Project 1

Step 1: Open the `sat_scores.csv` file. Investigate the data, and answer the questions below.

1. What does the data describe?

In [ ]:
```
This data describes the average Sat Verbal and Math scores by state.
```

2. Does the data look complete? Are there any obvious issues with the observations?

In [ ]:
```
It is not obvious what the state rate represents. What is it a measure of? Is it a SAT Completion Rate?
```

3. Create a data dictionary for the dataset.

In [110]:
```
import csv

with open("/Users/colinjclemence/Documents/DSI_SM_01/projects/01-projects-weekly/project-01/assets/sat_scores.csv", mode='r') as infile:
    reader = csv.reader(infile)
    #for row in reader:

    mydict = dict((rows[0],[rows[1],rows[2],rows[3]]) for rows in reader)
    print mydict
```

```
{'WA': ['53', '527', '527'], 'DE': ['67', '501', '499'], 'DC': ['56',
 '482', '474'], 'WI': ['6', '584', '596'], 'WV': ['18', '527', '512'],
 'State': ['Rate', 'Verbal', 'Math'], 'HI': ['52', '485', '515'], 'FL':
 ['54', '498', '499'], 'WY': ['11', '547', '545'], 'NH': ['72', '520',
 '516'], 'NJ': ['81', '499', '513'], 'NM': ['13', '551', '542'], 'TX':
 ['53', '493', '499'], 'LA': ['7', '564', '562'], 'NB': ['8', '562', '5
68'], 'NC': ['65', '493', '499'], 'ND': ['4', '592', '599'], 'TN': ['1
3', '562', '553'], 'NY': ['77', '495', '505'], 'PA': ['71', '500', '49
9'], 'RI': ['71', '501', '499'], 'NV': ['33', '509', '515'], 'VA': ['6
8', '510', '501'], 'CO': ['31', '539', '542'], 'AK': ['51', '514', '51
0'], 'AL': ['9', '559', '554'], 'AR': ['6', '562', '550'], 'VT': ['69',
 '511', '506'], 'IL': ['12', '576', '589'], 'GA': ['63', '491', '489'],
 'IN': ['60', '499', '501'], 'IA': ['5', '593', '603'], 'OK': ['8', '56
7', '561'], 'AZ': ['34', '523', '525'], 'CA': ['51', '498', '517'], 'I
D': ['17', '543', '542'], 'CT': ['82', '509', '510'], 'ME': ['69', '50
6', '500'], 'MD': ['65', '508', '510'], 'All': ['45', '506', '514'], 'M
A': ['79', '511', '515'], 'OH': ['26', '534', '439'], 'UT': ['5', '57
5', '570'], 'MO': ['8', '577', '577'], 'MN': ['9', '580', '589'], 'MI':
 ['11', '561', '572'], 'KS': ['9', '577', '580'], 'MT': ['23', '539',
 '539'], 'MS': ['4', '566', '551'], 'SC': ['57', '486', '488'], 'KY':
 ['12', '550', '550'], 'OR': ['55', '526', '526'], 'SD': ['4', '577',
 '582']}
```

Step 2: Load the data.

4. Load the data into a list of lists

In [99]:
```
import csv

data = []

with open("/Users/colinjclemence/Documents/DSI_SM_01/projects/01-projects-weekly/project-01/assets/sat_scores.csv",'r') as f:
    reader = csv.reader(f)
    for row in reader:
        data.append(row)
f.close()
```

5. Print the data

In [85]:
```
print data
```

```
[['State', 'Rate', 'Verbal', 'Math'], ['CT', '82', '509', '510'], ['N
J', '81', '499', '513'], ['MA', '79', '511', '515'], ['NY', '77', '49
5', '505'], ['NH', '72', '520', '516'], ['RI', '71', '501', '499'], ['P
A', '71', '500', '499'], ['VT', '69', '511', '506'], ['ME', '69', '50
6', '500'], ['VA', '68', '510', '501'], ['DE', '67', '501', '499'], ['M
D', '65', '508', '510'], ['NC', '65', '493', '499'], ['GA', '63', '49
1', '489'], ['IN', '60', '499', '501'], ['SC', '57', '486', '488'], ['D
C', '56', '482', '474'], ['OR', '55', '526', '526'], ['FL', '54', '49
8', '499'], ['WA', '53', '527', '527'], ['TX', '53', '493', '499'], ['H
I', '52', '485', '515'], ['AK', '51', '514', '510'], ['CA', '51', '49
8', '517'], ['AZ', '34', '523', '525'], ['NV', '33', '509', '515'], ['C
O', '31', '539', '542'], ['OH', '26', '534', '439'], ['MT', '23', '53
9', '539'], ['WV', '18', '527', '512'], ['ID', '17', '543', '542'], ['T
N', '13', '562', '553'], ['NM', '13', '551', '542'], ['IL', '12', '57
6', '589'], ['KY', '12', '550', '550'], ['WY', '11', '547', '545'], ['M
I', '11', '561', '572'], ['MN', '9', '580', '589'], ['KS', '9', '577',
 '580'], ['AL', '9', '559', '554'], ['NB', '8', '562', '568'], ['OK',
 '8', '567', '561'], ['MO', '8', '577', '577'], ['LA', '7', '564', '56
2'], ['WI', '6', '584', '596'], ['AR', '6', '562', '550'], ['UT', '5',
 '575', '570'], ['IA', '5', '593', '603'], ['SD', '4', '577', '582'],
 ['ND', '4', '592', '599'], ['MS', '4', '566', '551'], ['All', '45', '5
06', '514']]
```

6. Extract a list of the labels from the data, and remove them from the data.

In [42]:
```
#print data
label = data[0]
data.pop(0)
print label
```

```
['State', 'Rate', 'Verbal', 'Math']
```

7. Create a list of State names extracted from the data. (Hint: use the list of labels to index on the State column)

In [43]:
```
state_names = []
for i in data:
    state_names.append(i[0])
print state_names
```

```
['CT', 'NJ', 'MA', 'NY', 'NH', 'RI', 'PA', 'VT', 'ME', 'VA', 'DE', 'M
D', 'NC', 'GA', 'IN', 'SC', 'DC', 'OR', 'FL', 'WA', 'TX', 'HI', 'AK',
 'CA', 'AZ', 'NV', 'CO', 'OH', 'MT', 'WV', 'ID', 'TN', 'NM', 'IL', 'K
Y', 'WY', 'MI', 'MN', 'KS', 'AL', 'NB', 'OK', 'MO', 'LA', 'WI', 'AR',
 'UT', 'IA', 'SD', 'ND', 'MS', 'All']
```

8. Print the types of each column

In [44]:
```
for i in data[0]:
    print i + ": is of type: " + str(type(i))
```

```
CT: is of type: <type 'str'>
82: is of type: <type 'str'>
509: is of type: <type 'str'>
510: is of type: <type 'str'>
```

9. Do any types need to be reassigned? If so, go ahead and do it.

In [45]:
```
for i in data:
    i[1] = int(i[1])
    i[2] = int(i[2])
    i[3] = int(i[3])
print data
```

```
[['CT', 82, 509, 510], ['NJ', 81, 499, 513], ['MA', 79, 511, 515], ['N
Y', 77, 495, 505], ['NH', 72, 520, 516], ['RI', 71, 501, 499], ['PA', 7
1, 500, 499], ['VT', 69, 511, 506], ['ME', 69, 506, 500], ['VA', 68, 51
0, 501], ['DE', 67, 501, 499], ['MD', 65, 508, 510], ['NC', 65, 493, 49
9], ['GA', 63, 491, 489], ['IN', 60, 499, 501], ['SC', 57, 486, 488],
 ['DC', 56, 482, 474], ['OR', 55, 526, 526], ['FL', 54, 498, 499], ['W
A', 53, 527, 527], ['TX', 53, 493, 499], ['HI', 52, 485, 515], ['AK', 5
1, 514, 510], ['CA', 51, 498, 517], ['AZ', 34, 523, 525], ['NV', 33, 50
9, 515], ['CO', 31, 539, 542], ['OH', 26, 534, 439], ['MT', 23, 539, 53
9], ['WV', 18, 527, 512], ['ID', 17, 543, 542], ['TN', 13, 562, 553],
 ['NM', 13, 551, 542], ['IL', 12, 576, 589], ['KY', 12, 550, 550], ['W
Y', 11, 547, 545], ['MI', 11, 561, 572], ['MN', 9, 580, 589], ['KS', 9,
 577, 580], ['AL', 9, 559, 554], ['NB', 8, 562, 568], ['OK', 8, 567, 56
1], ['MO', 8, 577, 577], ['LA', 7, 564, 562], ['WI', 6, 584, 596], ['A
R', 6, 562, 550], ['UT', 5, 575, 570], ['IA', 5, 593, 603], ['SD', 4, 5
77, 582], ['ND', 4, 592, 599], ['MS', 4, 566, 551], ['All', 45, 506, 51
4]]
```

10. Create a dictionary for each column mapping the State to its respective value for that column.

In [46]:
```
curRate = []
curVerbal = []
curMath = []
for i in data:
    curRate.append({i[0] : i[1]})
    curVerbal.append({i[0] : i[2]})
    curMath.append({i[0] : i[3]})
rate = {label[1]: curRate}
verbal = {label[2]: curVerbal}
math = {label[3]: curMath}

output = {"Rate" : curRate,
          "Verbal" : curVerbal,
          "Math" : curMath}
print output
```

{'Rate': [{'CT': 82}, {'NJ': 81}, {'MA': 79}, {'NY': 77}, {'NH': 72},
 {'RI': 71}, {'PA': 71}, {'VT': 69}, {'ME': 69}, {'VA': 68}, {'DE': 6
7}, {'MD': 65}, {'NC': 65}, {'GA': 63}, {'IN': 60}, {'SC': 57}, {'DC':
 56}, {'OR': 55}, {'FL': 54}, {'WA': 53}, {'TX': 53}, {'HI': 52}, {'A
K': 51}, {'CA': 51}, {'AZ': 34}, {'NV': 33}, {'CO': 31}, {'OH': 26},
 {'MT': 23}, {'WV': 18}, {'ID': 17}, {'TN': 13}, {'NM': 13}, {'IL': 1
2}, {'KY': 12}, {'WY': 11}, {'MI': 11}, {'MN': 9}, {'KS': 9}, {'AL':
 9}, {'NB': 8}, {'OK': 8}, {'MO': 8}, {'LA': 7}, {'WI': 6}, {'AR': 6},
 {'UT': 5}, {'IA': 5}, {'SD': 4}, {'ND': 4}, {'MS': 4}, {'All': 45}],
 'Math': [{'CT': 510}, {'NJ': 513}, {'MA': 515}, {'NY': 505}, {'NH': 51
6}, {'RI': 499}, {'PA': 499}, {'VT': 506}, {'ME': 500}, {'VA': 501},
 {'DE': 499}, {'MD': 510}, {'NC': 499}, {'GA': 489}, {'IN': 501}, {'S
C': 488}, {'DC': 474}, {'OR': 526}, {'FL': 499}, {'WA': 527}, {'TX': 49
9}, {'HI': 515}, {'AK': 510}, {'CA': 517}, {'AZ': 525}, {'NV': 515},
 {'CO': 542}, {'OH': 439}, {'MT': 539}, {'WV': 512}, {'ID': 542}, {'T
N': 553}, {'NM': 542}, {'IL': 589}, {'KY': 550}, {'WY': 545}, {'MI': 57
2}, {'MN': 589}, {'KS': 580}, {'AL': 554}, {'NB': 568}, {'OK': 561},
 {'MO': 577}, {'LA': 562}, {'WI': 596}, {'AR': 550}, {'UT': 570}, {'I
A': 603}, {'SD': 582}, {'ND': 599}, {'MS': 551}, {'All': 514}], 'Verba
l': [{'CT': 509}, {'NJ': 499}, {'MA': 511}, {'NY': 495}, {'NH': 520},
 {'RI': 501}, {'PA': 500}, {'VT': 511}, {'ME': 506}, {'VA': 510}, {'D
E': 501}, {'MD': 508}, {'NC': 493}, {'GA': 491}, {'IN': 499}, {'SC': 48
6}, {'DC': 482}, {'OR': 526}, {'FL': 498}, {'WA': 527}, {'TX': 493},
 {'HI': 485}, {'AK': 514}, {'CA': 498}, {'AZ': 523}, {'NV': 509}, {'C
O': 539}, {'OH': 534}, {'MT': 539}, {'WV': 527}, {'ID': 543}, {'TN': 56
2}, {'NM': 551}, {'IL': 576}, {'KY': 550}, {'WY': 547}, {'MI': 561},
 {'MN': 580}, {'KS': 577}, {'AL': 559}, {'NB': 562}, {'OK': 567}, {'M
O': 577}, {'LA': 564}, {'WI': 584}, {'AR': 562}, {'UT': 575}, {'IA': 59
3}, {'SD': 577}, {'ND': 592}, {'MS': 566}, {'All': 506}]}

11. Create a dictionary with the values for each of the numeric columns

In [47]:
```
curRate = []
curVerbal = []
curMath = []
for i in data:
    curRate.append(i[1])
    curVerbal.append(i[2])
    curMath.append(i[3])
rate = {label[1]: curRate}
verbal = {label[2]: curVerbal}
math = {label[3]: curMath}

output = {"Rate" : curRate,
          "Verbal" : curVerbal,
          "Math" : curMath}
print output
```

```
{'Rate': [82, 81, 79, 77, 72, 71, 71, 69, 69, 68, 67, 65, 65, 63, 60, 5
7, 56, 55, 54, 53, 53, 52, 51, 51, 34, 33, 31, 26, 23, 18, 17, 13, 13,
 12, 12, 11, 11, 9, 9, 9, 8, 8, 8, 7, 6, 6, 5, 5, 4, 4, 4, 45], 'Math':
 [510, 513, 515, 505, 516, 499, 499, 506, 500, 501, 499, 510, 499, 489,
 501, 488, 474, 526, 499, 527, 499, 515, 510, 517, 525, 515, 542, 439,
 539, 512, 542, 553, 542, 589, 550, 545, 572, 589, 580, 554, 568, 561,
 577, 562, 596, 550, 570, 603, 582, 599, 551, 514], 'Verbal': [509, 49
9, 511, 495, 520, 501, 500, 511, 506, 510, 501, 508, 493, 491, 499, 48
6, 482, 526, 498, 527, 493, 485, 514, 498, 523, 509, 539, 534, 539, 52
7, 543, 562, 551, 576, 550, 547, 561, 580, 577, 559, 562, 567, 577, 56
4, 584, 562, 575, 593, 577, 592, 566, 506]}
```

Step 3: Describe the data

12. Print the min and max of each column

In [48]:
```
print "Max Rate: " + str(max(curRate))
print "Min Rate: " + str(min(curRate))
print "Max Verbal: " + str(max(curVerbal))
print "Min Verbal: " + str(min(curVerbal))
print "Max Math: " + str(max(curMath))
print "Min Math: " + str(min(curMath))
```

```
Max Rate: 82
Min Rate: 4
Max Verbal: 593
Min Verbal: 482
Max Math: 603
Min Math: 439
```

13. Write a function using only list comprehensions, no loops, to compute Standard Deviation. Print the Standard Deviation of each numeric column.

In [50]:
```
#standard_deviation = sqrt(mean(abs(x - x.mean())**2))
import math

def standard_deviation(curList):
    return math.sqrt(
        sum(
            [x for x in [
                    abs(x-(sum([x for x in curList])/len(curList)))**2 f
or x in curList]])
        /len(curList))

sdRate = standard_deviation(curRate)
sdVerbal = standard_deviation(curVerbal)
sdMath = standard_deviation(curMath)
print "Rate SD: " + str(sdRate) + ", Verbal SD: " + str(sdVerbal) + ", M
ath SD: " + str(sdMath)
```

```
Rate SD: 27.0370116692, Verbal SD: 32.9089653438, Math SD: 35.665109000
3
```

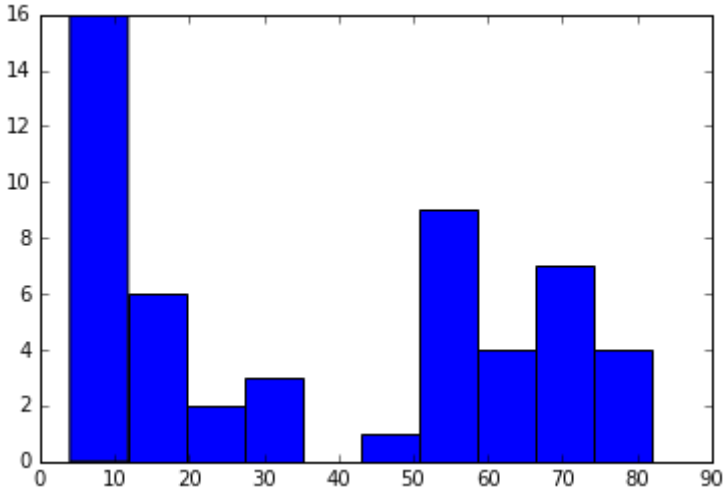
Step 4: Visualize the data

14. Using MatPlotLib and PyPlot, plot the distribution of the Rate using histograms.

In [62]:
```
import numpy as np
import matplotlib
import matplotlib.pyplot as plt
%matplotlib inline

plt.hist(curRate)
```

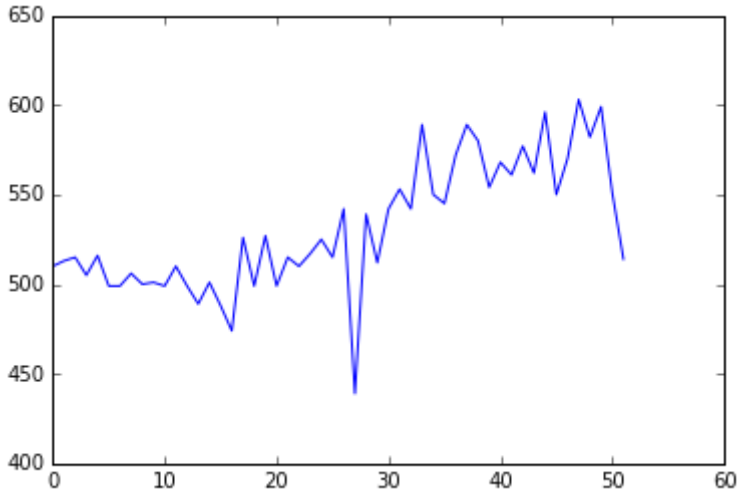
Out[62]:
```
(array([ 16.,   6.,   2.,   3.,   0.,   1.,   9.,   4.,   7.,   4.]),
 array([  4. ,  11.8,  19.6,  27.4,  35.2,  43. ,  50.8,  58.6,  66.4,
         74.2,  82. ]),
 <a list of 10 Patch objects>)
```



15. Plot the Math distribution

In [60]:
```
plt.plot(curMath)
```

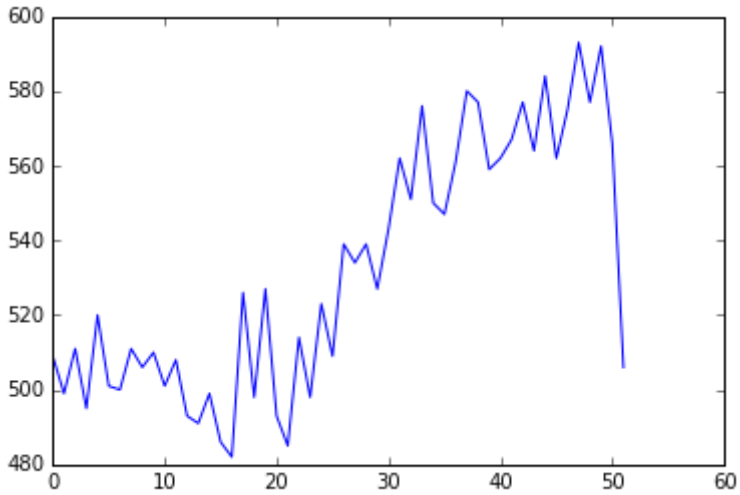
Out[60]:
```
[<matplotlib.lines.Line2D at 0x110052c10>]
```



16. Plot the Verbal distribution

In [61]:
```
plt.plot(curVerbal)
```

Out[61]:
```
[<matplotlib.lines.Line2D at 0x1100fd4d0>]
```



17. What is the typical assumption for data distribution?

In [ ]:
```
Typical assumption is that the data is normal.
```

18. Does that distribution hold true for our data?

In [ ]:
```
This is not true with our data as we have a positive skew on both math and verbal scores.
```

19. Plot some scatterplots. BONUS: Use a PyPlot `figure` to present multiple plots at once.

In [ ]:
```

```

20. Are there any interesting relationships to note?

In [ ]:
```

```

21. Create box plots for each variable.

In [ ]:
```

```

BONUS: Using Tableau, create a heat map for each variable using a map of the US.

In [ ]: