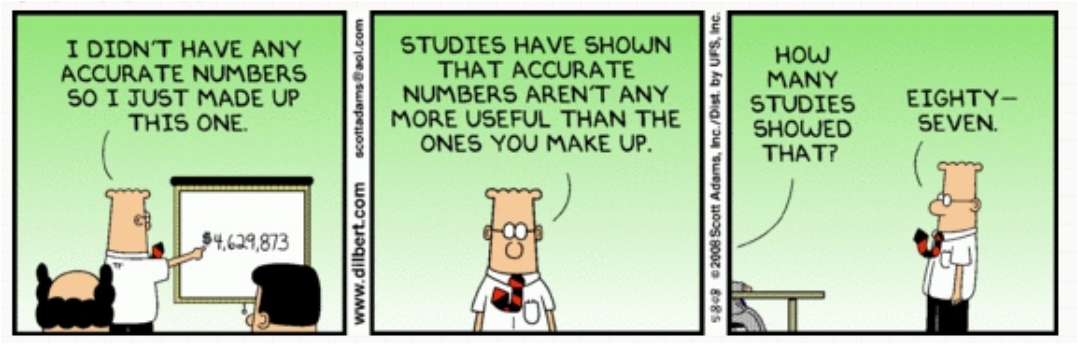
CS150 - Assignment 5
Data For Everyone
Due: Wednesday March 19, at the beginning of class



http://dilbert.com/fast/2008-05-08/

For this assignment we're going to implement some initial data analysis functions and then analyze some real data.

You are allowed (and encouraged) to work in pairs on this assignment. If you do, you must do *all* of the work together. Only turn in one copy of the assignment, but make sure both of your names are in the comments at the top of the file.

1 Creating/Editing Text Files

For this assignment, we will be reading data from files. The files must be of type ".txt" (no Microsoft Word docs, etc.). Also, recall from class that the files must be saved in the *same directory as your Python program* that will be reading them. There are many ways to do this, but one easy way is to just create them using Wing. To do this, just create a new file and then put whatever text data you want in the file (e.g. a list of numbers). When you save the file, give it a filename with ".txt" at the end and save it in the same directory as your Python program. `TextWrangler` is installed on all the lab machines and can also be used to edit and save text files.

2 Data Basics

Read through this whole section before starting!

Write a program that prompts the user for the name of a data file that contains one number per line and then prints the following statistics about the file:

- The number of entries in the file
- The largest value in the file
- The smallest value in the file
- The average of the values in the file (recall we did this one in class)
- The median of the values in the file. For an odd number of data elements, the median is the "middle" item if the data were in order. For an even number of data elements, the median is the average of the two middle items if the data were in order. Note, the median we examined in class was not complete/correct.
- The standard deviation of the values in the file. The standard deviation of n data points, is defined as:

$$std\_dev(data) = \sqrt{\frac{\sum_{i=1}^{n}(data_i - avg(data))^2}{n-1}} \tag{1}$$

  that is the square root of the sum of the differences between the data points and the average squared, divided by number of data points minus 1.

For example, if I have a file called "test.txt" that contains the following:

```
1
2.0
10
5
5
9
8
6
7
5
```

then a run of the program would output:

```
Enter file to analyze: test.txt
File contained 10 entries
Max: 10.0
Min: 1.0
Average: 5.8
Median: 5.5
Std. dev: 2.85968141194
```

Your program should only read data from the file *once*. Like the examples in class, read the data from the file once, then store the values in a list and use that list to calculate what you need.

I'm giving you a fair amount of flexibility regarding how you implement this, but use good style. For example, think about how to break your program into a number of functions instead of writing one giant piece of code.

3 Frequency

The following is a function that attempts to print out the frequency of each item in the data:

```
def frequencies(data):
    """Attempts to print the frequency of each item in the list data"""
    data.sort()

    count = 0
    previous = data[0]

    print "data\tfrequency"

    for d in data:
        if d == previous:
            # same as the previous, so just increment the count
            count += 1
        else:
            # we've found a new item so print out the old and reset the count
            print str(previous) + "\t" + str(count)
            count = 1

        previous = d
```

For example, given the list `[6, 5, 5, 1, 1, 2, 2, 3, 3, 4, 5, 5]` it should print:

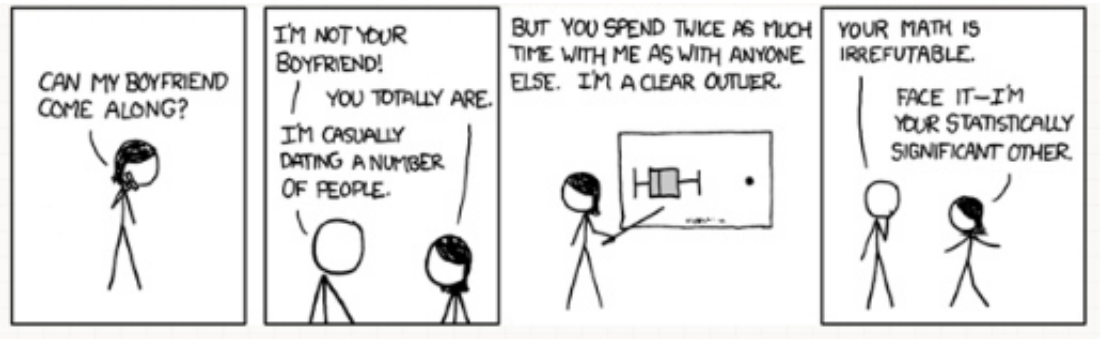
```
data frequency
1    2
2    2
3    2
4    1
5    4
6    1
```

Unfortunately, the program has a bug and doesn't do this. Go to:

`http://www.cs.middlebury.edu/~dkauchak/classes/cs150/assignments/assign5/frequencies.py`

and copy and paste this function into your program and then fix it so that it works properly. There are many ways to fix it, however, one straightforward way will only require adding/changing one line of code

4 Real Data



http://xkcd.com/539/

I've provided you with two real-world data sets at:

`http://www.cs.middlebury.edu/~dkauchak/classes/cs150/assignments/assign5/`

- `95census` is a folder containing some data from the Northeast region of the US from the 1995 census. There are three files in the folder corresponding to the age, number of kids and income of the surveyed participants. You can download any of these by right clicking and selecting "Save link as..." (or something similar depending on the browser).
- `3movie_reviews.xlsx` is an Excel file containing movie reviews from three different movies. To use this data, you'll need to open it in Excel and then grab the data you're interested in and copy and paste it into Wing and save as a ".txt" file.[1]

For each of these data sets, provide an analysis of the data using your program (one analysis for each data set, for a total of two analysis). For example, your two experiments might be:

- analyze income in 1995
- compare the data analysis numbers for two different movies

Be creative! But don't spend too much time on this part.

Provide the output from your experiments at the beginning of your program file (after your names, etc.) commented out *with* one or two sentences describing your analysis and results. If you look under the "Source" menu in Wing, there is an entry "Toggle Block Comment", which allows you to select some text and comment/uncomment the entire thing.

[1] If you're curious, this is just a snippet of data from http://www.grouplens.org/node/73

5 Extra Credit

You may earn up to 2 points of extra credit on this assignment by adding improvements to your program. If you do, include in your comments at the top of your program what you added. Below are some suggestions, but feel free to add you own:

- (1 point) Add a calculation of the *mode* of your data, that is the most frequently occurring element in the data. Notice this should have a similar feel to the `frequencies` function.
- (? points) Do additional experiments/analysis of the real world data sets. You'll be scored based on how creative you are. For example, just running an additional census file through the program won't be worth much.
- (? points) Add some other statistic to analyze your data with.
- (? points) Add your own ideas. Points will be awarded based on difficulty and innovativeness.

6 When you're done

Make sure that your program is properly commented:

- You should have comments at the very beginning of the file stating your name, course (including section number), assignment number and the date.
- Each function should have an appropriate *docstring*
- Other miscellaneous comments to make things clear

In addition, make sure that you've used good ***style***.

What to hand in:

You should have implemented:

- The program described in Section 2
- Your fixed `frequencies` function (in the same file)
- Your two analyses of the real-world data (commented out in the same file)

Submission procedure

Submit your .py file online using the digital submission link on the course web page. You must have submitted it online before the beginning of class. If you worked with a partner, you only need to submit one copy, but make sure both people's names are at the top of the submitted file.

Grading

		points
Data Basics		
	file reading	3
	# of entries, largest, smallest, average	2
	median	2
	standard deviation	3
`frequencies`		3
Data analysis		4
Comments, style		3
extra credit		2
total		20 (+2)