## Exercise #3 - Simple Data Analysis and Graphics

Submit exercises only in groups of two or three students – create such a group in moodle – and only upload once per group. Make sure you write the names and ID of all the people in the group in the script as a comment, as shown below).

Tutor in charge of this HW: Michael Sokoletsky (michael.sokoletsky@weizmann.ac.il)

**Instructions**

1. Do not submit a script that does not run properly. Before submission make sure to run the entire script from scratch.
2. Name your script "hw1\_< Name1\_ID number 1>\_< Name2\_ID number 2>.m".
3. Divide your code into **sections** (according to the questions or any other reasonable division). Use the '%% ' sign to divide your code into sections.
4. Give your variables **meaningful names**.
5. **Document** your work with comments in the script. (use the '%' sign).
6. Any number that appears in your script should be assigned to a variable (i.e. do NOT use "magic numbers").
7. Look at the documentation of the relevant functions you should use, and explore the function options before using it.
8. Use the following header and questions separator:

%%

% HW1 solution

% <your name1 & T.Z.1><your name1 & T.Z.2>

% for example:

% HW1 solution

% 1. Israel Israeli 123456789 2. Jane Doe 123456789

%%

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

% <short description of what your script is doing for

% example: this script load experiments data and

% analyzes it. The output is the statistics

% calculations results.

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%% Q1

% <Document the code: briefly describe the question and how you solved it>

disp('\*\*\* Question 1 \*\*\*');

<Q1 output>

disp(' ');

%% Q2

% <Document the code: briefly describe the question and how you solved it>

disp('\*\*\* Question 2 \*\*\*');

<Q2 output>

# disp(' ');

NOTE: All the following instructions should be implemented in your script *except* of sub-questions where it is indicated that you should use the command line.

**Data description**

Please download the files *conductivity.mat* and *conductivity.xls*. This file contains a table of electrical conductivity values as a function of concentration for various electrolytes in aqueous solution at 25 °C. In order to understand the dataset please review the spreadsheet in Excel before loading the \*.mat file into matlab. For section h you will need *pH\_in\_solution.mat* and *pH\_in\_solution.xls* which include pH values of several solutions as a function of temperature.

**Question 1- the plot**

1. Load the file q1a\_to\_f\_*conductivity.mat*. (after reviewing the Conductivity.xls)

You should have 3 variables:

*concentration* – electrolyte concentrations for which conductivity was measured.

*conductivity* – conductivity values of several electrolytes measured across concentration.

*electrolyte\_names* – names of the electrolytes.

1. Calculate the mean and the median conductivity of the electrolytes across concentration values (i.e. one mean/median value per concertation). Store the information in appropriate variables – one variable for al1 the means, and one for all the medians. Use short and meaningful names for your new variables. You can use the help command to learn about the mean and median functions.
2. Figure 1: Plot the mean and the median conductivity vs. concertation – in the same graph, each one with a different color. Line width should be 1.5. Add an appropriate title to this graph. Add an appropriate x label and y label. Add a legend in the lower right corner of the axes, so it won’t cover the data lines. (Hint: you can use the functions: title, legend, xlabel, ylabel, plot and hold on).
3. Figure 2: For all electrolytes, except for HCl (*Do not remove it from the data*, You are expected to use HCl in the next questions), plot the conductivity as a function of concentration in the same graph. Calculate and display in the command window the minimal and maximal conductivity values that are shown in the plotted data. Set the axis limits to be the {minimal\_value-15} and the {maximal\_value+15}. Add a title, axis labels and legend which contain the electrolyte names. Important note: Don’t *assume* the index of the HCl electrolyte in the list, *find* it using code!
4. Figure 3: plot the maximal conductivity per each electrolyte (in the given range of concentrations) using the *bar* function. Don’t forget to add individual labels to all the bars andthe axes, as well as a title.
5. Figure 4: Find the electrolyte that shows the largest change (Max-Min values) in conductivity across concentrations and plot its conductivity values (in red). In the same graph plot also the conductivity values for the electrolyte which changed the least (in blue). Add axis labels a title and a legend.
6. Figure 5: Load the file q1g\_pH\_in\_solution.mat (after reviewing the spread sheet pH\_in\_solution.xls)

You should have 3 variables:

*temp* – temperature of the solution in which the acidity (pH) was measured.

*pH* – acidity of each solution across different temperatures.

*solution\_names* – names of the solutions.

Use the function *subplot* to show two graphs on the same figure: the first graph should be pH as a function of temperature for all the acidic solutions (average pH<7) and the second graph should be pH as a function of temperature for all the alkaline solutions (average pH>7). Make sure to set the neutral pH value as the upper/lower ylimits accordingly (acidic/alkaline). Hint: don’t use the value 7 directly, assign it to a variable (i.e. avoid magic numbers).

**Question 2**

In this part, you will perform basic image manipulations. Remember that a color-image can be stored as a 3-dimensional array of size (height X width X 3). Each “layer” of this array is a matrix which store the pixel values in this channel (RGB: red, green, and blue). First, read the image “q2\_image\_hw3\_2022b.jpg” into your workspace, using the function ‘*imread*’. Display your image with ‘*imshow*’ (figure 6).

1. Now, we will create a 2X2 collage like we did in class (week 3). Duplicate the image (matrix) 4 times (test- the new variable size should be 2908x2400x3): for the first 3 images, keep only one channel from the RGB (your “layers”) per image (R, then G, then B) and put 0 for the other channels. The 4th image in your collage should show the original image. Explain briefly the result you get. What happens when instead of 0, you use the maximal value (255 for uint8)? (in this question you should plot figure 7 and 8)
2. Now we will play with the image colors. Create a new image (3-dimensional array) by switching between the first and the 3rd channels of the original image. Display your image (figure 9), and describe the result.

**Bonus question!**

1. Display the image and its vertical reflection as one vertically-elongated image. (Hint – The new matrix size should be 2908 x 1200 x 3).