Metropolia Ammattikorkeakoulu
Cognitive Systems Mathematics and Methods
Mathematics / Python Exercises 2

1. Create a matrix

$$X = \begin{bmatrix} 1 & 1 & 1^2 & 1^3 & 1^4 \\ 1 & 2 & 2^2 & 2^3 & 2^4 \\ 1 & 3 & 3^2 & 3^3 & 3^4 \\ 1 & 4 & 4^2 & 4^3 & 4^4 \\ 1 & 5 & 5^2 & 5^3 & 5^4 \end{bmatrix}$$

   using commands `np.ones`, `np.arange` and `np.concatenate`.

2. Let

x_i	1	2	3	4	5	6	7	8	9	10
y_i	14	12	34	24	43	38	41	57	55	50

   Create 10×1 column vectors x and y and a 10×2 matrix

$$X = \begin{bmatrix} 1 & x \end{bmatrix}.$$

   Regression coefficients θ_0 and θ_1 in

$$\hat{y}_i = \theta_0 + \theta_1 x_i$$

   can be calculated in matrix form by

$$M = (X^T X)^{-1} X^T Y,$$

   where

$$M = \begin{bmatrix} \theta_0 \\ \theta_1 \end{bmatrix}$$

   Plot the points (x, y) and the regression line.

```
xx = np.linspace(1, 10, 1000)
XX =     <----- insert code, XX = [1, xx]

yy = XX.dot(M)
plt.plot(x, y, 'o')
plt.plot(xx, yy)
```

3. Fit a second order polynomial

$$\hat{y}_i = \theta_0 + \theta_1 x_i + \theta_2 x_i^2$$

to the data in Exercise 2. Use the matrix

$$X = \begin{bmatrix} 1 & x & x^2 \end{bmatrix}$$

and calculate

$$M = \begin{bmatrix} \theta_0 \\ \theta_1 \\ \theta_2 \end{bmatrix}$$

by

$$M = (X^T X)^{-1} X^T Y.$$

Plot the polynomial and the data in the same figure.

4. Use the Python command `np.polyfit` to fit a higher order polynomial to the data and plot the polynomial.

```
coef = np.polyfit(x, y, 3)
p = np.poly1d(coef)

xx = np.linspace(1, 10, 1000)
yy = p(xx)
plt.plot(x, y, 'o')
plt.plot(xx, yy)
plt.show()
```

Try different degrees. Does the higher order polynomials fit the data better? Which model would you use for prediction?