NAME: VEDANT JADHAV
REG : 21BCE2796

Assignment 3

Penguin Classification Analysis

Problem Statement:

The Penguin Classification Analysis problem involves predicting the species of a penguin based on various physical characteristics. The dataset includes information about the body mass, culmen length, culmen depth, flipper length, and sex of different penguin species. The problem is typically approached as a classification problem, where the target variable is the penguin species, and the features are the physical characteristics of the penguins. Accurate classification of penguin species can also help researchers understand the effects of climate change and other environmental factors on penguin populations. The problem can also be useful for conservation efforts, as it can help identify and protect endangered penguin species.

Attribute Information:

- Species: penguin species (Chinstrap, Adélie, or Gentoo)
- Island: island name (Dream, Torgersen, or Biscoe) in Antarctica
- culmen_length_mm: culmen length (mm)
- culmen_depth_mm: culmen depth (mm)
- flipper_length_mm: flipper length (mm)
- body_mass_g: body mass (g)
- Sex: penguin sex

What is culmen?

The upper margin of the beak or bill is referred to as the culmen and the measurement is taken using calipers with one jaw at the tip of the upper mandible and the other at base of the skull or the first feathers depending on the standard chosen.

Perform the below Tasks to complete the Assignment:-

Clustering the data and performing classification algorithms

1. Download the dataset: Dataset
2. Load the dataset into the tool.
3. Perform Below Visualizations.

- Univariate Analysis

● Bi- Variate Analysis

● Multi-Variate Analysis

1. Perform descriptive statistics on the dataset.
2. Check for Missing values and deal with them.
3. Find the outliers and replace them outliers 7.Check the correlation of independent variables with the target
4. Check for Categorical columns and perform encoding.
5. Split the data into dependent and independent variables.
6. Scaling the data
7. Split the data into training and testing 12.check the training and testing data shape.

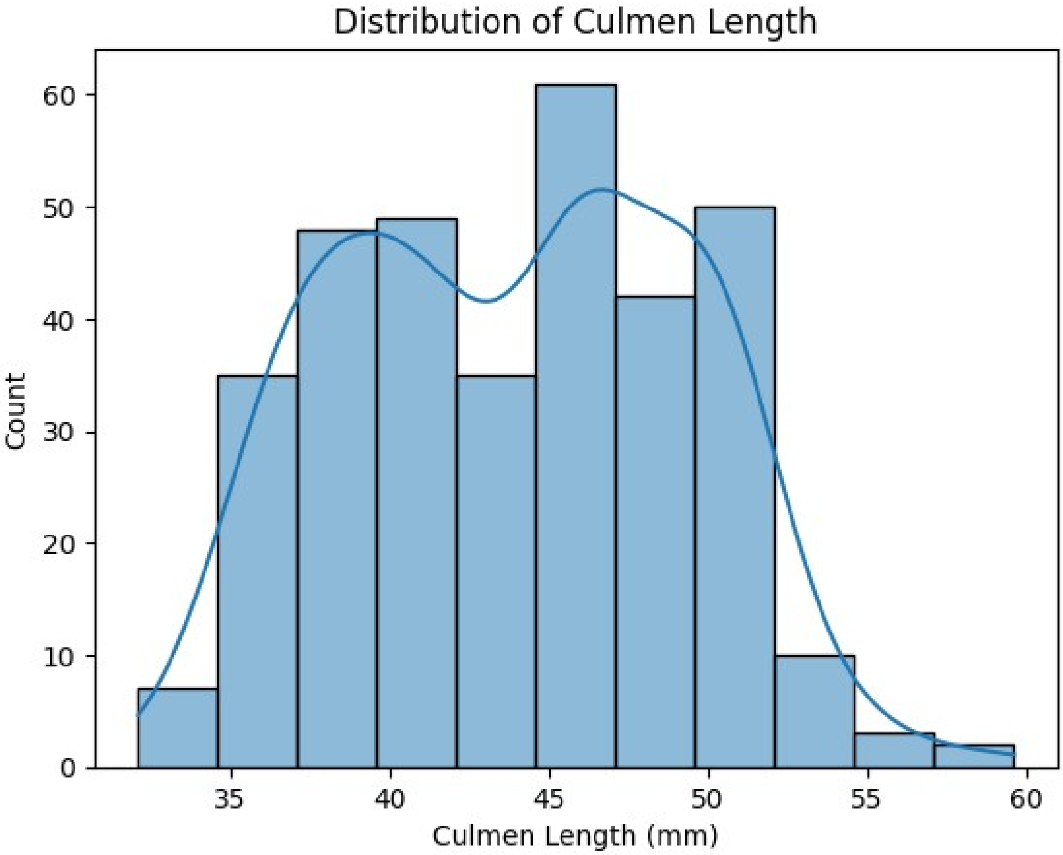
```
import pandas as pd

# Load the dataset
df = pd.read_csv('/content/penguins_size.csv')
df.head()

  species     island  culmen_length_mm  culmen_depth_mm
flipper_length_mm  \
0  Adelie  Torgersen              39.1             18.7
181.0
1  Adelie  Torgersen              39.5             17.4
186.0
2  Adelie  Torgersen              40.3             18.0
195.0
3  Adelie  Torgersen               NaN              NaN
NaN
4  Adelie  Torgersen              36.7             19.3
193.0

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

Perform Visualizations:

a. Univariate Analysis:

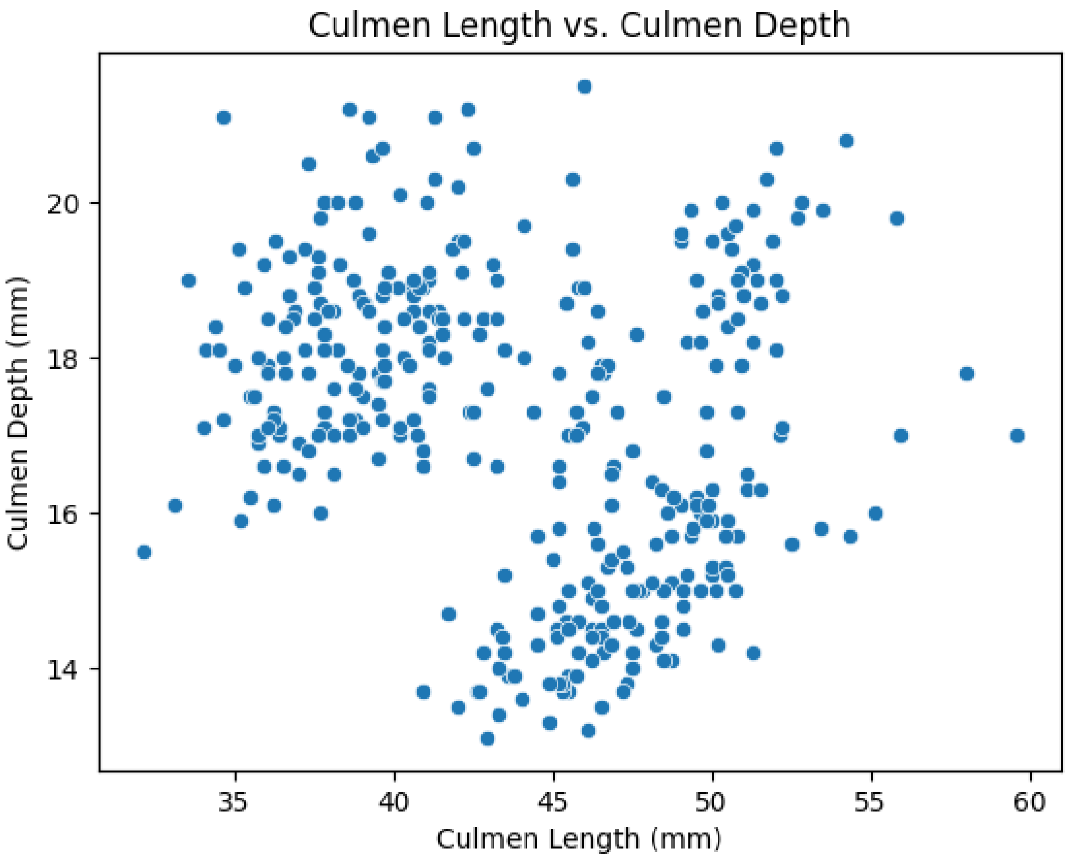
```
import matplotlib.pyplot as plt
import seaborn as sns
```

```
# Example for culmen_length_mm
sns.histplot(df['culmen_length_mm'], kde=True)
plt.xlabel('Culmen Length (mm)')
plt.ylabel('Count')
plt.title('Distribution of Culmen Length')
plt.show()
```



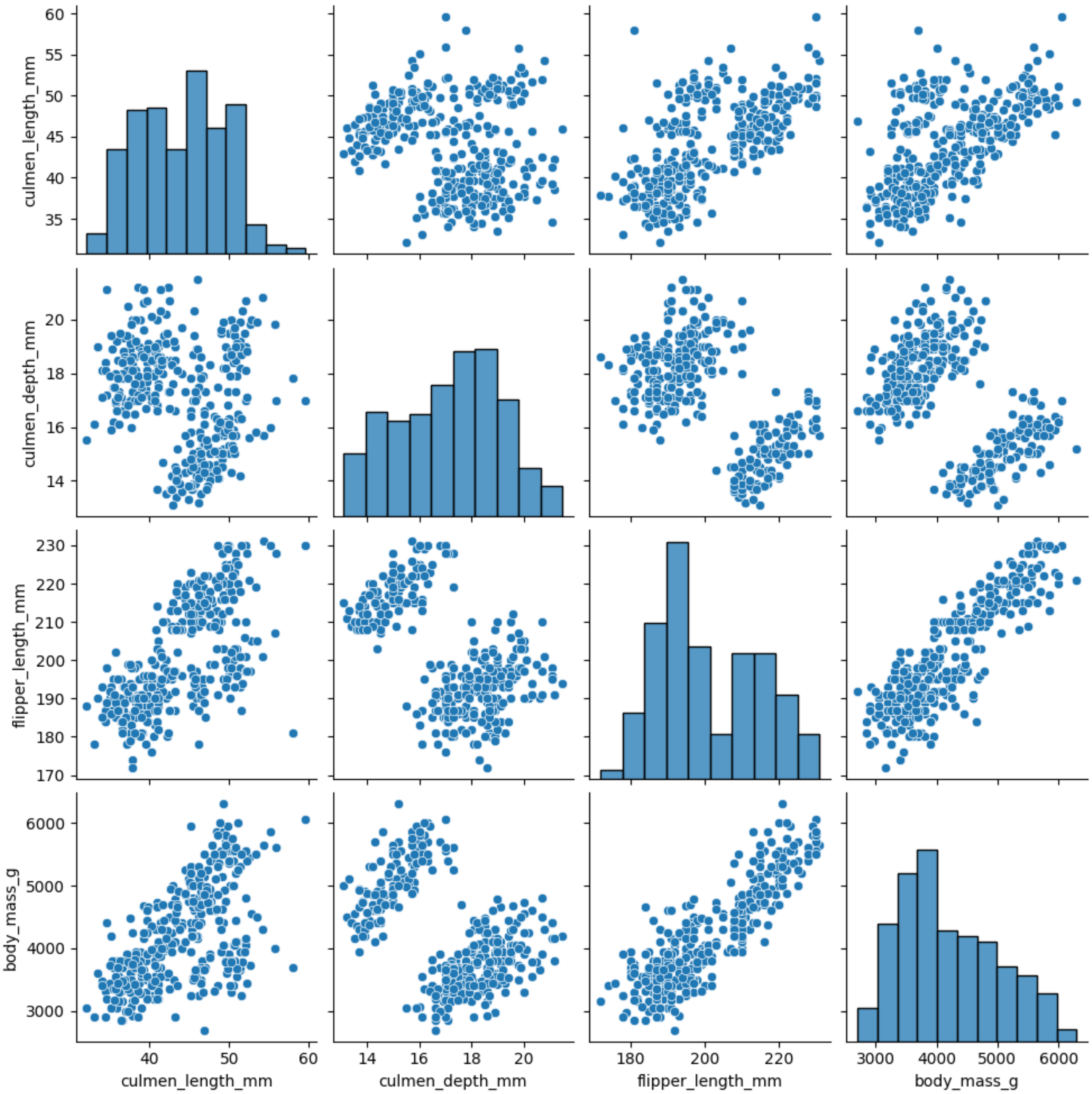
b. Bi-Variate Analysis:

```
# Example: Culmen Length vs. Culmen Depth
sns.scatterplot(x='culmen_length_mm', y='culmen_depth_mm', data=df)
plt.xlabel('Culmen Length (mm)')
plt.ylabel('Culmen Depth (mm)')
plt.title('Culmen Length vs. Culmen Depth')
plt.show()
```



c. Multi-Variate Analysis:

```
# Example: Pairplot
sns.pairplot(df)
plt.show()
```



Descriptive Statistics:

```
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
```

Check for Missing Values:

```
df.isnull().sum()
# Handle missing values if necessary
```

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

Find and Replace Outliers:

```
from scipy import stats

# Example: Z-score for 'culmen_length_mm'
z_scores = stats.zscore(df['culmen_length_mm'])
outliers = (z_scores > 3) | (z_scores < -3)

# Replace outliers with a suitable strategy (e.g., mean or median)
df.loc[outliers, 'culmen_length_mm'] = df['culmen_length_mm'].median()
```

Check Correlation:

```
correlation_matrix = df.corr()
correlation_matrix
```

```
<ipython-input-120-f471181e404f>: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_matrix = df.corr()
```

```
                   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
sex_FEMALE                -0.323210        -0.355333          -
0.244215
sex_MALE                   0.348378         0.368696
0.251283

                   body_mass_g  sex_FEMALE  sex_MALE
culmen_length_mm      0.595110   -0.323210  0.348378
culmen_depth_mm      -0.471916   -0.355333  0.368696
flipper_length_mm     0.871202   -0.244215  0.251283
body_mass_g           1.000000   -0.409315  0.422023
sex_FEMALE           -0.409315    1.000000 -0.938024
sex_MALE              0.422023   -0.938024  1.000000
```

Categorical Encoding:

```
df = pd.get_dummies(df, columns=['sex'], drop_first=True)

df.head()

  species     island  culmen_length_mm  culmen_depth_mm  
flipper_length_mm  \
0  Adelie  Torgersen              39.1             18.7
181.0
1  Adelie  Torgersen              39.5             17.4
186.0
2  Adelie  Torgersen              40.3             18.0
195.0
3  Adelie  Torgersen               NaN              NaN
NaN
4  Adelie  Torgersen              36.7             19.3
193.0

   body_mass_g  sex_FEMALE  sex_MALE
0       3750.0           0         1
1       3800.0           1         0
2       3250.0           1         0
3          NaN           0         0
4       3450.0           1         0
```

Split Data:

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

Scaling Data:

```
# Drop the 'Island' column before scaling
X = df.drop(['species', 'island'], axis=1)



# Scale the numeric features
scaler = StandardScaler()
X_scaled = scaler.fit_transform(X)
```

Split into Training and Testing 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=42)
```

Check Data Shape:

```
print("X_train shape:", X_train.shape)
print("X_test shape:", X_test.shape)

X_train shape: (275, 6)
X_test shape: (69, 6)
```