

Programming and Visualisation

This exercise is to familiarize yourself with one of the programming environments and introduce standard ways of visualisation. Possible options are Python, R, Julia or MATLAB. For more details, see the programming resources on ISIS.

The submission procedure (upload to ISIS) can be found in the organizational slides.

1.1 Scatter and Line Plots (3 points)

- (a) Read the data in the file `expDat.txt` (available on ISIS). Each row contains the number of events (of 20 observed processes) counted in one of 100 subsequent time intervals. The first column contains the respective time index. Each other column corresponds to one of 20 different observed processes.
- (b) Create a line plot – one line for each column where the X-axis refers to time index.
- (c) Create a scatter plot matrix for the first 5 columns (i.e., 25 pairwise scatter plots).
- (d) Compute the covariance matrix by centering & matrix multiplication – without looping over the data. Plot this matrix with a heatmap and compare it to the result when using inbuilt functions such as `cov` or `cor`.

1.2 Plotting 3d (5 points)

- (a) Read the data contained in the file `pca-data-3d.txt`.
- (b) Make a scatter plot matrix for all the columns.
- (c) Visualize the data in 3d. How do you interpret your results?
- (d) Pick the first two dimensions and plot the data in a scatter plot. Project the 2d data onto the unit vectors with angles 0, 15, ..., 180 degrees from the x-axis and compute the variance along these directions. Plot variance against angle.

1.3 Data Processing: Image data (2 points)

- (a) Read in the data from the file `natIMG.jpg`.
- (b) Plot the data using a heatmap with a 'color' scale ranging from black (minimal value of all image pixels) to white (maximal occurring pixel value).
- (c) Sample 100 image patches of size 10x10=100 pixels.
- (d) Plot these patches using the same color scheme as above.

- (e) Plot these patches using the inverted color scheme (white minimal, black maximal pixel value).

Total points: 10