

Assignment 4: Analyzing Isometric Strength Data Due: November 5

You will analyze some isometric strength data. All input data is contained in the file 'isok_data_6803.csv' on Canvas in the 'Get Data In, Process Data, Get Data Out' Module. The file contains subject ids, ages, genders, weights, and three days of isometric maximum strength data for some participants in a strength training research protocol.

Please make sure your function names match the ones specified in this document. Make sure that all necessary .m files you create are pushed to github. Do NOT push your results spreadsheet to your github account.

Requirements:

1. Create a new script and save it as **assignment4.m**. This will be your master script in which will run your analysis. This is the only file you should have to run. You will create several functions for certain tasks that you will call from this file.
2. Create a function to import your data. (You can use the import data tool if you wish). Name this function **importfile.m**
3. Using your import function, import each column as a separate vector and use the column headers as the variable names.
4. Create a function (name it **genderIsoCalc.m**) that returns four outputs. The first set of outputs (one for males and one for females) should return a matrix with individual mean isometric strength values across all 3 days of lifting for the appropriate group. The second set of outputs will return the single mean value for each group (mean of subject means). Call this function in your main script and store the results with variable names **maleIsoIndMeans**, **femaleIsoIndMeans**, **maleGroupIsoMean** and **femaleGroupIsoMean**. These data should not be normalized.
5. Create a function (name it **dayComparer.m**) that takes the subject ids and two days as inputs and returns a matrix with the subject IDs of the subjects who had an increase from the first day to the second day (it should also work for the second day to the third day). Run this function in your main script twice (be sure to update your inputs when you call the function the second time) and store the outputs as **day1toDay2** and **day2toDay3**.

6. Weight normalize the isokinetic data and calculate the group means for each day. Store these values with the names **normDay1mean**, **normDay2mean**, **normDay3mean**. You do not need to write a function for this, but if you'd like to feel free.
7. Export your results to a csvfile using an appropriate built-in function. Name the file **iso_results.csv**. (You do not need to submit this file because your script should create this file when I run your code.)
8. Don't forget to write lots of comments!