## Intro to MATLAB

## Learning Objectives

* Get to know the MATLAB environment
* Assign values to variables
* Learn about MATLAB arrays
* Load data, perform operations on arrays of data
* Display simple graphs.

**Part 1 – The MATLAB Environment**

Current Folder

Command Window

Workspace => Temporary memory

Search Documentation

**Part 2 – Assigning Variables**

Assigning a variable, explain semi-colon

X = 1;

Y = 1;

Z = X + Y;

Variable names must begin with a letter, and can contain numbers or underscores

Variable arithmetic (plus, negative, divide, multiply)

Vectors (row, column, transpose, size) - use [ ]

X = [1 2 3];

Matrices

X = [1 2 3; 4 5 6];

Characters and strings – use ‘ ‘

X = 'hello Matlab';

%% Challenge 1

% Make a variable called 'age\_2015' and set it equal to your current age.

% Make another variable called 'age\_2025', set this variable to Age\_2015

% plus 10.

% Make another variable (Choose a name) and set it equal to half your age.

Explain how random numbers are generated, use of rand and randi command

random\_nums = rand(4)

random\_ints = randi([0 10],3,3)

Count how many participants are present and use randperm command to divide them in teams (Valour, Mystic, Instinct). For instance, if 20 participants are present, give each participant a unique number between 1 and 20, then use

teams = randperm(20)

%% Challenge 2

% create a square matrix in your workspace that is filled with random

% numbers

% HINT: randi([a b],Nrows,Ncols), creates a random matrix filled

% with numbers between a and b, and of the size specified by Nrows x Ncols

% randi([-10 10],2,3) is a matrix with 2 rows, and 3 columns,

% containing numbers between -10 and 10

% try it out! (highlight the following line and press F9)

random\_matrix = randi([-10 10],2,3)

**Part 3 – Loading Data**

Pokemon Evolution data (each row: individual pokemon, first column: Pokemon Name, second column: Pokedex Number, third column: Combat Power (CP) before evolution, fourth column: CP after evolution, fifth column: player level)

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | Name | Pokedex Num CP before | | CP after | Player level |
| Pokemon 1 | … |  |  |  |  |
| Pokemon 2 |  | … |  |  |  |
| Pokemon 3 |  |  | … |  |  |
| … |  |  |  | … |  |
|  |  |  |  |  | … |

Show how to import data from excel file using import data button in home ribbon

Import data directly, also show how to generate code. Go through generated code, explain xlsread and show in documentation

**Concatenating Data – “Putting Numeric Data together”**

**Indexing Data – “Taking Slices”**

Look at the first pokemon – use matrix\_name(row index, column index )

Pokemon1\_Data = PokemonCP(1,:);

Look at second pokemon, all days

Pokemon2\_Data = PokemonCP(2,:);

Look at all pokemons, CP before

AllPokemons\_CP\_before = PokemonCP(:,2);

Look at all pokemons before and after CP together

AllPatient\_5Days = PokemonCP(:,2:3);

More vector making

AllNumbers = 1:10; % start:step:end

EverySecondNumber = 1:2:10;

Look at CP before, every second pokemon

All\_CP\_before\_AlternatePokemons = PokemonCP(:,1:2:end);

CHALLENGE:

%% NEED A BETTER CHALLENGE

**RECAP**

Define vectors and matrices with [ ]

Define strings and characters with ‘ ‘, and squish them together with [ ]

Index a matrix or vector with ( )

**Part 4 – Analysing Data**

Finding the mean of the inflammation on all days – use documentation

mean(patient\_data(:))

 maximum, minimum and standard deviation

max(patient\_data(:))

min(patient\_data(:))

std(patient\_data(:))

We can also save multiple values to variables

Display the max for patient 1

Patient1\_Max = max(patient\_data(1,:))

**Part 5 – Displaying Data**

We have a lot of dimensions to look at. A good place to start is visualizing

Show them the plot tab

Heat map

imagesc(patient\_data)

Of course we can also perform calculations on different dimensions

This is each patients average inflammation calculated over 40 days

mean(patient\_data, 1)

Dimension 2

This is the average inflammation across the 60 patients on each day

mean(patient\_data, 2)

Look at avg inflammation over time using the plot GUI

ave\_inflammation = mean(patient\_data);

CHALLENGE:

%% Challenge 3 - Part 3

% You saw how to calculate the mean inflammation for different

% dimensions

% Now create a variable that contains the maximum inflammation on each day

% and another with the maximum for each patient

% HINT: Using max on different dimensions is not exactly the same as mean -

% check the documentation!!

%% EXTENSION

% Plot your result

% Repeat for min and std.