ASSIGNMENT 3 (AIML)

DIVYANSH JAIN
21BCE2072

(I HAVE USED SCREENSHOT BECAUSE I WAS HAVING ISSUES WITH LINK PUSHING INTO REPO)

DivyanshJainAct3.ipynb

File Edit View Insert Runtime Tools Help All changes saved

+ Code + Text

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
```

Importing the dataset

```
[ ] df = pd.read_csv("penguins_size.csv")
    df.head()
```

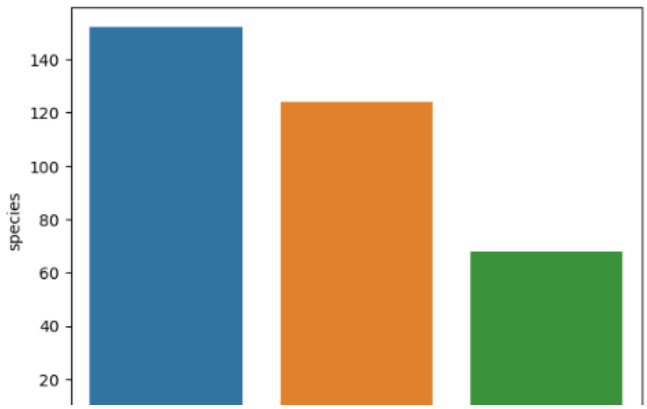
	species	island	culmen_length_mm	culmen_depth_mm	flipper_length_mm	body_mass_g	sex
0	Adelie	Torgersen	39.1	18.7	181.0	3750.0	MALE
1	Adelie	Torgersen	39.5	17.4	186.0	3800.0	FEMALE
2	Adelie	Torgersen	40.3	18.0	195.0	3250.0	FEMALE
3	Adelie	Torgersen	NaN	NaN	NaN	NaN	NaN
4	Adelie	Torgersen	36.7	19.3	193.0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

Perform Below Visualizations. • Univariate Analysis • Bi- Variate Analysis • Multi-Variate Analysis

Univariate Analysis

```
[ ] sns.barplot(x=df.species.value_counts().index,y=df.species.value_counts())
```

```
<Axes: ylabel='species'>
```



DivyanshJainAct3.ipynb

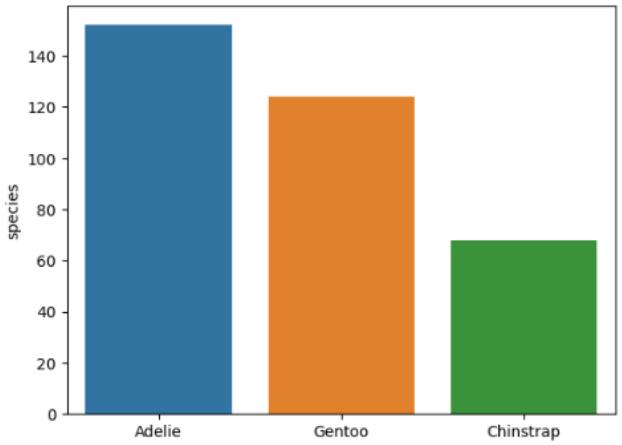
File Edit View Insert Runtime Tools Help All changes saved

+ Code + Text

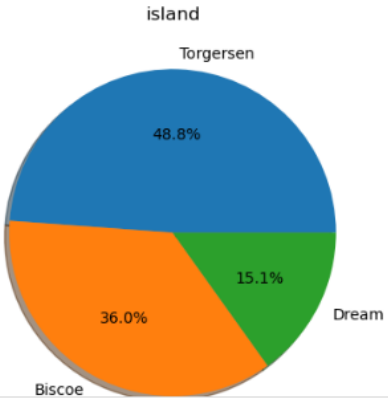
Univariate Analysis

```
sns.barplot(x=df.species.value_counts().index,y=df.species.value_counts())
```

```
<Axes: ylabel='species'>
```




```
[ ] plt.pie(df.island.value_counts(),[0,0,0],labels = ['Torgersen','Biscoe','Dream'],autopct='%1.1f%%',shadow=True)
    plt.title("island")
    plt.show()
```



DivyanshJainAct3.ipynb ☆

File Edit View Insert Runtime Tools Help All changes saved

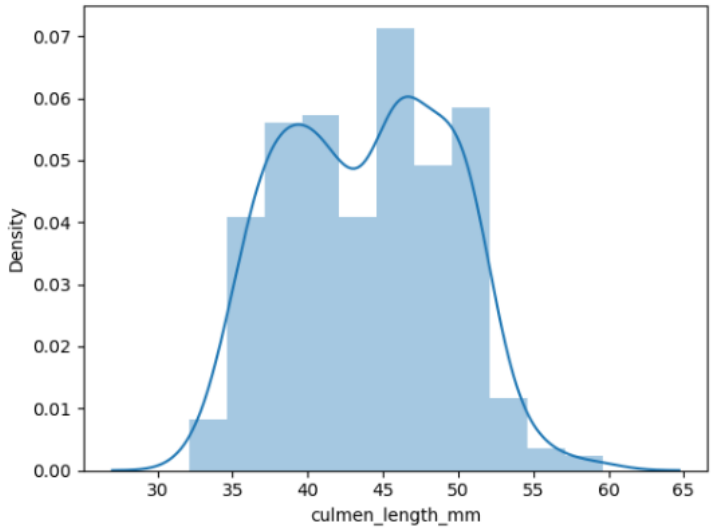
+ Code + Text

[ ]
```
`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_length_mm)
<Axes: xlabel='culmen_length_mm', ylabel='Density'>
```



Bivariate Analysis

```
sns.lineplot(x=df.culmen_length_mm,y=df.culmen_depth_mm)
```

```
<Axes: xlabel='culmen_length_mm', ylabel='culmen_depth_mm'>
```

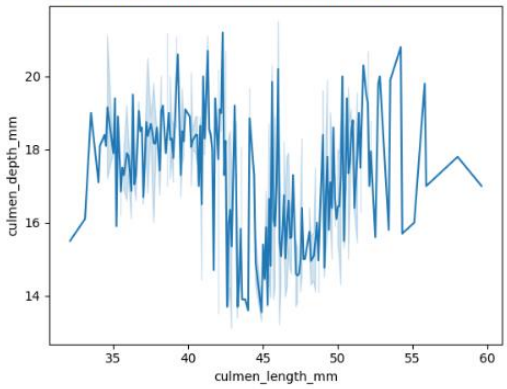
DivyanshJainAct3.ipynb ☆

File Edit View Insert Runtime Tools Help All changes saved

+ Code + Text

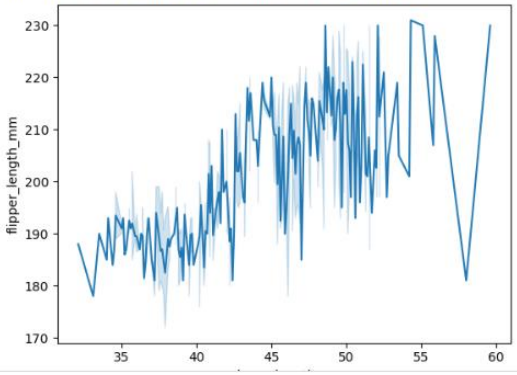
```
[ ] sns.lineplot(x=df.culmen_length_mm,y=df.culmen_depth_mm)
```

```
<Axes: xlabel='culmen_length_mm', ylabel='culmen_depth_mm'>
```




```
sns.lineplot(x=df.culmen_length_mm,y=df.flipper_length_mm)
```

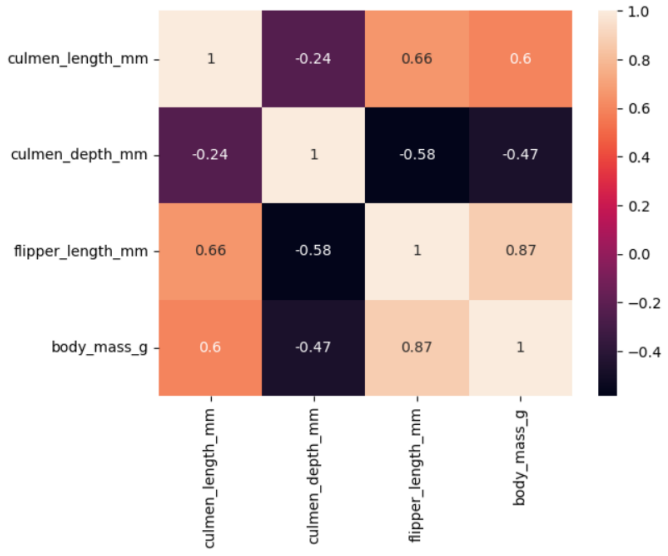
```
<Axes: xlabel='culmen_length_mm', ylabel='flipper_length_mm'>
```



Multivariate Analysis

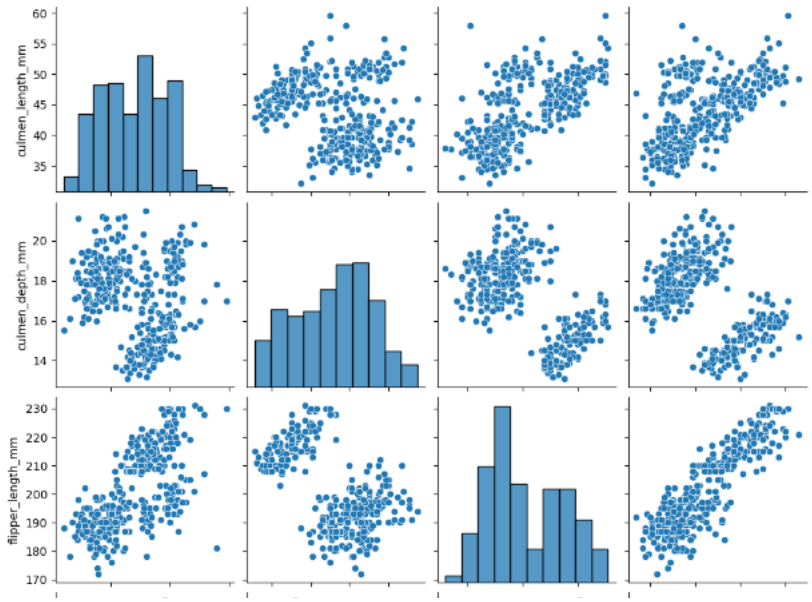
```
[ ] sns.heatmap(df.corr(),annot=True)
```

```
<Axes: >
```



```
[ ] sns.pairplot(df)
```

```
<seaborn.axisgrid.PairGrid at 0x2b9bd61e890>
```

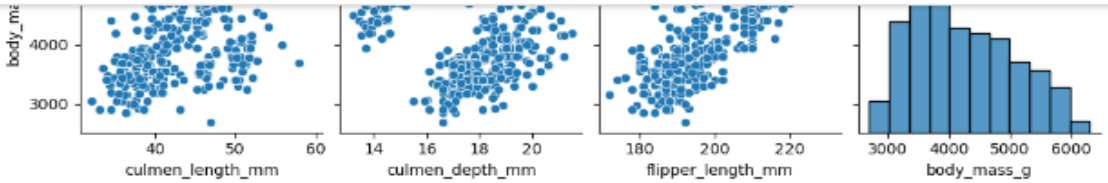


DivyanshJainAct3.ipynb ☆

File Edit View Insert Runtime Tools Help All changes saved

+ Code + Text

[ ]



Descriptive Statistics of the dataset

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

	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

Checking for missing values and dealing with them

```
df.isnull().sum()
```

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

```
[ ] 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']=df['sex'].replace('.','MALE')
```

```
[ ] df.sex.value_counts()
```

```
MALE      169
FEMALE    165
Name: sex, dtype: int64
```

DivyanshJainAct3.ipynb ☆

File Edit View Insert Runtime Tools Help All changes saved

+ Code + Text

```
[ ] df['sex'].fillna('MALE',inplace=True)
```

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

```
species              0
island               0
culmen_length_mm     0
culmen_depth_mm      0
flipper_length_mm    0
body_mass_g          0
sex                  0
dtype: int64
```

```
df.island.value_counts()
```

```
Biscoe       168
Dream        124
Torgersen     52
Name: island, dtype: int64
```

```
[ ] df.species.value_counts()
```

```
Adelie       152
Gentoo       124
Chinstrap     68
Name: species, dtype: int64
```

DivyanshJainAct3.ipynb ☆

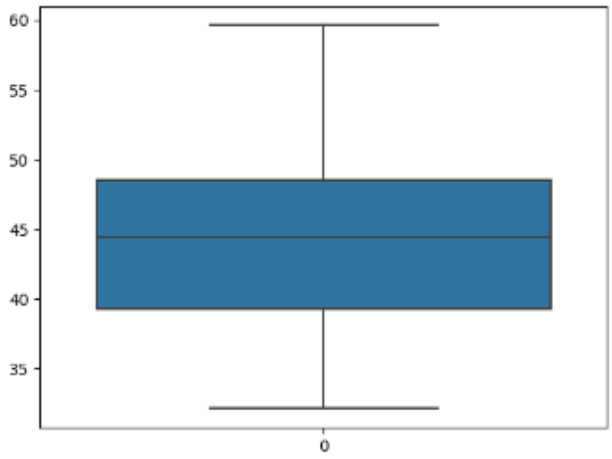
File Edit View Insert Runtime Tools Help All changes saved

+ Code + Text

Finding the outliners and replacing them

```
[ ] sns.boxplot(df.culmen_length_mm)
```

```
<Axes: >
```



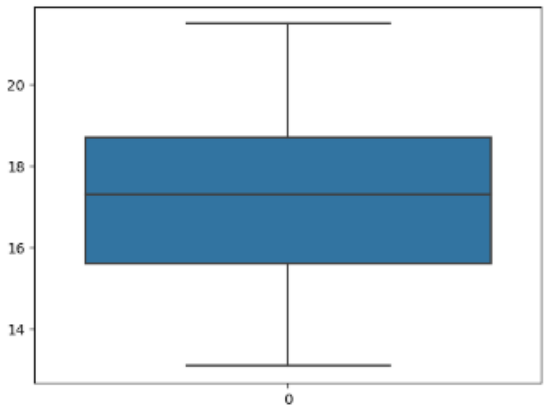
DivyanshJainAct3.ipynb ☆

File Edit View Insert Runtime Tools Help All changes saved

+ Code + Text

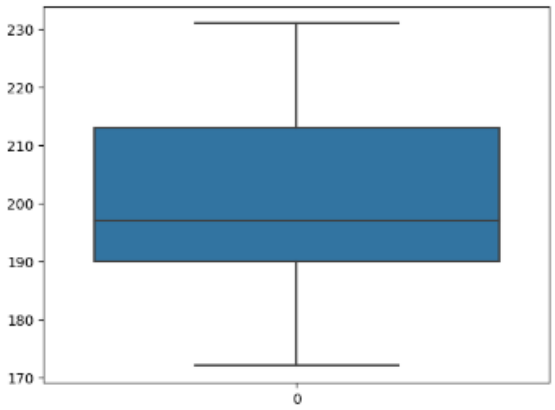
```
[ ] sns.boxplot(df.culmen_depth_mm)
```

```
<Axes: >
```




```
[ ] sns.boxplot(df.flipper_length_mm)
```

```
<Axes: >
```



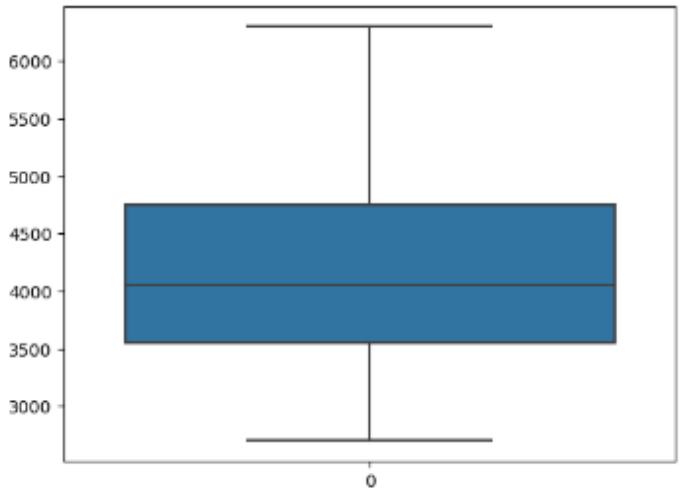
DivyanshJainAct3.ipynb ☆

File Edit View Insert Runtime Tools Help All changes saved

+ Code + Text


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

```
<Axes: >
```



DivyanshJainAct3.ipynb ☆

File Edit View Insert Runtime Tools Help All changes saved

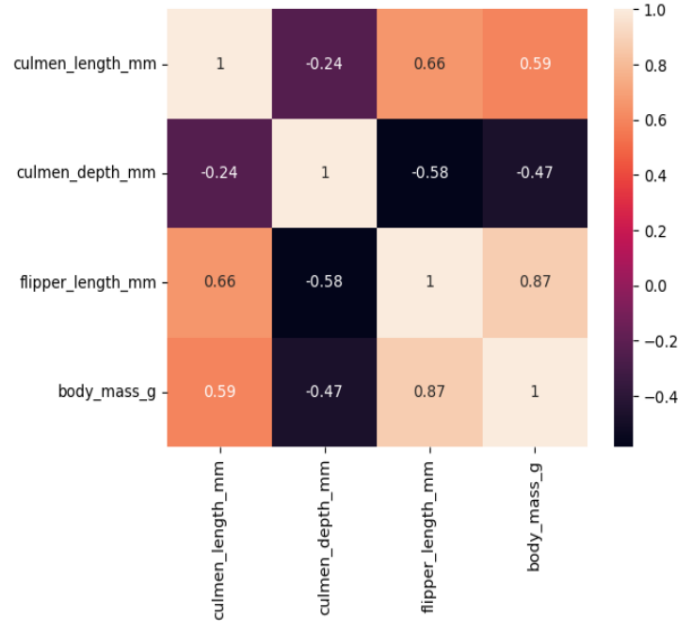
+ Code + Text

Checking the correlation of independent variables with the target

	culmen_length_mm	culmen_depth_mm	flipper_length_mm	body_mass_g
culmen_length_mm	1.000000	-0.235000	0.655858	0.594925
culmen_depth_mm	-0.235000	1.000000	-0.583832	-0.471942
flipper_length_mm	0.655858	-0.583832	1.000000	0.871221
body_mass_g	0.594925	-0.471942	0.871221	1.000000

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

<Axes: >



DivyanshJainAct3.ipynb ☆

File Edit View Insert Runtime Tools Help All changes saved

+ Code + Text

Checking for categorical columns and performing encoding

```
[ ] from sklearn.preprocessing import LabelEncoder
    le = LabelEncoder()
```

```
[ ] df.species=le.fit_transform(df.species)
    df.sex=le.fit_transform(df.sex)
    df.island=le.fit_transform(df.island)
```

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

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

DivyanshJainAct3.ipynb ☆

File Edit View Insert Runtime Tools Help All changes saved

+ Code + Text

Splitting the data into dependent and independent variables

```
[ ] X = df.drop(columns =['species'],axis=1)
    X.head()
```

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

```
y = df.species
df.head()
```

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

Scaling the data

```
[ ] from sklearn.preprocessing import StandardScaler
    scale = StandardScaler()
```

```
[ ] X_scaled = pd.DataFrame(scale.fit_transform(X),columns=X.columns)
    X_scaled.head()
```

	island	culmen_length_mm	culmen_depth_mm	flipper_length_mm	body_mass_g	sex
0	1.844076	-0.887622	0.787289	-1.420541	-0.564625	0.960098
1	1.844076	-0.814037	0.126114	-1.063485	-0.502010	-1.041561
2	1.844076	-0.666866	0.431272	-0.420786	-1.190773	-1.041561
3	1.844076	0.096581	0.075255	-0.277964	-0.188936	0.960098
4	1.844076	-1.329133	1.092447	-0.563608	-0.940314	-1.041561

Splitting the data into testing and training data

```
[ ] 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=0)
```

Checking the training and testing data shape

```
[ ] x_train.shape
```

```
(275, 6)
```

```
[ ] x_test.shape
```

```
(69, 6)
```

```
[ ] y_train.shape
```

```
(275,)
```

```
[ ] y_test.shape
```

```
(69,)
```

```
[ ]
```

DivyanshJainAct3.ipynb ☆

File Edit View Insert Runtime Tools Help All changes saved

+ Code + Text

Splitting the data into dependent and independent variables

```
[ ] X = df.drop(columns =['species'],axis=1)
    X.head()
```

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

```
y = df.species
df.head()
```

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

Scaling the data

```
[ ] from sklearn.preprocessing import StandardScaler
    scale = StandardScaler()
```

```
[ ] X_scaled = pd.DataFrame(scale.fit_transform(X),columns=X.columns)
    X_scaled.head()
```

	island	culmen_length_mm	culmen_depth_mm	flipper_length_mm	body_mass_g	sex
0	1.844076	-0.887622	0.787289	-1.420541	-0.564625	0.960098