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 values of the first 4 elements for the first training sample in Xtrn_nm [-3.137e-06, -2.268e-05, -1.180e-04, -4.071e-04]
The values of the first 4 elements for the last training sample in Xtrn_nm [-3.137e-06, -2.268e-05, -1.180e-04, -4.071e-04]
The values are the same. Values are rounded to 3 decimal places.

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 in the most left column correspond to the main features in each class and they are the most significant features even though the images are not clear enough. Besides, it can be observed that the images of the first (1st column) and second (2nd column) closest samples to the mean are similar to the images of the mean vectors, whereas the first (4th column) and second (3rd column) furthest samples to the mean have nothing to do with the images of the mean vectors.

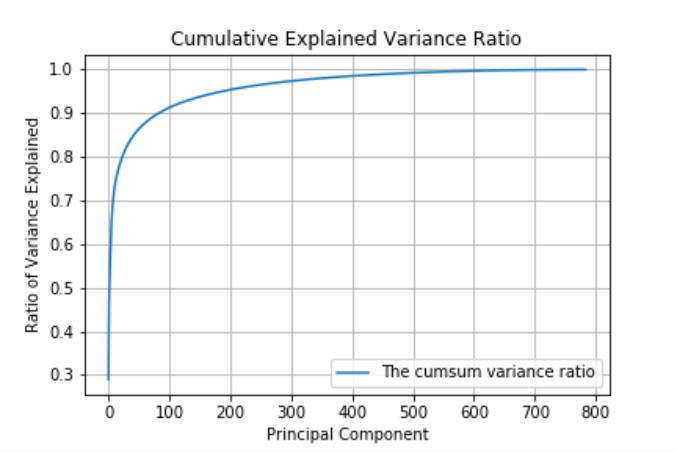
1.3 (3 points) Apply Principal Component Analysis (PCA) to the data of `Xtrn_nm` using sklearn.decomposition.PCA, and report the variances of projected data for the first five principal components in a table. Note that you should use `Xtrn_nm` instead of `Xtrn`.

	1st pc	2nd PC	3rd PC	4th PC	5th PC
Variance	19.8098	12.1122	4.1062	3.3818	2.6248

The variances of projected data for the first five principal components are rounded to 4 decimal places.

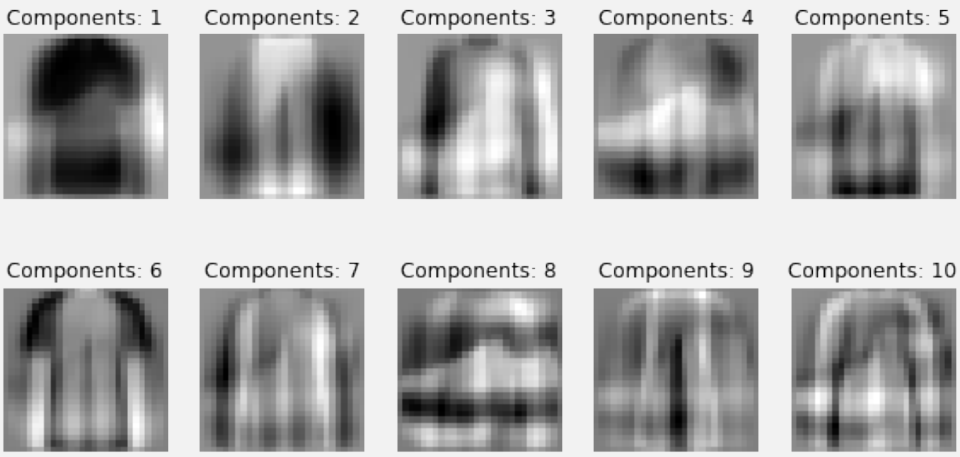
Question 1 continued on next page...

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 ratio rapidly increases to approximately 0.7 at first and then increases slowly. The ratio seems like not increasing much after the principal component being equal to 500. Thus, for the first few principal components, they have a great gap between their variances and the gap starts to reduce after that. Moreover, the graph indicates that for the first 100 components, they contain approximately 90% of the cumulative variance, which means they are more significant for showing information on the graph. Consequently, the greater principal component is, the lower variance would be and with the greater principal component, the difference between each principal component would be smaller.

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 black and white pixels represent high variance whereas the grey pixels represent low variance. Besides, positive values are in the black pixels and negative values are in the white values. So for instance, for the second image that contains a black sleeve and a white trouser. They are distinguished clearly because the signs of the values that refer to each item are the opposite. Thus, the first few principal components are better to classify and distinguish what the image shows as they have greater variances and the gap of variance between each principal component is relatively large. Whereas for the images with the higher principal component, they are not doing well in classifying the features in the images as they have relatively lower variances.

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.

	K = 5	K = 20	K = 50	K = 200
Class 0	0.256	0.150	0.127	0.061
Class 1	0.198	0.140	0.095	0.038
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	0.089
Class 6	0.129	0.096	0.072	0.046
Class 7	0.165	0.128	0.107	0.064
Class 8	0.223	0.145	0.124	0.091
Class 9	0.185	0.151	0.122	0.072

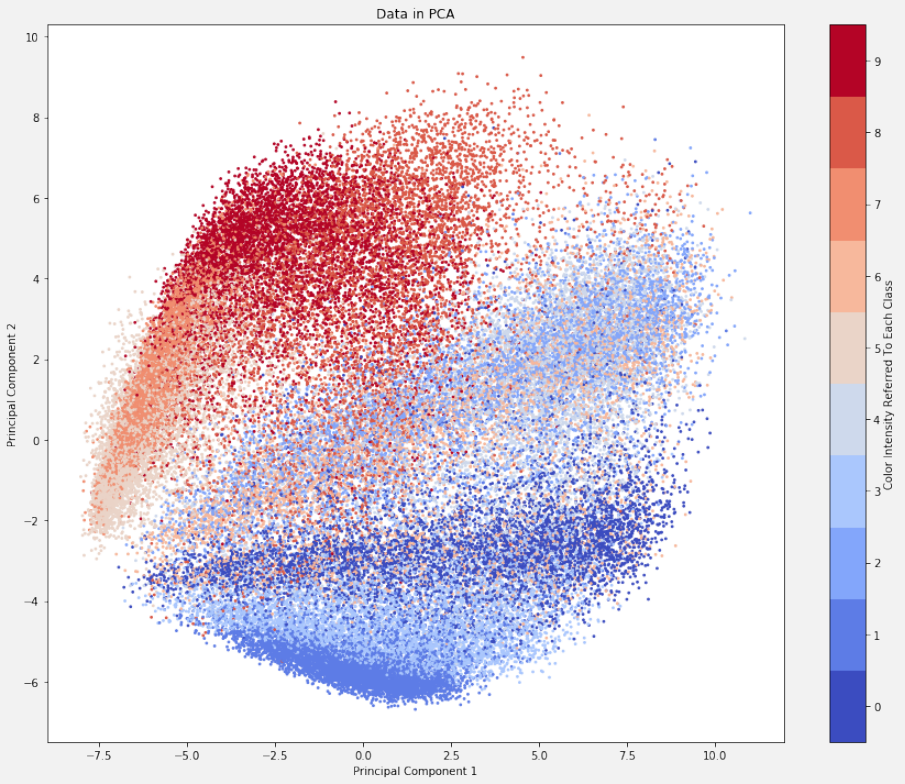
All the values in the table above are rounded to 3 decimal places.

Question 1 continued on next page...

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 $K = 5,\ 20,\ 50,\ 200$.

The images with a smaller number of principal components are not as clear as those with a larger number of principal components. The reason is that for a class with more number of principal components, it would have a higher variance and consequently show more features on the image. For instance, the cumulative variance for K equals to 200 is approximately 0.95 and there are approximately 90% of variances are included, whereas the cumulative variance for K equals to 50 is about 0.85. So the gap of the cumulative variance decreases as K increases and the variance defines the clarity of an image. Therefore, the larger K is, the clearer the image would be.

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.



From the plot, the classes in warm dots are closer to the second component whereas the classes in cool dots are closer to the first component. The dots from each class stick together. Besides, it can be observed that the scattered classes are mixed, which would lead to loss of information. This is caused by dimensional reduction from a higher dimension and it is not the best choice for a sample of 784 features. Thus, I reckon that applying multi-dimensional PCA would be better in this case. Moreover, the separation of the nine classes is clear to observe even though some classes are mixed. Class 0, class 1, class 2 and class 3 are scattered at the bottom whereas class 7, class 8 and class 9 are located at the top left corner. So the classification gives a clear visualisation of shoes and bags being at the bottom while clothes being at the top left corner.

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.

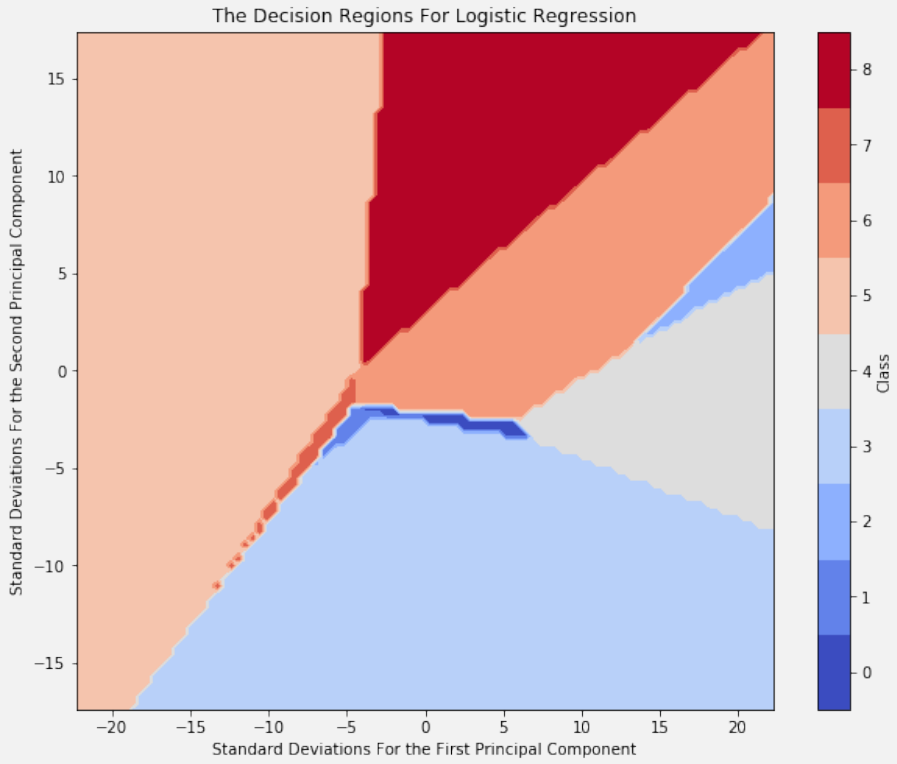
2.1 (3 points) Carry out a classification experiment with multinomial logistic regression, and report the classification accuracy and confusion matrix (in numbers rather than in graphical representation such as heatmap) for the test set.

```
Accuracy: 84.01%
Confusion matrix:
 [[819   3  15  50   7   4  89   1  12   0]
 [  5 953   4  27   5   0   3   1   2   0]
 [ 27   4 731  11 133   0  82   2   9   1]
 [ 31  15  14 866  33   0  37   0   4   0]
 [  0   3 115  38 760   2  72   0  10   0]
 [  2   0   0   1   0 911   0  56  10  20]
 [147   3 128  46 108   0 539   0  28   1]
 [  0   0   0   0   0  32   0 936   1  31]
 [  7   1   6  11   3   7  15   5 945   0]
 [  0   0   0   1   0  15   1  42   0 941]]
```

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.

```
Accuracy: 84.61%
Confusion matrix:
 [[845   2   8  51   4   4  72   0  14   0]
 [  4 951   7  31   5   0   1   0   1   0]
 [ 15   2 748  11 137   0  79   0   8   0]
 [ 32   6  12 881  26   0  40   0   3   0]
 [  1   0  98  36 775   0  86   0   4   0]
 [  0   0   0   1   0 914   0  57   2  26]
 [185   1 122  39  95   0 533   0  25   0]
 [  0   0   0   0   0  34   0 925   0  41]
 [  3   1   8   5   2   4  13   4 959   1]
 [  0   0   0   0   0  22   0  47   1 930]]
```

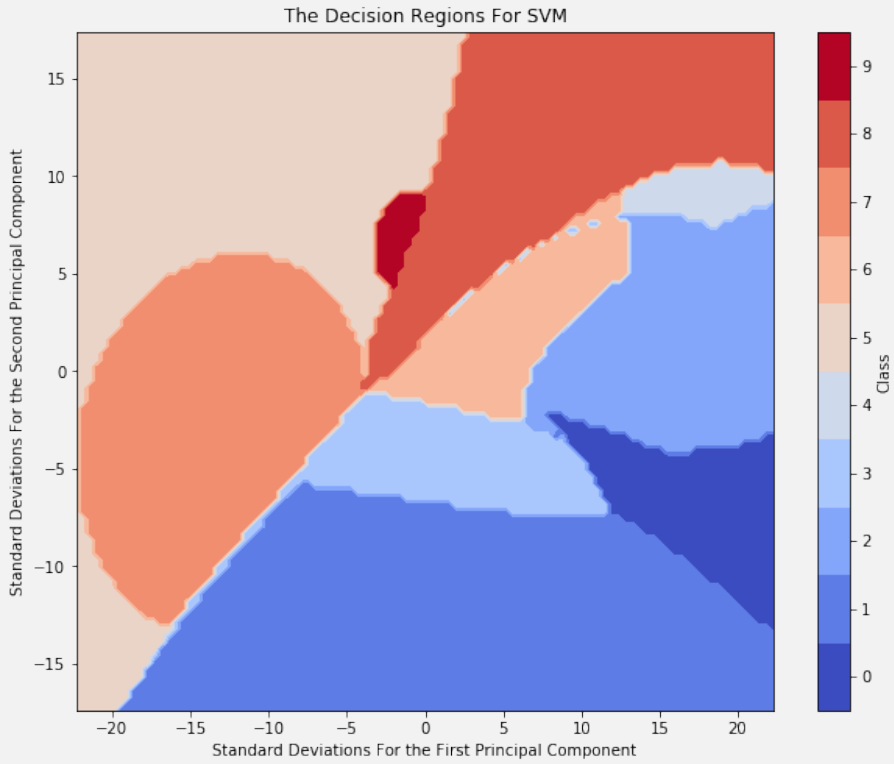
2.3 (6 points) We now want to visualise the decision regions for the logistic regression classifier we trained in Question 2.1.



The plot of the decision regions for the logistic regression classifier has one class less which leads to loss of information. This is caused by dimensional reduction. Additionally, the decision regions have linear boundaries only. For example, class 0, class 1 and class 7 are almost not shown on the graph. As a result, the logistic regression classifier is not one of the best choices.

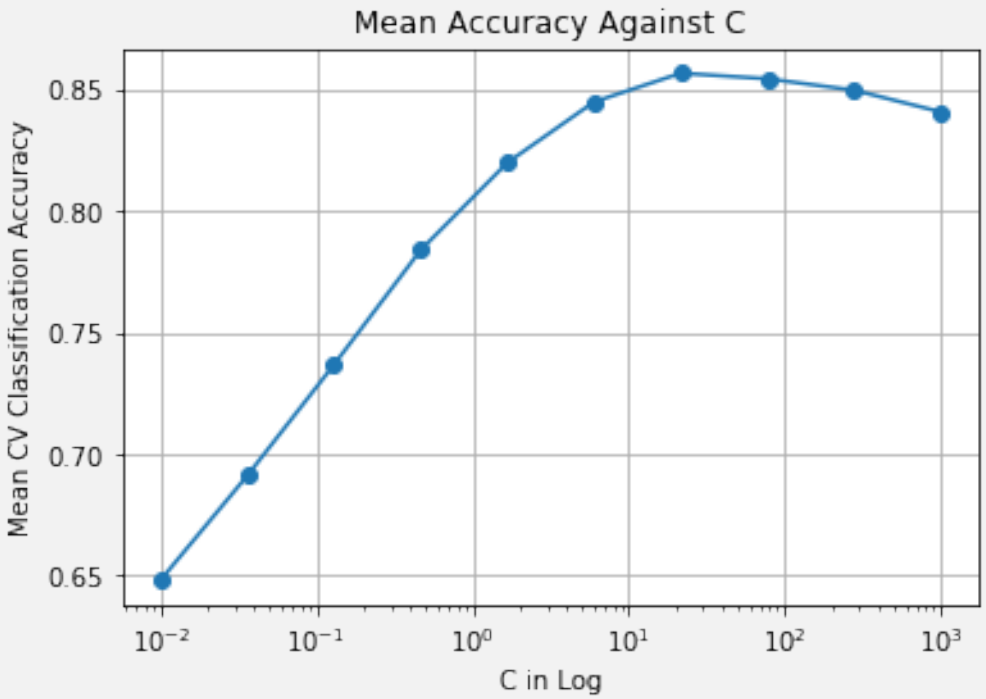
Question 2 continued on next page...

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.



From the plot of the decision regions, the SVM classifier seems to work better than the logistic regression classifier as it classifies the entire ten classes of which the logistic regression classifier could not have done. The decision regions are bounded by curves instead of straight lines anymore. The SVM classifier separate the plane by maximising the the distance between each points. Therefore, the SVM classifier would be a better choice for showing more information and details.

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 highest mean accuracy score is 0.857 with the value of C being 21.544 rounded to 3 decimal places.

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.

Training Accuracy: 90.84%
Tesing Accuracy: 87.65%

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

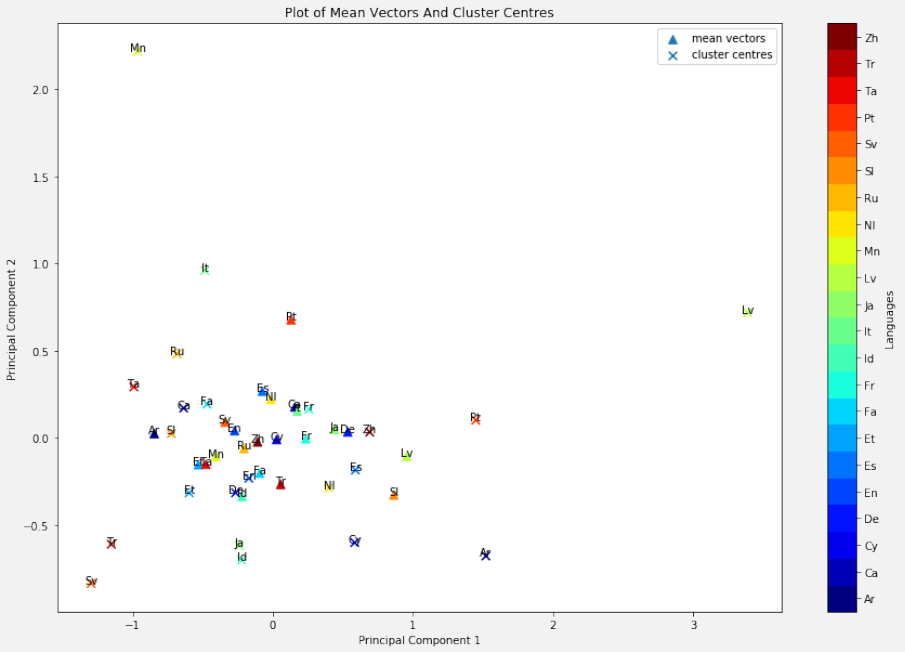
In this question we will explore K-means clustering, hierarchical clustering, and GMMs.

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.

Sum of squared distances	38185.817
Number of samples for cluster 0	1018
Number of samples for cluster 1	1125
Number of samples for cluster 2	1191
Number of samples for cluster 3	890
Number of samples for cluster 4	1162
Number of samples for cluster 5	1332
Number of samples for cluster 6	839
Number of samples for cluster 7	623
Number of samples for cluster 8	1400
Number of samples for cluster 9	838
Number of samples for cluster 10	659
Number of samples for cluster 11	1276
Number of samples for cluster 12	121
Number of samples for cluster 13	152
Number of samples for cluster 14	950
Number of samples for cluster 15	1971
Number of samples for cluster 16	1251
Number of samples for cluster 17	845
Number of samples for cluster 18	896
Number of samples for cluster 19	930
Number of samples for cluster 20	1065
Number of samples for cluster 21	1466

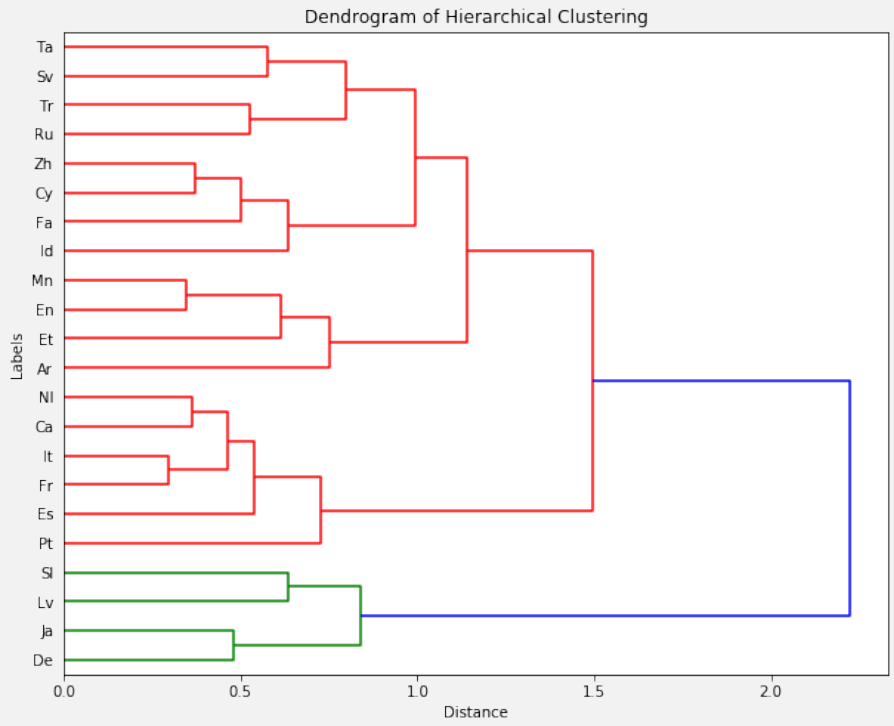
Question 3 continued on next page...

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.



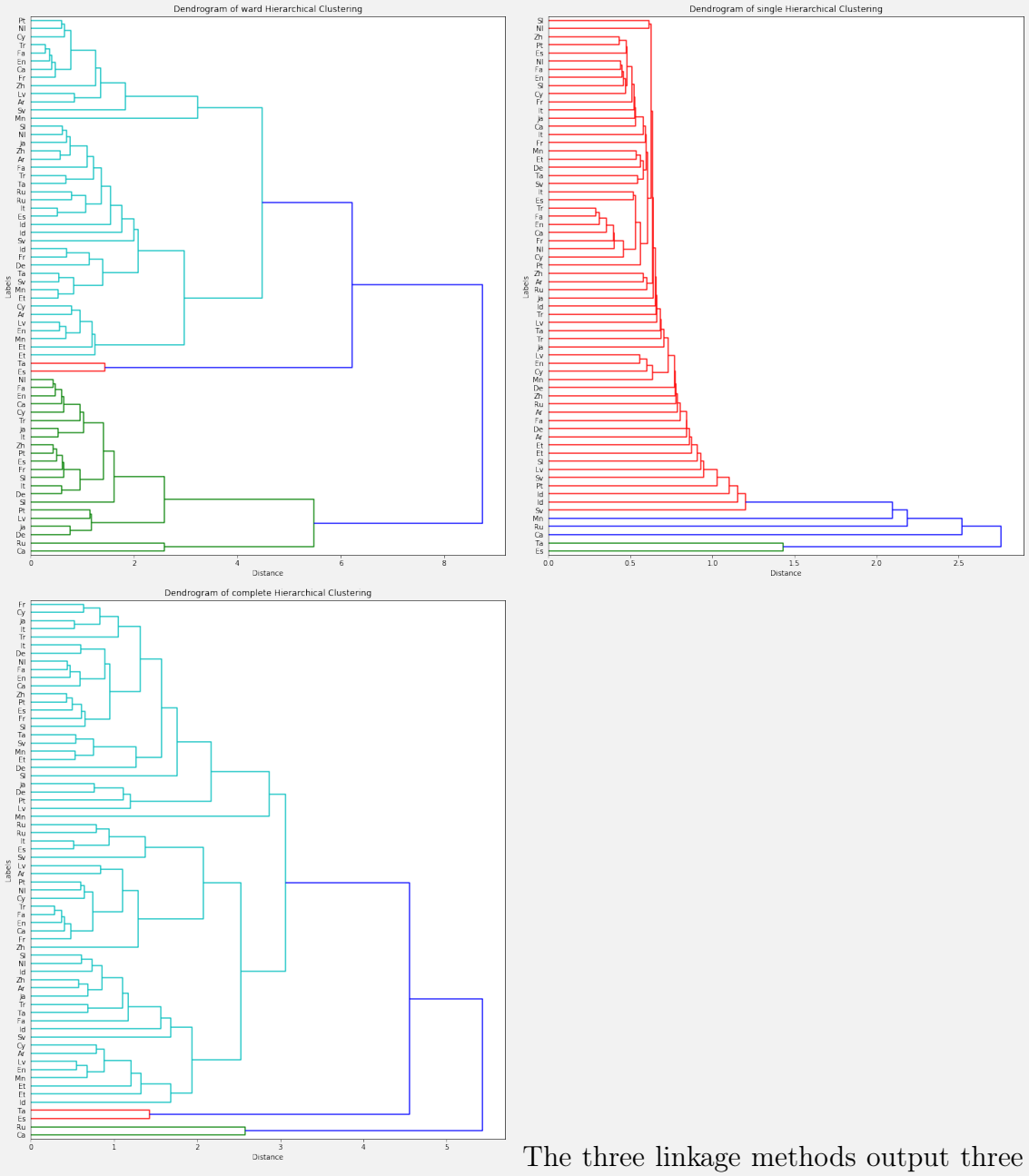
The mean vectors and cluster centres are not similar as the cluster centres are separated more widely whereas the mean vectors are more concentrated. However, most of the mean vectors are concentrated around the origin (0,0). I suppose there probably exist outliers among the dataset because the cluster centres of language 'Mn' and 'Lv' are far from others. K-means clustering is sensitive to outliers so it is not a choice for fitting the 22 languages. Additionally, I reckon that K-means clustering is not working as expected to divide each language after PCA dimension reduction. The performance could be improved by applying a higher parameter K in PCA to avoid loss of crucial information.

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.



The languages are clustered according to their mean vectors. It can be observed that the higher the distance threshold, the fewer clusters the method will output. From the clusters, the bottom four languages in green (Sl, Lv, Ja, De) are grouped with one cluster, where the rest of the languages in red are clustered together.

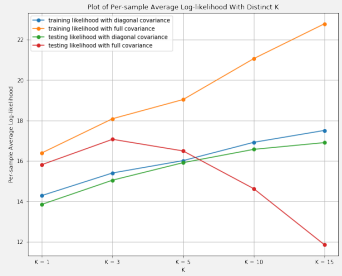
3.4 (5 points) We here extend the hierarchical clustering done in Question 3.3 by using multiple samples from each language.



The three linkage methods output three individual images with different clusters. But whatever the method is, Es and Ta are always in the same cluster, which the algorithm believes they are similar and different from other languages. The distance threshold between each cluster is different for the three methods. 'Single' has the smallest distance, 'Ward' has the largest distance and 'Complete' is in the middle.

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.,



Data	Type	K = 1	K = 3	K = 5	K = 10	K = 15
Training	Diagonal	14.280	15.398	16.010	16.917	17.505
Testing	Diagonal	13.843	15.041	15.909	16.567	16.902
Training	Full	16.394	18.086	19.036	21.062	22.786
Testing	Full	15.811	17.066	16.489	14.622	11.848

The red line (testing likelihood with full covariance matrix) has expected value for K equals to 1 and 3. However, the line performs as reduction for K equals to the rest three values. For full type covariance matrix, the Gaussian mixture mode is working well when fitting the training data, however, it seems not working well for the testing data with higher number of mixture components. The size of Gaussian mixture model will be reduced and there will be loss of generality due to large number of components. For diagonal type covariance matrix, the model is working similar to both of the training data and the testing data. There will be loss of correlation while applying the diagonal type. So the results will not be sensitive as the full type and will be less affected.