Com S 573	Spring Semester 2021
Machine Learning	
Homework 4	
Instructor: Hongyang Gao	Submitted by: Ibne Farabi Shihab

Question no 1

(1) Min link hierarchical clustering: Similarity of two clusters is determined based on the two most similar (closest) points in the different clusters.

Step 1 :Cluster P2 and P5 cluster can be merged to one one cluster as they have the most similarity 0.98. The new similarity matrix will be(based on the definition of Single link):

Table 1: Similarity matrix.

	p1	**p2,p5**	**p3**	**p4**
p1	1.00	0.35	0.41	0.55
p2+p5	0.35	1.00	0.85	0.76
p3	0.41	0.85	1.00	0.44
p4	0.55	0.76	0.44	1.00

Step 2 : Cluster P3 and (P2,P5) cluster can be merged to one one cluster because they have the most similarity of 0.85. The new similarity matrix will be:

Table 2: Similarity matrix.

	p1	**p2,p5,p3**	**p4**
p1	1.00	0.41	0.55
p2+p5+p3	0.41	1.00	0.76
p4	0.55	0.76	1.00

Step 3 : (P2,P5,p3) and P4 clusters can be merged to one one cluster as they have the most similarity of 0.76. The new similarity matrix will be: Step 4 : Lastly, all the clusters will be merged to one cluster. The similarity value is 0.55.

(2) Max link hierarchical clustering: Similarity of two clusters is determined based on the two least similar points in the different clusters.

Step 1 : Cluster P2 and P5 cluster can be merged to one cluster as they have the most similarity of 0.98.

The new similarity matrix will be:

Step 2 : Cluster P3 and (P2,P5) can be merged to one cluster because they have the most similarity of 0.64. The new similarity matrix will be:

Step 3 : At this point, cluster P1 and P4 cluster can be merged to one cluster as they have the most similarity 0.55. The new similarity matrix will be:

Step 4 : At the end, all the clusters will be merged to one one cluster. The similarity value is (**0.10**).

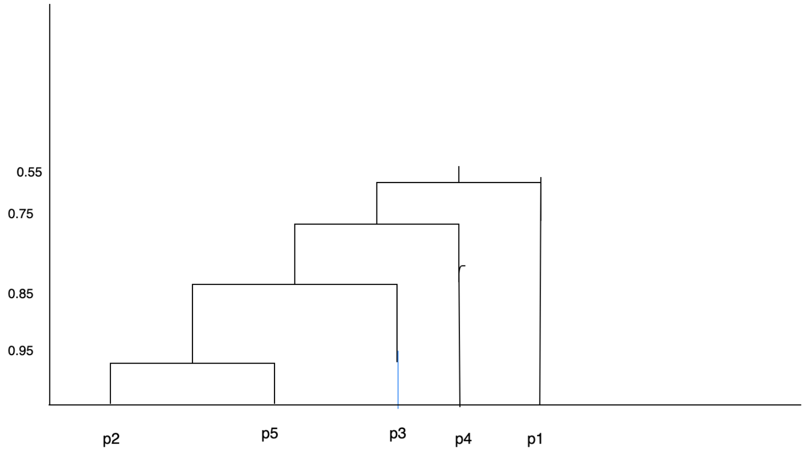

Figure 1: Min link dendrogram

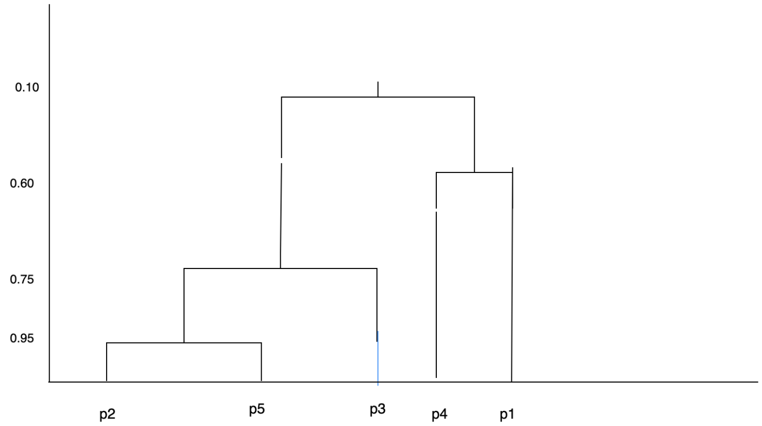

Figure 2: Man link dendrogram

Table 3: Similarity matrix.

	p1	**p2,p5,p3,p4**
p1	1.00	0.55
p2+p5+p3+p4	0.55	1.00

Table 4: Similarity matrix.

	p1	**p2,p5**	**p3**	**p4**
p1	1.00	0.10	0.41	0.55
p2+p5	0.10	1.00	0.64	0.47
p3	0.41	0.64	1.00	0.44
p4	0.55	0.47	0.44	1.00

Table 5: Similarity matrix.

	p1	**p2,p5,p3**	**p4**
p1	1.00	0.10	0.55
p2+p5+p3	0.10	1.00	0.44
p4	0.55	0.44	1.00

Table 6: Similarity matrix.

	p1,p4	**p2,p5,p3**
p1+p4	1.00	0.10
p2+p5+p3	0.10	1.00

Question no 2
2(a):
The algorithm of K-median are shown below:

```
1. Select K points as the initial centroids (same as K means).
2. repeat
3.     Form K clusters by assigning all points to the closest centroid (same as K means).
4.     Recompute the median using the median of each individual feature
5. until The centroids don't change (centroid that have a minimum loss- same a k means).
```

2(b):
K-medians uses Manhattan distance for the assignment of points. In every iteration, the centriod of each cluster is moved to a new point and the Manhattan distance from each point to the centroid is being calculated. Then the distance is being ordered numerically in the ascending order and the median is being pointed out.

The idea is to move the median in any direction by the value of step which provides a better approximation for the median. If it does, then the median need to be updated and continue the same process.

2(c):
Yes, K-medians does help to get rid of the outlier. In k- means, the centroids do not necessarily need be a point in a cluster. However, in K-median, centroids has to be points in a cluster. As we can assume from that, means are impacted by the outliers while medians are not that much . For an instance, there are 7 points in a cluster and they are in the following distance: 1,2,4,6,9,11,40. Now, for mean, the distance is 10.42. The centroid will reflect this mean distance and should be in between two points which are at the corner which does not actually show real truth. On the other hand, if we use median than the median of this distances is 6 which is not being impacted by the outlier (the point at distance 40). Thus, we can conclude that K-medians helps to get rid of the outlier problem.

Question no 3(a):

1st PC is a minimum distance fit to a line in horizontal space X. PCA's are a series of linear least squares fits to a sample, each orthogonal to the previous. These signifies the maximum variance in a particular direction of data. Each observation may be projected to a line in order to get a coordinate value in the PC line. This value is known as score.

The given data points are, say : x_1=(-1,-1), x_2 =(0,0) and x_3 =(1,1) (seems to be normalized)

The first principal component is $PC^T = [\frac{\sqrt{2}}{2}, \frac{\sqrt{2}}{2}]$ (Sum of square distance divided by number of points-1)

Question no 3(b): The data after the transform to a 1-D space are:

x_1P=$x_1^T PC$=$[-1,-1][\frac{\sqrt{2}}{2}, \frac{\sqrt{2}}{2}]^T$= $-\sqrt{2}$
x_2P=$x_2^T PC$=$[0,0][\frac{\sqrt{2}}{2}, \frac{\sqrt{2}}{2}]^T$= 0
x_3P=$x_3^T PC$=$[1,1][\frac{\sqrt{2}}{2}, \frac{\sqrt{2}}{2}]^T$= $\sqrt{2}$

(5 points) Run “main.py” to see the reconstruction results and summarize your observations from the results into a short report. When you run the “main.py” file, a subset (the first two) of the reconstructed images based on p = 10, 50, 100, 200 principal components will be automatically saved on the “code” folder. Please attach these images into your report also.

Question no 4(c):

P=10	P=50	P=100	P=200	
394.14	202.55	119.59	52.49	

If we look at the error for the components(in the table), it is decreasing with the increase of of the number of components. Taking a look at the generated picture also justify the claim as with the increase of the components the images are getting more sharper and clear. In the matrix we had 256 components while taking the 200 components give us a good image to look at with a comparatively less error(52.49). In a larger context or with large data, this small reduction will help to reduce the computation while also losing less information. In addition to that, it is also helping us to get rid of some unnecessary features while retaining most of the important informations.