

# TFL115 - Exercise 5

Jonas Heer, Einar Johnsen

September 2018

## 1 Introduction

In this exercise there are assignments which should help you understand what we covered in the lecture. This exercise is not mandatory and should not be delivered, however it will help you manage the mandatory exercise which is introduced later on. Please let us know if there are any problems with the exercise or you feel unable to solve the tasks.

Feel free to use Github or Bitbucket when solving the tasks, but please do use a private repository for the solutions.

## 2 Numpy Basics

- Install the numpy package in PyCharm (or the environment you are using).
- Create a numpy array which contains two lists of respectively 1,2,3,4,5 and 6,7,8,9,10. The array shape should be (2, 5).
- Create another numpy array which contains 6,7,8,9,10 and 11,12,13,14,15. Should also have shape (2, 5).
- Multiply the two arrays together.

## 3 Remembering files

- Create a file which contains the numbers 1-15.
- Read the file and store the numbers in a list.
- Create a numpy array of the numbers you have stored.
- Run `np.random.shuffle(my_list)` and see what happens.

## 4 Numpy Helper Functions

- Create a function which takes in one argument (numpy array).
- Find the **index** of the maximum value in the numpy array. *Hint:* `np.argmax()`
- Find the maximum **value** in the numpy array. *Hint:* `max()` and `np.max()` is not the same
- Let the function return both the index and the value. *Hint:* `return a,b`
- Download the file *exercise\_5\_numbers.txt* from Canvas and generate a numpy array from the values. *Hint:* `np.genfromtxt()`
- Use your function and find the greatest value in the numpy array, as well as it's index.

## 5 Numpy Slicing

Try to solve each task with a single slice/selection. E.g. `a[:, -1]`.

- Create a numpy array as shown in Figure 1.
- Select the values marked in green.
- Select the values with a blue outline.
- Select the values with a red outline.

*Hint:* lecture notes

1	2	3	4	5	6	7
8	9	10	11	12	13	14
15	16	17	18	19	20	21
22	23	24	25	26	27	28
29	30	31	32	33	34	35
36	37	38	39	40	41	42
43	44	45	46	47	48	49

Figure 1:  $7 \times 7$  array

## 6 Solutions

### 6.1 Numpy helper function: 10004, 39