CARNEGIE MELLON UNIVERSITY

10601a: Homework #3 - “hello, matlab” (autolab)

TAs-in-charge:
Daniel Ribeiro Silva (drsilva@andrew)
Seth Flaxman (sflaxman@cs)

Assigned: Tuesday, 14 January 2014.
Due: 10am on Tuesday, 21 January 2014.
Late Penalty: 10% per day.

Policy on Collaboration among Students

The purpose of student collaboration is to facilitate learning, not to circumvent it. Studying the material in groups is strongly encouraged. It is also allowed to seek help from other students in understanding the material needed to solve a particular homework problem, provided no written notes are shared, or are taken at that time, and provided learning is facilitated, not circumvented. The actual solution must be done by each student alone, and the student should be ready to reproduce their solution upon request. The presence or absence of any form of help or collaboration, whether given or received, must be explicitly stated and disclosed in full by all involved, on the first page of their assignment. Specifically, **each assignment must contain a file named collaboration.txt where you will answer the following questions**:

- Did you receive any help whatsoever from anyone in solving this assignment? Yes / No. If you answered ‘yes’, give full details? (e.g.“Jane explained to me what is asked in Question 3.4").
- Did you give any help whatsoever to anyone in solving this assignment? Yes / No. If you answered ‘yes’, give full details? (e.g. “I pointed Joe to section 2.3 to help him with Question 2").

Collaboration without full disclosure will be handled severely, in compliance with CMU?s Policy on Cheating and Plagiarism. All violations (even first one) will carry severe penalties, up to failure in the course, and will in addition always be reported to the university authorities.
Some of the homework assignments used in this class may have been used in prior versions of this class, or in classes at other institutions. Avoiding the use of heavily tested assignments will detract from the main purpose of these assignments, which is to reinforce the material and stimulate thinking. Because some of these assignments may have been used before, solutions to them may be (or may have been) available online, or from other people. It is explicitly forbidden to use any such

sources, or to consult people who have solved these problems before. You must solve the homework assignments completely on your own. I will strictly enforce this policy, and if a violation is detected it will be dealt with harshly. Collaboration with other students who are currently taking the class is allowed, but only under the conditions stated above.

General Instructions

The goal of this assignment is to introduce you to Matlab as a powerful tool for doing statistics and to introduce you to the autolab environment for homework submission. We've described functions that we want you to implement. Be sure to test your code on linux.andrew.cmu.edu (using Octave) before submitting. Following the software engineering principle of "unit tests," we'll give you sample input and output, and when we grade it, we'll use other inputs and outputs.

All four questions in this assignment are very basic, introductory questions about Matlab/Octave. So don't be concerned if your implemented function has a single line or two of code. Also, you are allowed to use built in Matlab/Octave functions in this assignment.

For example: if we asked you to implement `function [z] = add(x,y)` where $z = x + y$, we want you to write a function add to take as input two vectors and return their sum. Put the following in a file called `add.m`:

```
function [z] = add(x,y)
    z = x + y;
end
```

You can test `add.m` by call `addTest()`, which returns a score between 0 (all tests were incorrect) and 1 (all tests were correct). It'll also tell you which tests, if any, failed. (Don't worry about the details of how `addTest` is implemented).

1 Mean

Implement `function [a] = myMean(v)` to take a vector $\vec{v}$ and return the average of its entries:

$$\frac{1}{n}\sum_{i=1}^{n} v_i$$

(Don't use a for loop!)

You can test your function by calling `meanTest()`. Your delivered file must be named **myMean.m**.

2 Dot products

Here is a particularly slow way to take the dot product of two vectors (you can verify this yourself with the `cputime` variable):

```
function [c] = dotp(x,y)
```

```
    c = 0;
    for i = 1:size(x,1)
        c = c + x(i) * y(i);
    end
end
```

Rewrite this function without a for loop (this is called “vectorizing” your code) in `dotp.m`. Before you rewrite it, check that it works first by calling `dotpTest()`. Then, after you rewrite it, check that it still works by calling `dotpTest()` again. Your delivered file must be named **dotp.m**.

3 Matrix algebra

Implement `function [x] = solve(A,b)` to solve the equation $Ax = b$ for x where A is a matrix and x and b are vectors. You can test your function with `testSolve()`. Your delivered file must be named **solve.m**.

4 Standardizing Data

Implement `function [z] = rescale(x)` to take a vector of data in x and rescale it so that it has mean 0 and standard deviation 1:

$$z = \frac{x - \bar{x}}{\mathrm{sd}(x)}$$

where $\bar{x} = \sum_i x_i / n$. You can test your function with `testRescale()`. Use the Matlab function `std` to calculate the standard deviation. Your delivered file must be named **rescale.m**.

5 Autolab Submission

You must submit a .tgz file named **hw3.tgz** containing all your files (the four .m files + collaboration.txt).
You can create that file by running “ *tar -cvf hw3.tgz *.m collaboration.txt*”.
DO NOT put the above files in a folder and then tar gzip the folder.
You must submit this file to the “homework3" link on Autolab (https://autolab.cs.cmu.edu/10601a-s14/homework3/). Since you have a limited number of attempts, please make sure your code is working under Octave before you submit it to Autolab.