

# Week 9 Assignment

## Importing the data into Python and Graphing the 4 plots with regression lines.

First we need to connect to the database. Make sure to change the path to the location of the file on your machine.

```
In [5]: import sqlite3
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
%matplotlib inline
conn = sqlite3.connect('D:/Erik/cunyweek9.sqlite')
cur = conn.cursor()
```

Next I created a function that takes the table name and returns a list of tuples which represent the x and y coordinates. The second function creates a scatter plot and fits a linear regression line to it.

```
In [39]: #Takes the data from the SQLite database and generates a set of points
def read_table(tablename):
    table = str(tablename)
    cur.execute('SELECT * FROM %s' %table)
    points = []
    for row in cur:
        points.append(row)
    return points

#Takes a list of points(as a tuple or list) and title for the plot as a character
#and generates a scatterplot and linear regression line for the data.
def plot_test(ax, points, title):
    x = []
    y = []
    for point in points:
        x.append(point[0])
        y.append(point[1])
    m,b = np.polyfit(x,y,1) #Generates the m,b values for the regression
    yp = np.polyval([m,b],x) #generates a set of y values based of the x-values and the m and b values
    ax.scatter(x, y) #Scatter plot of the data
    ax.plot(x,yp) #Plots the regression line
    ax.set_xlabel('x')
    ax.set_ylabel('y')
    ax.set_title(title)
```

Finally we call the function for each table in the database and assign each to it's own list.

```
In [7]: points1 = read_table('I')
points2 = read_table('II')
points3 = read_table('III')
points4 = read_table('IV')
```

Plotting the 4 data sets with regression lines.

```
In [40]: #fig allows you to put multiple plots in one resizable figure
fig, ((ax1, ax2), (ax3, ax4)) = plt.subplots(nrows=2, ncols=2)
plot_test(ax1, points1, 'Data Set I')
plot_test(ax2, points2, 'Data Set II')
plot_test(ax3, points3, 'Data Set III')
plot_test(ax4, points4, 'Data Set IV')
plt.tight_layout()
```

