

Lab 1

The goal of this lab is to provide exercises that will help students reinforce and improve knowledge of NumPy data structures.

Note: For each exercise enclose your source code and provide a short description of your solution, e.g. in case you have used an existing numPy function or written your own, explain briefly why/how the function helped you solve the exercise.

Exercise 1

Write a NumPy program that creates a 6x5 matrix and fills it with values 10 till 39.

Exercise 2

Write a NumPy program that converts kilograms (kg) into pounds (lbs) for the given input. The kilogram values are stored in NumPy array. Use the following conversion formula:

$$m_{(\text{lb})} = m_{(\text{kg})} / 0.45359237$$

The input array is [50, 100, 200, 1000, 1500]

The expected output is:

```
Values in kilograms:
[ 50  100  200 1000 1500]
Values in pounds:
[ 110.23113109  220.46226218  440.92452437 2204.62262185 3306.93393277]
```

Exercise 3

Write a NumPy program that generates 3x4 array of random numbers and prints out the maximum and minimum values for each row.

The expected output is:

```
Random array values:
[[0.77053771 0.51728205 0.55575297 0.45551608]
 [0.51758592 0.20422145 0.94800162 0.1028391 ]]
```

MatchIT – Python for Data Science

```
[0.30456961 0.25779675 0.55474463 0.70870617]]  
Max and Min values of row at index: 0 are: 0.7705377132377478  
0.45551608399300936  
Max and Min values of row at index: 1 are: 0.9480016240341845  
0.10283910003980368
```

```
Max and Min values of row at index: 2 are: 0.7087061667455995  
0.25779675262944257
```

Exercise 4

Use the array of randomly generated numbers from exercise 3 and print it out first sorted along the axis=0 and then along the axis=1. Write below the output of the first and the second sort (round floating point numbers to 3 decimals). Discuss the obtained output with respect to the axis specified for each sort.

Output, sort along axis=0

Output, sort along axis=1

Exercise 5

Write numPy program that calculates a sum of numbers divisible by 3 or 5 in interval from (0, 100).

Exercise 6

In this exercise you will create NumPy structured array for the following input data:

```
data = [(1, "Albert Einstein", 1.2),  
        (2, "Muhammad ibn Musa al Khwarizmi", 1.3),  
        (3, "Kurt Godel", 1.4)]
```

Data type of the structured array is integer of length 4, string of length 40, and float with precision 4. Save the structured array to a file, and load the file.

Exercise 7

In this exercise you will use NumPy structured array to load data from database.csv file and perform some simple data manipulation. The file contains comma-separated values pertaining to earthquakes that have occurred around the world since 1965. The file can be downloaded from the course Moodle page.

- a) Read in comma separated file database.csv using NumPy genfromtxt() function and print out the generated structured array. You will see many values marked as 'nan' – why? What data type(s) are by default assigned to your struct (integer, string, float, datetime ..)? What are the column headers?
- b) From the database.csv file read in the Date, Depth, and Magnitude columns. Ensure that all the elements within the struct are properly formatted.
- c) Write code to calculate minimum, maximum and mean Magnitude values. Print out the values.
- d) Write a program that will create total number of earthquakes for each year. You can store this information in a key, value dictionary, e.g. result[1965]=339. Use plot library matplotlib.pyplot to draw a bar graph where x axis will display a year and they y axis corresponding number of earthquakes for the year.