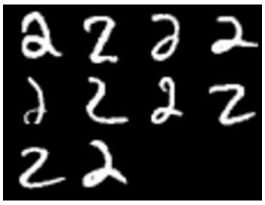
Report 1 - s1614102

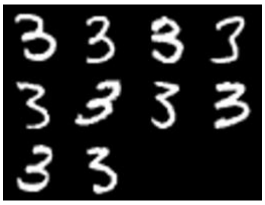
Task 1.1

Class 1

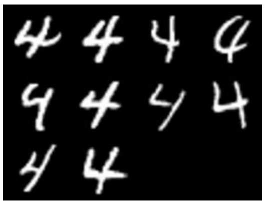
Class 2



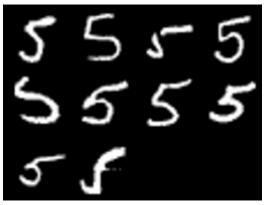
Class 3



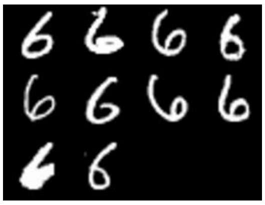
Class 4



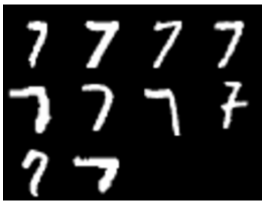
Class 5



Class 6

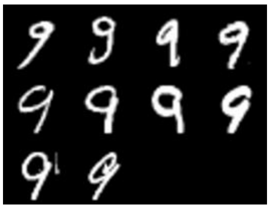


Class 7



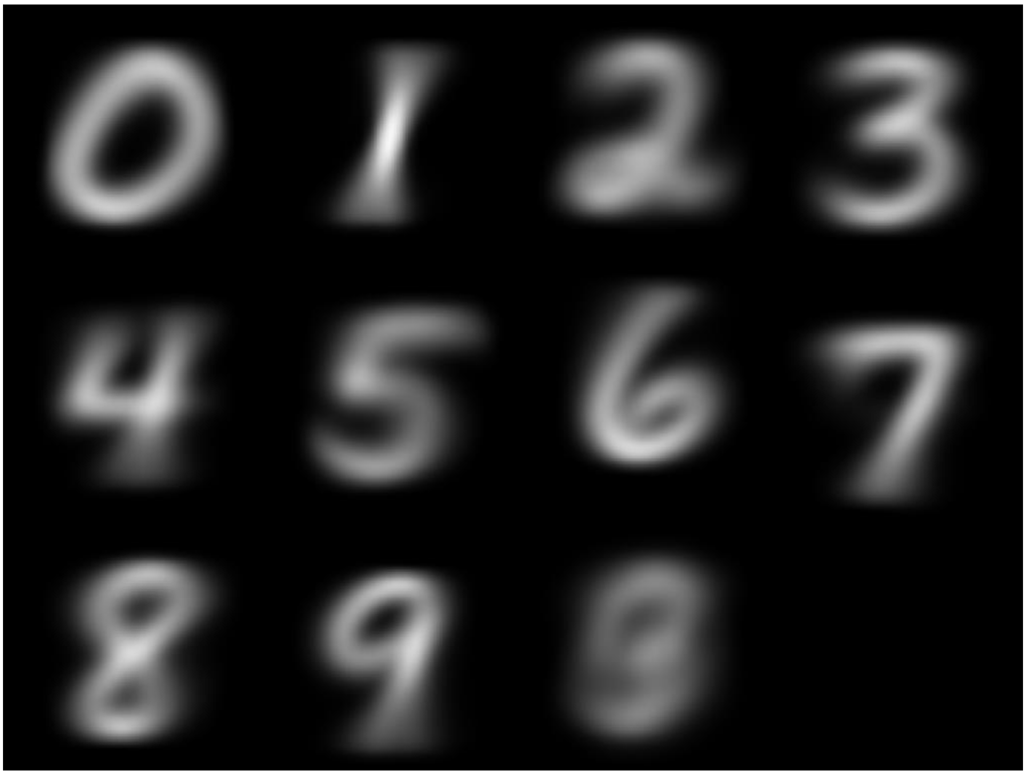
Class 8

Class 9

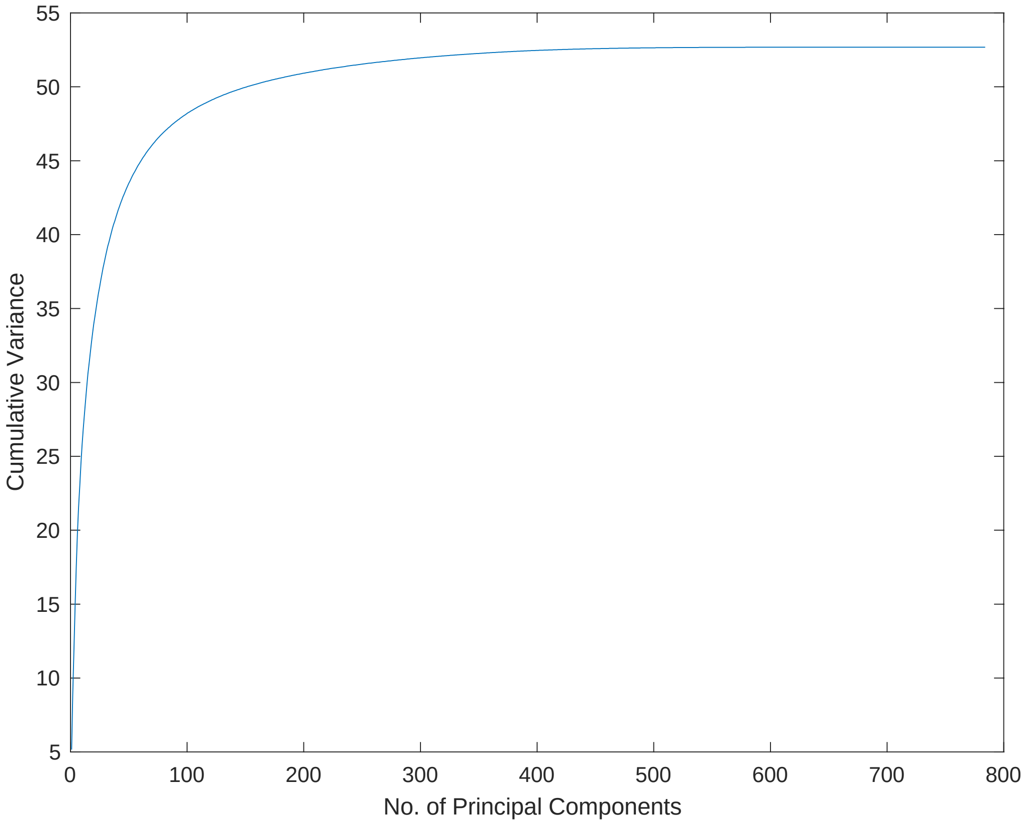


Class 10

Task 1.2

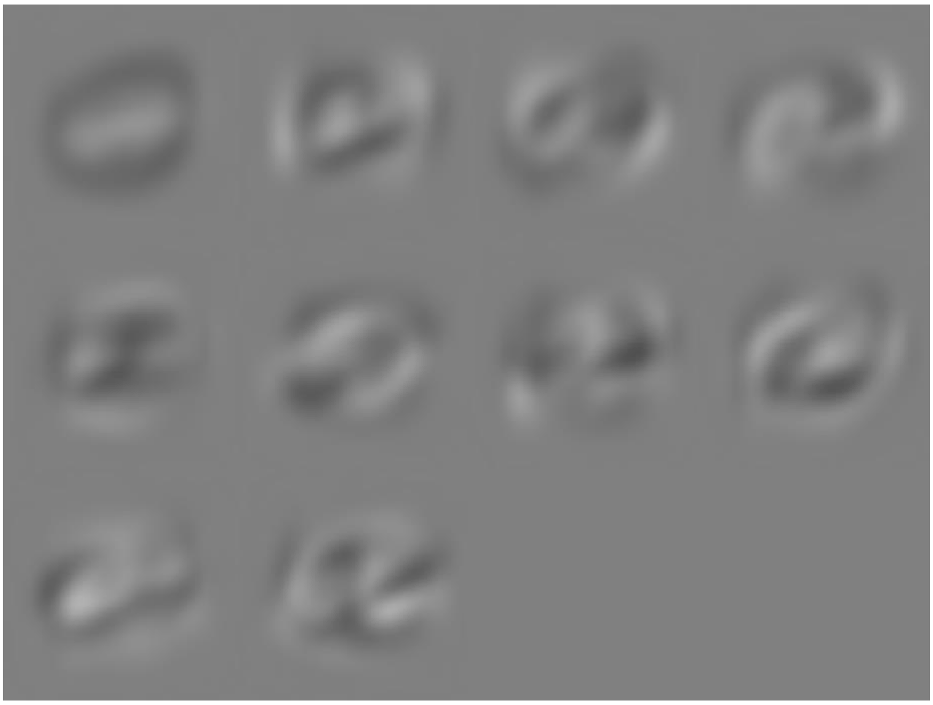


Task 1.3

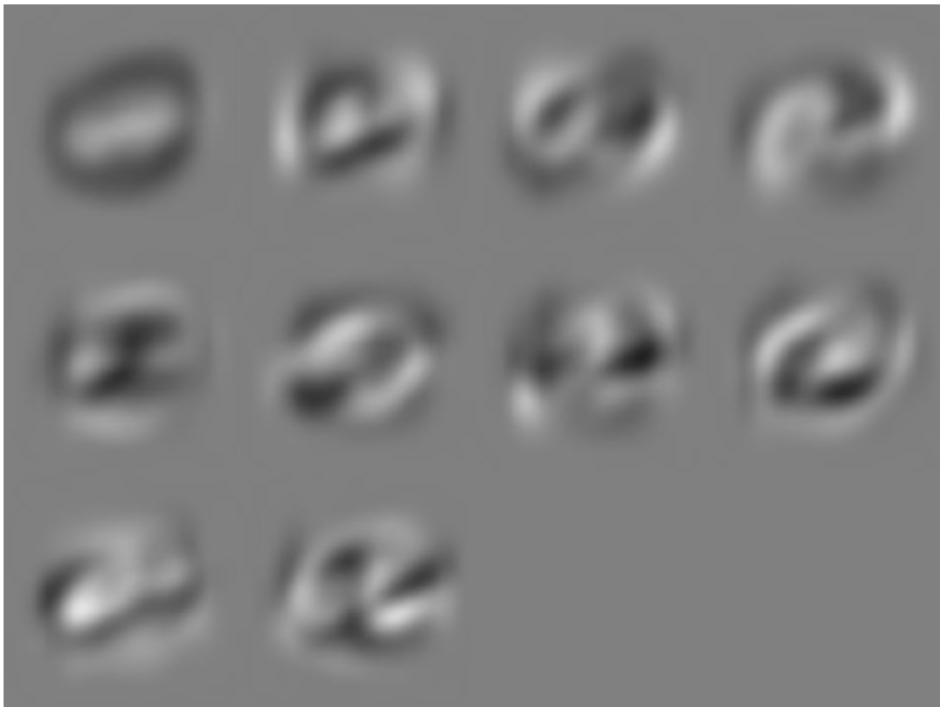


Cumulative variance	70%	80%	90%	95%
Minimum number of principal components	26	44	87	154

Task 1.4

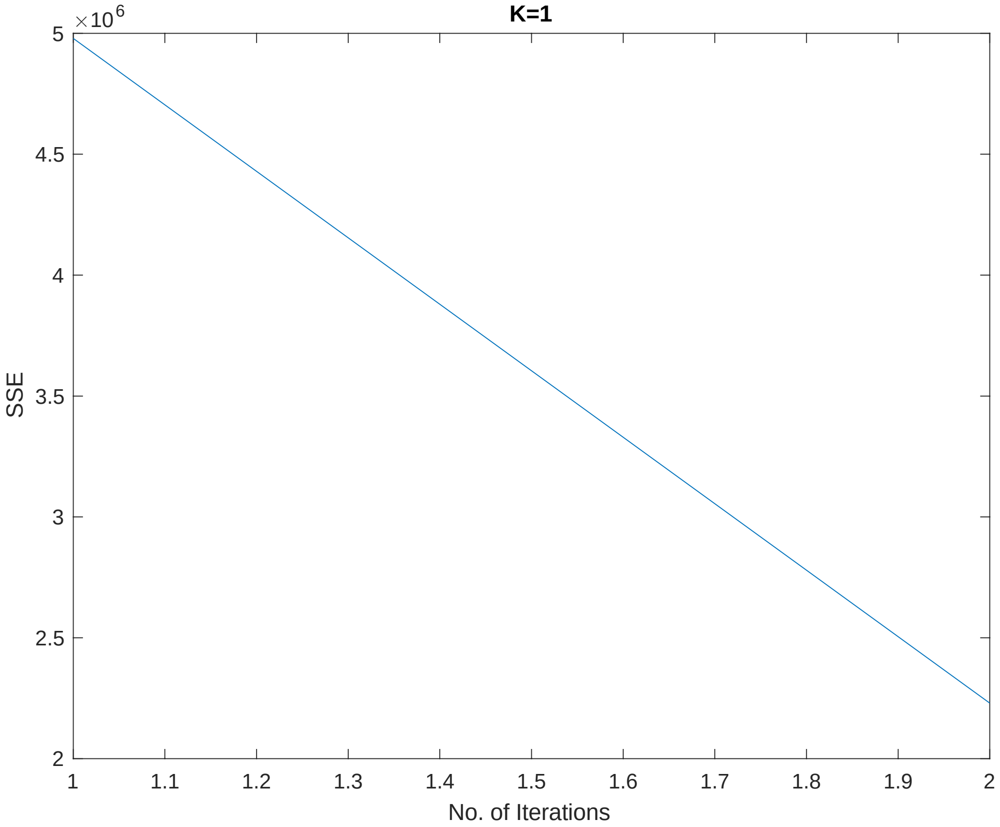


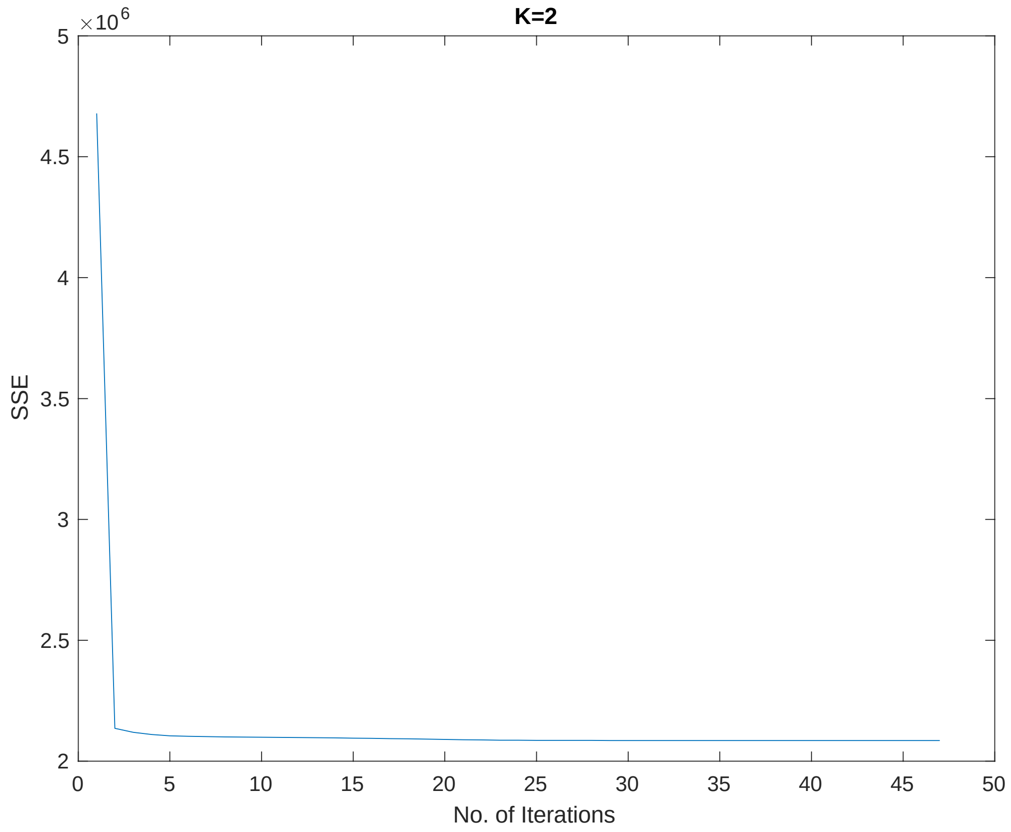
First ten principle components (Image range -0.5 to 0.5)

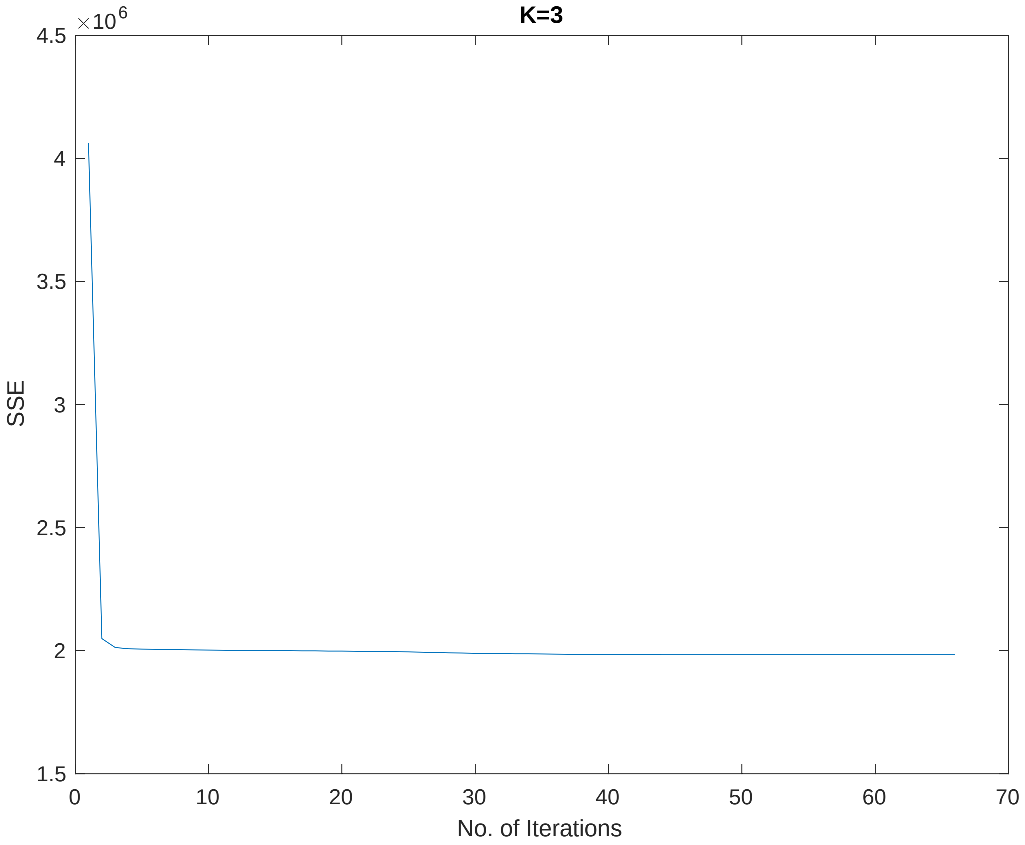


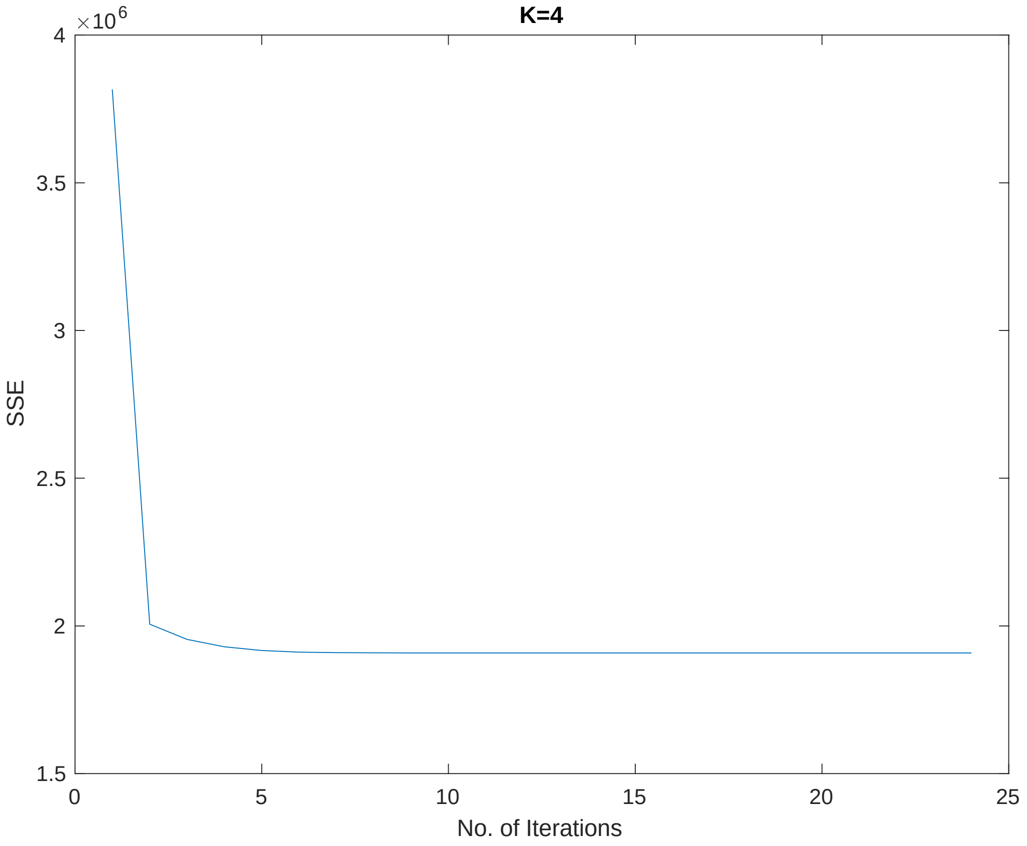
First 10 principle components (Image range -0.25 to 0.25)

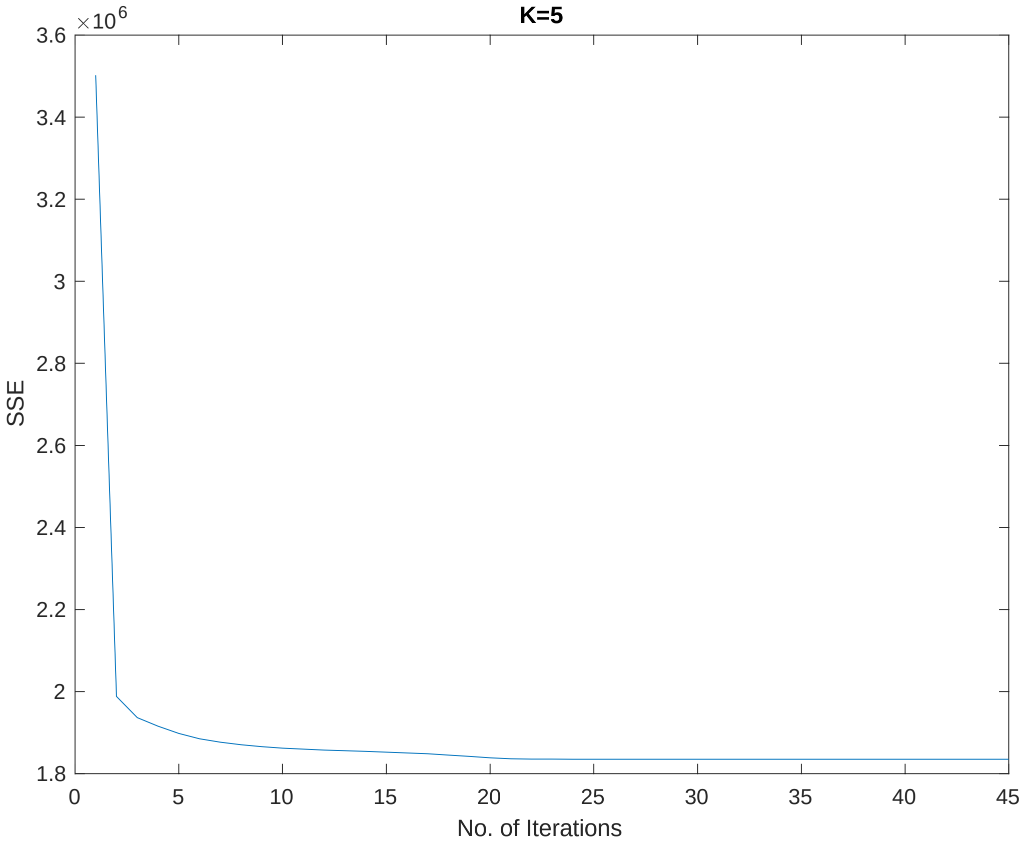
Task 1.5

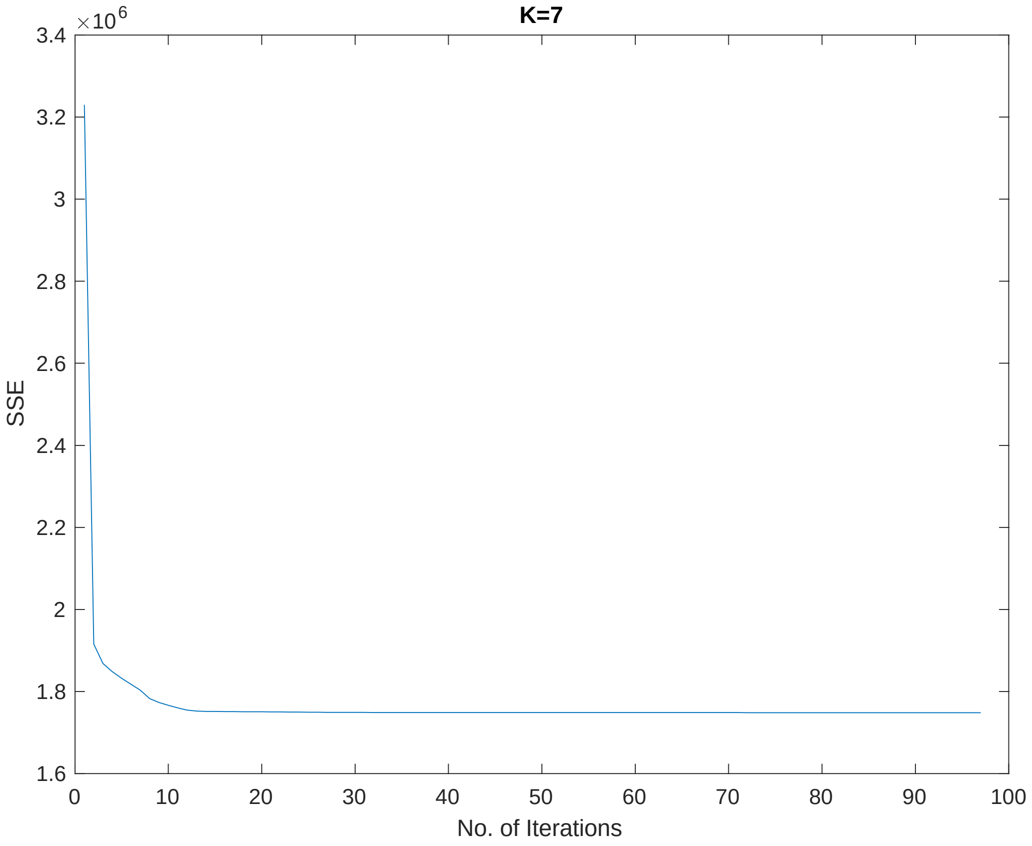

K=1
×10^6
5
4.5
4
3.5
3
2.5
2
SSE
1
1.1
1.2
1.3
1.4
1.5
1.6
1.7
1.8
1.9
2
No. of Iterations

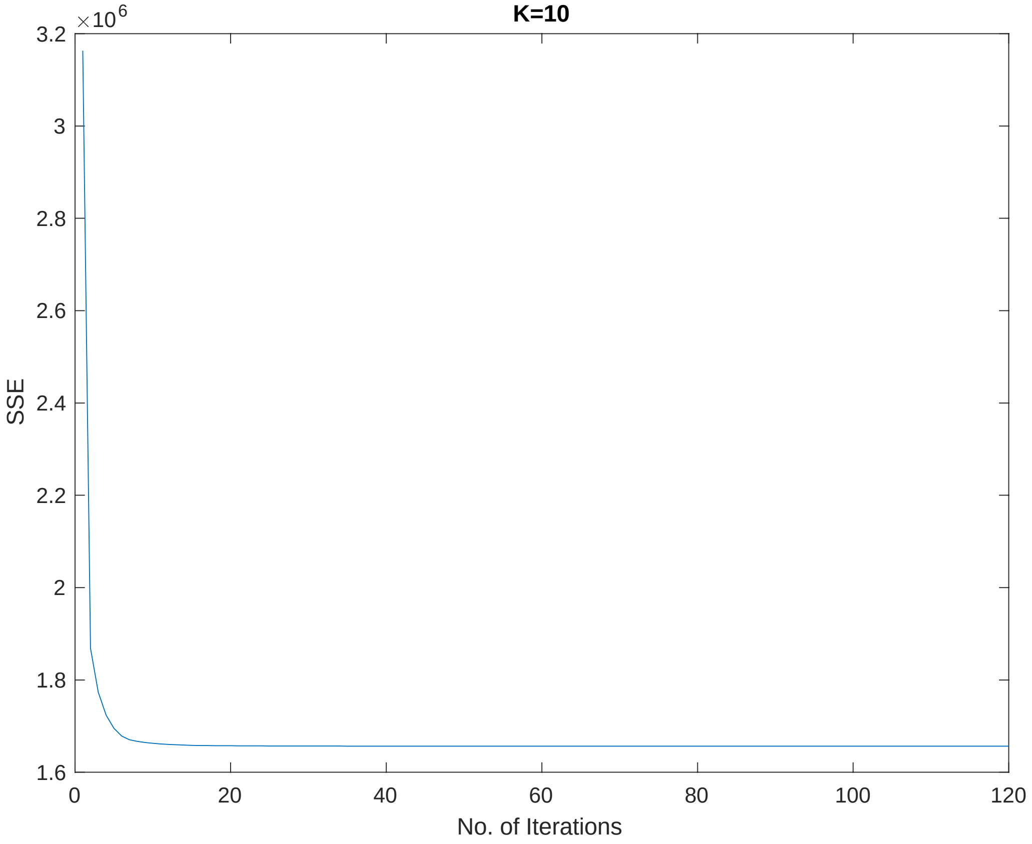

K=2
×10^6
5
4.5
4
3.5
3
2.5
2
SSE
0
5
10
15
20
25
30
35
40
45
50
No. of Iterations

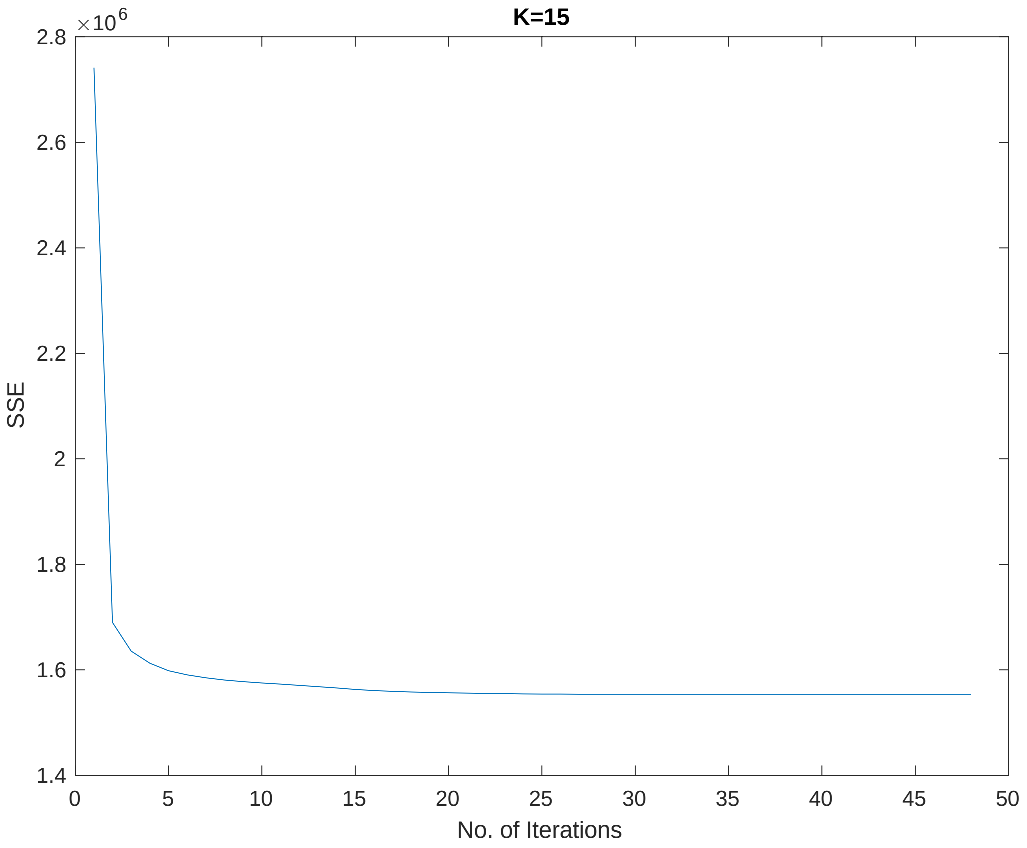

K=3
×10^6
4.5
4
3.5
3
2.5
2
1.5
SSE
0
10
20
30
40
50
60
70
No. of Iterations

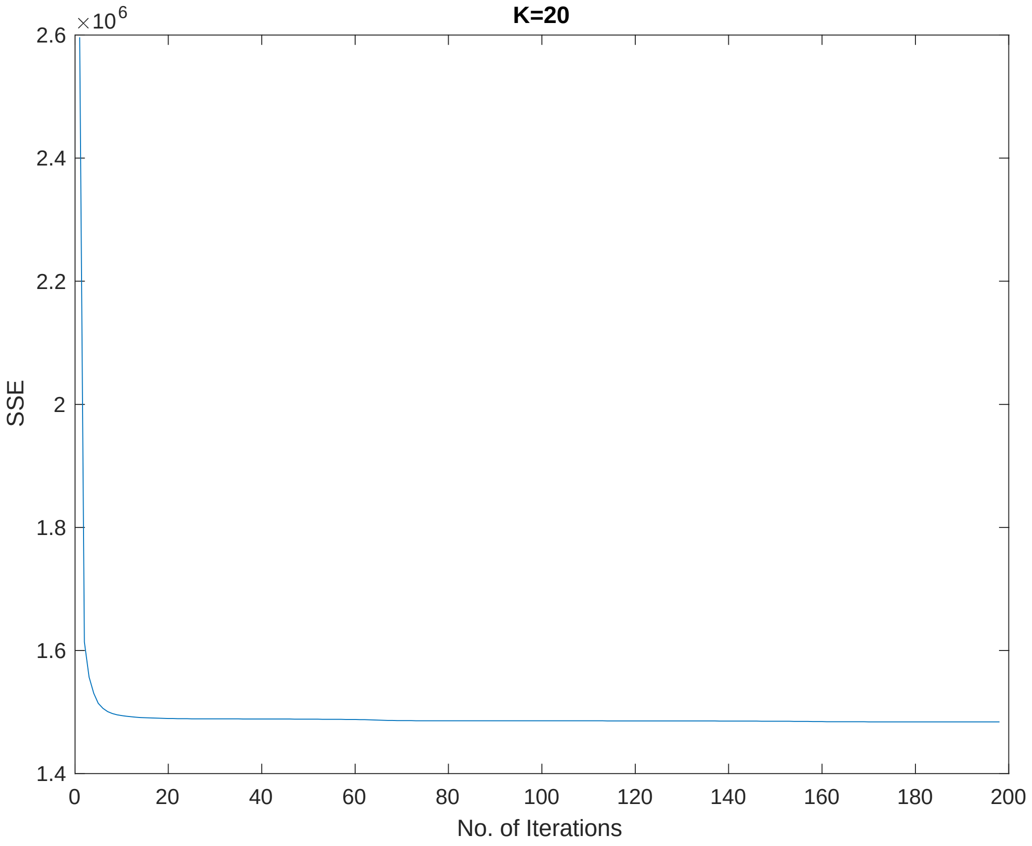

K=4
×10^6
4
3.5
3
2.5
2
1.5
SSE
0
5
10
15
20
25
No. of Iterations

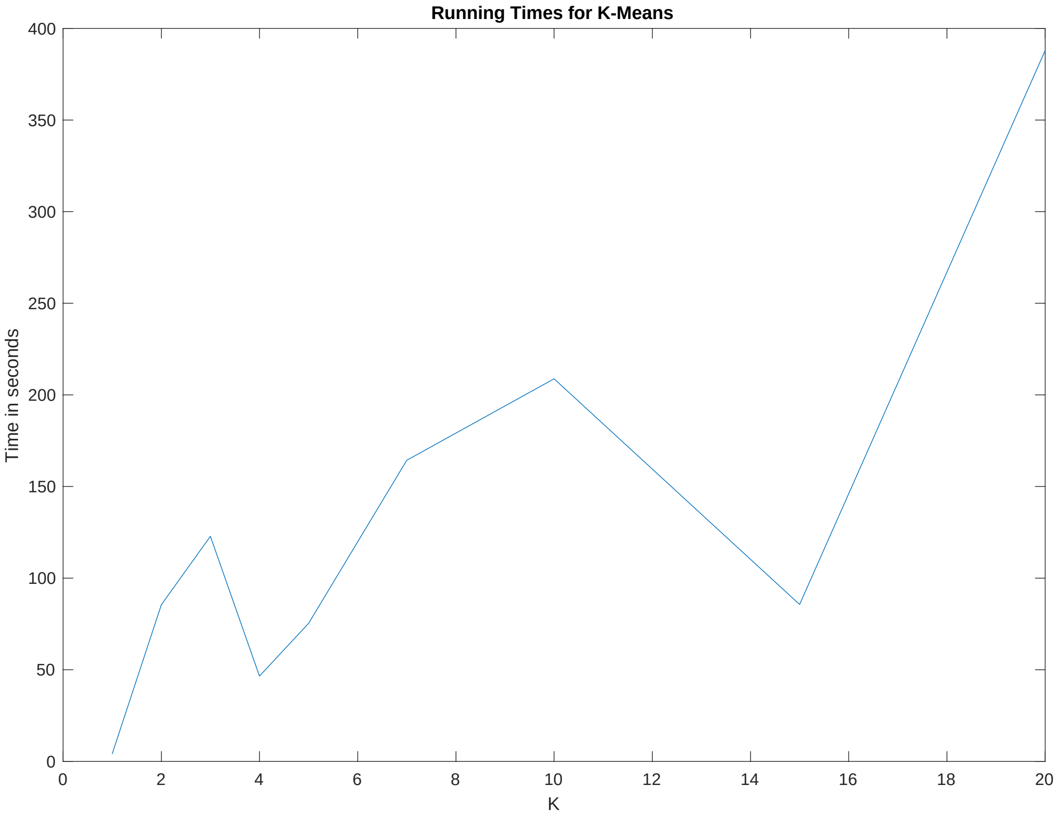

K=5
×10^6
3.6
3.4
3.2
3
2.8
2.6
2.4
2.2
2
1.8
SSE
0
5
10
15
20
25
30
35
40
45
No. of Iterations


K=7
×10^6
3.4
3.2
3
2.8
2.6
2.4
2.2
2
1.8
1.6
SSE
0
10
20
30
40
50
60
70
80
90
100
No. of Iterations

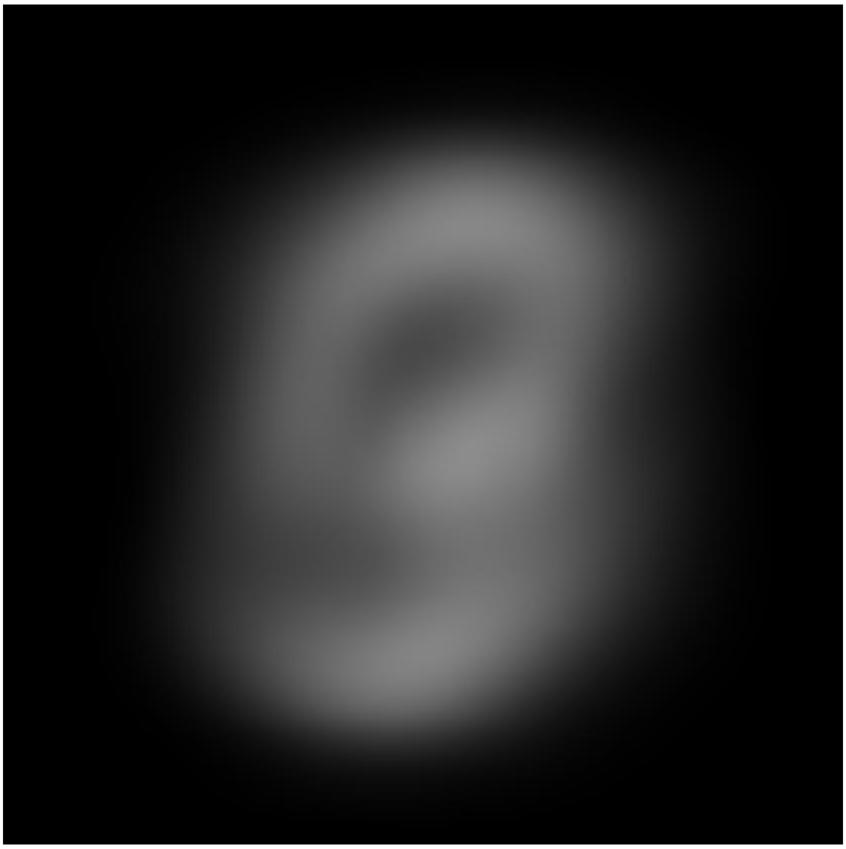

K=10
×10^6
3.2
3
2.8
2.6
2.4
2.2
2
1.8
1.6
SSE
0
20
40
60
80
100
120
No. of Iterations


K=15
×10^6
2.8
2.6
2.4
2.2
2
1.8
1.6
1.4
SSE
0
5
10
15
20
25
30
35
40
45
50
No. of Iterations


K=20
×10^6
2.6
2.4
2.2
2
1.8
1.6
1.4
SSE
0
20
40
60
80
100
120
140
160
180
200
No. of Iterations

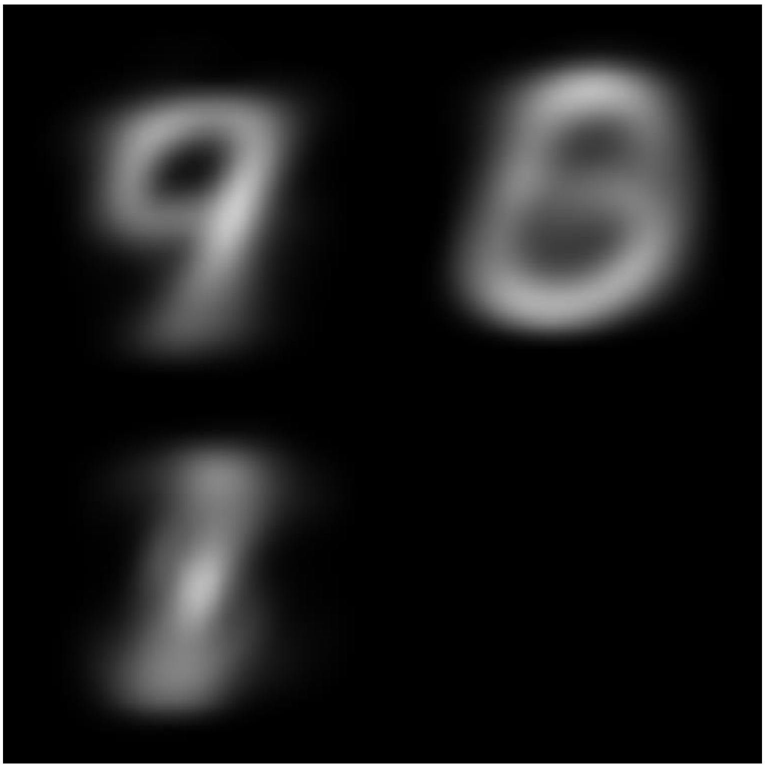

Running Times for K-Means
400
350
300
250
200
150
100
50
0
Time in seconds
0
2
4
6
8
10
12
14
16
18
20
K

Task 1.6

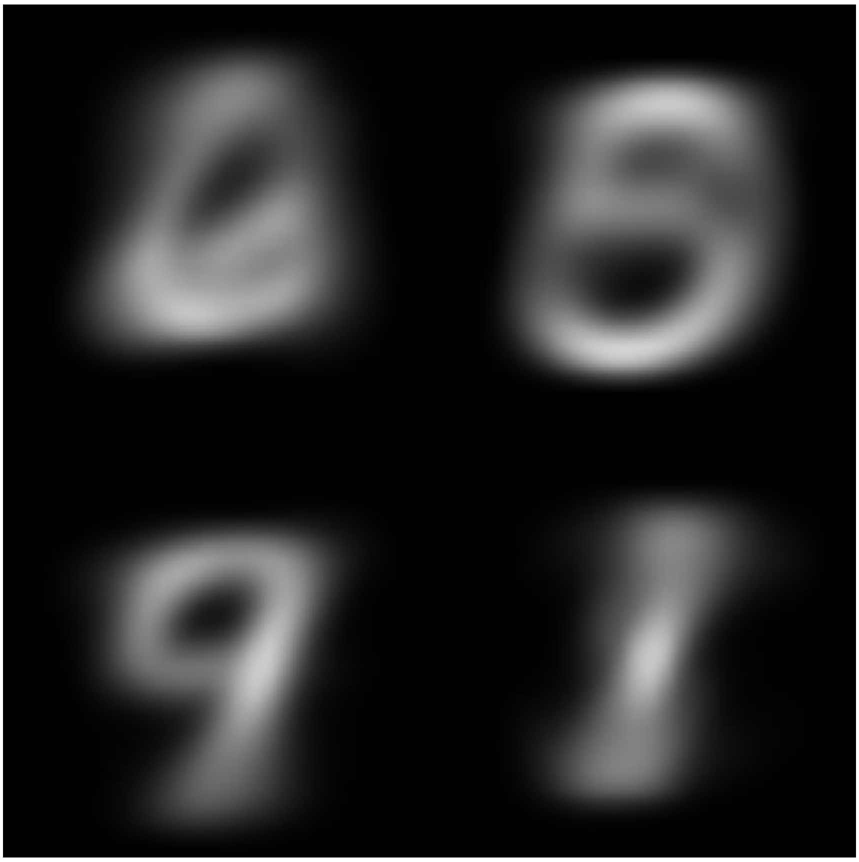


b) *K=1*

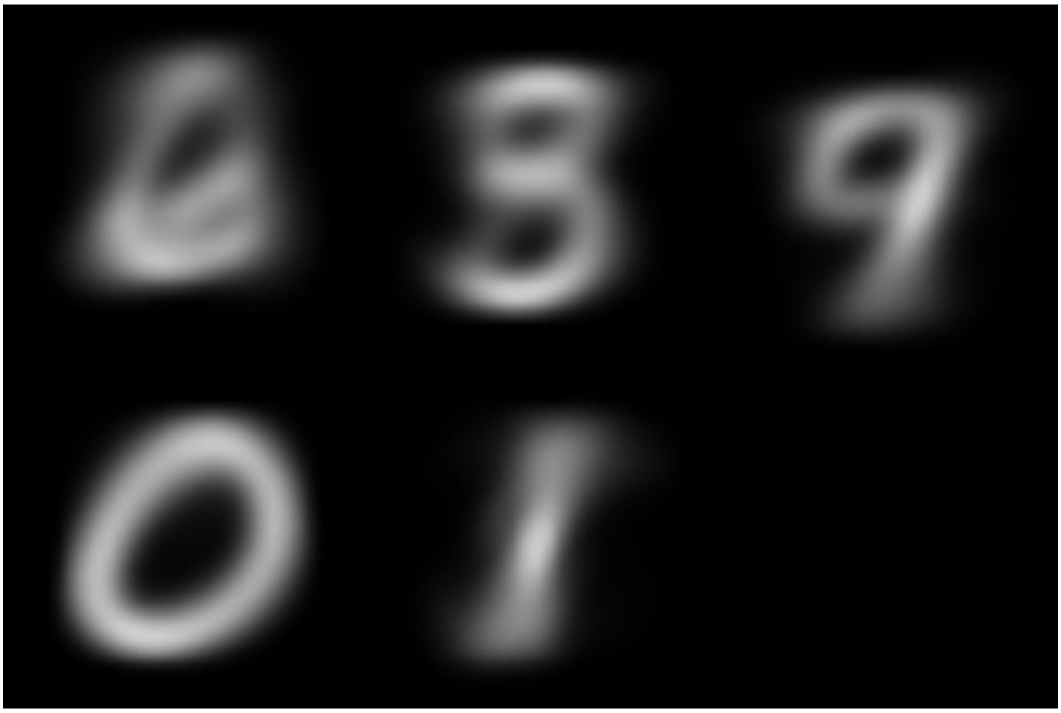
a) *K=2*



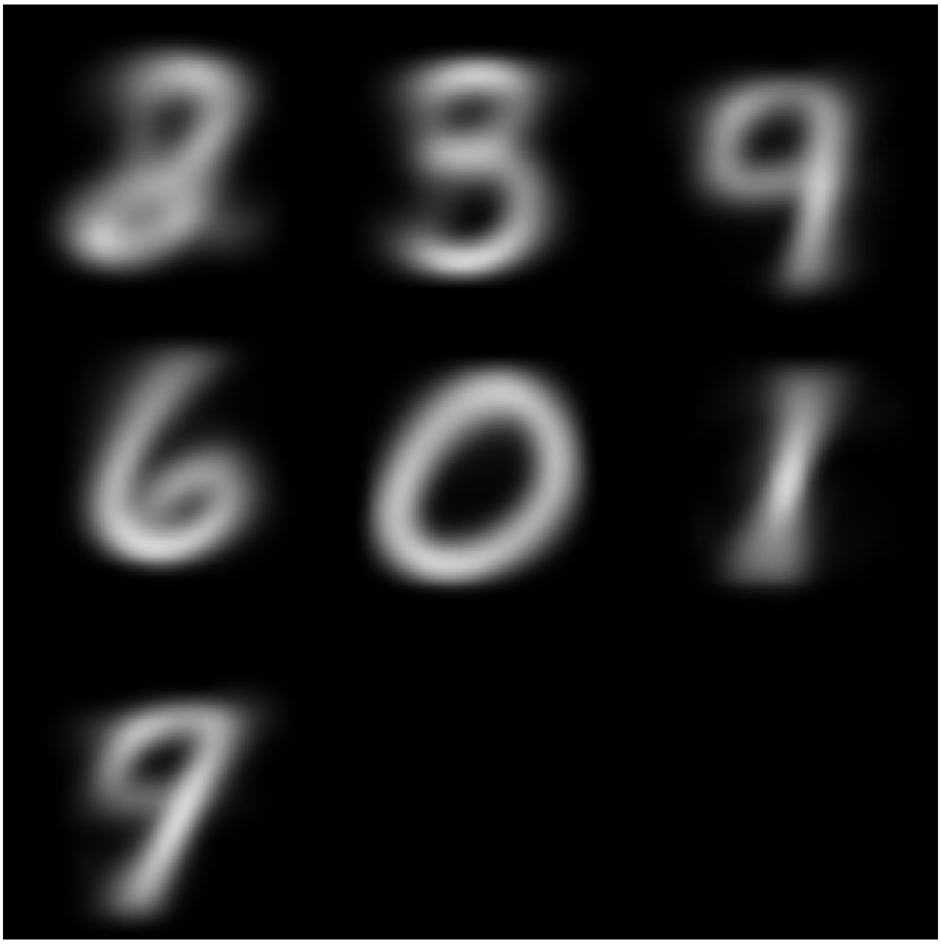
b) K=3



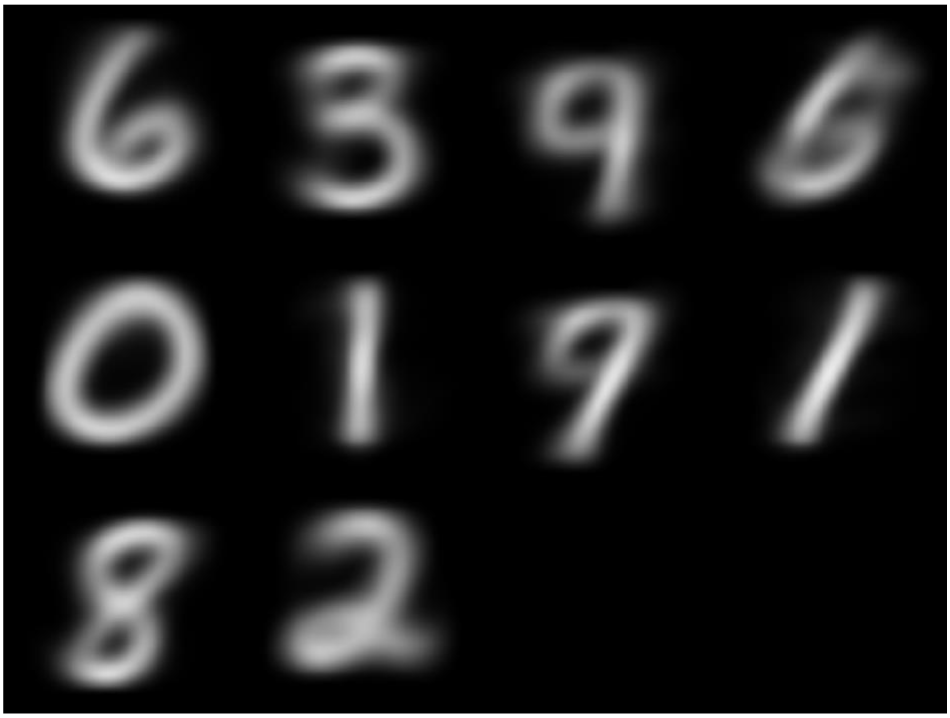
c) K = 4



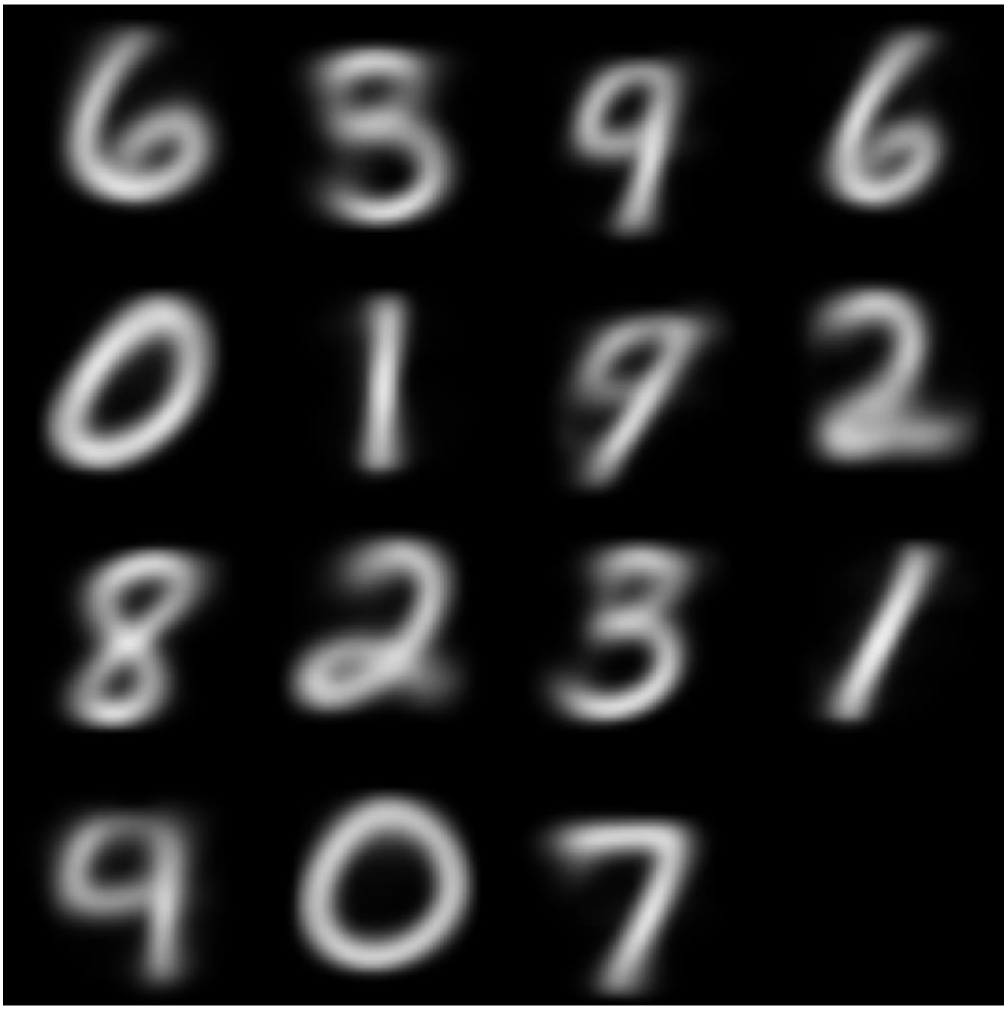
d) K=5



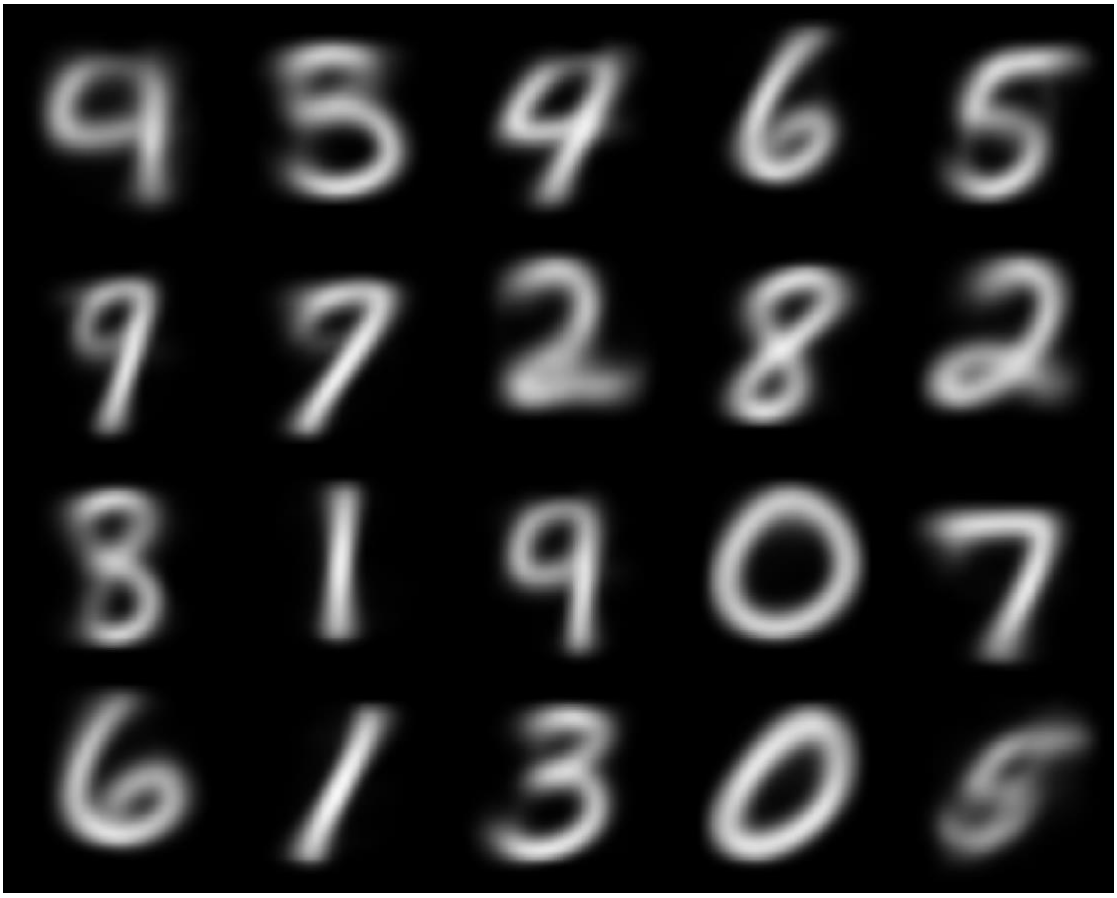
e) K=7



f) K=10



g) K=15

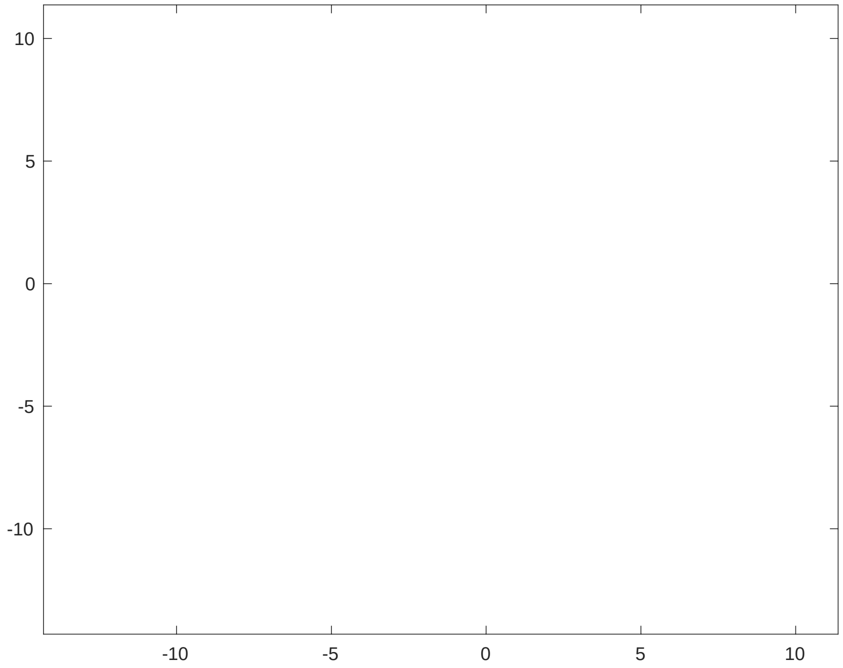


h) K=20

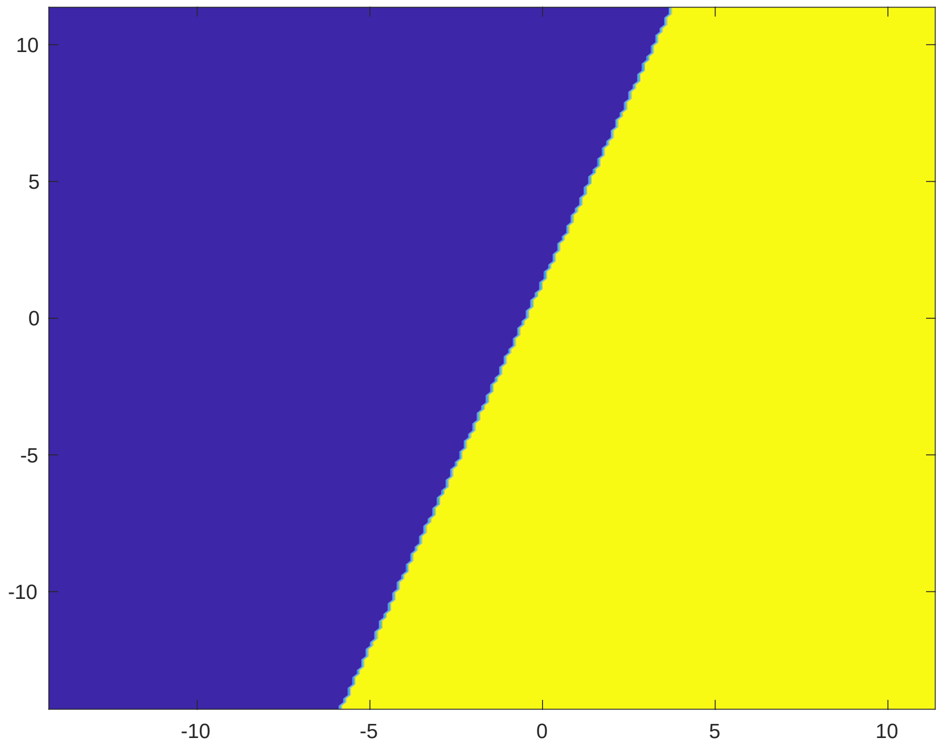
Task 1.7

The decision regions were visualised using the following method:
1. Create a grid of points on the 2D-PCA plane.
2. Reconstruct the points by mapping them back to the high-dimensional space.
3. Run the 1-NN Algorithm to assign each of those points to the nearest cluster centroid.
4. Draw the decision boundaries using *contourf*.

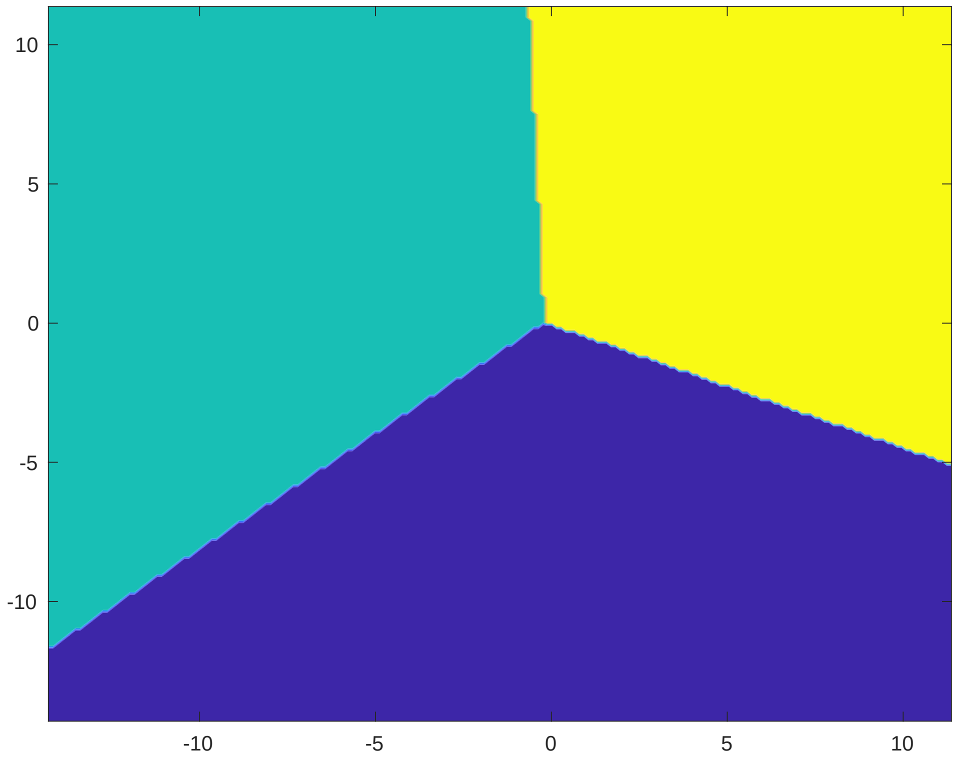
For k=1 the figure is obviously blank since all points are assigned to the same and only cluster.



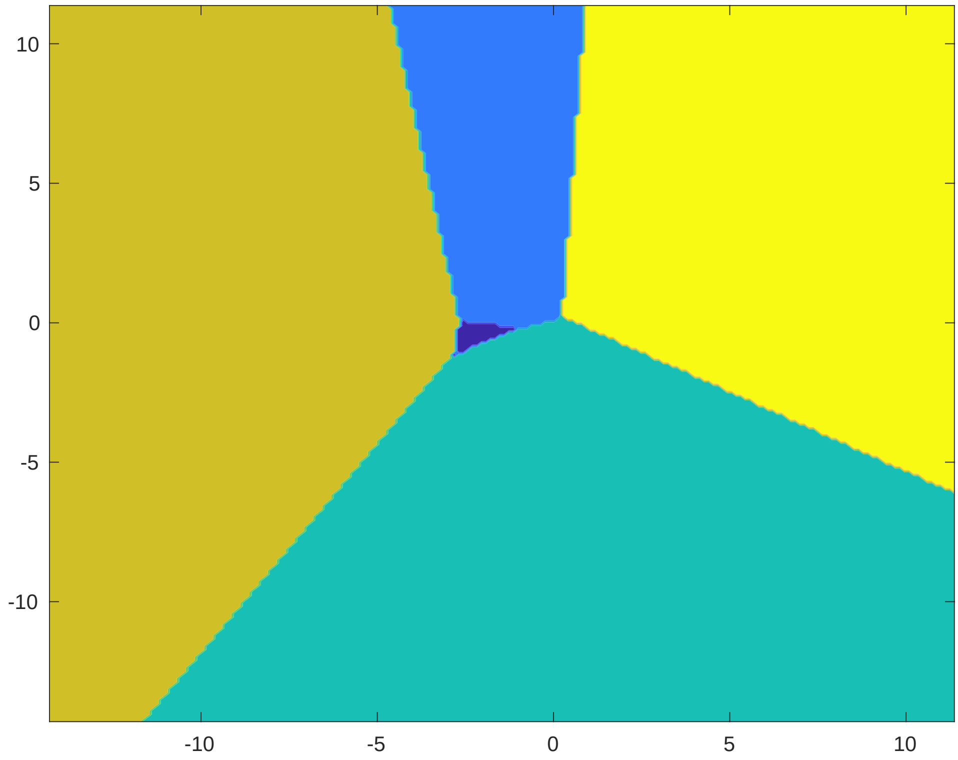
i) k=1



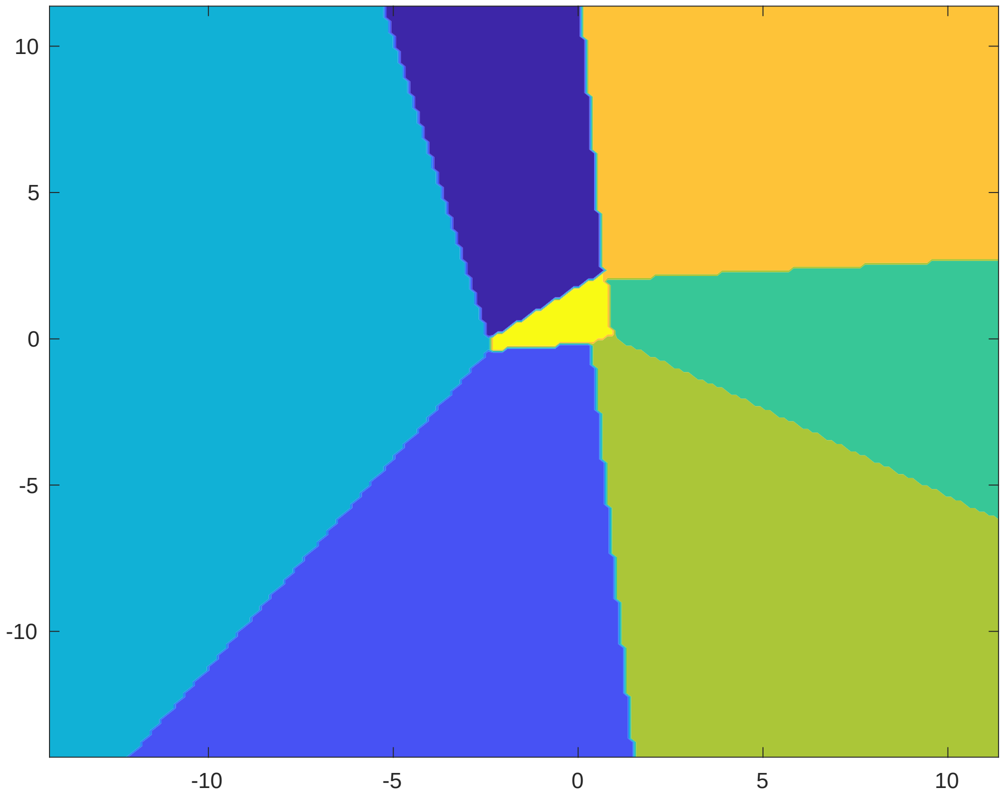
j) k=2



k) k=3



l) k=5

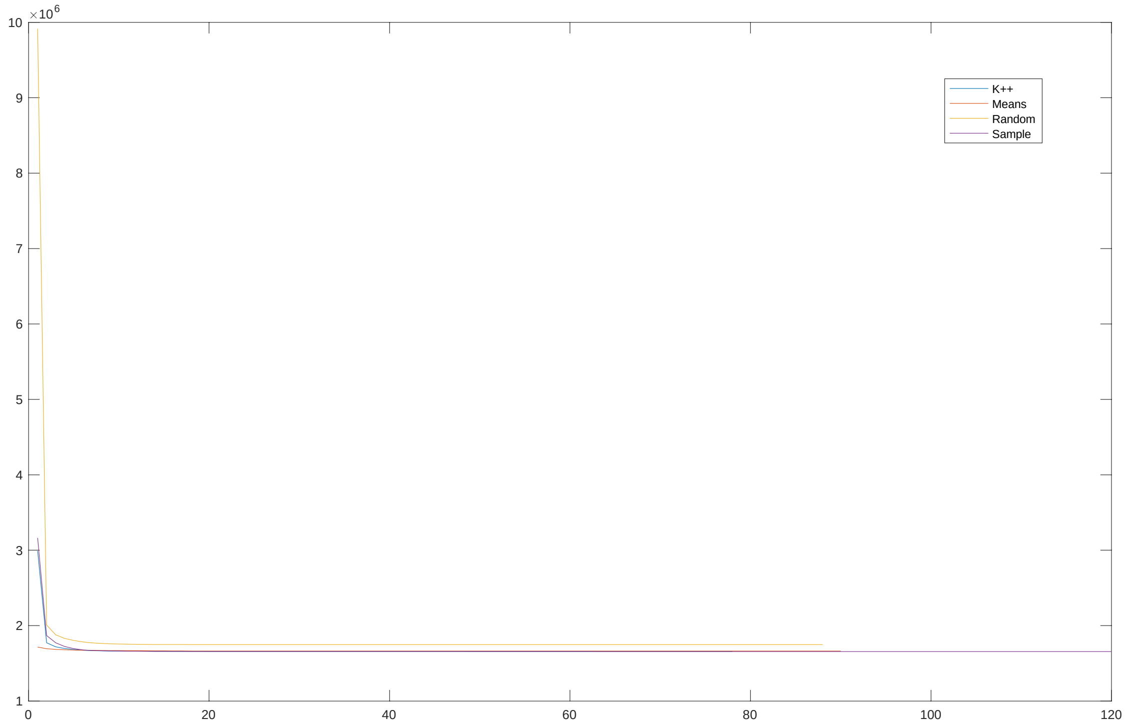


m) k=10

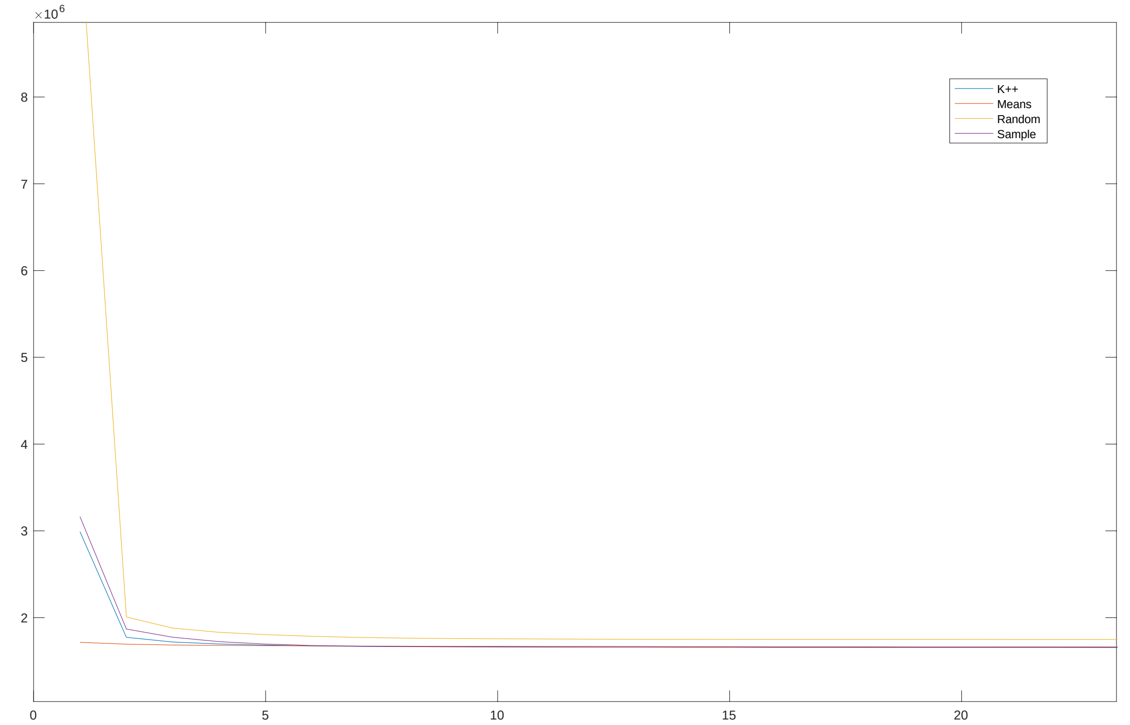
Task 1.8

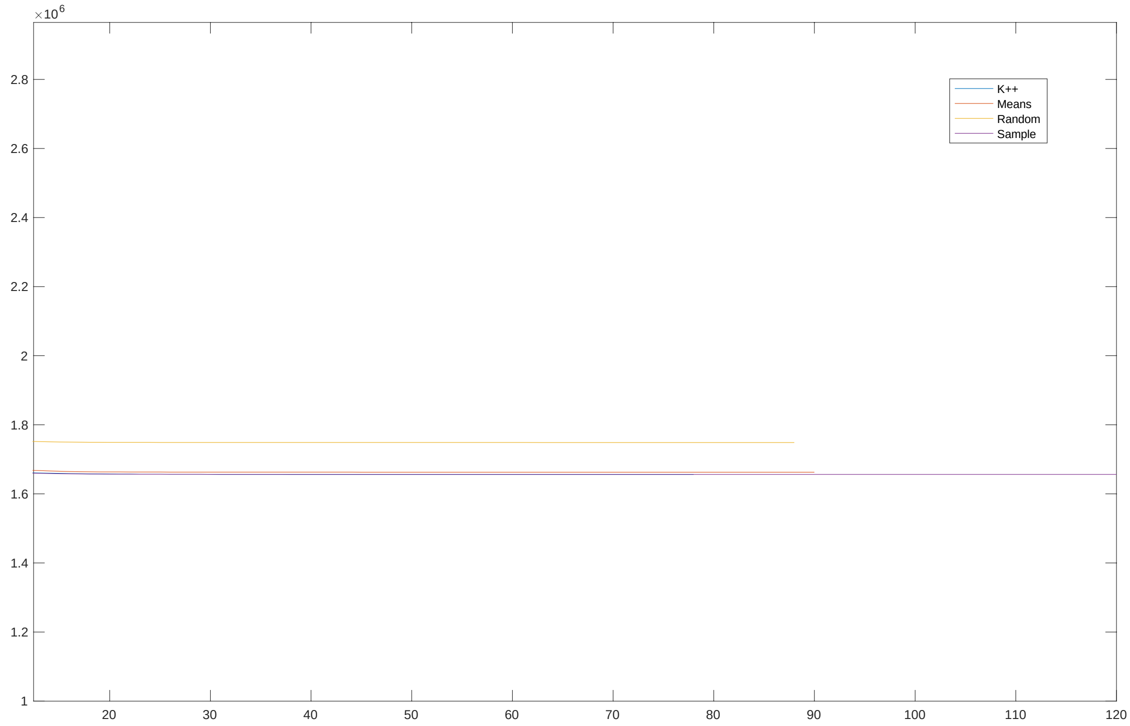
Using k=5, four different initial centers were evaluated with regards to their clustering performances. The clustering performance was determined based on the SSE and the number of iterations needed to converge.
The initial centers were chosen
1. by uniformly randomly generating k vectors (denoted in the following by *Random*).
2. by taking the first k vectors from the sample (denoted in the following by *Sample*).
3. by taking the mean vectors of k classes respectively (denoted in the following by *Mean*).
4. using the K-means++ algorithm as suggested by Artur et al[1] (denoted in the following by *K++*).

The following plots demonstrate the SSE (y-axis) for the different numbers of iterations (x-axis).
The number of iterations needed to find the cluster were similar among *K++*, *Means* and *Random* whereas *Sample* performed significantly worse.
In terms of SSE, *K++*, *Means* and *Sample* achieved comparable results whereas *Random* had a higher SSE after convergence.
It is also interesting to see that the SSE with *Mean*-Initialisation hardly changes from beginning to convergence.

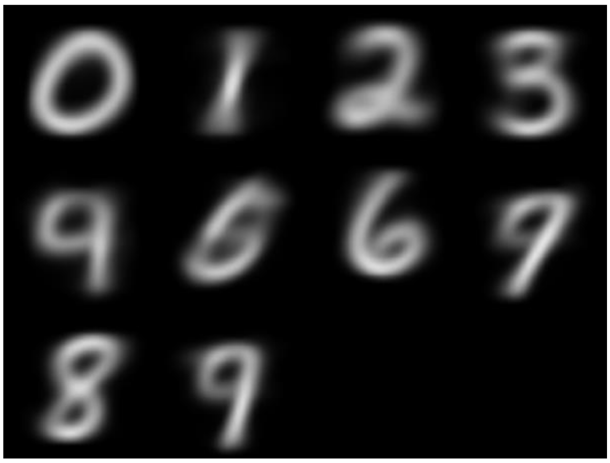


[1] http://ilpubs.stanford.edu:8090/778/1/2006-13.pdf

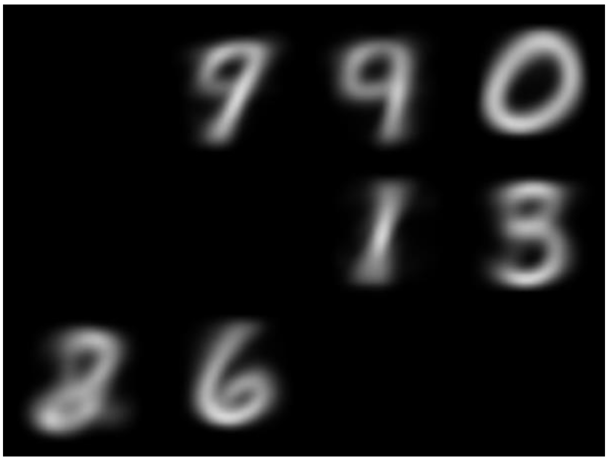




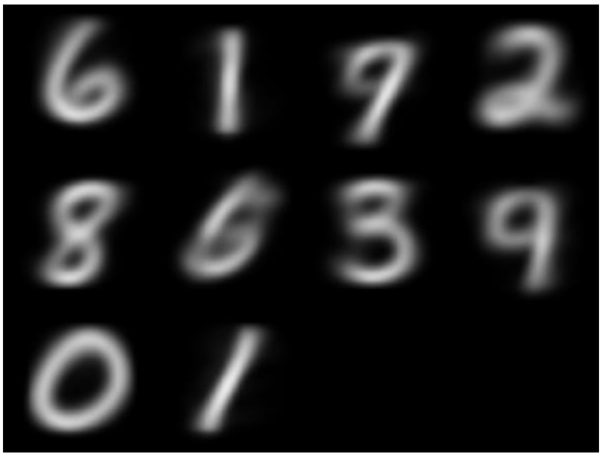
The final centroids were also plotted and compared with regards to their similarity to the real class labels. *Means* yielded the most similar images whereas *Random* yielded the most dissimilar images. *K++* and *Sample* achieved comparable results.



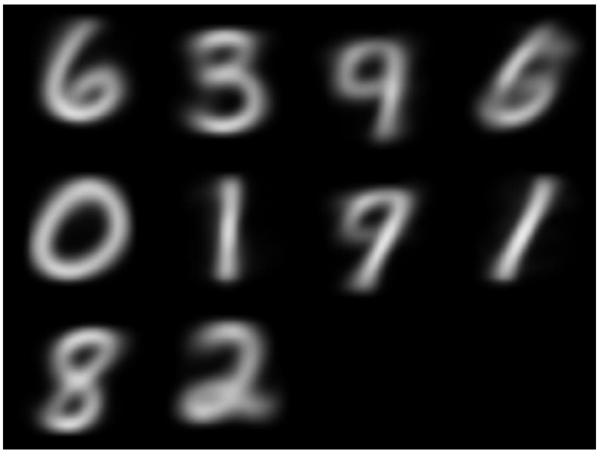
Means



Random



K++



Sample