### **Task 7**

**Machine Learning**

Upload .py or Ipynb extension file on GitHub public repo “100DaysofBytewise" and share the link in the submission form by 2 July 2024.

1. **Exercise: Create a simple line plot using Matplotlib to display the trend of a list of values over time.**

**Code:** import matplotlib.pyplot as plt

values = [1, 3, 2, 5, 7, 8, 6, 9, 11, 10]

time = range(1, 11)

plt.figure(figsize=(10, 10))

plt.plot(time, values, marker='o', linestyle='--', color='b')

plt.title('Trend of Values Over Time')

plt.xlabel('Time')

plt.ylabel('Values')

plt.show()

1. **Exercise: Plot a bar chart using Matplotlib to show the frequency of different categories in a dataset.**

**Code:** import pandas as pd

import matplotlib.pyplot as plt

data={'a':['A', 'B', 'A', 'C', 'B', 'A', 'D', 'C', 'B', 'C', 'D', 'A']}

df=pd.DataFrame(data)

freqz=df['a'].value\_counts()

freqz.plot( kind='bar',color='black')

plt.title('Bar chart ')

plt.xlabel('frequency')

plt.ylabel('Values')

1. **Exercise: Create a scatter plot using Matplotlib to visualize the relationship between two variables in a dataset.**

**Code :** import matplotlib.pyplot as plt

x=[1,2,1,1,2]

y=[3,4,5,6,7]

plt.scatter(x, y, color='b')

plt.title('Scatter plot ')

plt.xlabel('x-values')

plt.ylabel('y-values')

plt.show()

1. **Exercise: Load a dataset using Seaborn's built-in dataset functions and create a pairplot to visualize the relationships between all pairs of features.**

**Code:** import seaborn as sns

import matplotlib.pyplot as plt

df=sns.load\_dataset('iris')

sns.pairplot(df,hue='species',markers=["o", "s", "D"])# o = oval s= squre d= diamond

plt.show()

1. **Exercise: Create a box plot using Seaborn to show the distribution of values for different categories in a dataset.**

**Code:** import seaborn as sns

import matplotlib.pyplot as plt

df=sns.load\_dataset('iris')

plt.figure(figsize=(10, 6))

sns.boxplot(df)# o = oval s= squre d= diamond

plt.show()

1. **Exercise: Plot a heatmap using Seaborn to visualize the correlation matrix of a dataset.**

**Code:** # correlation heat map

import matplotlib.pyplot as plt

import seaborn as sns

df=sns.load\_dataset('iris')

matrix=df.select\_dtypes(include=('number')).corr()

sns.heatmap(matrix,annot=True,cmap='cool',vmin=-1,vmax=1)

plt.show()

1. **Exercise: Use Matplotlib to create a subplot grid that displays multiple charts in a single figure.**

**Code:** import numpy as np

import matplotlib.pyplot as plt

x=np.linspace(0,10,100) # random array of 100 elements

a=np.sin(x)

b=np.cos(x)

c=np.tan(x)

d=np.exp(x)

fig,axes=plt.subplots(2,2,figsize=(12,8))

axes[0, 0].plot(x, a, color='blue')

axes[0, 1].plot(x, b, color='red')

axes[1,0].plot(x,c,color='black')

axes[1,1].plot(x,d,color='yellow')

1. **Exercise: Customize the appearance of a Seaborn plot by changing the color palette, adding titles, and modifying axis labels.**

**Code:** import seaborn as sns

import matplotlib.pyplot as plt

df=sns.load\_dataset('iris')

sns.boxplot(df)

plt.title('Sepal Length vs. Sepal Width')

plt.xlabel('Sepal Length (cm)')

plt.ylabel('Sepal Width (cm)')

plt.legend(title='Species')

plt.grid(True)

plt.show()

1. **Exercise: Create a violin plot using Seaborn to visualize the distribution of a dataset across different categories.**

**Code:** import seaborn as sns

import matplotlib.pyplot as plt

df = sns.load\_dataset('iris')

plt.figure(figsize=(12, 5))

sns.violinplot( df)

plt.title('Violin Plot')

plt.show()

1. **Exercise: Combine Matplotlib and Seaborn to create a complex visualization, such as overlaying a KDE plot on a histogram.**

**Code:** import pandas as pd

import matplotlib.pyplot as plt

df=sns.load\_dataset('iris')

plt.figure(figsize=(10,5))

sns.histplot(df,kde=True,color='b')

plt.show()