**Octave Tutorial**

**Nicholas Wilkinson**

<https://www.youtube.com/watch?v=TqwSlEsbObg>

The video I used was “Octave Tutorial for Absolute Beginners: Learn Octave in 1 hr and 30 min” by Mr. STEM EDU TV on YoutTube. The video starts off by giving a brief summary of what is to be covered and how to download and install Octave. STEM then went over the basics of the GUI itself, such as which windows show what, how to adjust their sizes, and how to input commands into the command window. Afterwards, he covers simple commands, such as ‘clc’, ‘help’, ‘lookfor’, ‘who’, ‘whos’, and ‘clear all’. He then discusses basic arithmetic, variables, value types, vectors and matrices.

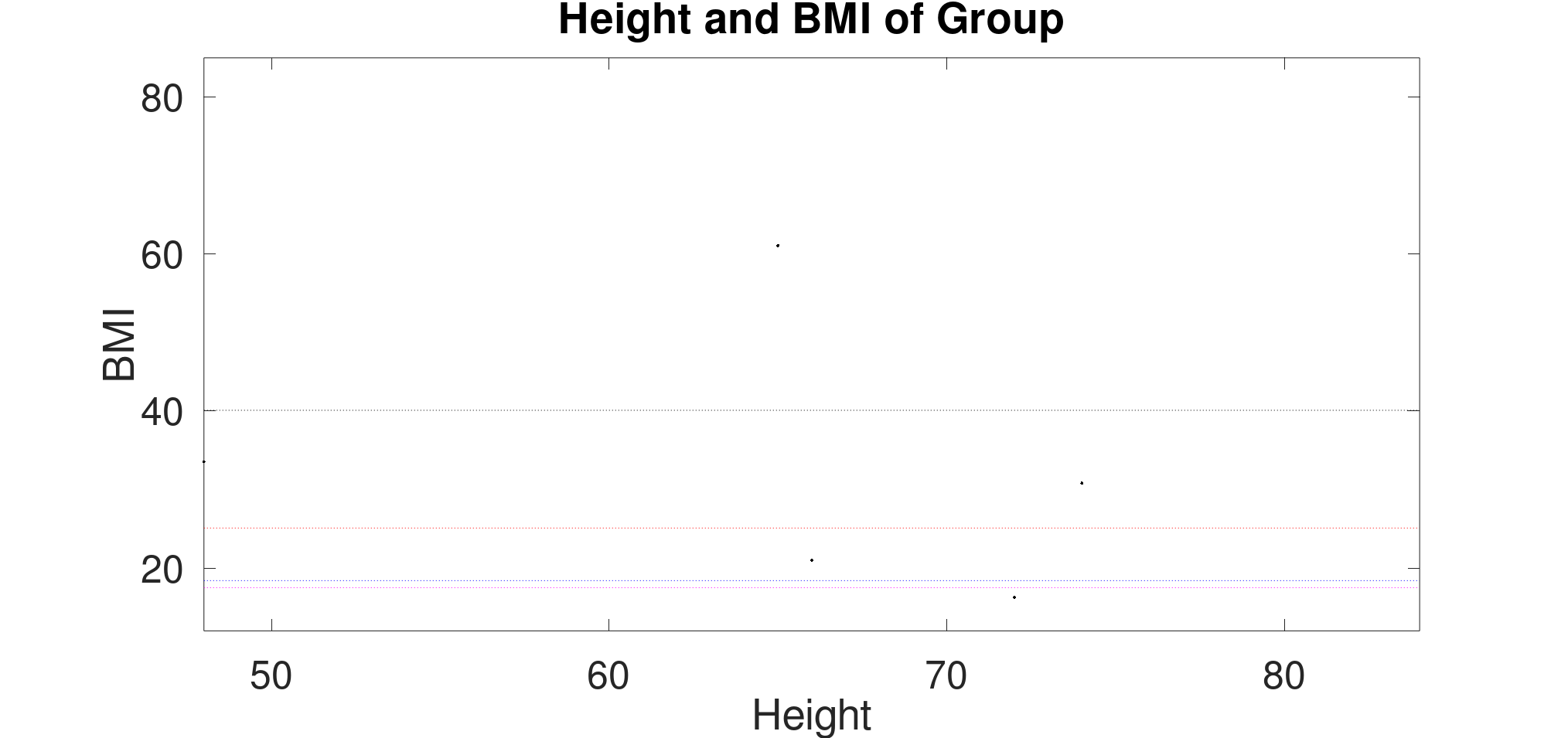
Then, STEM explains how to plot points and the different properties that can be assigned to a graph, such as the line style, marker type, the color, and the marker size. He then goes over how to adjust the properties of the graph, such as the title and axis minimums and maximums. Further, he shows how to plot multiple graphs at a time, since each use of the ‘plot’ command clears the previous plot. Afterwards, he goes over how to make custom ticks for a plot, such as using pi as the ticks for a sine or cosine graph.

After that, STEM covers how to make scripts and functions, as well as functions with multiple return variables. As an example, he makes a function that finds the area and volume of a circle given its radius. He also discusses how to use functions within functions, also known as nested functions, and shows this by making a function to find the area of a circle using the radius and nests it into the function to find the area and volume of a circle. He briefly goes over what conditional statements are, and then shows how to make conditional statements by making a function to see what letter grade someone has based on their number grade, followed by explaining for and while loops. The explanation for the while loop is given at the end of the video, but is introduced in concept with the for loop. After going over the for loop, STEM creates a for loop to calculate the BMI of a list of individuals. Lastly, he covered the ‘pause’ command while making a countdown timer.

Since there wasn’t a large project throughout the tutorial that utilized everything learned, I decided to create my own project that required most of the knowledge gained in the video. I expanded upon STEM’s BMI for loop project, challenging myself to create several functions to do with BMIs. First, I made a function, heightWeightMatrix, that would take user input and create a height and weight matrix. The user will first input the number of individuals, followed by the height and weight for each in inches and pounds respectively. The returned matrix will have height as the first column and weight as the second.

Then, I made another function, bmiMatrix, to calculate BMIs from a matrix of height and weight. heightWeightMatrix will be nested within bmiMatrix to get the heights and weights. The return will be a new matrix, with the first column being the height, the second the weight, the third the BMI, and the fourth will state if the individual is severely underweight, underweight, healthy, overweight, or obese. This is formated as numbers, with -2 being severely underweight, -1 being underweight, 0 being healthy, 1 being overweight, and 2 being obese.

I then made a third function, bmiGraph, that uses bmiMatrix as a nested function to graph the BMI values based on their height (so height would be the x values and BMI would be the y values). Additionally, there will be botted lines that show the thresholds for being obese, overweight, underweight, and severely underweight with the lines for each being black, red, blue, and magenta respectively. If the dot is in between the overweight and underweight line, they are healthy. The function has a static axis from 4 feet to 8 feet and a BMI of 12 to 90.



The resulting graph from my bmiGraph function.

There were only a few commands that I needed to look up outside of the video tutorial I used, such as how to increment variables and how to do a formatless print with the ‘disp’ command. I ran into some trouble programming these functions, but all of them were due to unfamiliarity with Octave. The mistake I made the most was that I kept adding colons to the ends of my conditional statements. The first time it happened it took a while for me to notice what was wrong since Octave was saying the error was the line after the conditional statement was initiated.