**SVKM’s NMIMS**

**Mukesh Patel School of Technology Management & Engineering**

Program: B.Tech\MBA.Tech

**Course: Machine Learning**

**Experiment No.02**

PART A

(PART A : TO BE REFFERED BY STUDENTS)

**A.1 Aim: To understand and implement data visualization techniques**

**A.2 Prerequisite:**

Python Programming, Pandas library, Numpy Library, MatplotLib, Seaborn Library

**A.3 Outcome:**

**After successful completion of this experiment students will be able to:**

* 1. Read different types of data files (csv, excel, text file etc.)
  2. Understand usage of different types of Python libraries for plotting data
  3. Plotting of data using different types of plots

**A.4 Theory:**

Data visualization is a form of visual communication. It involves the creation and study of the visual representation of data. It translates the data to a more natural form for the human mind to comprehend and pick out patterns or points of interest.

**Matplotlib and seaborn** are among the common libraries for visualizing data in Python.

Matplotlib is a python library used extensively for the visualization of data. While Seaborn is a python library based on matplotlib. Seaborn provides a high-level interface for drawing attractive and informative statistical graphics.

Most common types of plots used in data visualization:

* Scatter plot
* Pair plot
* Box plot
* Violin plot
* Distribution plot
* Joint plot
* Bar chart
* Line plot

**Scatter Plot:**

It is one of the most commonly used plots for simple data visualization. It gives us a representation of where each point in the entire dataset are present with respect to any 2 or 3 features (or columns)

**Pair plot:**

Pair plot will help us create us a (n x n) figure where the diagonal plots will be histogram plot of the feature corresponding to that row and rest of the plots are the combination of feature from each row in y axis and feature from each column in x axis.

**Box plot:**

A box plot (or box-and-whisker plot) shows the distribution of quantitative data in a way that facilitates comparisons between variables or across levels of a categorical variable. The box shows the quartiles of the dataset while the whiskers extend to show the rest of the distribution.

**Violin Plot**

The violin plots can be inferred as a combination of Box plot at the middle and distribution plots (Kernel Density Estimation) on both side of the data. This can give us the details of distribution like whether the distribution is multimodal, Skewness etc.

**Joint Plot:**

Join plots can do both univariate as well as bivariate analysis. The main plot will give us a bivariate analysis, whereas on the top and right side we will get univariate plots of both the variables that were considered. It makes our job easy by getting both scatter plots for bivariate and Distribution plot for univariate, both in a single plot.

**Tasks:**

**Task 1:**

1. Plot a scatter plot for X (random numbers from 0 to 10) and Y (random numbers from 11 to 21). Save the scatter plot
2. Create a subplot with different markers and different line colors
3. Plot a bar plot for below X and Y values.

X= [2,8,10] Y= [11,16,9]

1. Plot a box plot for the values given below and state your inference.

A= [3,4,5,7,9,8,12,13,7,8,19,90,12,15]

Task 2:

1. Read “seeds.csv” file into data frame. ( Dataset description is found at <https://archive.ics.uci.edu/ml/datasets/seeds>)
2. Explore the dataset by using head and describe.
3. Find the number of samples per type.
4. Plot a scatter plot for Kernel Width vs Length. Write your inference.
5. Different type should have different colours in above scatter plot
6. Plot a Jointplot to understand relation between Perimeter and Compactness. Write your inference.
7. Plot a Box plot to understand correlation between compactness and type.
8. Plot a pair plot to understand all characteristics with type being the main parameter. State your inference
9. Plot a Violin plot to understand correlation between compactness and type. State your inference
10. Plot a Kernel Density Estimation plots to understand correlation between compactness and type. State your inference
11. Plot a pair plot to understand all characteristics with type being the main parameter. (the main parameter, with KDE instead of histogram in diagonal subplots)
12. An Andrews curve to display separability of data according to Type.

PART B

(PART B : TO BE COMPLETED BY STUDENTS)

***(Students must submit the soft copy as per following segments within two hours of the practical.)***

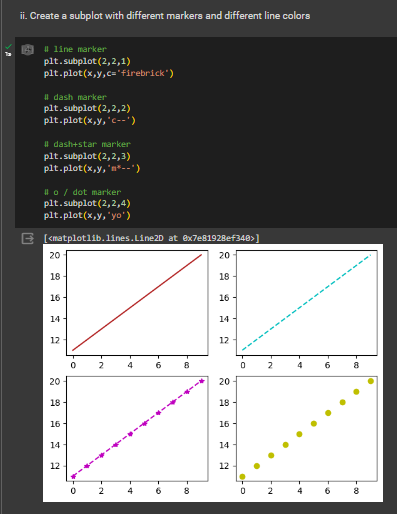
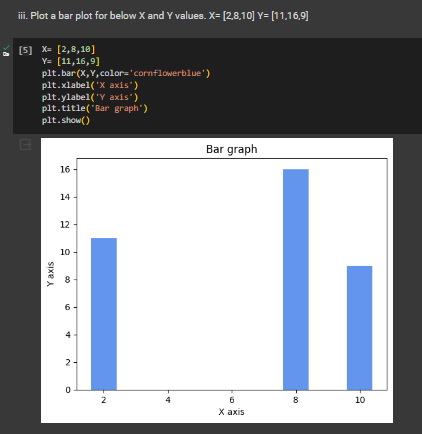
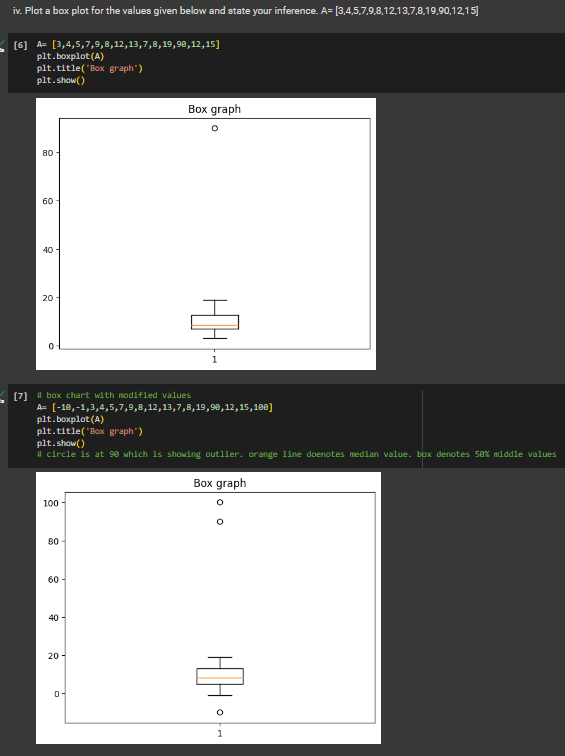
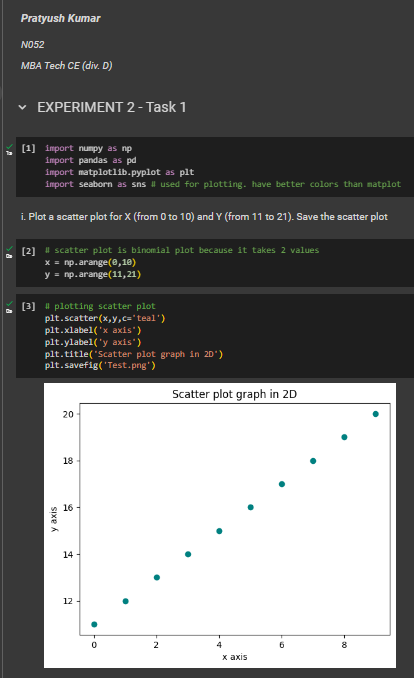
|  |  |
| --- | --- |
| Roll No. N052 | Name: Pratyush Kumar |
| Class : MBA Tech CE (div. D) | Batch : B2 |
| Date of Experiment: 21-12-2023 | Date of Submission: 06-01-2023 |
| Grade : |  |

**B.1 Task 1**

**Colab link:** <https://colab.research.google.com/drive/19zag3G0H2e7cjLhyx6hQpZOGahwW8JNw?usp=sharing>

* **Source Code**

*"""  
 \* This file contains code snippets for using different types of Python libraries for plotting data  
 \* ML-E2-Task1  
 \*  
 \* Original file is located at: https://colab.research.google.com/drive/19zag3G0H2e7cjLhyx6hQpZOGahwW8JNw  
 \* @author Pratyush Kumar (github.com/pratyushgta)  
"""*import numpy as np  
import pandas as pd  
import matplotlib.pyplot as plt  
import seaborn as sns # used for plotting. have better colors than matplot  
  
"""i. Plot a scatter plot for X (from 0 to 10) and Y (from 11 to 21). Save the scatter plot"""  
  
# scatter plot is binomial plot because it takes 2 values  
x = np.arange(0,10)  
y = np.arange(11,21)  
  
# plotting scatter plot  
plt.scatter(x,y,c='teal')  
plt.xlabel('x axis')  
plt.ylabel('y axis')  
plt.title('Scatter plot graph in 2D')  
plt.savefig('Test.png')  
  
"""ii. Create a subplot with different markers and different line colors"""  
  
# line marker  
plt.subplot(2,2,1)  
plt.plot(x,y,c='firebrick')  
  
# dash marker  
plt.subplot(2,2,2)  
plt.plot(x,y,'c--')  
  
# dash+star marker  
plt.subplot(2,2,3)  
plt.plot(x,y,'m\*--')  
  
# o / dot marker  
plt.subplot(2,2,4)  
plt.plot(x,y,'yo')  
  
"""iii. Plot a bar plot for below X and Y values.  
X= [2,8,10] Y= [11,16,9]  
"""  
  
X= [2,8,10]  
Y= [11,16,9]  
plt.bar(X,Y,color='cornflowerblue')  
plt.xlabel('X axis')  
plt.ylabel('Y axis')  
plt.title('Bar graph')  
plt.show()  
  
"""iv. Plot a box plot for the values given below and state your inference.  
A= [3,4,5,7,9,8,12,13,7,8,19,90,12,15]  
"""  
  
A= [3,4,5,7,9,8,12,13,7,8,19,90,12,15]  
plt.boxplot(A)  
plt.title('Box graph')  
plt.show()  
  
# box chart with modified values  
A= [-10,-1,3,4,5,7,9,8,12,13,7,8,19,90,12,15,100]  
plt.boxplot(A)  
plt.title('Box graph')  
plt.show()  
# circle is at 90 which is showing outlier. orange line doenotes median value. box denotes 50% middle values

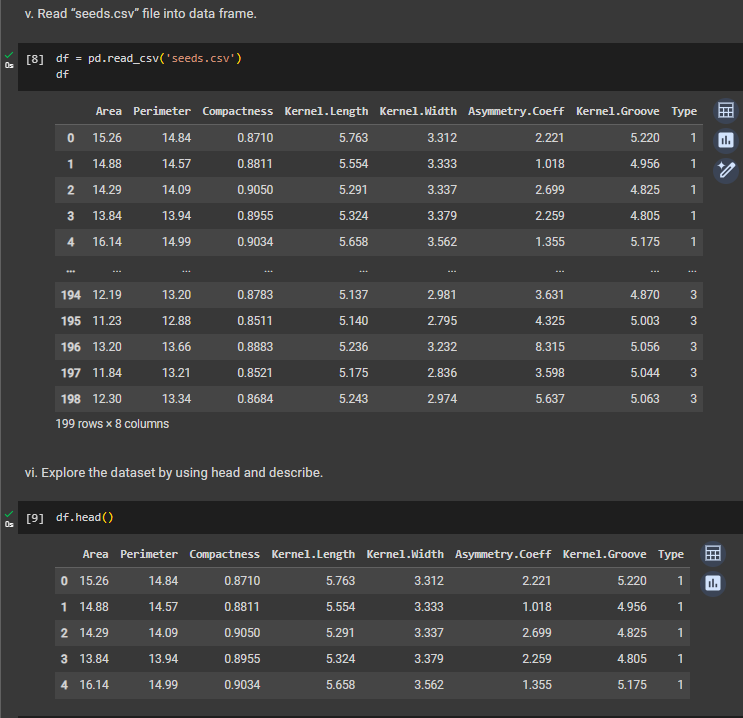
* **Input/ Output**

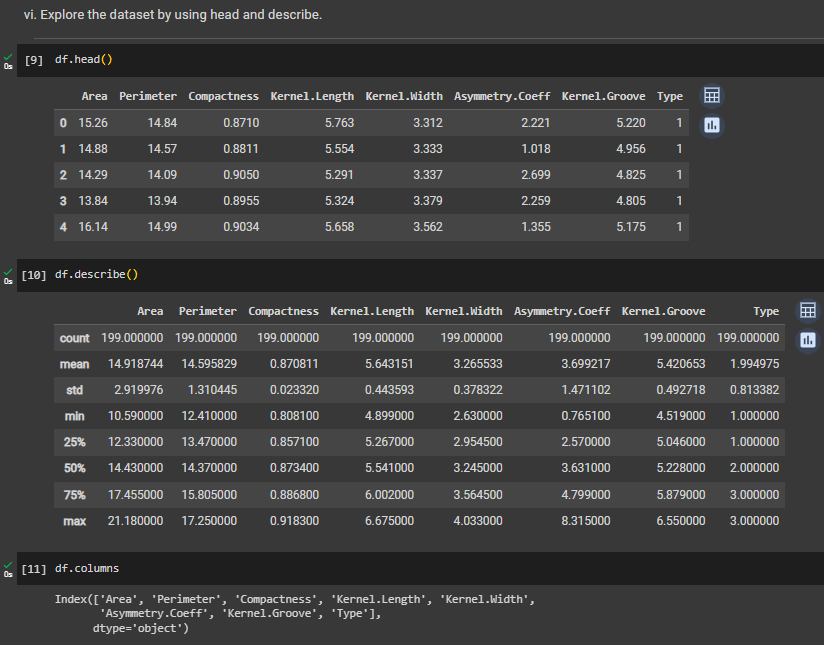
**B.2 Task 2**

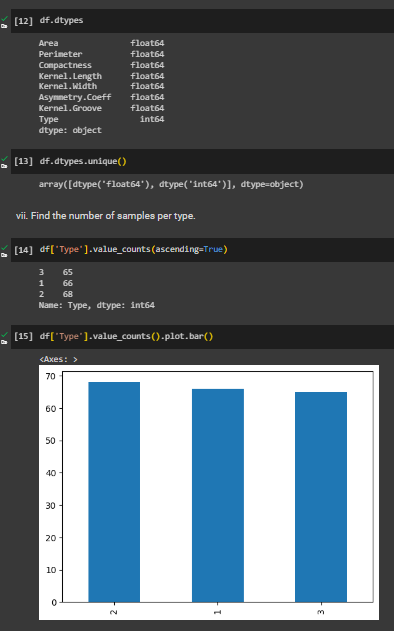
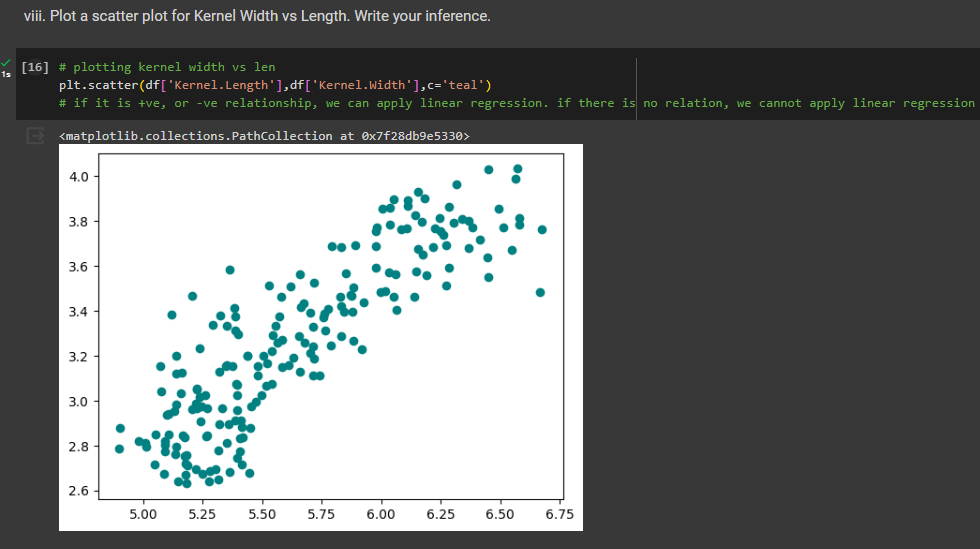
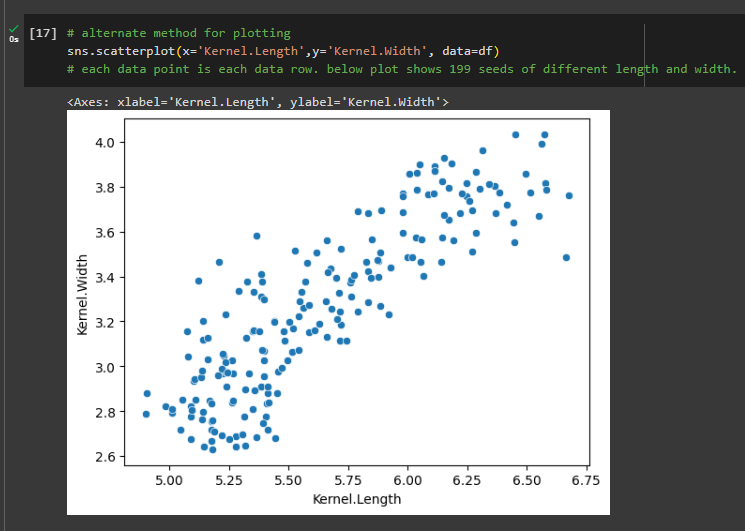
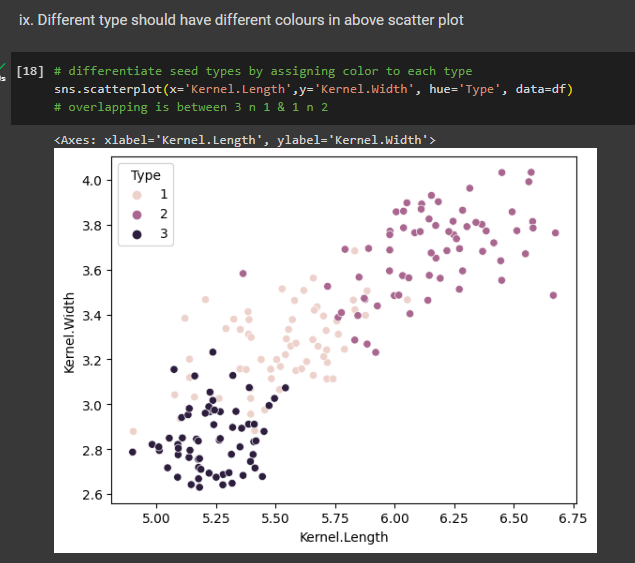
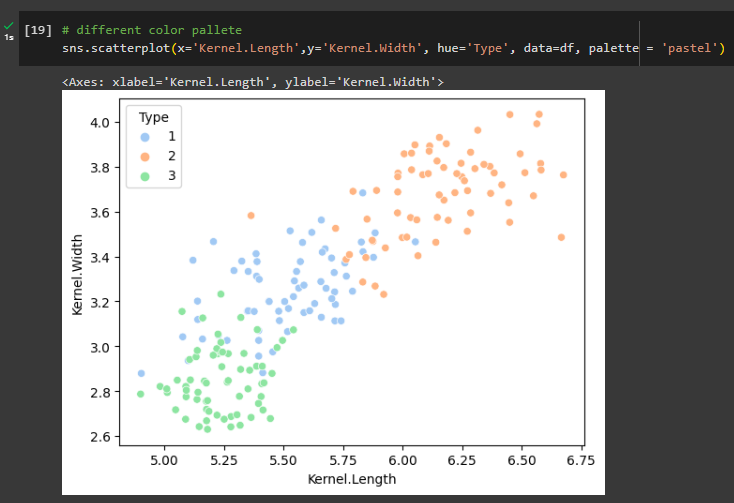
**Colab link:** <https://colab.research.google.com/drive/19zag3G0H2e7cjLhyx6hQpZOGahwW8JNw?usp=sharing>

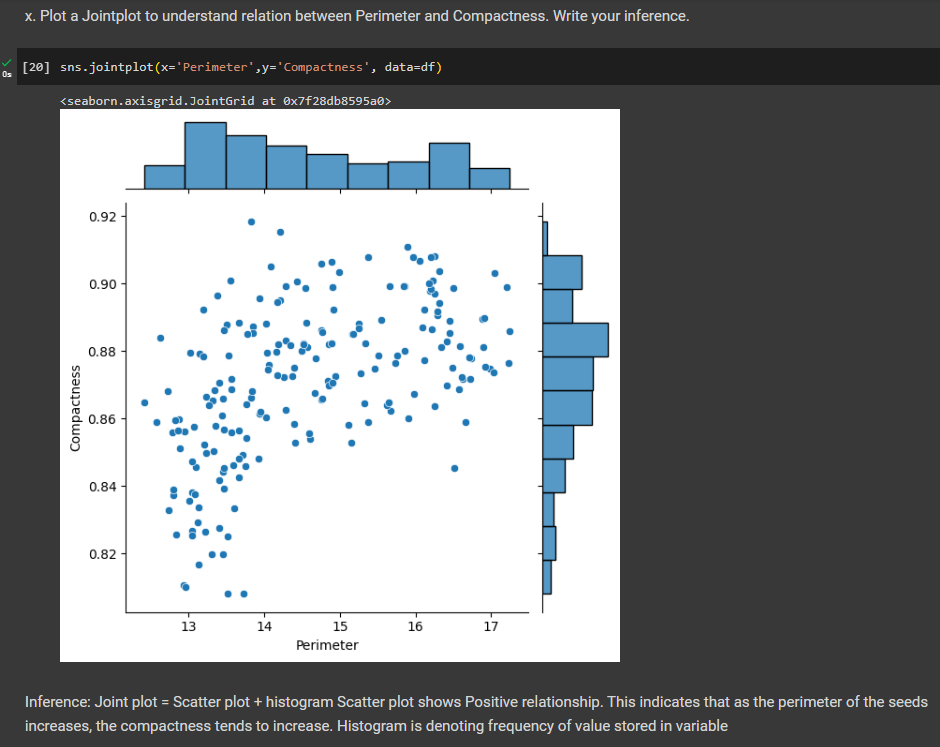
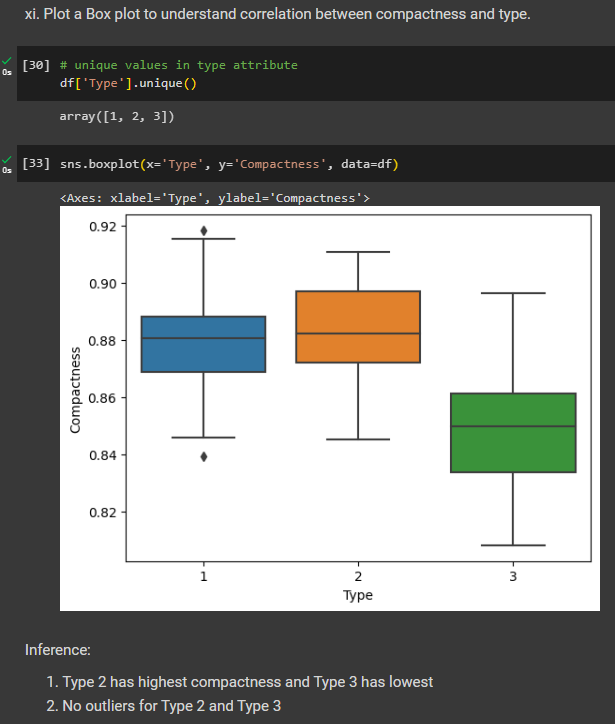
* **Source Code**

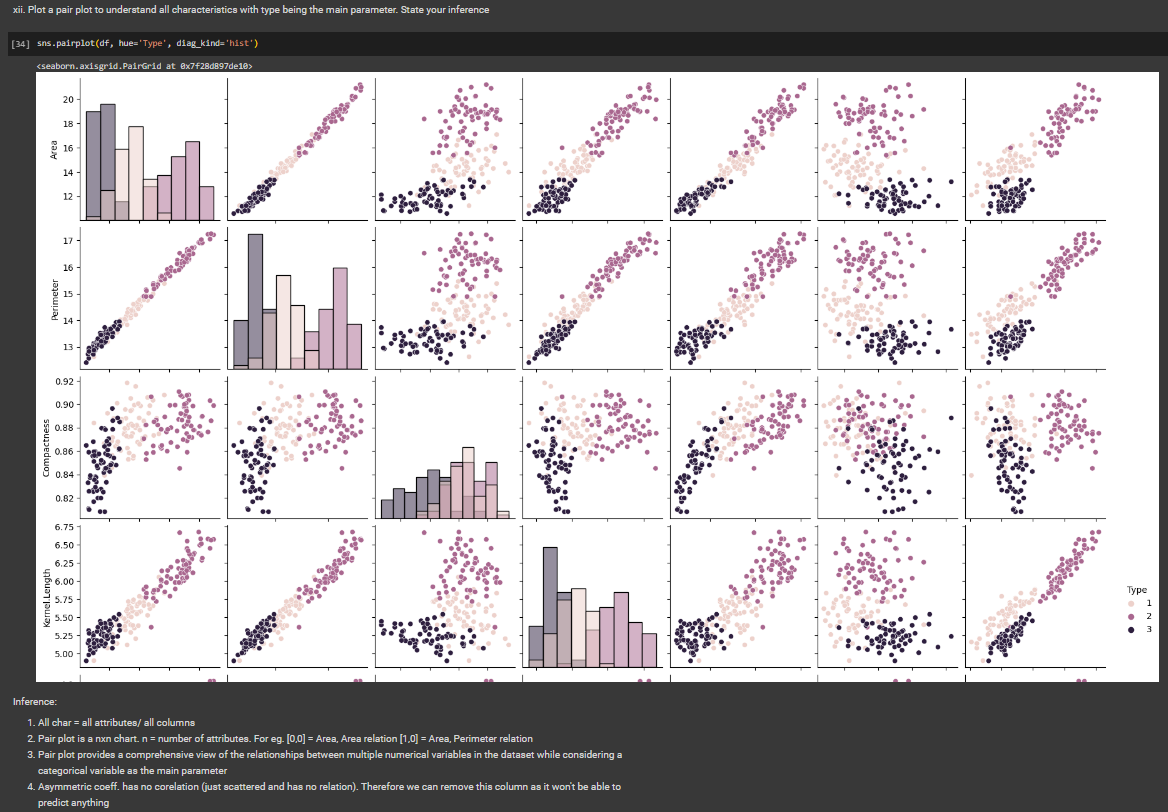
*"""  
 \* This file contains code snippets for performing exploratory data analysis on seeds dataset  
 \* ML-E2-Task2  
 \*  
 \* Original file is located at: https://colab.research.google.com/drive/19zag3G0H2e7cjLhyx6hQpZOGahwW8JNw  
 \* @author Pratyush Kumar (github.com/pratyushgta)  
"""*import numpy as np  
import pandas as pd  
import matplotlib.pyplot as plt  
import seaborn as sns # used for plotting. have better colors than matplot  
  
"""## EXPERIMENT 2 - Task 2  
v. Read “seeds.csv” file into data frame.  
"""  
  
df = pd.read\_csv('seeds.csv')  
df  
  
"""vi. Explore the dataset by using head and describe."""  
  
df.head()  
  
df.describe()  
  
df.columns  
  
df.dtypes  
  
df.dtypes.unique()  
  
"""vii. Find the number of samples per type."""  
  
df['Type'].value\_counts(ascending=True)  
  
df['Type'].value\_counts().plot.bar()  
  
"""viii. Plot a scatter plot for Kernel Width vs Length. Write your inference."""  
  
# plotting kernel width vs len  
plt.scatter(df['Kernel.Length'],df['Kernel.Width'],c='teal')  
# if it is +ve, or -ve relationship, we can apply linear regression. if there is no relation, we cannot apply linear regression  
  
# alternate method for plotting  
sns.scatterplot(x='Kernel.Length',y='Kernel.Width', data=df)  
# each data point is each data row. below plot shows 199 seeds of different length and width.  
  
"""ix. Different type should have different colours in above scatter plot"""  
  
# differentiate seed types by assigning color to each type  
sns.scatterplot(x='Kernel.Length',y='Kernel.Width', hue='Type', data=df)  
# overlapping is between 3 n 1 & 1 n 2  
  
# different color pallete  
sns.scatterplot(x='Kernel.Length',y='Kernel.Width', hue='Type', data=df, palette = 'pastel')  
  
"""x. Plot a Jointplot to understand relation between Perimeter and Compactness. Write your inference."""  
  
sns.jointplot(x='Perimeter',y='Compactness', data=df)  
  
"""Inference:  
Joint plot = Scatter plot + histogram  
Scatter plot shows Positive relationship. This indicates that as the perimeter of the seeds increases, the compactness tends to increase.  
Histogram is denoting frequency of value stored in variable  
  
xi. Plot a Box plot to understand correlation between compactness and type.  
"""  
  
# unique values in type attribute  
df['Type'].unique()  
  
sns.boxplot(x='Type', y='Compactness', data=df)  
  
"""Inference:  
1. Type 2 has highest compactness and Type 3 has lowest  
2. No outliers for Type 2 and Type 3  
  
xii. Plot a pair plot to understand all characteristics with type being the main parameter. State your inference  
"""  
  
sns.pairplot(df, hue='Type', diag\_kind='hist')  
  
"""Inference:  
1. All char = all attributes/ all columns  
2. Pair plot is a nxn chart. n = number of attributes. For eg. [0,0] = Area, Area relation [1,0] = Area, Perimeter relation  
3. Pair plot provides a comprehensive view of the relationships between multiple numerical variables in the dataset while considering a categorical variable as the main parameter  
4. Asymmetric coeff. has no corelation (just scattered and has no relation). Therefore we can remove this column as it won't be able to predict anything  
  
xiii. Plot a Violin plot to understand correlation between compactness and type. State your inference  
"""  
  
# Center = box plot. White dot in center = median (Q2)  
# Curved shape is nothing but a histogram.  
# Curve is made by first forming a histogram, gets a smooth curve, rotates the histogram, shifts bars to get central axis. Based on central axis, divides the histograph  
sns.violinplot(x='Type', y='Compactness', data=df)  
  
"""Inference:  
- Wider sections in the plot indicate a higher frequency of data points, while narrower sections represent lower frequency.  
- Differences in the shapes (spread of the violins) indicates varying compactness characteristics among different seed types.  
  
xiv. Plot a Kernel Density Estimation plots to understand correlation between compactness and type. State your inference  
"""  
  
# KDE plots provide a smooth estimation of the probability density function of a continuous variable (Compactness)  
# across different categories (Type)  
  
#sns.displot(df, x='Compactness', hue='Type', kind='kde', fill=True)  
sns.FacetGrid(df,hue='Type',height=5).map(sns.kdeplot,'Compactness').add\_legend()  
  
"""xv. Plot a pair plot to understand all characteristics with type being the main parameter. (the main parameter, with KDE instead of histogram in diagonal subplots)"""  
  
# Creating pair plot with KDE on diagonal  
# by default it takes kde  
sns.pairplot(df, hue='Type', height=3)  
  
"""xvi. An Andrews curve to display separability of data according to Type."""  
  
from pandas.plotting import andrews\_curves  
# if attributes are overlapping too much, it will be difficult to classify. So, we use andrews curve to check extent of overlapping  
# Andrews curves is used to visualize the structure of data by transforming each row of a dataset into a curve  
andrews\_curves(df, class\_column='Type', colormap='viridis')

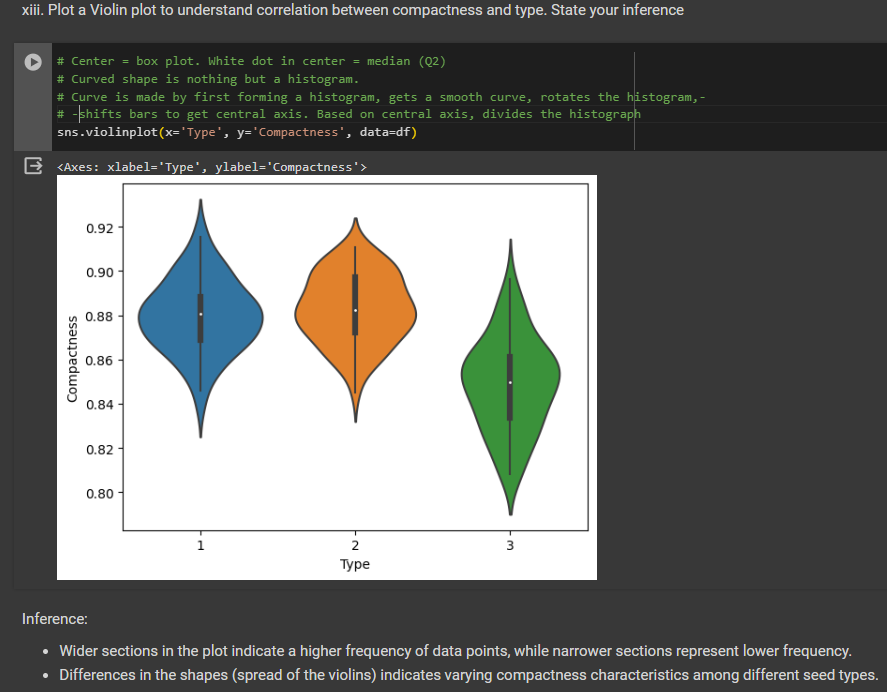
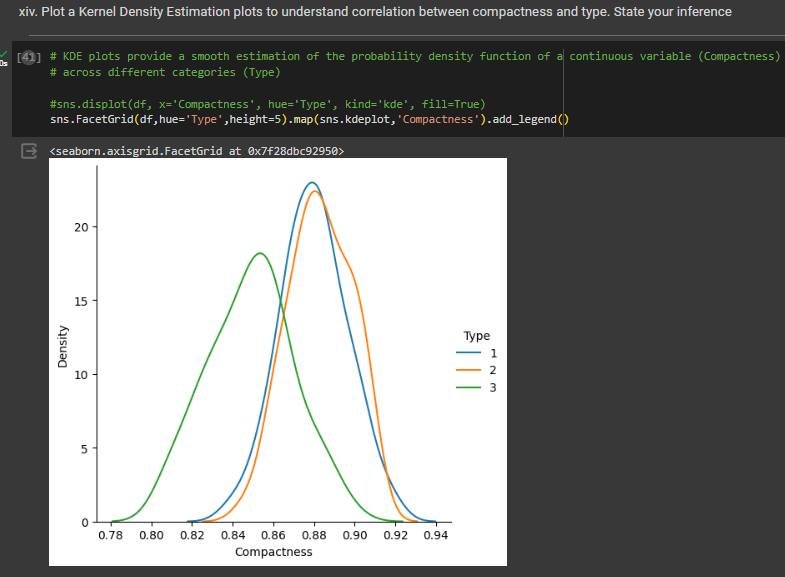
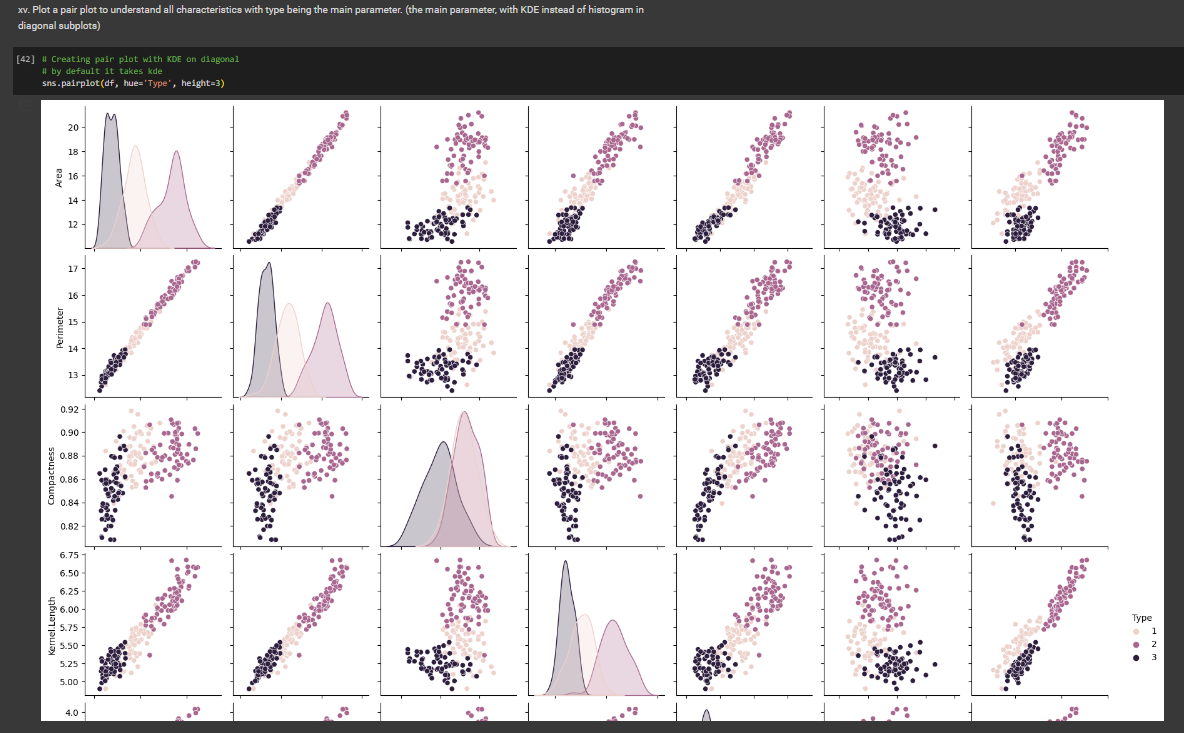
* **Input/ Output**

****

****

****

****

****

**B.3 Conclusion:**

*(Students must write the conclusion in their own words.)*

Implemented data visualization techniques by using two Python libraries- matplotlib and seaborn, for plotting data different types of plots. Explored the ‘seeds’ dataset and performed various visualizations and analyses to understand its various attributes, relationships between variables, and the distinctive characteristics associated with each seed type.