University of Wisconsin-Madison | Spring 2018
BMI/CS 776: Advanced Bioinformatics | **Homework #0**
Prof. Anthony Gitter | Due: **Thu, Feb 8th 2018, 11:59 PM**

Assignment goals:

i. Make sure you can log in to your course account and turn in files
ii. Set up your Python environment for future homework
iii. Learn to write basic Python code

Submission instructions:

To turn in your assignment, copy any relevant files to the directory:

```
/u/medinfo/handin/bmi776/hw0/<USERNAME>
```

where `<USERNAME>` is your account name for the biostat server. You can test running your code in that directory as well. You will likely want to use an SFTP client to transfer files to the server.

Part 1: Changing initial password

To change your password, log on to the server `adhara.biostat.wisc.edu` via `SSH`. If you have a new account, change your initial temporary password using the `yppasswd` command. You do not need to change your password if you already had a biostat account.

Once you have changed your password on `adhara`, use the servers `mi1.biostat.wisc.edu` or `mi2.biostat.wisc.edu` for submitting your assignments. It may take several hours for your new password to propagate from `adhara` to the other servers, so keep your initial password handy. Note that you will need to run WiscVPN (http://www.doit.wisc.edu/network/vpn/) in order to access these servers if your computer is not already on the UW network.

Part 2: Set up Python environment

We will be using our own BMI 776 Python environment for the homework instead of the default version on the server so that we can control which packages are available. The exact package versions are specified on the course website. In this step, we will test that you can run Python code in the BMI 776 environment.

Set the `PATH` environment variable so that the BMI 776 version of Python is detected before the default version.

a. Change to your home directory
```
cd ~
```
b. Edit your `.profile` file
```
vi .profile
```
c. Insert the following after the line `fi` and save the file
```
export PATH=/u/medinfo/bmi776-miniconda3/bin:$PATH
```
d. Exit the vi editor and type
```
source .profile
```

e. Test that the correct version of Python is found

   ```
   type -a python
   ```

f. You should see `/u/medinfo/bmi776-miniconda3/bin/python` listed first if the `PATH` was set correctly. As further confirmation, when you run the command `python --version` you should see Python 3.6.2 from Continuum Analytics, Inc. instead of Python 2.7, the default on the biostat servers.

Part 3: Python code practice

Download `hw0.py` and fill in the missing code based on the prompts in the comments. You will get points for successively uploading your code, attempts to finish the code, and producing the expected output when it is run on the biostat server (the most important part).

Here are some requirements and hints:

1. Use the command

   ```
   python hw0.py --inputfile=data.txt --outputfile=out.txt
   ```

   to test your code on the biostat server. The input file should be in the same directory as `hw0.py`.
2. For the instructions to print information or variables, you should print into the output text file and not to the screen (stdout). See the examples at http://www.python-course.eu/python3_print.php
3. The Python cheat sheet (https://ipgp.github.io/scientific_python_cheat_sheet/) will be quite useful if you are not familiar with Python. When you search the Python API (https://docs.python.org/3.6/library/), make sure it is the API for Python 3 not Python 2.
4. We will check your code. Although you can easily compute the simple operations in your head and print the requested output, you have to follow the steps in your code.
5. For lists and dictionaries, you should print them directly by `print(your_list)` or `print(your_dict)`.
6. Every printed output should be separated by a blank line.
7. Parts 4 and 5 are designed to introduce important parts of the NumPy API. Check the NumPy Reference (https://docs.scipy.org/doc/numpy-1.13.0/reference/) before re-implementing standard operations on your own, especially the `digitize`, `matmul`, and `percentile` functions.

8. For matrix printing, you should use the following format:

```
-1 2 -3
0 4 2

1 3 5
4 1 8
1 1 1

……
……
```

That is, separate elements with a space and matrices by blank lines.