Department of Computer Science University of Bristol

COMS21202: Symbols, Patterns and Signals

Lab 2: Introduction to Matlab (Part II)

NOTE: You will need to refer to the Matlab help pages to complete most of these examples.

1. Create a function called 'testNormal' that generates a random sequence of 10000 numbers from $N(0,1)$ and returns the `mean` and `var` of the sequence.

   Call the function multiple times and observe the output.

2. For a 2D normal distribution $N(\mu, \mathbf{C})$

$$\mu = \begin{bmatrix} 2 \\ 2 \end{bmatrix} \qquad \mathbf{C} = \begin{bmatrix} 4 & 2 \\ 2 & 6 \end{bmatrix}$$

   Estimate the probability that the data members $x_1 = (1,2)$, $x_2 = (2,8)$, $x_3 = (5,4)$ are sampled from this distribution (Hint: use `mvnpdf`)

3. Generate a sequence of 100 random vectors from the distribution above. (Hint: use `mvnrnd`)

4. Generate a scatter plot of the sequence (Hint: use `scatter`). Compute the mean and covariance for the sequence using the commands `mean` and `cov`.

5. Increase the length of the sequence to 4000. Plot the new data and compute the new mean and covariance.
   Discuss with your lab partner and write down your conclusion.

6. Create your own function in matlab that takes a matrix and returns its inverse.

   Do not use a built-in function for this exercise.

   The inverse of the matrix $\vec{A} = \begin{bmatrix} a & b \\ b & c \end{bmatrix}$ is $\vec{A}^{-1} = \frac{1}{ac-b^2} \begin{bmatrix} c & -b \\ -b & a \end{bmatrix}$.

   Call the function from the command line with different matrix configurations.

   Are there any conditions that you need to check for and include in your function?

   Compare the results of your function with the built-in function in matlab `inv`.

7. Learn how to debug your code one line at a time using breakpoints. You can also check the values of your variables during debugging

8. Load the image ‘flower.png’ into matlab using the command `imread` and display the image using the command `imshow`.

   Check the size of the image matrix.

   Discuss with your lab partner what this size stands for and write down your conclusions. Check with a lab assistant if you are not sure.

9. Convert the image to black and white using the command `rgb2gray`, check the new size.

   What do you conclude?

   Change the size of the image to half its current size using the command `imresize`.

10. Plot the histogram of your image’s pixel values using the command `imhist`.

11. Use the command `histeq` to change the image.

    Study the resulting image and the histogram. Understand the effect of this command on the image quantitatively and qualitatively. Write down your conclusions.