Final review

1. Numpy array

>>> import numpy as np

>>> myArray=np.array(mylist)

>>> myArray=np.zeros((3,3))

>>> myArray= np.arange(0,10,0.2)

>>> myArray =np.linspace(2.0, 3.0, num=5)

# NumPy Array Slicing

>>> myArray[0:2,0:2]

#Transposing an array,

>>> np.transpose(myArray)

>>> myarray.T

#Finding dimensions of an array

>>> myArray.shape

# Reshaping an array,

>>> myArray\_1= myArray.reshape(2,6)

#Find elements in an array satisfying a condition

>>> myArray[myArray>5]

#Array math

>>> Z=X+Y

>>> Z=X-Y

>>> Z=X\*Y

>>> Z=X/Y

>>>np.sin, np.cos, np.exp, np.sqrt, …

#mean

>>> np.mean(x)

>>> np.mean(x,axis=0)

>>> np.mean(x,axis=1)

>>> np.nanmean(x)

>>> np.nanmean(x,axis=0)

>>> np.nanmean(x,axis=1)

#minimum, maximum

>>> np.amin(x)

>>> np.amin(x,axis=0)

>>> np.amin(x,axis=1)

>>> np.nanmin(x)

>>> np.nanmin(x,axis=0)

>>> np.nanmin(x,axis=1)

>>> np.amax(x)

>>> np.amax(x,axis=0)

>>> np.amax(x,axis=1)

>>> np.nanmax(x)

>>> np.nanmax(x,axis=0)

>>> np.nanmax(x,axis=1)

# Finding array index of minimum and maximum

>>>i\_min=np.argmin(a)

>>>i\_min=np.argmin(a,axis=0)

>>>i\_min=np.argmin(a,axis=1)

>>>i\_max=np.argmax(a)

>>>i\_max=np.argmax(a,axis=0)

>>>i\_max=np.argmax(a,axis=1)

#standard deviation

>>> np.std(x)

>>> np.std(x,axis=0)

>>> np.std(x,axis=1)

>>> np.nanstd(x)

>>> np.nanstd(x,axis=0)

>>> np.nanstd(x,axis=1)

2. Graphics

>>>import matplotlib.pyplot as plt

# plt.plot; # labels and title; #label ticks; #define axis range; #legend #text; #define figures; #subplot; #figure layout; #save figures; #plt.show()

Examples: plt.plot\_example1.py, plt.plot\_example2.py, plt.plot\_example3.py.

#errorbar

Example: plt.plot\_example4.py

# Histogram

Example: plt.plot\_example5.py, plt.plot\_example6.py, plt.plot\_example7.py

#pie chart

Example: plt.plot\_example8.py

#contour

Example: contour\_example.py

#contourf

Example: contourf\_example.py

#quiver

Example: quiver\_example.py

#3d plot

>>> from mpl\_toolkits.mplot3d import Axes3D

>>> ax = fig.add\_subplot(111, projection='3d')

>>> ax.plot(x, y, z)

>>> ax.view\_init(30, 90)

Example: plt\_3dplot\_exam1.py

>>> ax.plot\_wireframe(X, Y, Z)

>>> ax.plot\_surface(X, Y, Z)

Example: plt\_3dplot\_exam2.py

>>> ax.contour(X,Y,Z)

>>> ax.contourf(X,Y,Z)

>>> ax.quiver(x, y, z, u, v, w)

Example: plt\_3dplot\_exam3.py, plt\_3dplot\_exam4.py

#Basemap

>>> from mpl\_toolkits.basemap import Basemap

Example: basemap\_example\_1.py

3. FFT(graduate)

Lect13\_Fourier\_transfer.ppt

Examples: numpy\_fft\_example\_1.py, numpy\_fft\_example\_2.py

4. Wavelet (graduate)

Lect14\_wavelet.ppt

Examples: wavelet\_example\_4.py

5. def function

Example: def\_func\_example.py, def\_func\_exercise\_1.py

6. IO

Read python, matlab and netcdf data

Lect7\_IO.docx

7. Solving differential equations (graduate)

Lect15\_Poisson\_eq.ppt; Lect16\_Runge-Kutta.ppt

Examples: poission\_excerise.py, Runge-Kutta-1order.py, Runge-Kutta-2order.py, Runge-Kutta-4order.py

8. ‘for’ and ‘while’ loop

Examples: test7\_for\_1.py, test7\_for\_2.py, test7\_for\_3.py, test7\_for\_4.py

test4\_while\_loop.py

9. ‘if’, ‘break’, and ‘continue’ statement

Examples: test2\_if\_statement.py, test3\_if\_statement.py, test5\_break.py, test6\_continue.py

Lecture3-python\_flow\_control.docx