Name: Durva Kadam Roll No: 11

Subject: AI-ML in Healthcare Lab

Experiment – 2: To implement Data Visualization/Data Exploratory Analysis using Matplotlib and Seaborn on Healthcare database

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

# Numpy array
a = np.array([[1, 2, 3], [4, 5, 6], [7, 8, 9]])
print(a)

# Sinusoidal plot
x = np.arange(0.0, 2.0, 0.01)
y = np.sin(2 * np.pi * x)
plt.figure()
plt.plot(x, y)
plt.title("Sinusoidal Plot")
plt.grid(True)
plt.show()

# Barplot
plt.figure()
x = [1, 2, 3, 4, 5, 6, 7]
y = [23, 43, 65, 32, 87, 45, 90]
plt.bar(x, y, color='orange')
plt.title('Average Score of Students')
plt.xlabel("Students")
plt.ylabel("Avg. Score")
plt.show()

# Barplot with error bars (stacked)
plt.figure()
```

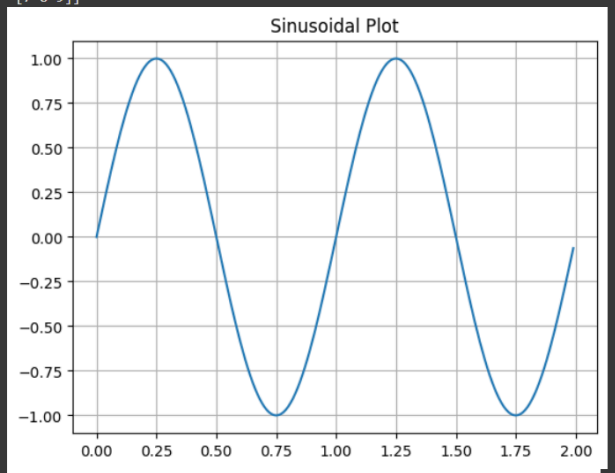
```
women = (25, 32, 34, 20, 25)
mstd = (2, 2, 2, 2, 2)
i = np.arange(N)
p1 = plt.bar(i, men, width=0.35, yerr=mstd)
p2 = plt.bar(i, women, width=0.35, bottom=men, yerr=mstd)
plt.xticks(i, ('G1', 'G2', 'G3', 'G4', 'G5'))
plt.yticks(np.arange(0, 90, 10))
plt.legend((p1[0], p2[0]), ('Men', 'Women'))
plt.title("Barplot with Error Bars")
plt.show()

# Histogram
plt.figure()
m = 100
std = 15
x = m + std * np.random.randn(480)
y = 50
plt.hist(x, y, color='blue', density=False)
plt.grid(color='gray', linestyle='--', linewidth=2)
plt.title("Histogram")
plt.show()

# Pie Chart
plt.figure()
labels = 'cricket', 'Hockey', 'Tennis', 'Football'
v = [40, 20, 10, 30]
e = (0, 0, 0.1, 0)
plt.pie(v, explode=e, labels=labels, autopct='%1.3f%%', shadow=True, startangle=0)
plt.title("Average Score of Students")
plt.show()
```

```
# Sin(x - shift) with different line styles
x = np.linspace(0, 10, 1000)
plt.figure()
plt.plot(x, np.sin(x), linestyle='solid')
plt.plot(x, np.sin(x - 1), linestyle='dashed')
plt.plot(x, np.sin(x - 2), linestyle='dashdot')
plt.plot(x, np.sin(x - 3), linestyle='dotted')
plt.plot(x, np.sin(x - 4), linestyle=':')
plt.title("Sine Waves with Line Styles")
plt.show()

# Scatterplot of sine wave
x = np.linspace(0, 10, 50)
y = np.sin(x)
plt.figure()
plt.scatter(x, y)
plt.title("Sine Wave - Scatter Plot")
plt.show()

# Line with markers
x = np.linspace(0, 10, 50)
plt.figure()
plt.plot(x, np.sin(x), '-o')
plt.title("Sine Wave with Markers")
plt.show()

# Seaborn plots
data = sns.load_dataset("iris")

plt.figure()
```

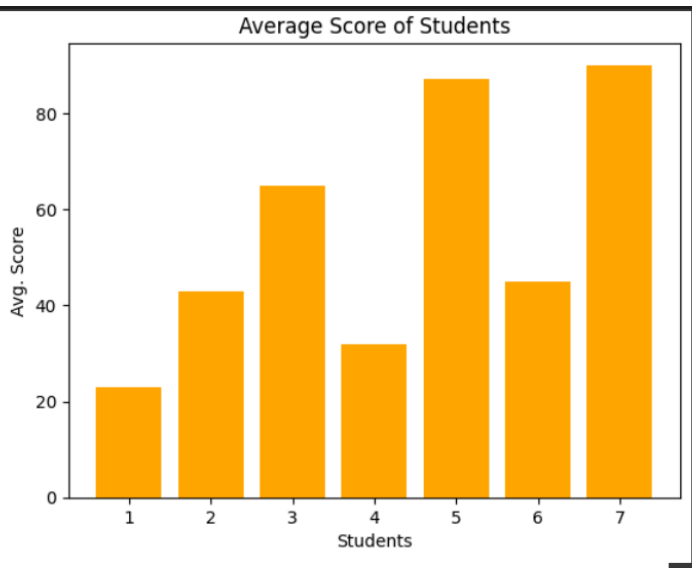
```
plt.figure()
sns.lineplot(x="sepal_length", y="sepal_width", data=data)
plt.title("Seaborn Lineplot")
plt.show()

plt.figure()
sns.scatterplot(x="sepal_length", y="sepal_width", data=data)
plt.title("Seaborn Scatterplot")
plt.show()

plt.figure()
sns.stripplot(x="species", y="sepal_length", data=data)
plt.title("Seaborn Stripplot")
plt.show()

plt.figure()
sns.violinplot(x="species", y="sepal_length", data=data)
plt.title("Seaborn Violinplot")
plt.show()

plt.figure()
sns.swarmplot(x="species", y="sepal_length", data=data)
plt.title("Seaborn Swarmplot")
plt.show()
```

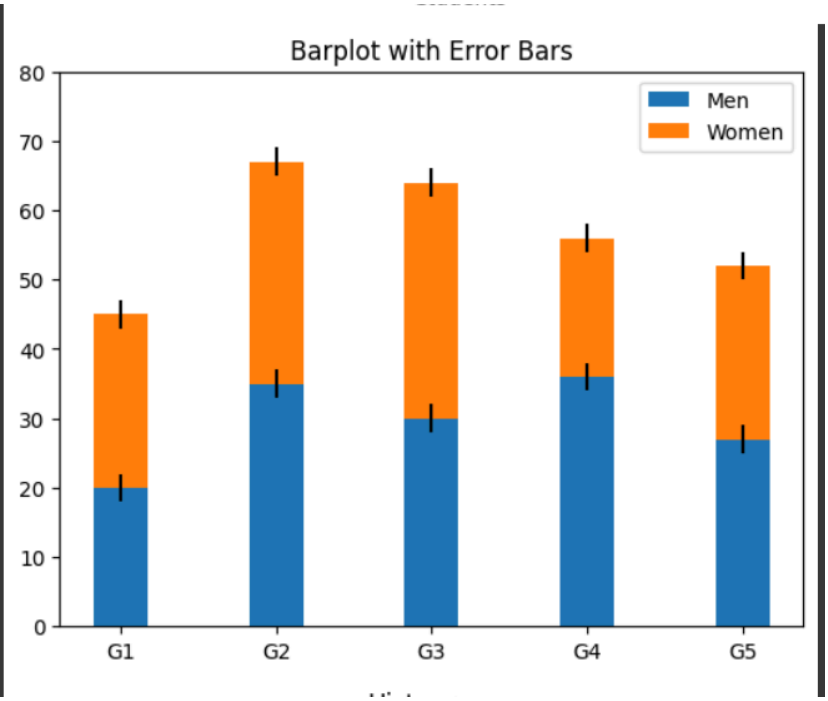
Name: Durva Kadam Roll No: 11

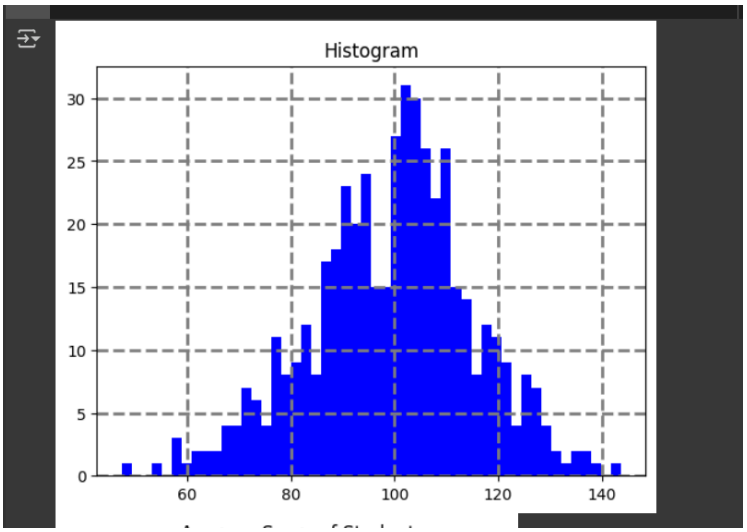
```
[[1 2 3]
 [4 5 6]
 [7 8 9]]
```



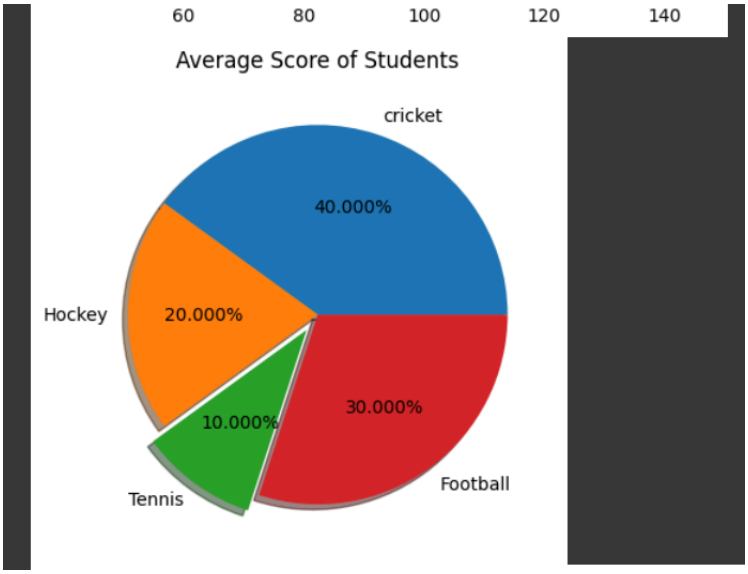
Average Score of Students

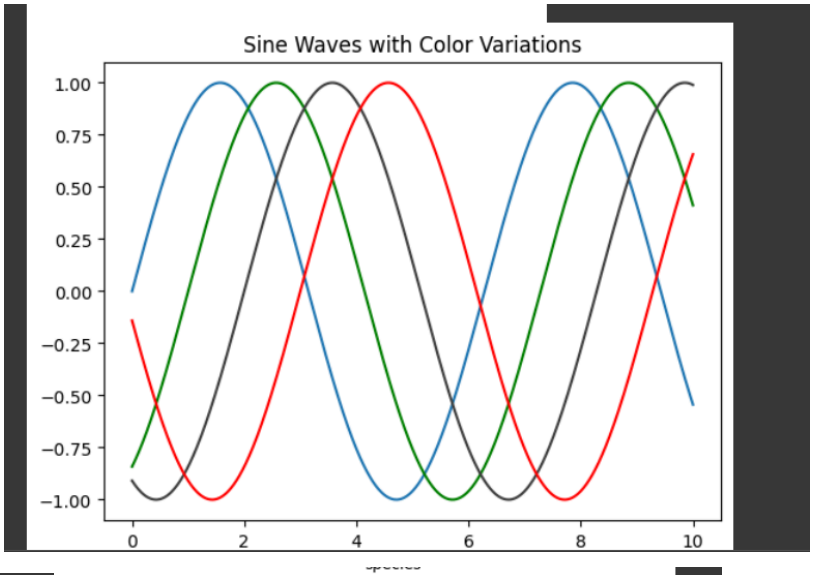


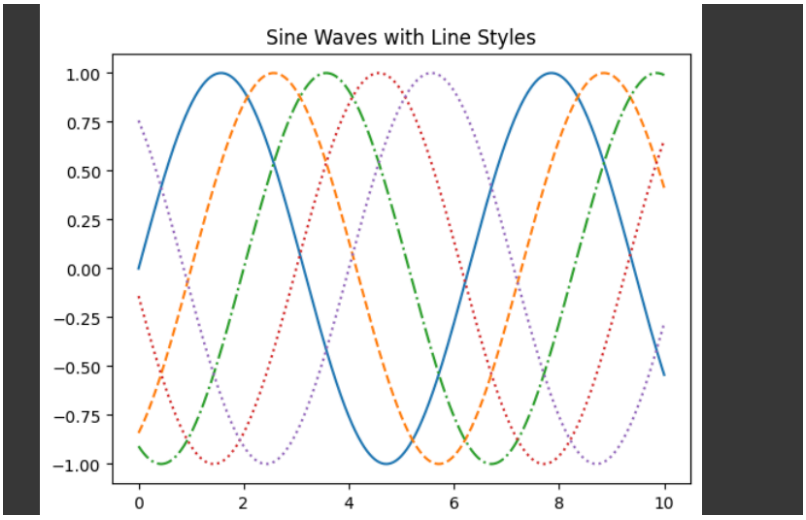


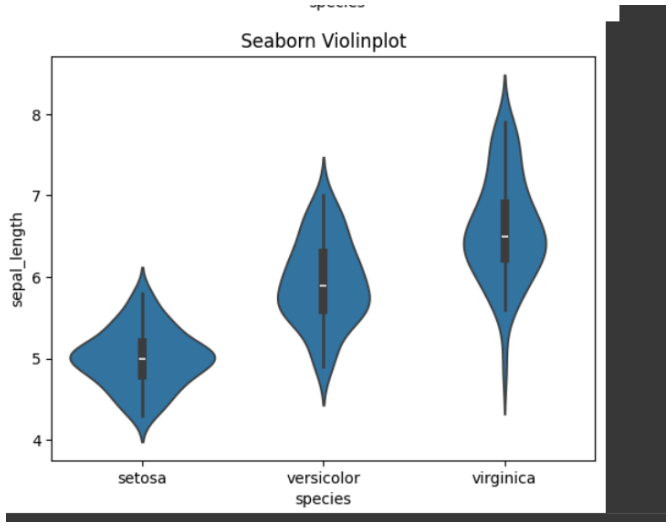


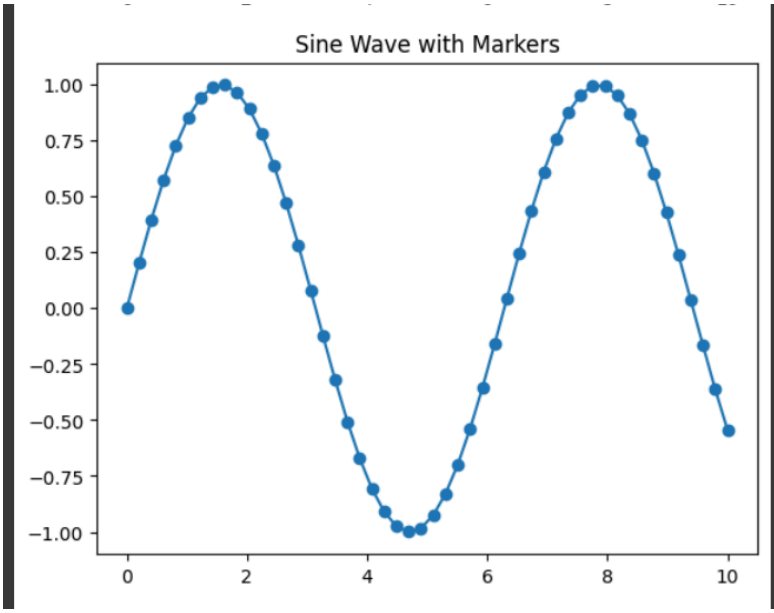
Name: Durva Kadam Roll No: 11

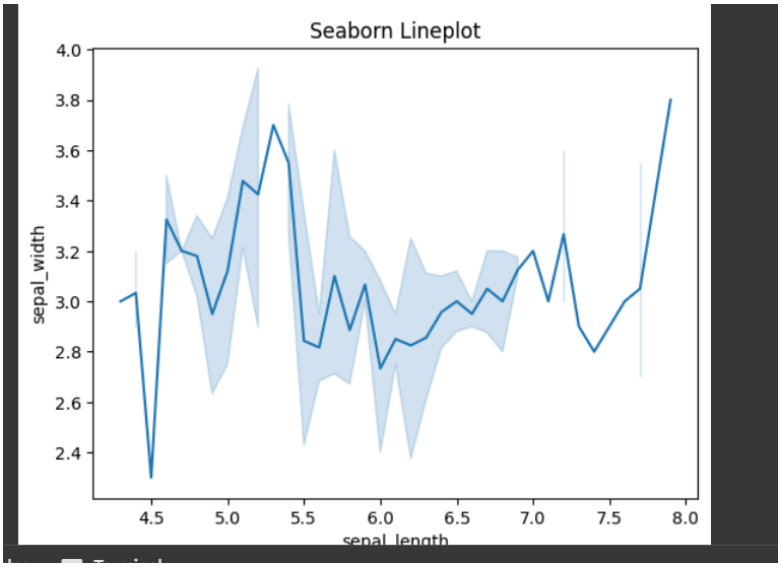












Name: Durva Kadam Roll No: 11

```python
import numpy as np
# Load the dataset
file_path = "pima-indians-diabetes.data.csv" # Update path if needed
df = pd.read_csv(file_path, header=None)
# Assign column names
df.columns = [
"Pregnancies", "Glucose", "BloodPressure", "SkinThickness",
"Insulin",
"BMI", "DiabetesPedigreeFunction", "Age", "Outcome"
]
# Set figure size for all plots
plt.figure(figsize=(14, 10))
# 1. Sinusoidal Plot (synthetic data)
plt.subplot(2, 2, 1)
x = np.linspace(0, 2 * np.pi, 100)
y = np.sin(x)
plt.plot(x, y, label='sin(x)', color='blue')
plt.title('Sinusoidal Plot')
plt.xlabel('x')
plt.ylabel('sin(x)')
plt.grid(True)
plt.legend()
# 2. Bar Plot (Count of Diabetic vs Non-Diabetic)
plt.subplot(2, 2, 2)
diabetes_counts = df['Outcome'].value_counts()
plt.bar(['Non-Diabetic', 'Diabetic'], diabetes_counts, color=['green',
'red'])
plt.title('Bar Plot: Diabetes Outcome')
plt.ylabel('Count')
# 3. Histogram (Age Distribution)
plt.subplot(2, 2, 3)
```