Question 1 : (30 total points) Image data analysis with PCA

In this question we employ PCA to analyse image data

1.1 (3 points) Once you have applied the normalisation from Step 1 to Step 4 above, report the values of the first 4 elements for the first training sample in `Xtrn_nm`, i.e. `Xtrn_nm[0,:]` and the last training sample, i.e. `Xtrn_nm[-1,:]`.

The first 4 elements for the first training sample and the last training sample in `Xtrn_nm` are the same, that is (from first element to fourth element): -3.137×10^{-6}, -2.268×10^{-5}, -1.180×10^{-4}, -4.071×10^{-4}.

1.2 (4 points) Using `Xtrn` and Euclidean distance measure, for each class, find the two closest samples and two furthest samples of that class to the mean vector of the class.

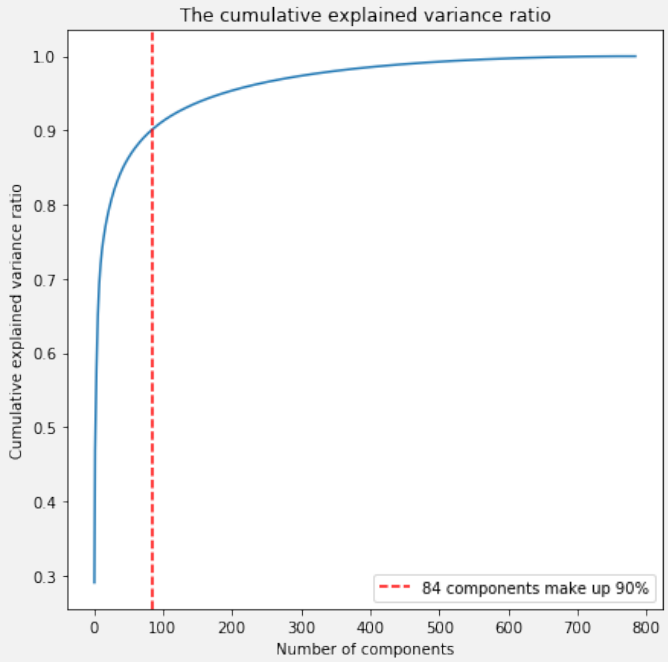
The images of the mean vectors are more blurry than that of the samples because it is the mean of all the samples in the class. The images for the two closest samples look similar to that of the mean vector and the images for the two furthest samples look different from that of the mean vector.

1.3 (3 points) Apply Principal Component Analysis (PCA) to the data of `Xtrn_nm` using sklearn.decomposition.PCA, and report the variances of projected data for the first five principal components in a table. Note that you should use `Xtrn_nm` instead of `Xtrn`.

The variances of projected data for the first 5 principal components are given in the table below.

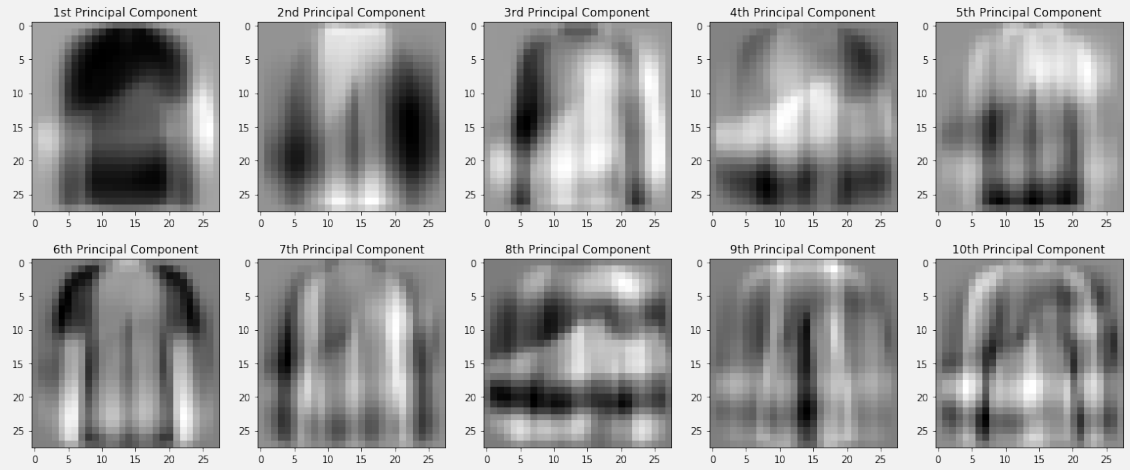
Principal Component	Variance
1	19.81
2	12.11
3	4.11
4	3.38
5	2.62

1.4 (3 points) Plot a graph of the cumulative explained variance ratio as a function of the number of principal components, K, where $1 \leq K \leq 784$. Discuss the result briefly.



The cumulative explained variance ratio increases logarithmically as the number of principal components increase, which is as expected because the components are in descending order of their variances. The number of components which collectively explain at least 90% of the total variance is 84 (shown above in red dotted line).

1.5 (4 points) Display the images of the first 10 principal components in a 2-by-5 grid, putting the image of 1st principal component on the top left corner, followed by the one of 2nd component to the right. Discuss your findings briefly.



The first principal component (PC) accounts for as much of the variability in the data as possible, and the following components account for as much of the remaining variability as possible. Each additional dimension added to the PCA captures less and less of the variance in the model. With more PCs used to reconstruct an image, the closer it will be to the original image as each PC adds information to distinguish between classes. For example, from our plots above, using second PC in addition of first PC will allow images of trousers to be distinguished better.

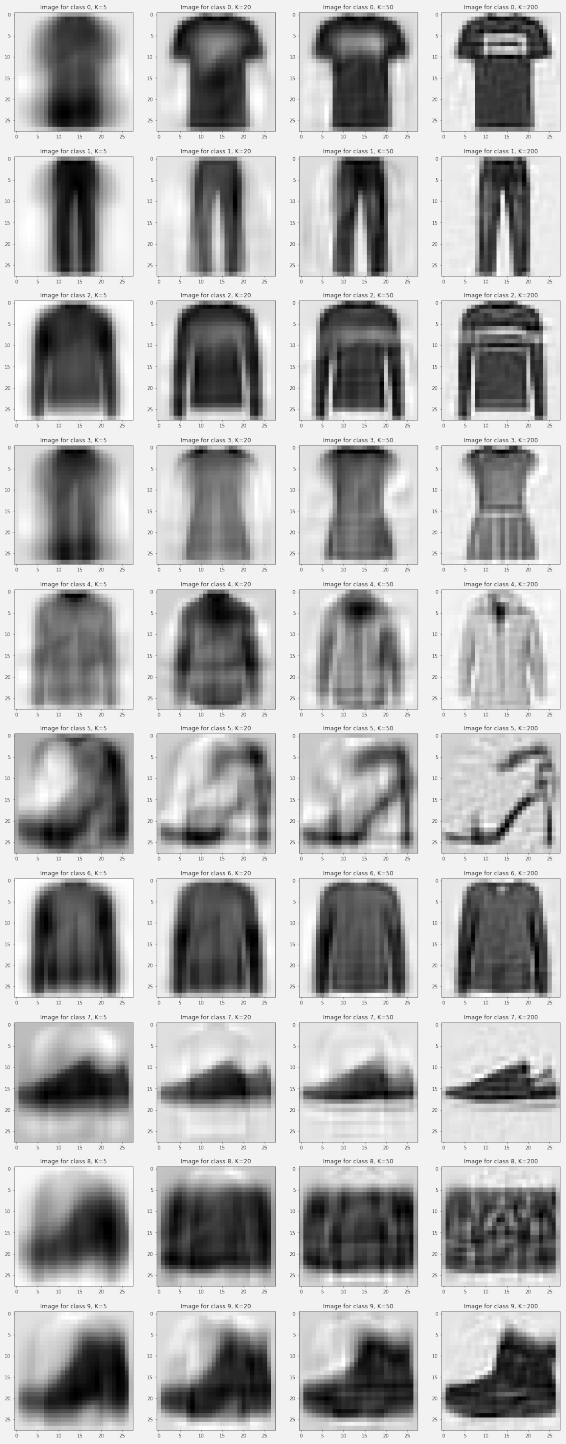
1.6 (5 points) Using `Xtrn_nm`, for each class and for each number of principal components $K = 5, 20, 50, 200$, apply dimensionality reduction with PCA to the first sample in the class, reconstruct the sample from the dimensionality-reduced sample, and report the Root Mean Square Error (RMSE) between the original sample in `Xtrn_nm` and reconstructed one.

The RMSE between the original sample in `Xtrn_nm` and reconstructed one for each class and for `K` principal components are shown below:

	K			
	5	20	50	200
Class 0	0.256	0.150	0.127	0.061
Class 1	0.198	0.140	0.096	0.036
Class 2	0.199	0.146	0.124	0.080
Class 3	0.146	0.107	0.083	0.056
Class 4	0.118	0.103	0.088	0.047
Class 5	0.181	0.159	0.143	0.090
Class 6	0.129	0.096	0.072	0.047
Class 7	0.166	0.128	0.107	0.063
Class 8	0.223	0.145	0.124	0.091
Class 9	0.184	0.151	0.122	0.072

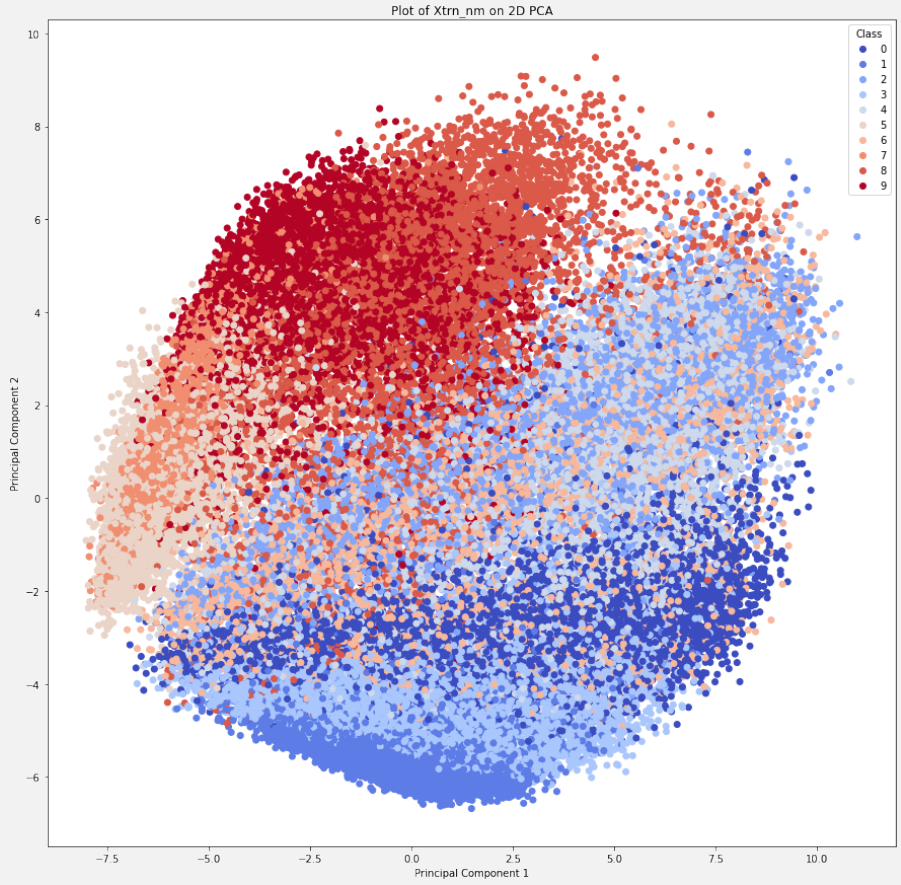
1.7 (4 points) Display the image for each of the reconstructed samples in a 10-by-4 grid, where each row corresponds to a class and each row column corresponds to a value of $K = 5,\ 20,\ 50,\ 200$.

As value of K increases, the image gets clearer and can better distinguish between the classes as a higher percentage of the variance is retained. However, the image for class 8 is still not reconstructed well even with K=200.



1.8 (4 points) Plot all the training samples (`Xtrn_nm`) on the two-dimensional PCA plane you obtained in Question 1.3, where each sample is represented as a small point with a colour specific to the class of the sample. Use the 'coolwarm' colormap for plotting.

With the 'coolwarm' colormap, it can be seen that there is some separation between the 'blue' classes (classes 0 to 4) and the 'red' classes (classes 5 to 9), where the red classes appear to cluster on the top left region while the blue classes appear to cluster on the bottom right region. This could be because classes 0-4 are tops/pants while classes 5-9 are shoes/bags with the exception of class 6 (which is a label for shirts). As a result, it can be seen that a lot of class 6 points seem to overlap in the region where blue points seem to be the dominate.



Question 2 : (25 total points) Logistic regression and SVM

In this question we will explore classification of image data with logistic regression and support vector machines (SVM) and visualisation of decision regions.

2.1 (3 points) Carry out a classification experiment with multinomial logistic regression, and report the classification accuracy and confusion matrix (in numbers rather than in graphical representation such as heatmap) for the test set.

The classification accuracy for the test set is 0.840.
The confusion matrix for the test set (where the columns indicate predicted labels and the rows indicate actual labels) is:

	0	1	2	3	4	5	6	7	8	9
0	819	3	15	50	7	4	89	1	12	0
1	5	953	4	27	5	0	3	1	2	0
2	27	4	731	11	133	0	82	2	9	1
3	31	15	14	866	33	0	37	0	4	0
4	0	3	115	38	760	2	72	0	10	0
5	2	0	0	1	0	911	0	56	10	20
6	147	3	128	46	108	0	539	0	28	1
7	0	0	0	0	0	32	0	936	1	31
8	7	1	6	11	3	7	15	5	945	0
9	0	0	0	1	0	15	1	42	0	941

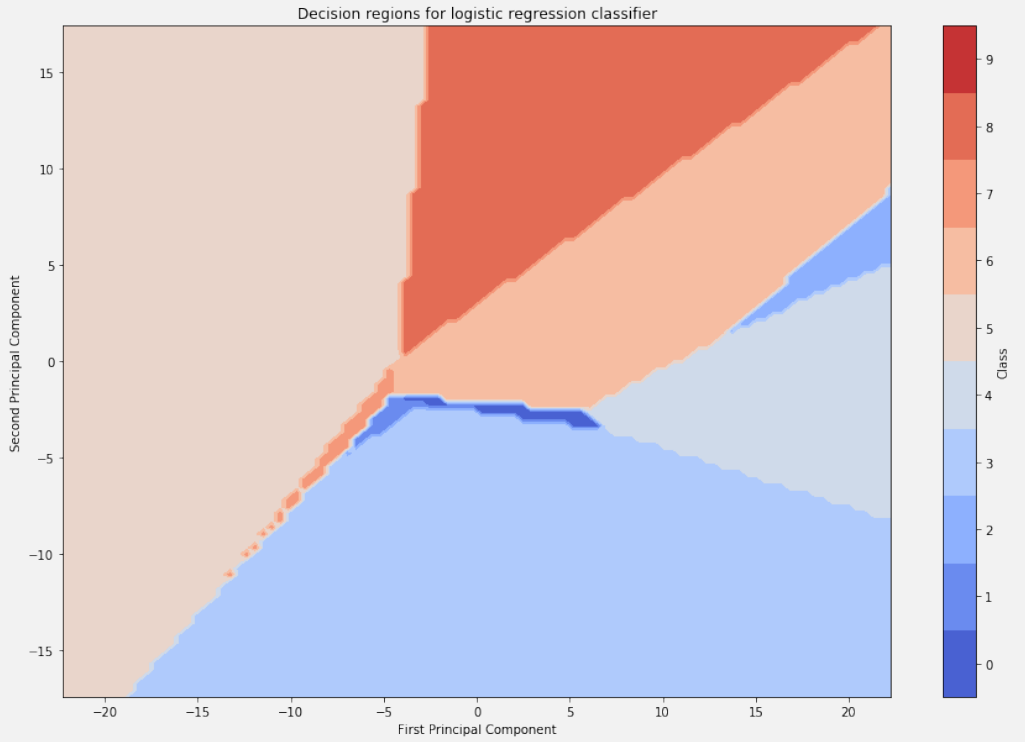
Question 2 continued on next page...

2.2 (3 points) Carry out a classification experiment with SVM classifiers, and report the mean accuracy and confusion matrix (in numbers) for the test set.

The classification accuracy for the test set is 0.846.
The confusion matrix for the test set (where the columns indicate predicted labels and the rows indicate actual labels) is:

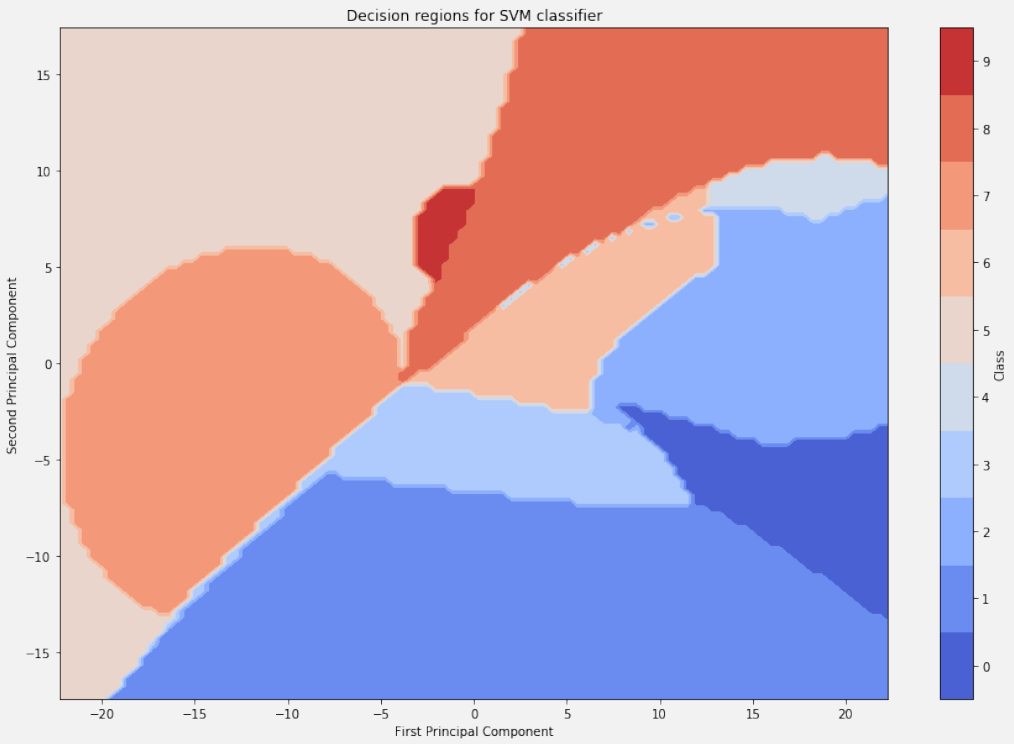
	0	1	2	3	4	5	6	7	8	9
0	845	2	8	51	4	4	72	0	14	0
1	4	951	7	31	5	0	1	0	1	0
2	15	2	748	11	137	0	79	0	8	0
3	32	6	12	881	26	0	40	0	3	0
4	1	0	98	36	775	0	86	0	4	0
5	0	0	0	1	0	914	0	57	2	26
6	185	1	122	39	95	0	533	0	25	0
7	0	0	0	0	0	34	0	925	0	41
8	3	1	8	5	2	4	13	4	959	1
9	0	0	0	0	0	22	0	47	1	930

2.3 (6 points) We now want to visualise the decision regions for the logistic regression classifier we trained in Question 2.1.



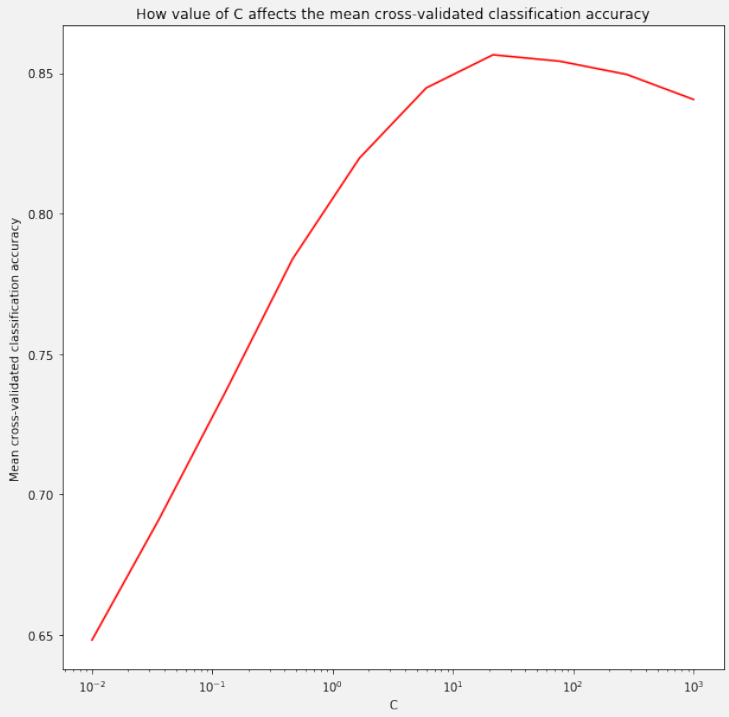
From the plot, it can be seen that the classifier does not classify any points to be of class 9. As expected, the decision boundaries are straight lines. There are some similarities with the plot we obtained for Q1.8, where the red classes (classes 5-9) are on the top left region and the blue classes (classes 0-4) are on the bottom right region.

2.4 (4 points) Using the same method as the one above, plot the decision regions for the SVM classifier you trained in Question 2.2. Comparing the result with that you obtained in Question 2.3, discuss your findings briefly.



Compared to the straight line decision boundaries obtained in Q2.3, the decision boundaries for SVM classifier are non-linear. It also classifies some points to be of class 9. Furthermore, the plot above is more similar to the plot we obtained for Q1.8 than the plot for Q2.3, particularly the positions for the decision regions for classes 0, 1, 7 and 9. In other words, it better captures the shape and position of clusters of each class in plot Q1.8.

2.5 (6 points) We used default parameters for the SVM in Question 2.2. We now want to tune the parameters by using cross-validation. To reduce the time for experiments, you pick up the first 1000 training samples from each class to create `Xsmall`, so that `Xsmall` contains 10,000 samples in total. Accordingly, you create labels, `Ysmall`.



The plot has a single maxima, where the highest obtained mean accuracy score is 0.857 and the value of C which yielded this is 21.54.

2.6 (3 points) Train the SVM classifier on the whole training set by using the optimal value of C you found in Question 2.5.

The classification accuracy on the training set and test set is given below:

	Accuracy
Training Set	0.908
Testing Set	0.877

Question 3 : (20 total points) Clustering and Gaussian Mixture Models

In this question we will explore K-means clustering, hierarchical clustering, and GMMs.

3.1 (3 points) Apply k-means clustering on `Xtrn` for $k = 22$, where we use sklearn.cluster.KMeans with the parameters `n_clusters=22` and `random_state=1`. Report the sum of squared distances of samples to their closest cluster centre, and the number of samples for each cluster.

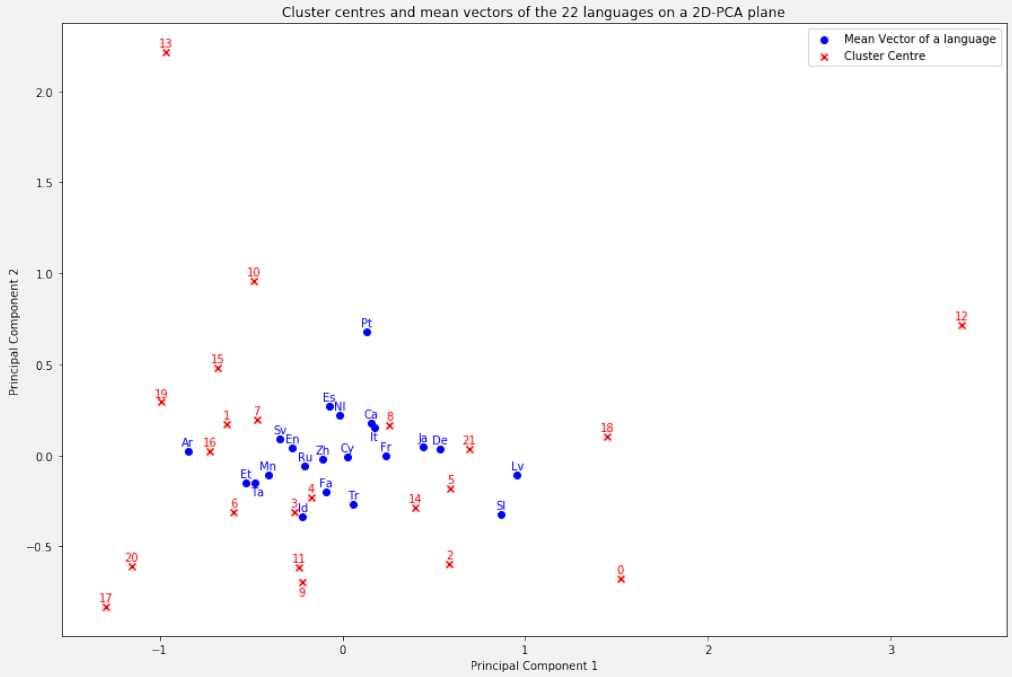
The sum of squared distances of samples to their closest cluster centre is 38185.8. The number of samples for each cluster is given in the table below:

Cluster	Number of samples
0	1018
1	1125
2	1191
3	890
4	1162
5	1332
6	839
7	623
8	1400
9	838
10	659
11	1276
12	121
13	152
14	950
15	1971
16	1251
17	845
18	896
19	930
20	1065
21	1466

Question 3 continued on next page...

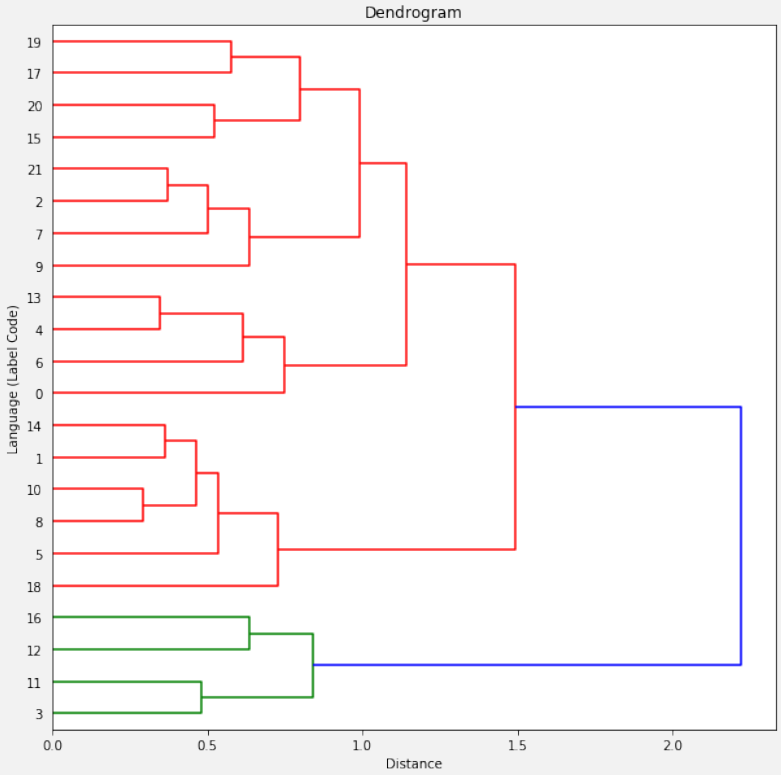
3.2 (3 points) Using the training set only, calculate the mean vector for each language, and plot the mean vectors of all the 22 languages on a 2D-PCA plane, where you apply PCA on the set of 22 mean vectors without applying standardisation. On the same figure, plot the cluster centres obtained in Question 3.1.

The numbers above/below the blue points (which represent the mean vectors of a language) correspond to the abbreviation of the language.



There are some cluster centres that are close to the mean vectors of a language (i.e. clusters 3, 4 and 8), but there are clearly also some cluster centres that are relatively far away from all the mean vectors (i.e. clusters 12 and 13). The mean vectors seem to be relatively close to each other, considering there are cluster centers that are far from them.

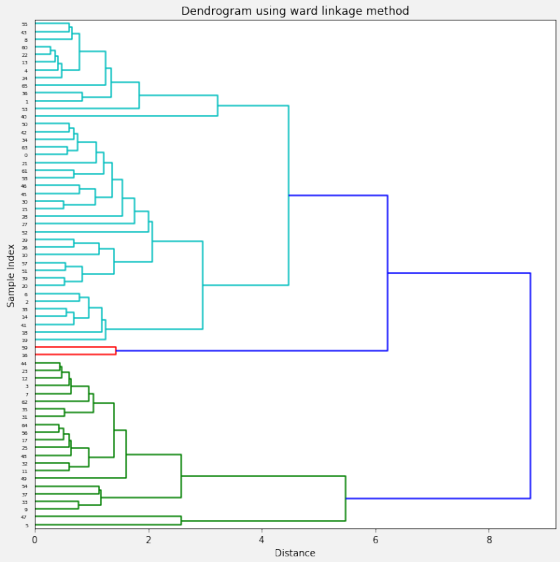
3.3 (3 points) We now apply hierarchical clustering on the training data set to see if there are any structures in the spoken languages.

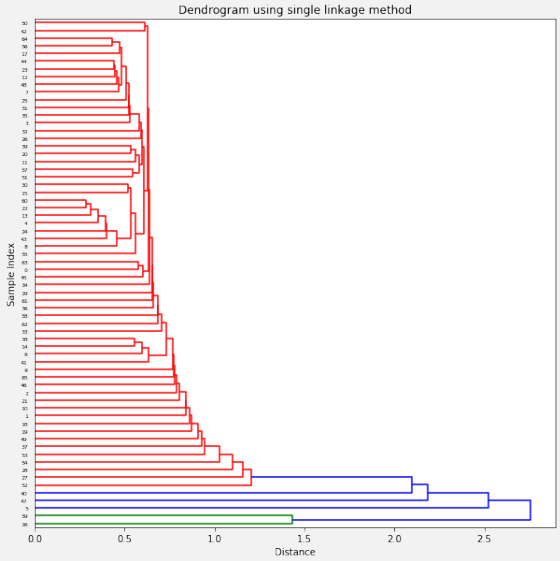


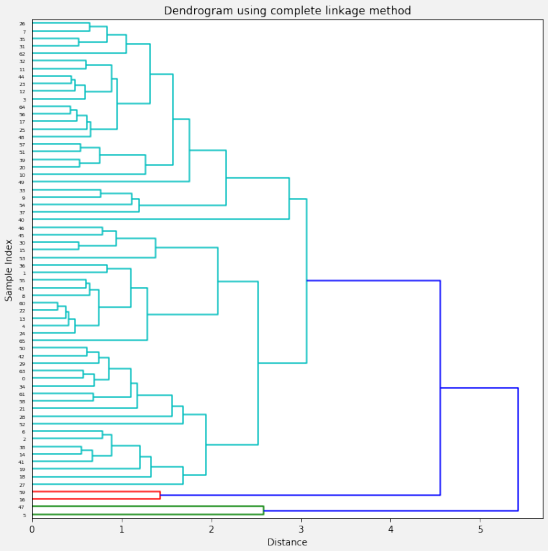
From the dendrogram, it can be seen that the languages whose label codes are 8 and 10 are the most similar. The clusters which include languages whose label codes are 3, 11, 12, 16 (denoted above in green) seem to be relatively dissimilar to the other languages.

3.4 (5 points) We here extend the hierarchical clustering done in Question 3.3 by using multiple samples from each language.

The dendrograms for each method is given below:



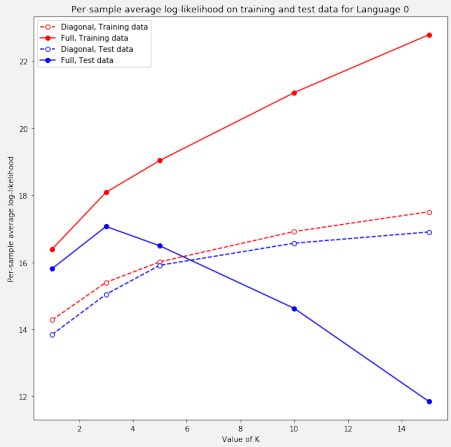




It can be seen that samples 22 and 60 are the most similar as it has the shortest distance in all 3 dendograms. Furthermore, the cluster that includes samples 56 and 59 are relatively far from other clusters. According to the dendograms using single linkage and complete linkage, the cluster that includes sample 5 and 47 also seem to be relatively far from other clusters.

Question 3 continued on next page...

3.5 (6 points) We now consider Gaussian mixture model (GMM), whose probability distribution function (pdf) is given as a linear combination of Gaussian or normal distributions, i.e.,



	Training Set		Testing Set	
	Diagonal	Full	Diagonal	Full
K=1	14.28	16.39	13.84	15.81
K=3	15.40	18.09	15.04	17.07
K=5	16.01	19.04	15.91	16.49
K=10	16.92	21.06	16.57	14.62
K=15	17.50	22.79	16.90	11.85

The lines show an increasing trend except for full covariance matrix on the test data, where the per-sample average log likelihood decreases after K=3. This may be caused by an overfitting of training data. On the training set, the likelihood using full covariance matrix is higher than that using diagonal covariance matrix.