Fall 2021: CS 4435/5435 and DASE 4435 Data Mining Homework 1

Due on September 16, 2021

David Vadnais

NOTE: The questions marked with *asterisk is for graduate students only. Undergraduates are free to attempt it as a bonus point. However, the bonus is awarded only if the question is answered correctly. Also, note that poor presentation of plots will lead to point detection. A good plot would include, plot legend, title, axis labels and proper figure labelling

A. Explain the difference and similarity between discrimination and classification, between characterization and clustering, and between classification and regression. Give one example to describe your point. Please don't use the examples in the lecture slides. (10 points)

- discrimination and classification
  - Similarity
    - Measure nominal data
  - Difference
    - Comparing columns in a data set vs comparing groups of a data set
  - Examples
    - Discrimination are you wearing pants?
    - Classification what type pf leg wear are you wearing
- characterization and clustering
  - Similarity
    - Collecting groups of data together
  - Difference
    - Characterization represents data in an easier to understand form cluistering groups data into pools of similar attributes
  - Examples
    - Standard deviation on a graph vs three groups of unkown meaning
- classification and regression
  - Similarity
    - Both supervised learning
  - Difference
    - Regression predict rea numbers while classification groups
  - Examples
    - Regression example predicting house value on market.
    - Classification predicting wealth bracket by house value.

B. Write a program in R, python or other programming tools to solve the following. Provide the code as a separate attachment and comment your code appropriately.

a. Suppose that the data for analysis includes the attribute age. The age values for the data tuples are (in increasing order) 13, 15, 16, 16, 19, 20, 20, 21, 22, 22, 25, 25, 25, 25, 30, 33, 33, 35, 35, 35, 35, 36, 40, 45,46, 52, 70. (25 points)

Note: Don't use programming library for questions a to c below.

i. What is the mean, median, and mode of the data?

1. Mean 29.96296
2. Median 25
3. Mode 25 and 35

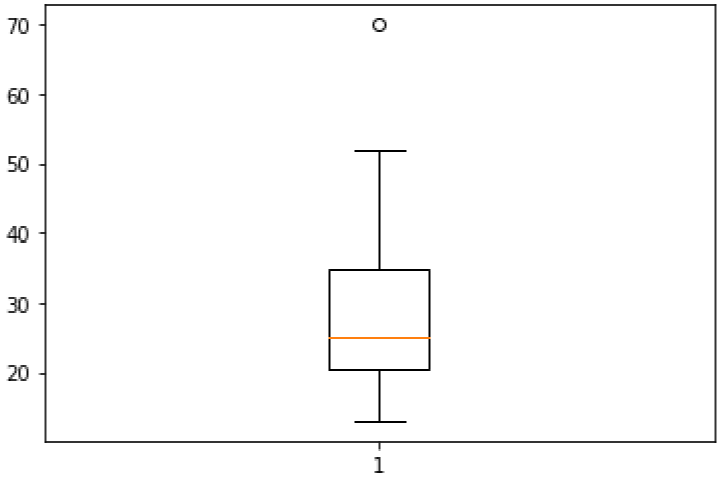
ii. Can you find (roughly) the first quartile (Q1) and the third quartile (Q3) of the data?

1. Q1 21
2. Q3 36

iii. Give the five-number summary of the data.

1. Q1 21
2. Q3 36
3. Median 25
4. Min 13
5. Max 70

iv. Show a boxplot of the data. *

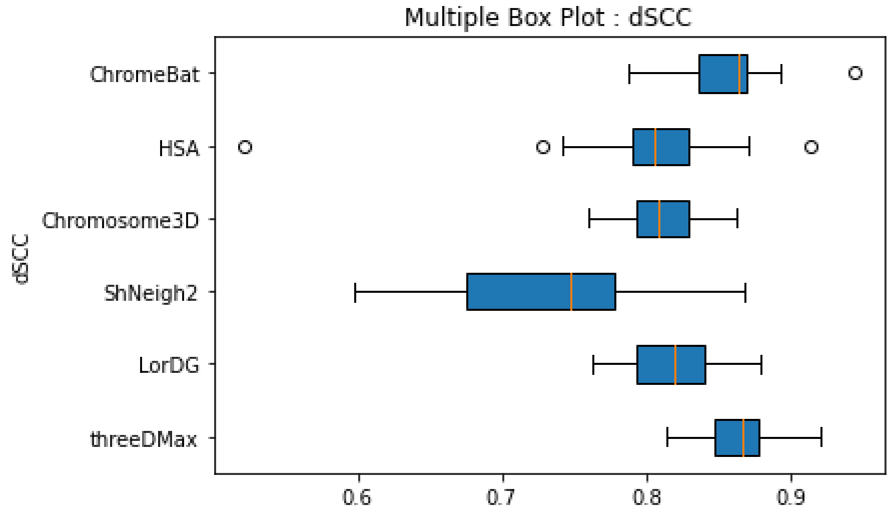


b. The data here contains the Spearman Correlation Coefficient (SCC) result score obtained by accessing the three-dimensional(3D) structures of Chromosome 1 to 23 generated by different 3D chromosome reconstruction methods−3DMax, LorDG, ShNeigh2, Chromosome3D, HSA, and ChromeBat− for a GM06990 cell line. GM06990 is a cell line derived from some human lymphoblastoid cells. SCC scores are in the range -1 to 1, the closer to 1 the better. Using this data (45 points):

i. Calculate the mean, median, and standard deviation* of the SCC scores. Note: Don't use programming library for this question.

	threeDMax	LorDG	Shneigh2	Chromosome3d	hsa	chromebat
Mean	0.8613681570869566	0.8200192904347827	0.7317263030869566	0.7994456550869565	0.8126086956521739	0.8569479670434784
Median	0.866727134	0.819727285	0.746826564	0.805482936	0.808	0.864169052
std	0.02448753816919791,	0.0350287480669858 74, ,	0.07850725710464285,	0.07196208463616	0.0322172416815210 55	0.0282407955461825 53

ii. Show a boxplot for of the SCC data. Note: show the boxplot side-by-side in the same figure not separately.



iii. Comment on the box plot results:

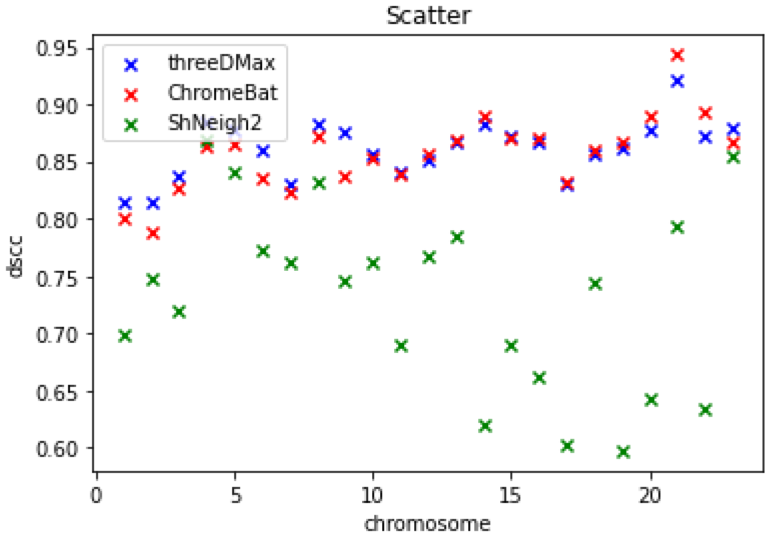
1. Which method(s) would you regard as the best performing ones, why?

ThreedMax and ChromeBat are the best because they have low variance and the highest median. Also, there mins are better.

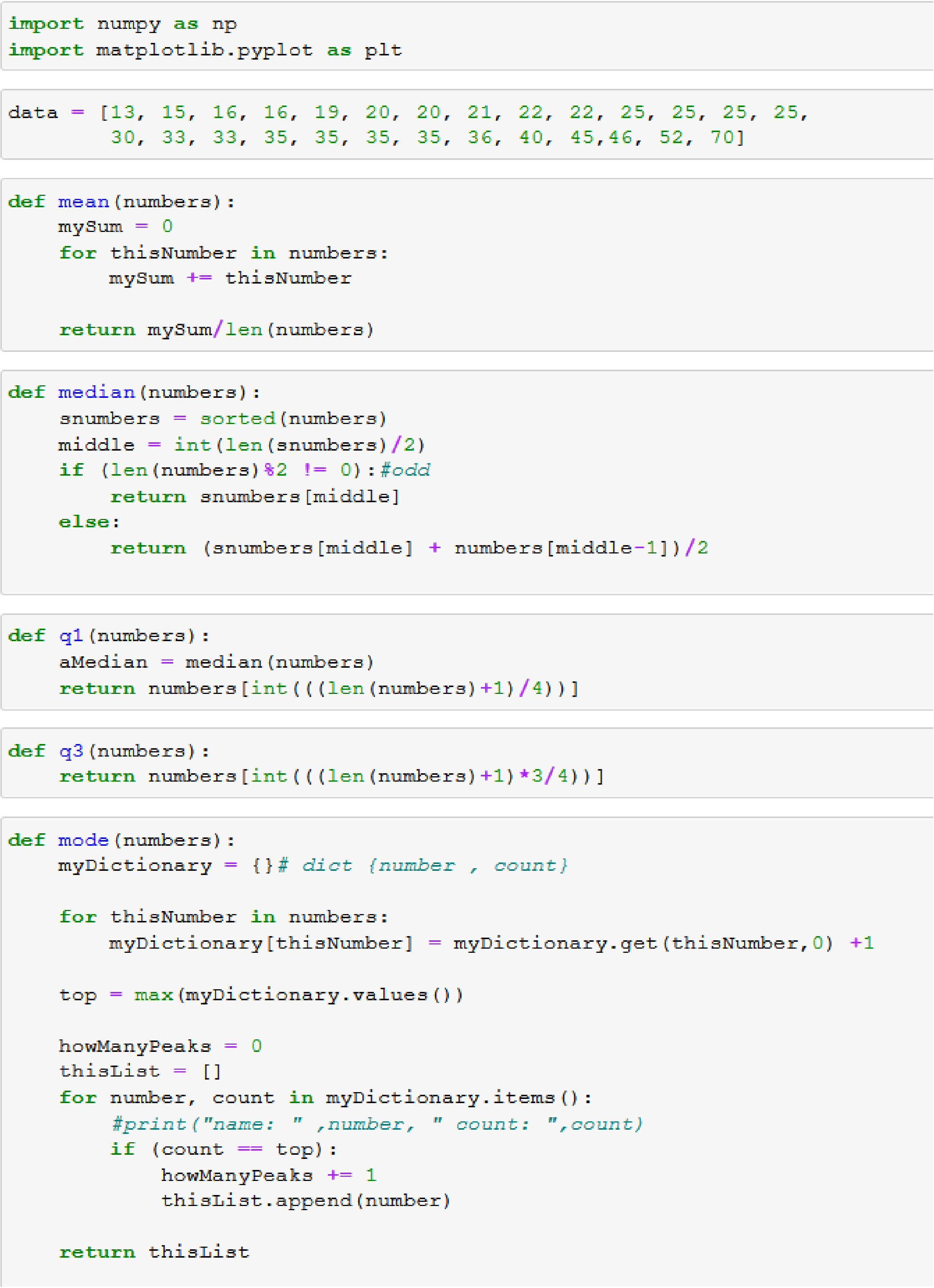
2. Which method(s) would you regard as the least performing ones, why?

ShNeight has the lowest median, and a very low min. It is extremely variable but mostly in a bad way.

iv. Using a Scatter plot of any three methods' result, what is the pattern of performance you can observe from these methods' results on some chromosome(s)? Provide the scatter plots in your report as well.



Code for question 1 (10 points)



```python
import numpy as np
import matplotlib.pyplot as plt
```

```python
data = [13, 15, 16, 16, 19, 20, 20, 21, 22, 22, 25, 25, 25, 25,
        30, 33, 33, 35, 35, 35, 35, 36, 40, 45,46, 52, 70]
```

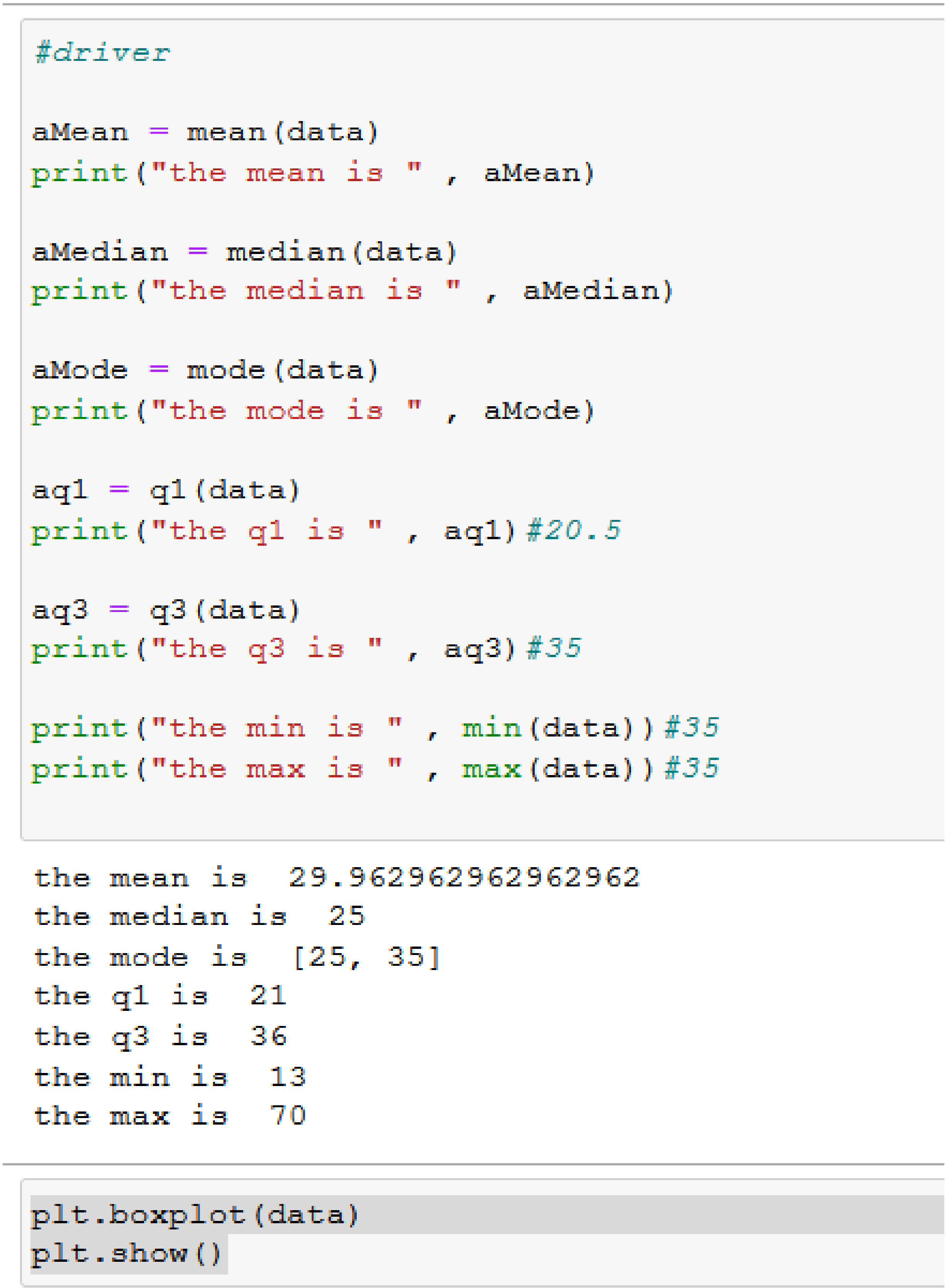
```python
def mean(numbers):
    mySum = 0
    for thisNumber in numbers:
        mySum += thisNumber

    return mySum/len(numbers)
```

```python
def median(numbers):
    snumbers = sorted(numbers)
    middle = int(len(snumbers)/2)
    if (len(numbers)%2 != 0):#odd
        return snumbers[middle]
    else:
        return (snumbers[middle] + numbers[middle-1])/2
```

```python
def q1(numbers):
    aMedian = median(numbers)
    return numbers[int(((len(numbers)+1)/4))]
```

```python
def q3(numbers):
    return numbers[int(((len(numbers)+1)*3/4))]
```

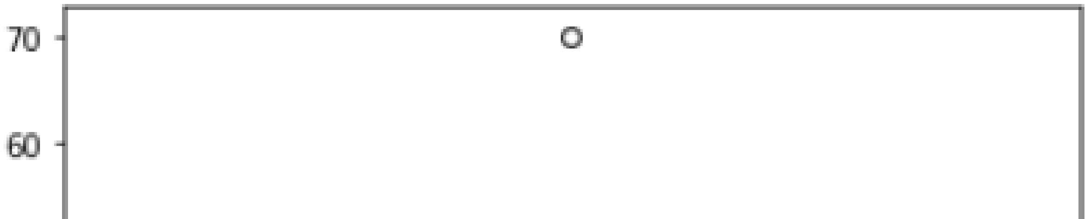
```python
def mode(numbers):
    myDictionary = {}# dict {number , count}

    for thisNumber in numbers:
        myDictionary[thisNumber] = myDictionary.get(thisNumber,0) +1

    top = max(myDictionary.values())

    howManyPeaks = 0
    thisList = []
    for number, count in myDictionary.items():
        #print("name: " ,number, " count: ",count)
        if (count == top):
            howManyPeaks += 1
            thisList.append(number)

    return thisList
```



```
#driver

aMean = mean(data)
print("the mean is " , aMean)

aMedian = median(data)
print("the median is " , aMedian)

aMode = mode(data)
print("the mode is " , aMode)

aq1 = q1(data)
print("the q1 is " , aq1)#20.5

aq3 = q3(data)
print("the q3 is " , aq3)#35

print("the min is " , min(data))#35
print("the max is " , max(data))#35
```

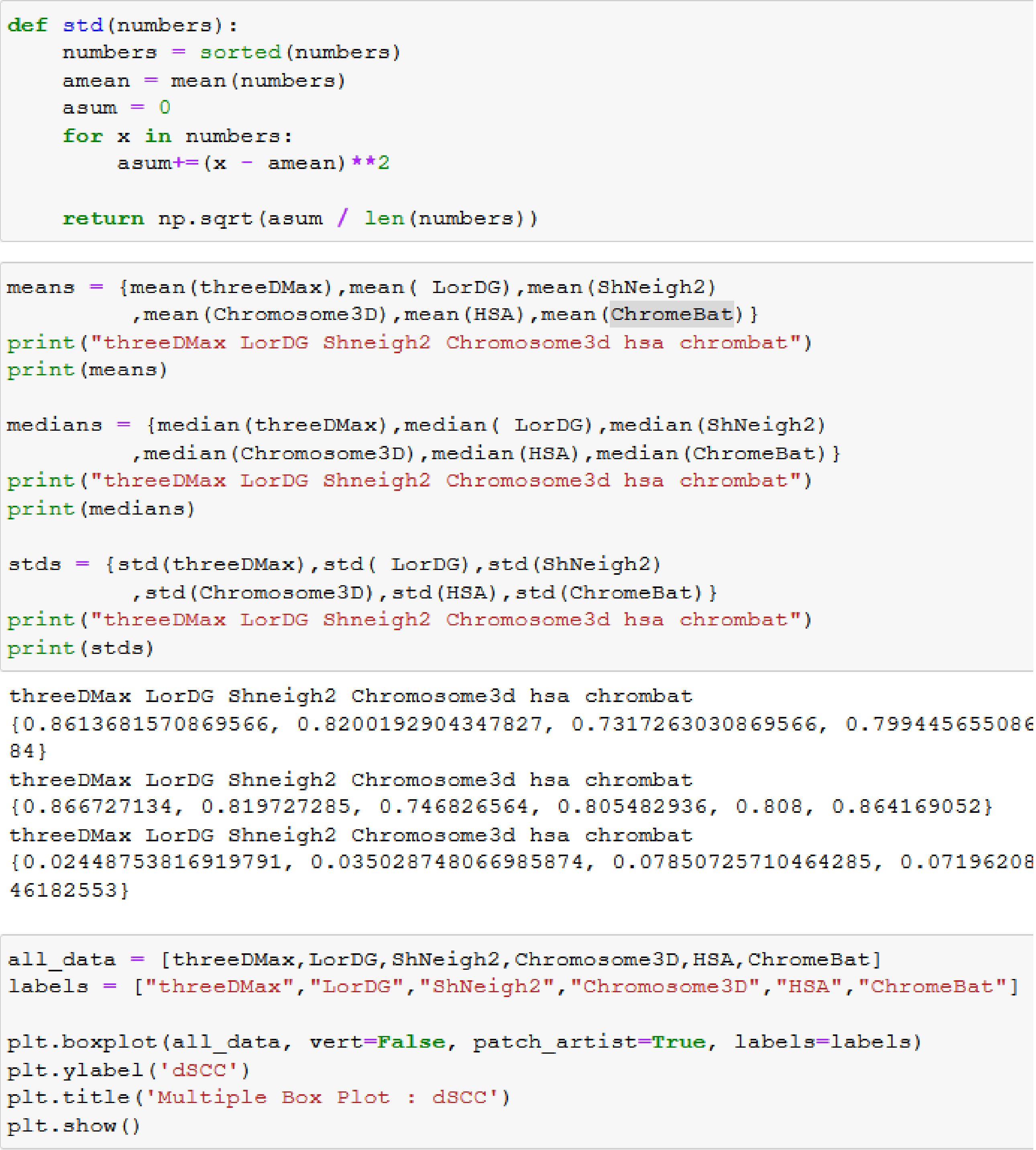
```
the mean is  29.962962962962962
the median is  25
the mode is  [25, 35]
the q1 is  21
the q3 is  36
the min is  13
the max is  70
```

```
plt.boxplot(data)
plt.show()
```



Code for question 2 (10 points)

```
threeDMax = [0.814102492,0.813621567,0.837803087,0.884940199,0.877041883,
0.859534182,0.830665735,0.881768757,0.87626118,0.856736255,0.841230581,0.85
1892957,0.867383432,0.881965904,
0.871448337,0.866727134,0.83017034,0.856242545,0.861872089,0.877643617,0.92
0786384,0.872886391,0.878742565]
LorDG =[0.777194373,0.777538274,0.770454362,0.878532687,0.862151096,
0.829818105,0.798790212,0.861752213,0.840972832,0.824618349,0.808859457,0.8
19727285,0.840077056,0.813496136,
0.830323238,0.788029869,0.766565944,0.817562874,0.815141153,0.82283662,0.87
8305902,0.763239881,0.874455762]

ShNeigh2 = [0.698225414,0.747572568,0.720008492,0.86791376,0.841163842,
0.772470936,0.76254506,0.831940359,0.746826564,0.761287418,0.69028024,0.767
227128,0.78372401,0.620457587,
0.689074742,0.662148418,0.602634335,0.744066759,0.597254155,0.642785561,0.7
92785159,0.633150342,0.854162122]

Chromosome3D = [0.766,0.777,0.795,0.861,0.852,0.826,0.788,0.852,0.803,0.825,
0.808,0.83,0.791,0.8,0.807,0.788,0.76,0.815,0.827,0.843,0.863,0.809,0.804]
HSA=[0.741951816,0.754669462,0.78290956,0.871315788,0.851475568,
0.813031772,0.788436774,0.853537768,0.521889059,0.804888328,0.791621627,
0.805482936,0.822652737,0.799115946,0.80661521,0.806632018,0.72770912,
0.800800322,0.802493463,0.828791598,0.913868944,0.830617256,0.866742995]
ChromeBat=[0.801017424,0.78773319,0.826161388,0.864169052,0.864943758,
0.835211755,0.823470031,0.871247459,0.837529292,0.852688073,0.838961086,
0.855582767,0.868645744,0.889928286,0.86966815,0.870228275,0.831012149,
0.860491375,0.866458315,0.890406712,0.943462868,0.893232298,0.867553795]
```

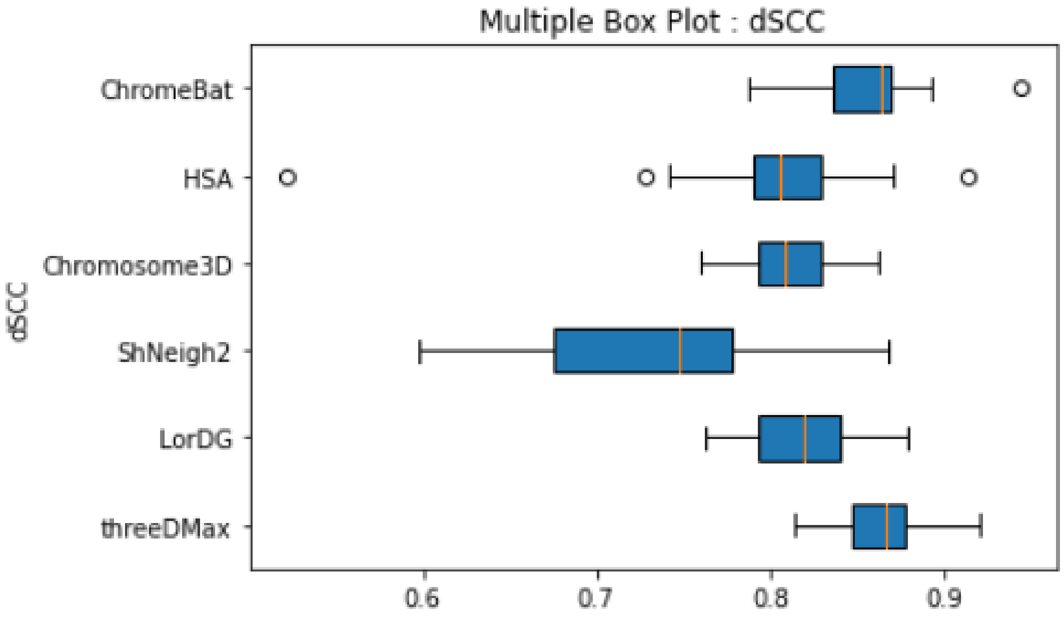


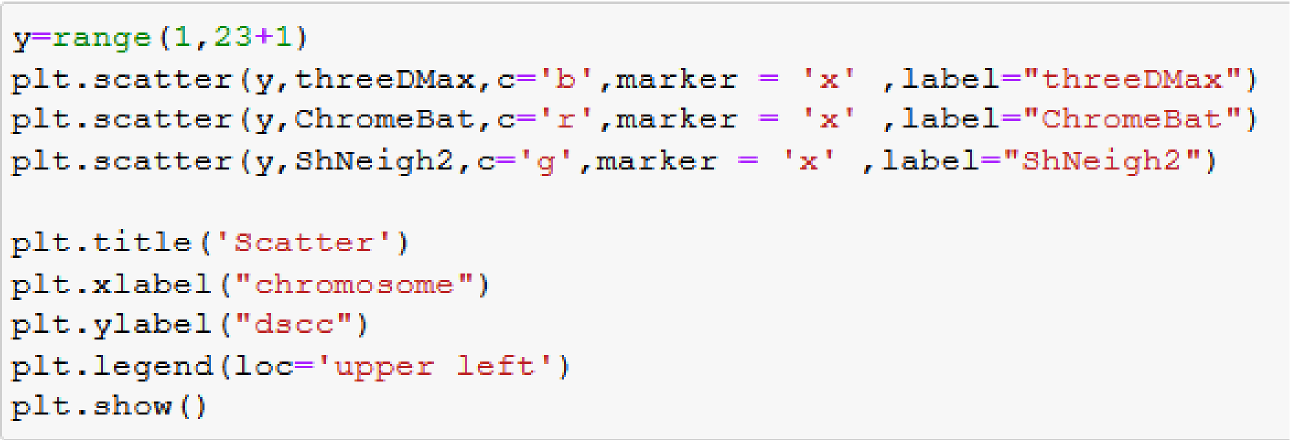
```
def std(numbers):
    numbers = sorted(numbers)
    amean = mean(numbers)
    asum = 0
    for x in numbers:
        asum+=(x - amean)**2

    return np.sqrt(asum / len(numbers))
```

```
means = {mean(threeDMax),mean( LorDG),mean(ShNeigh2)
         ,mean(Chromosome3D),mean(HSA),mean(ChromeBat)}
print("threeDMax LorDG Shneigh2 Chromosome3d hsa chrombat")
print(means)

medians = {median(threeDMax),median( LorDG),median(ShNeigh2)
         ,median(Chromosome3D),median(HSA),median(ChromeBat)}
print("threeDMax LorDG Shneigh2 Chromosome3d hsa chrombat")
print(medians)

stds = {std(threeDMax),std( LorDG),std(ShNeigh2)
         ,std(Chromosome3D),std(HSA),std(ChromeBat)}
print("threeDMax LorDG Shneigh2 Chromosome3d hsa chrombat")
print(stds)
```

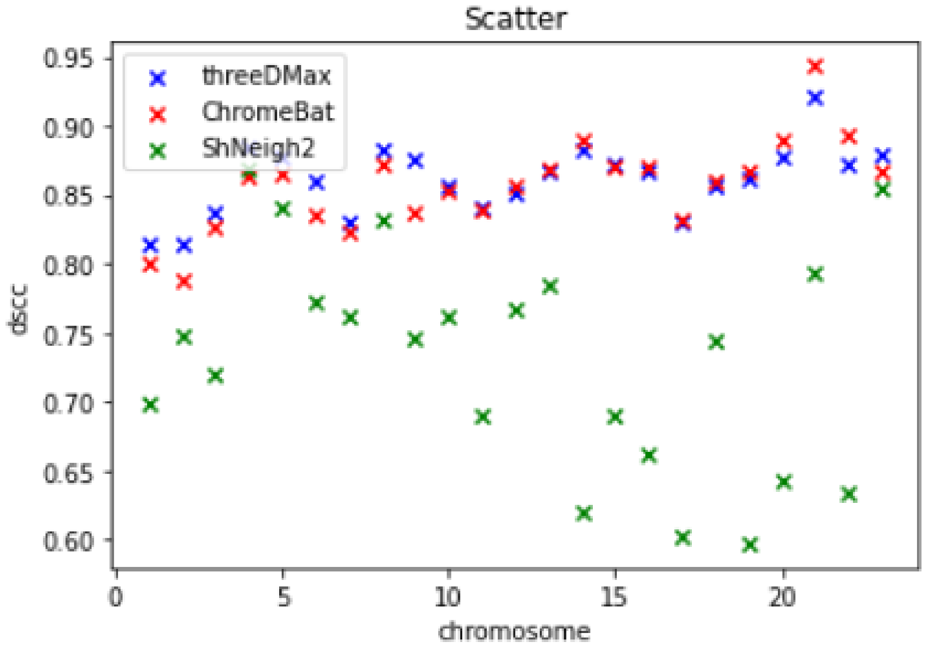
```
threeDMax LorDG Shneigh2 Chromosome3d hsa chrombat
{0.8613681570869566, 0.8200192904347827, 0.7317263030869566, 0.799445655086
84}
threeDMax LorDG Shneigh2 Chromosome3d hsa chrombat
{0.866727134, 0.819727285, 0.746826564, 0.805482936, 0.808, 0.864169052}
threeDMax LorDG Shneigh2 Chromosome3d hsa chrombat
{0.02448753816919791, 0.035028748066985874, 0.07850725710464285, 0.07196208
46182553}
```

```
all_data = [threeDMax,LorDG,ShNeigh2,Chromosome3D,HSA,ChromeBat]
labels = ["threeDMax","LorDG","ShNeigh2","Chromosome3D","HSA","ChromeBat"]

plt.boxplot(all_data, vert=False, patch_artist=True, labels=labels)
plt.ylabel('dSCC')
plt.title('Multiple Box Plot : dSCC')
plt.show()
```





```
y=range(1,23+1)
plt.scatter(y,threeDMax,c='b',marker = 'x' ,label="threeDMax")
plt.scatter(y,ChromeBat,c='r',marker = 'x' ,label="ChromeBat")
plt.scatter(y,ShNeigh2,c='g',marker = 'x' ,label="ShNeigh2")

plt.title('Scatter')
plt.xlabel("chromosome")
plt.ylabel("dscc")
plt.legend(loc='upper left')
plt.show()
```


Scatter
threeDMax
ChromeBat
ShNeigh2
dscc
chromosome