OPEN SOURCE SOFTWARE LAB (15B17CI575)

Lab Assignment 5 (Practice Lab)

Odd 2021

Week 5&6: 23 Sept-5 Oct

Topic Coverage: Python- Matplotlib and SciPy package

Matplotlib Practice Questions

1. Creating simple plots of sin(x) andcos(x)

```
import numpy as np

X = np.linspace(-np.pi, np.pi, 256, endpoint=True)
C, S = np.cos(X), np.sin(X)
import matplotlib.pyplot as plt
plt.plot(X, C)
plt.plot(X, S)
plt.show()
```

   Here, X is numpy array with 256 values ranging from -π to +π. C is cosine (256 values) and S is sine (256 values).

2. Exploring all the figure settings that influence the appearance of theplot.

```
# Create a figure of size 8x6 inches, 80 dots per inch
plt.figure(figsize=(8, 6), dpi=80)

# Create a new subplot from a grid of 1x1
plt.subplot(1, 1, 1)

# Plot cosine with a blue continuous line of width 1 (pixels)
plt.plot(X, C, color="blue", linewidth=1.0, linestyle="-")


# Plot sine with a green continuous line of width 1 (pixels)
plt.plot(X, S, color="green", linewidth=1.0, linestyle="-")

# Set x limits
plt.xlim(-4.0, 4.0)

# Set x ticks
plt.xticks(np.linspace(-4, 4, 9, endpoint=True))
```

```
# Set y limits
plt.ylim(-1.0, 1.0)

# Set y ticks
plt.yticks(np.linspace(-1, 1, 5, endpoint=True))

# Save figure using 72 dots per inch
plt.savefig("exercise_2.png", dpi=72)
```

3. Addinglegends

```
plt.plot(X, C, color="blue", linewidth=2.5, linestyle="-", label="cosine")
plt.plot(X, S, color="red", linewidth=2.5, linestyle="-", label="sine")
plt.legend(loc='upper left')
```

4. RegularPlots

```
n = 256
X = np.linspace(-np.pi, np.pi, n, endpoint=True)
Y = np.sin(2 * X)
plt.plot(X, Y + 1, color='blue', alpha=1.00)
plt.plot(X, Y - 1, color='blue', alpha=1.00)
```

5. Scatter Plot

```
n =1024
X = np.random.normal(0,1,n)
Y = np.random.normal(0,1,n)
plt.scatter(X,Y)
```

6. Bar Chart

```
n = 12
X = np.arange(n)
Y1 = (1 - X / float(n)) * np.random.uniform(0.5, 1.0, n)
plt.bar(X, +Y1, facecolor='#9999ff', edgecolor='white')
plt.show()
Y2 = (1 - X / float(n)) * np.random.uniform(0.5, 1.0, n)
plt.bar(X, -Y2, facecolor='#ff9999', edgecolor='white')
plt.show()
```

7. Plot tan(x), cot(x), sec(x) and cosec(x) for the values of x= [-pi,-pi/4, -pi/2, 0, pi/4, pi/2,pi]

8. Represent the following table using barchart

Method	**Result1**	**Result2**
A	2	3
B	5	2
C	8	5
D	5	7

SciPy Practice Questions

SciPy:

- SciPyisbuiltintopoftheNumPy
- SciPyisafully-featuredversionofLinearAlgebrawhileNumpycontainsonly a fewfeatures.
- MostnewDataSciencefeaturesareavailableinScipyratherthanNumpy.
- SciPy is organized into subpackages covering different scientific computing domains. These are summarized in the followingtable:

Subpackage	**Description**
cluster	Clustering algorithms
constants	Physical and mathematical constants
fftpack	Fast Fourier Transform routines
integrate	Integration and ordinary differential equation solvers
interpolate	Interpolation and smoothing splines
io	Input and Output
linalg	Linear algebra
ndimage	N-dimensional image processing
odr	Orthogonal distance regression
optimize	Optimization and root-finding routines
signal	Signal processing
sparse	Sparse matrices and associated routines
spatial	Spatial data structures and algorithms
special	Special functions

Subpackage	**Description**
stats	Statistical distributions and functions

- SciPy sub-packages need to be imported separately, forexample:

```
>>>
>>>from scipy import linalg,optimize
```

1. Importtheessentiallibraryscipywithi/opackageandNumpy. Create4x4,dimensional one'sarray.Storearrayin**test.text**file.Getdatafrom**test.text**fileandprintthe output.

2. Findcubicrootof27,64,891usingsciPyspecialpackage.

3. Create two matrices with 2x2 dimensions. Initialize them with values [4,5], [3,2]. Calculate determinant of a two-dimensional matrix using scipy.linalg.

4. CalculatetheinverseofamatrixinQ3.

5. Define two-dimensional array with values {(5,4),(6,3)}. Output eigen values and eigenvectors of the matrix.

6. CreateSparsematricesAandBandanalyzevariousfunctionsofsciPysparse package.