Homework #3

Assign date: 2015-06-21

Due date: 2015-07-01, 6pm

Submission:

1. **Please submit your results in email to the grader: 130301039@svuca.edu Chi Zhang**
2. **Please separate the written answers from the python code: you should submit 2 files in your email – cs596-29-hw2_yourID#.doc & cs596-29-hw2_yourID#.py**
3. **30 pts per day will be deducted for late submission**

Problem 1) K-means clustering (20 pts.)

You are given the following 10 data points of height & weight:

ID	1	2	3	4	5	6	7	8	9	10
Height	66	73	72	70	74	68	65	64	63	67
Weight	170	210	165	180	185	155	150	120	125	140

Manually apply k-means algorithms to get 2 clusters. Please produce the center and grouping step by step, using the following parameters:

a) Initialize with ID=1 and ID=2
b) Assume Euclidean distance

Solution

In each iteration (EPOCH), we will calculate the Euclidean distance matrix from the 2 seeds to each node. The distance matrix will be used for comparison and inferring if a node belongs to Cluster 1 (represented by Seed 1) or Cluster 2 (represented by Seed 2), the result is recorded in last line of each matrix. Base on the result, new seeds will be calculated and used for next iteration until the Clusters do not change.

EPOCH 1

Seed 1: [66, 170] ∈ C1
Seed 2: [73, 210] ∈ C2
Distance between 2 seeds to each element:

	ID=1	ID=2	ID=3	ID=4	ID=5	ID=6	ID=7	ID=8	ID=9	ID=10
Seed-1	0	40.6	7.8	10.8	17	15.1	20.0	50.0	45.1	30.0
Seed-2	40.6	0	45.0	30.1	25.0	55.2	60.5	90.5	85.6	70.3
Cluster	C1	C2	C1	C1	C1	C1	C1	C1	C1	C1

-> Base on above result of new clusterings, new seeds (mean) are computed: [67.66666666666667, 154.44444444444446], and [73.0, 210.0]

EPOCH 2

Seed 1: [67.66666666666667, 154.44444444444446] ∈ C1
Seed 2: [73.0, 210.0] ∈ C2
Distance between 2 seeds to each element:

	ID=1	ID=2	ID=3	ID=4	ID=5	ID=6	ID=7	ID=8	ID=9	ID=10
Seed-1	15.6	55.8	11.4	25.7	31.2	0.7	5.2	34.6	29.8	14.5
Seed-2	40.6	0	45.0	30.1	25.0	55.2	60.5	90.5	85.6	70.3
Cluster	C1	C2	C1	C1	C2	C1	C1	C1	C1	C1

-> Base on above result of new clusterings, new seeds (mean) are computed: [66.875, 150.625], and [73.5, 197.5]

EPOCH 3

Seed 1: [66.875, 150.625] ∈ C1
Seed 2: [73.5, 197.5] ∈ C2
Distance between 2 seeds to each element:

	ID=1	ID=2	ID=3	ID=4	ID=5	ID=6	ID=7	ID=8	ID=9	ID=10
Seed-1	19.4	59.7	15.3	29.5	35.1	4.5	2.0	30.8	25.9	10.6
Seed-2	28.5	12.5	32.5	17.9	12.5	42.9	48.2	78.1	73.3	57.9
Cluster	C1	C2	C1	C2	C2	C1	C1	C1	C1	C1

-> Base on above result of new clusterings, new seeds (mean) are computed: [66.42857142857143, 146.42857142857142], and [72.33333333333333, 191.66666666666666]

EPOCH 4

Seed 1: [66.42857142857143, 146.42857142857142] ∈ C1
Seed 2: [72.33333333333333, 191.66666666666666] ∈ C2
Distance between 2 seeds to each element:

	ID=1	ID=2	ID=3	ID=4	ID=5	ID=6	ID=7	ID=8	ID=9	ID=10
Seed-1	23.6	63.9	19.4	33.8	39.3	8.7	3.9	26.5	21.7	6.5
Seed-2	22.6	18.4	26.7	11.9	6.9	36.9	42.3	72.2	67.3	51.9
Cluster	C2	C2	C1	C2	C2	C1	C1	C1	C1	C1

-> Base on above result of new clusterings, new seeds (mean) are computed: [66.5, 142.5], and [70.75, 186.25]

EPOCH 5

Seed 1: [66.5, 142.5] ∈ C1
Seed 2: [70.75, 186.25] ∈ C2
Distance between 2 seeds to each element:

	ID=1	ID=2	ID=3	ID=4	ID=5	ID=6	ID=7	ID=8	ID=9	ID=10

Seed-1	27.5	67.8	23.2	37.7	43.2	12.6	7.7	22.6	17.9	2.5
Seed-2	16.9	23.9	21.3	6.3	3.5	31.4	36.7	66.6	61.7	46.4
Cluster	C2	C2	C2	C2	C2	C1	C1	C1	C1	C1

-> Base on above result of new clusterings, new seeds (mean) are computed: [65.4, 138.0], and [71.0, 182.0]

EPOCH 6

Seed 1: [65.4, 138.0] ∈ C1
Seed 2: [71.0, 182.0] ∈ C2
Distance between 2 seeds to each element:

	ID=1	ID=2	ID=3	ID=4	ID=5	ID=6	ID=7	ID=8	ID=9	ID=10
Seed-1	32.0	72.4	27.8	42.2	47.8	17.2	12.0	18.1	13.2	2.6
Seed-2	13	28.1	17.0	2.2	4.2	27.2	32.6	62.4	57.6	42.2
Cluster	C2	C2	C2	C2	C2	C1	C1	C1	C1	C1

-> Base on above result of new clusterings, new seeds (mean) are computed: [65.4, 138.0], and [71.0, 182.0]

After we **Epoch 6**, the Cluster members do not change anymore, so we know Epoch 5/6 has the final result with 2 clusters:

- **Cluster 1: Node {6, 7, 8, 9, 10}**
- **Cluster 2: Node {1, 2, 3, 4, 5}**

Problem 2) Agglomerative clustering (20 pts.)
Using the same 10 data points of height & weight as in Problem-1, apply agglomerative clustering manually. Produce the distance matrix and the resulting dendogram. Use the following assumptions:

a) Euclidean distance
b) Single linkage for cluster dissimilarity

Solution

Agglomerative: initially every point is a cluster of its own and we merge cluster until we end-up with one unique cluster containing all points.

Single link: distance between two clusters is the shortest distance between a pair of elements from the two clusters.

For each k – we recalculate the distance matrix, and merge 2 nodes into 1 node (representing new cluster).

We start with k=10, i.e. each point is a cluster.

k=10

0	40.6 0788 1008 5	7.81 0249 6759 1	10.7 7032 9614 3	17.0	15.1 3274 5950 4	20.0 2498 4394 5	50.0 3998 4012 8	45.0 9988 9135 1	30.0 1666 2039 6
40.6 08	0	45.0 11	30.1 50	25.0 20	55.2 27	60.5 31	90.4 49	85.5 86	70.2 57
7.81 0	45.0 11	0	15.1 33	20.1 00	10.7 70	16.5 53	45.7 06	41	25.4 95
10.7 70	30.1 50	15.1 33	0	6.40 3	25.0 80	30.4 14	60.2 99	55.4 44	40.1 12
17	25.0 20	20.1 00	6.40 3	0	30.5 94	36.1 39	65.7 65	61	45.5 41
15.1 33	55.2 27	10.7 70	25.0 80	30.5 94	0	5.83 1	35.2 28	30.4 14	15.0 33
20.0 25	60.5 31	16.5 53	30.4 14	36.1 39	5.83 1	0	30.0 17	25.0 80	10.1 98
50.0 40	90.4 49	45.7 06	60.2 99	65.7 65	35.2 28	30.0 17	0	5.09 9	20.2 24
45.1 00	85.5 86	41	55.4 44	61	30.4 14	25.0 80	5.09 9	0	15.5 24
30.0 17	70.2 57	25.4 95	40.1 12	45.5 41	15.0 33	10.1 98	20.2 24	15.5 24	0

k=9

0	40.60 78810 085	7.810 24967 591	10.77 03296 143	17.0	15.13 27459 504	20.02 49843 945	61.72 69089 274	30.01 66620 396
40.60 8	0	45.01 1	30.15 0	25.02 0	55.22 7	60.53 1	115.6 57	70.25 7
7.810	45.01 1	0	15.13 3	20.10 0	10.77 0	16.55 3	56.10 4	25.49 5
10.77 0	30.15 0	15.13 3	0	6.403	25.08 0	30.41 4	75.46 2	40.11 2
17	25.02 0	20.10 0	6.403	0	30.59 4	36.13 9	82.81 0	45.54 1
15.13 3	55.22 7	10.77 0	25.08 0	30.59 4	0	5.831	42.06 1	15.03 3
20.02 5	60.53 1	16.55 3	30.41 4	36.13 9	5.831	0	35.03 1	10.19 8
61.72 7	115.6 57	56.10 4	75.46 2	82.81 0	42.06 1	35.03 1	0	22.13 2
30.01	70.25	25.49	40.11	45.54	15.03	10.19	22.13	0

7	7	5	2	1	3	8	2	

k=8

0	40.607881	7.8102496	10.770329	17.0	21.494265	61.726908	30.016620
	40.608	0	45.011	30.150	25.020	75.228	115.657	70.257
	7.810	45.011	0	15.133	20.100	16.272	56.104	25.495
	10.770	30.150	15.133	0	6.403	35.052	75.462	40.112
	17	25.020	20.100	6.403	0	42.545	82.810	45.541
	21.495	75.228	16.272	35.052	42.545	0	54.904	14.877
	61.727	115.657	56.104	75.462	82.810	54.904	0	22.132
	30.017	70.257	25.495	40.112	45.541	14.877	22.132	0

k=7

0	40.607881	7.8102496	16.379178	21.494836	61.726908	30.016662
	40.608	0	45.011	34.645	75.228	115.657	70.257
	7.810	45.011	0	21.354	16.272	56.104	25.495
	16.379	34.645	21.354	0	54.996	115.503	54.968
	21.495	75.228	16.272	54.996	0	54.904	14.877
	61.727	115.657	56.104	115.503	54.904	0	22.132
	30.017	70.257	25.495	54.968	14.877	22.132	0

k=6

0	54.47591060	24.39472644	24.41990187	84.46809174	34.40442317
	54.476	0	34.645	75.228	115.657	70.257
	24.395	34.645	0	54.996	115.503	54.968
	24.420	75.228	54.996	0	54.904	14.877
	84.468	115.657	115.503	54.904	0	22.132
	34.404	70.257	54.968	14.877	22.132	0

k=5

0	54.4759106068	24.3947264468	34.2276795506	84.4680917433
54.476	0	34.645	87.830	115.657
24.395	34.645	0	71.027	115.503
34.228	87.830	71.027	0	51.252
84.468	115.657	115.503	51.252	0

k=4

0	48.5938237752	64.7269108126	125.182321436
48.594	0	87.830	115.657
64.727	87.830	0	51.252
125.182	115.657	51.252	0

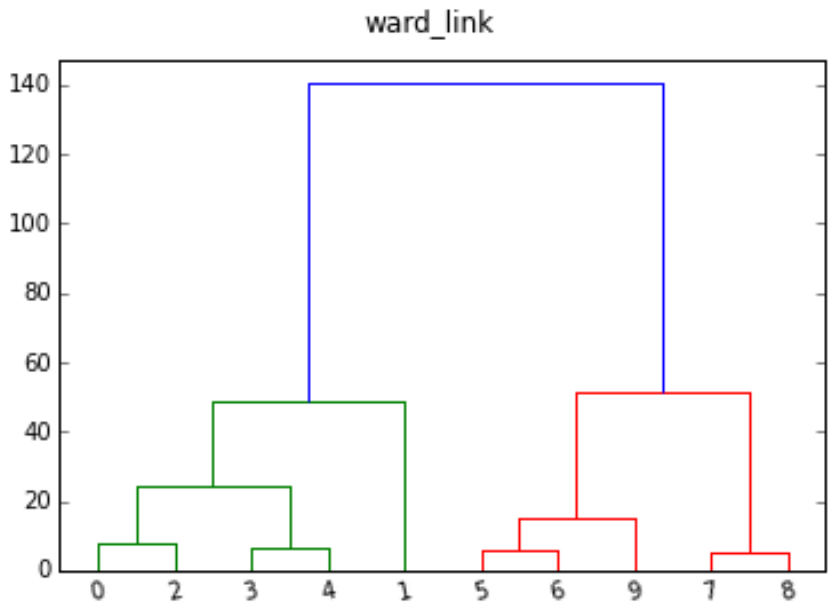
k=3

0	82.3284543958	142.982492917
82.328	0	51.252
142.982	51.252	0

k=2

0	140.324632695
140.325	0

Dendogram



Problem 3) Model Evaluation (20 pts.)
We are given the following classification results for a cancer screening test. Please derive the evaluation parameters:

a) Accuracy
b) Specificity
c) Sensitivity
d) Precision
e) Recall

Actual / Prediction	test = positive	test = negative	Total
cancer = yes	**90**	**210**	300
cancer = no	**140**	**9560**	9700
Total	230	9770	10000

Solution

Actual / Prediction	test = positive	test = negative	Total
cancer = yes	**TP = 90**	**FN = 210**	300
cancer = no	**FP = 140**	**TN = 9560**	9700
Total	230	9770	10000

TP = 90
FN = 210
FP = 140

TN = 9560

Accuracy **= #correct / N**
= (TP + TN) / (TP + FP + TN + FN)
= (90 + 9560) / (90 + 140 + 9560 + 210)
= 0.965

Specificity **= TN / (TN + FP)**
= 9560 / 9700
= 0.986

Sensitivity **= TP / (TP + FN)**
= 90 / 300
= 0.3

Precision **= TP / (TP + FP)**
= 90 / 230
= 0.391

Recall **= Sensitivity**
= 0.3

F1 **= P*R / 2(P+R)**
= 0.084

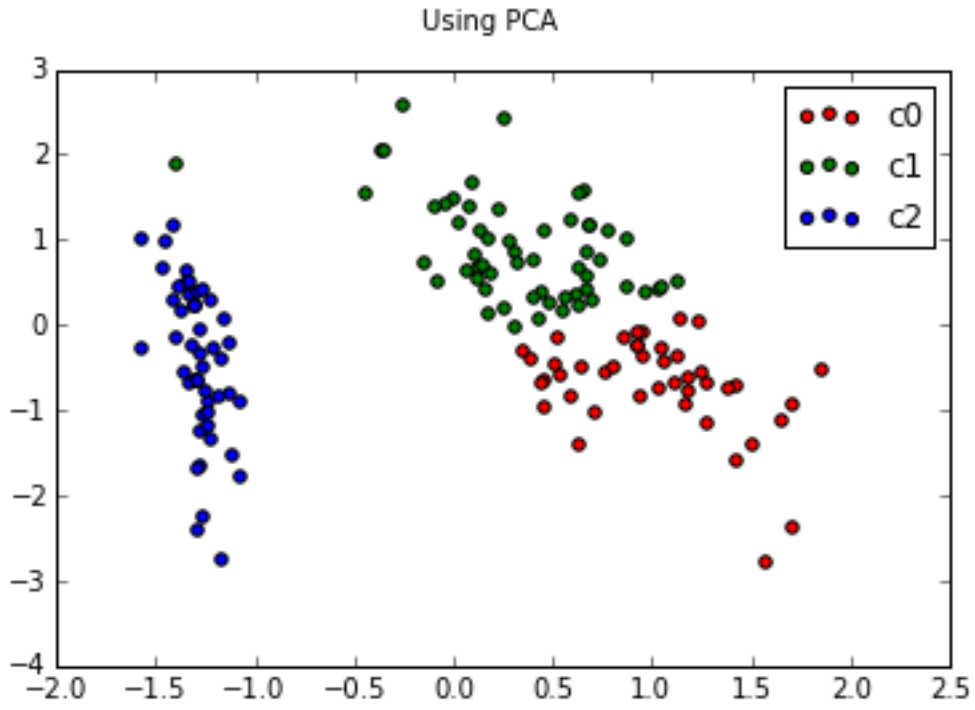
Problem 4) Python program (40 pts)
Take the sk-learn sample code *cluster_kmeans.py* discussed in the class that is used for clustering iris dataset. Make the following changes:

1. In addition to using PCA to reduce features from 4 to 2, also evaluate using the original feature pairs (1,2) and (3, 4).
2. For all the 3 clustering settings (original, (1,2), (3, 4)), calculate the clustering quality CQ = IE / EV as defined in the class:

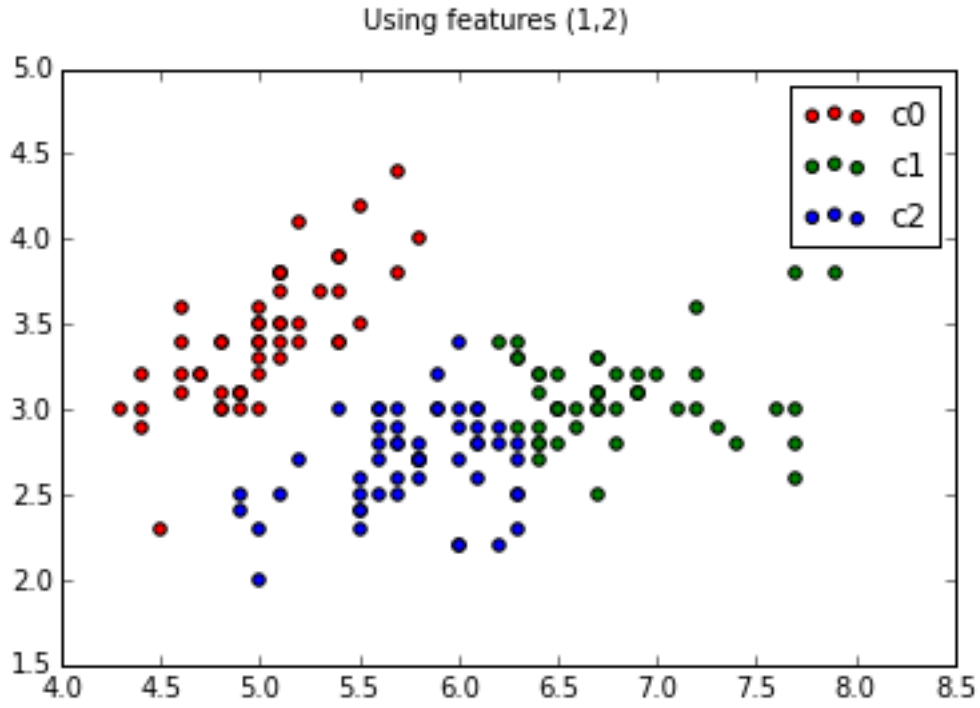
$$IV = \sum_{C} \sum_{x \in C} d(x, c) \quad \text{and} \quad EV = \frac{1}{N} \sum_{i} \sum_{j} \delta(C(x_i) \neq C(x_j)) d(x_i, x_j)$$

Output:
3 plots of k-means clustering results for PCA, features (1,2), and features (3,4).
3 CQ values for PCA, features (1,2), and features (3,4).

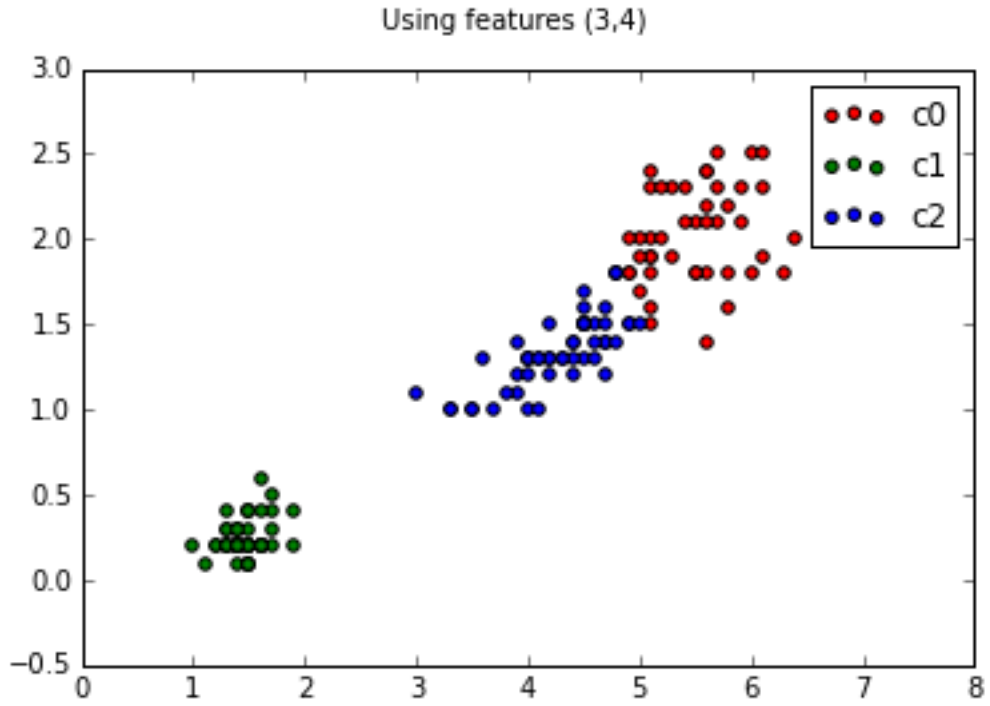
Solution




```
IV: 19831.5213
EV: 2.2027
CQ: 9003.2777
```




```
IV: 12872.9967
EV: 1.4206
CQ: 9061.7792
```




```
IV: 24422.1216
EV: 3.0034
CQ: 8131.4968
```