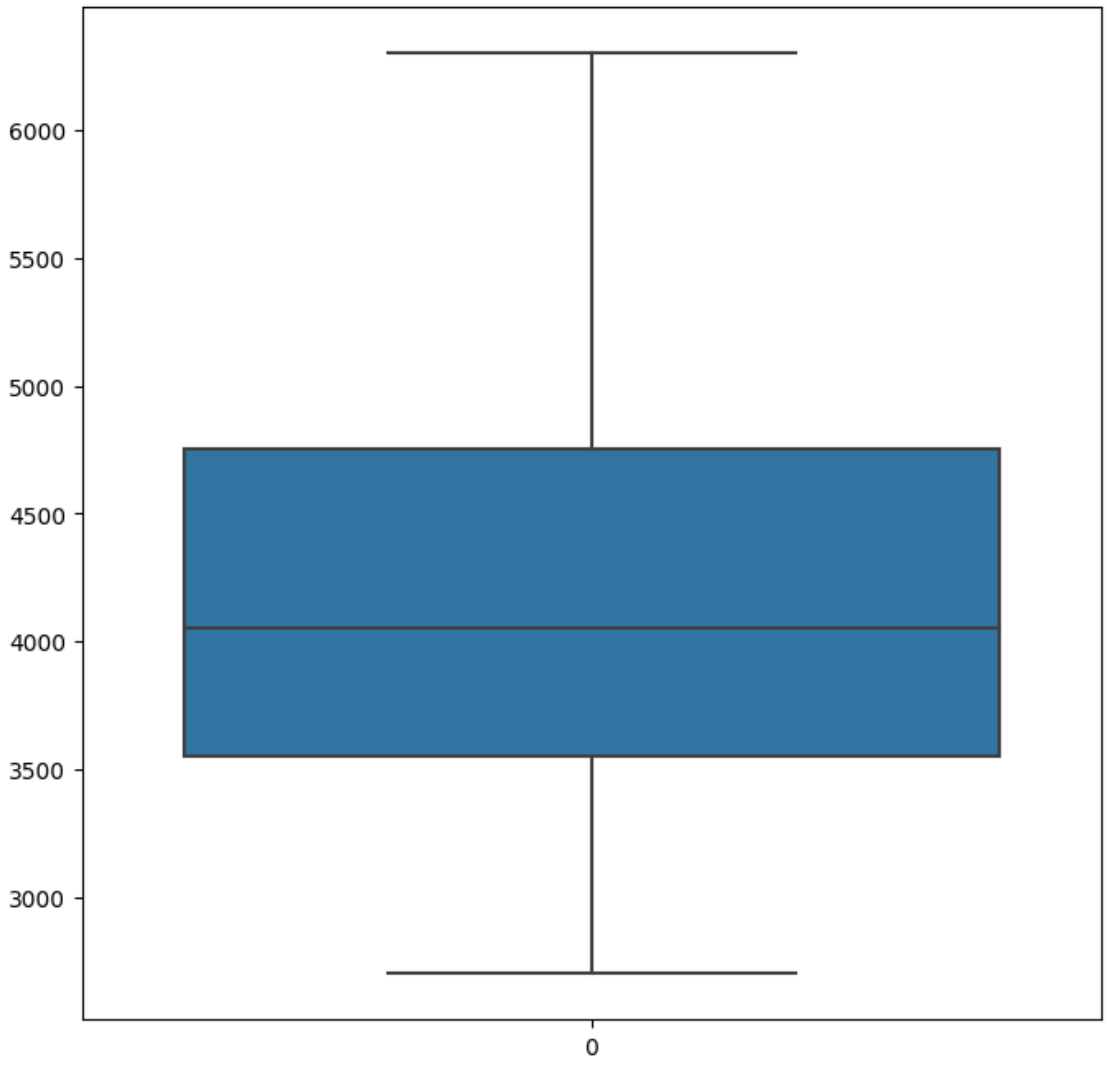
[22]:
```
species               0
island                0
culmen_length_mm      2
culmen_depth_mm       2
flipper_length_mm     2
body_mass_g           2
sex                  10
dtype: int64
```

[26]:
```
df['culmen_length_mm'].fillna(df['culmen_length_mm'].median(),inplace=True)
df['culmen_depth_mm'].fillna(df['culmen_depth_mm'].median(),inplace=True)
df['flipper_length_mm'].fillna(df['flipper_length_mm'].median(),inplace=True)
df['body_mass_g'].fillna(df['body_mass_g'].median(),inplace=True)
#df['sex'].fillna(df['sex'].median(),inplace=True)
most_frequent_category = df['sex'].mode()[0]
df['sex'].fillna(most_frequent_category, inplace=True)
df.head()
```

[26]:
```
  species     island  culmen_length_mm  culmen_depth_mm  flipper_length_mm  \
0  Adelie  Torgersen              39.10             18.7              181.0
1  Adelie  Torgersen              39.50             17.4              186.0
2  Adelie  Torgersen              40.30             18.0              195.0
3  Adelie  Torgersen              44.45             17.3              197.0
4  Adelie  Torgersen              36.70             19.3              193.0

   body_mass_g     sex
0       3750.0    MALE
1       3800.0  FEMALE
2       3250.0  FEMALE
3       4050.0    MALE
4       3450.0  FEMALE
```

[27]:
```
sns.boxplot(df.body_mass_g)
```

[27]:
```
<Axes: >
```



[28]:
```
q1 = df.body_mass_g.quantile(0.25)
q3 = df.body_mass_g.quantile(0.75)
IQR = q3-q1
upper_limit = q3+1.5*IQR
lower_limit = q1-1.5*IQR
df.median()
```

```
<ipython-input-28-619b8a44c144>:6: FutureWarning: The default value of
numeric_only in DataFrame.median is deprecated. In a future version, it will
default to False. In addition, specifying 'numeric_only=None' is deprecated.
Select only valid columns or specify the value of numeric_only to silence this
warning.
  df.median()
```

[28]:
```
culmen_length_mm       44.45
culmen_depth_mm        17.30
flipper_length_mm     197.00
body_mass_g          4050.00
dtype: float64
```

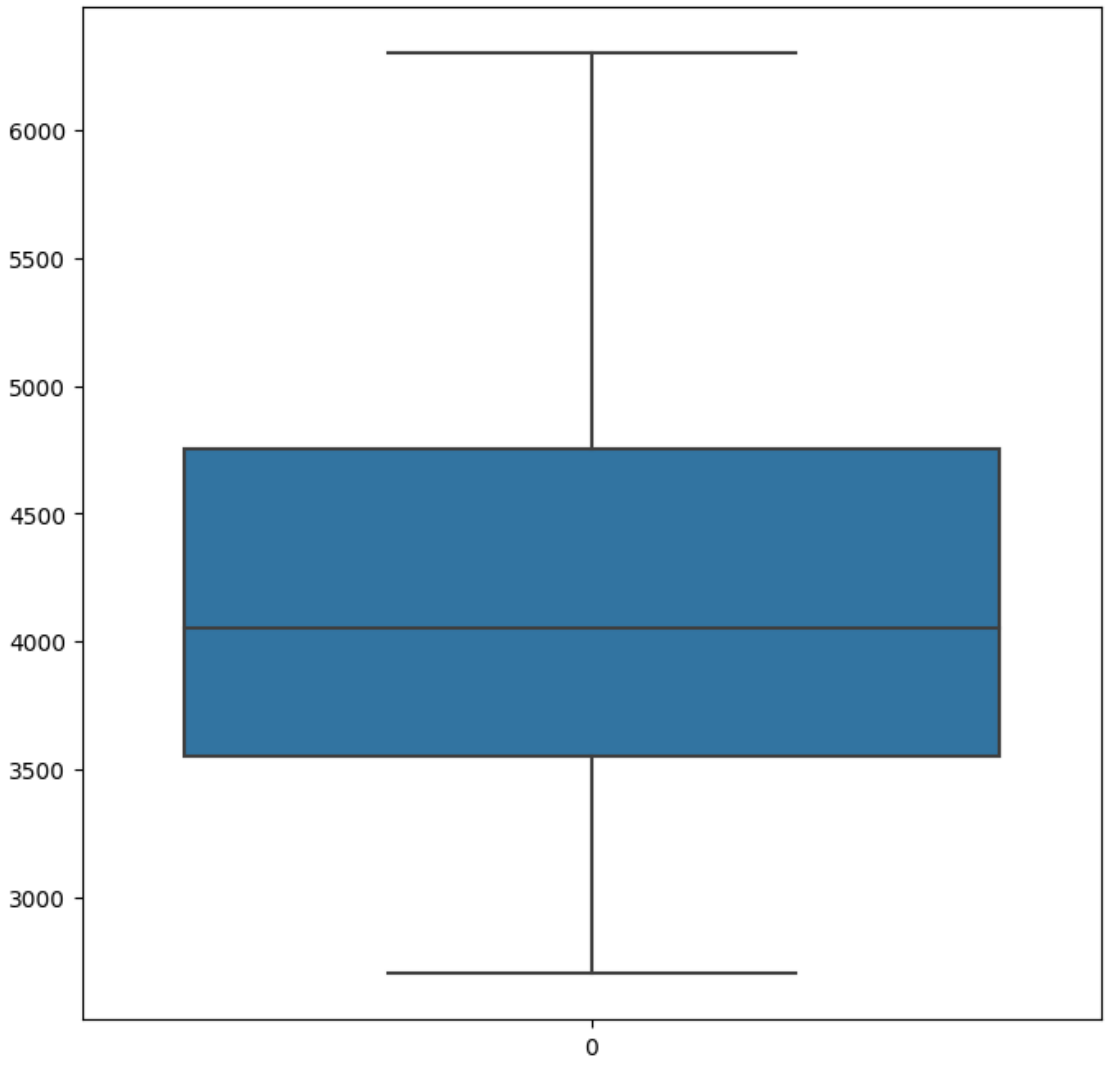
[30]:
```
df['body_mass_g'] = np.where(df['body_mass_g']>upper_limit,30,df['body_mass_g'])
df.head()
```

[30]:

	species	island	culmen_length_mm	culmen_depth_mm	flipper_length_mm	body_mass_g	sex
0	Adelie	Torgersen	39.10	18.7	181.0	3750.0	MALE
1	Adelie	Torgersen	39.50	17.4	186.0	3800.0	FEMALE
2	Adelie	Torgersen	40.30	18.0	195.0	3250.0	FEMALE
3	Adelie	Torgersen	44.45	17.3	197.0	4050.0	MALE
4	Adelie	Torgersen	36.70	19.3	193.0	3450.0	FEMALE

[32]:
```
sns.boxplot(df.body_mass_g)
```

[32]:
```
<Axes: >
```



[33]:
```
correlation_with_target = df.corr()['body_mass_g']
print(correlation_with_target)
```

```
culmen_length_mm     0.594925
culmen_depth_mm     -0.471942
flipper_length_mm    0.871221
body_mass_g          1.000000
Name: body_mass_g, dtype: float64
```

```
<ipython-input-33-e5b928a105f7>:1: FutureWarning: The default value of
numeric_only in DataFrame.corr is deprecated. In a future version, it will
default to False. Select only valid columns or specify the value of numeric_only
to silence this warning.
  correlation_with_target = df.corr()['body_mass_g']
```

[42]:
```
from sklearn.preprocessing import LabelEncoder
le=LabelEncoder()
df.sex=le.fit_transform(df.sex)
df.species=le.fit_transform(df.species)
df.island=le.fit_transform(df.island)
```

[43]:
```
X=df.drop('body_mass_g',axis=1)
y=df['body_mass_g']
```

[44]:
```
from sklearn.preprocessing import StandardScaler
sc=StandardScaler()
X_scaled= pd.DataFrame(sc.fit_transform(X),columns =X.columns)
X_scaled.head()
```

[44]:

	species	island	culmen_length_mm	culmen_depth_mm	flipper_length_mm	\
0	-1.029802	1.844076	-0.887622	0.787289	-1.420541	
1	-1.029802	1.844076	-0.814037	0.126114	-1.063485	
2	-1.029802	1.844076	-0.666866	0.431272	-0.420786	
3	-1.029802	1.844076	0.096581	0.075255	-0.277964	
4	-1.029802	1.844076	-1.329133	1.092447	-0.563608	

	sex
0	0.960230
1	-1.017729
2	-1.017729
3	0.960230
4	-1.017729

[45]:
```
from sklearn.model_selection import train_test_split
X_train,X_test,y_train,y_test=train_test_split(X_scaled,y,test_size=0.
↪2,random_state=42)
X_train.shape
```

[45]: (275, 6)

[46]:
```
X_test.shape
```

[46]: (69, 6)

[ ]:
```

```