

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

In this question we employ PCA to analyse image data

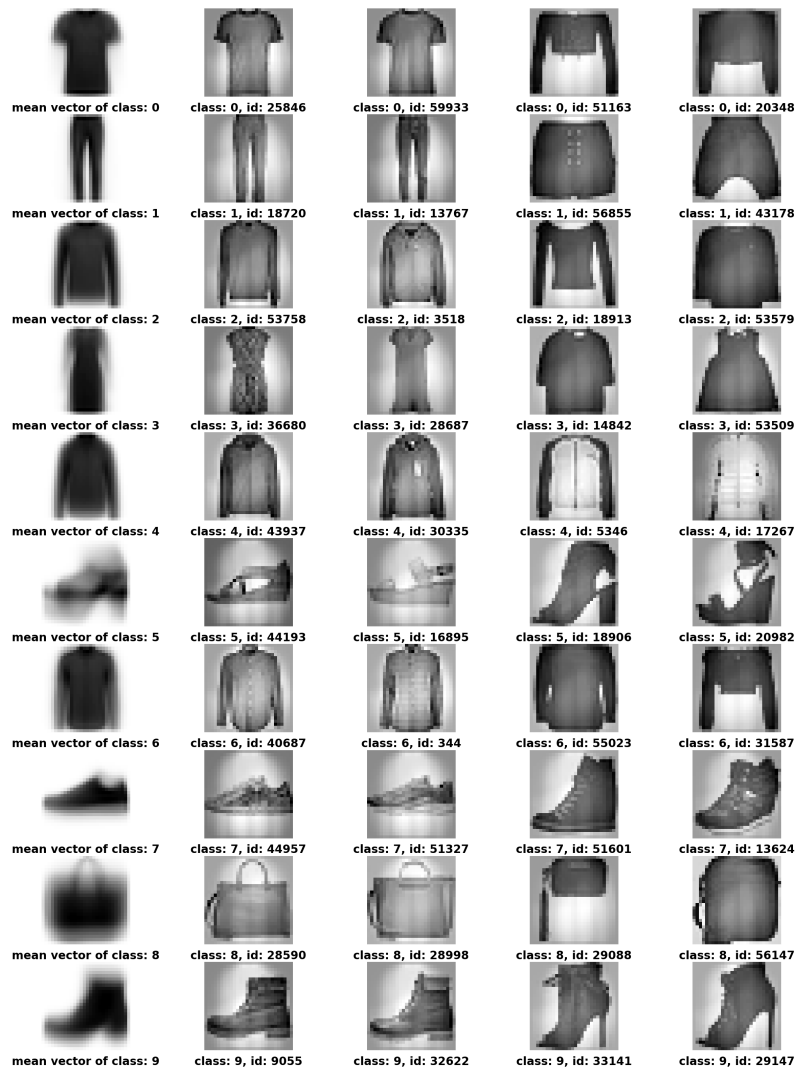
**1.1** (3 points) Once you have applied the normalisation from Step 1 to Step 4 above, report the values of the first 4 elements for the first training sample in `Xtrn_nm`, i.e. `Xtrn_nm[0,:]` and the last training sample, i.e. `Xtrn_nm[-1,:]`.

The first four elements for the first training sample in `Xtrn_nm` is:  $-3.137 * 10^{-6}$ ,  $-2.268 * 10^{-5}$ ,  $-1.180 * 10^{-4}$ ,  $-4.071 * 10^{-4}$

The first four elements for the last training sample in `Xtrn_nm` is:  $-3.137 * 10^{-6}$ ,  $-2.268 * 10^{-5}$ ,  $-1.180 * 10^{-4}$ ,  $-4.071 * 10^{-4}$

First four elements are the same in the first training sample and in the last training sample.

**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.



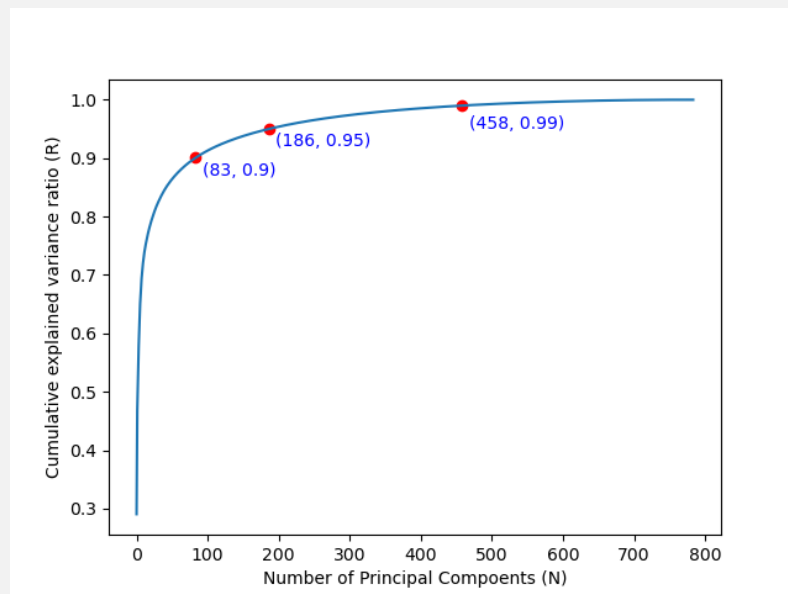
The closet two samples are typical cases in that class since they are close to the mean vector. However the furthest two samples are special cases in that class(not common). Sample id and class number are shown below each image.

**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`.

Principal Components(PC)	Explained Variances
PC 1	19.81
PC 2	12.11
PC 3	4.106
PC 4	3.382
PC 5	2.625

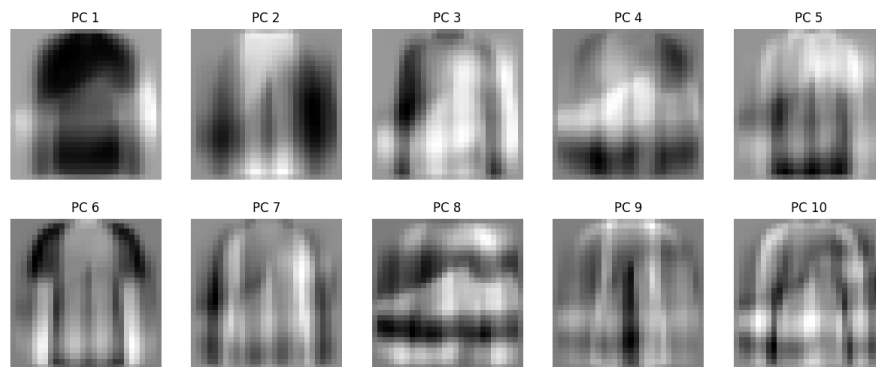
Each answer reserves four effective bit numbers.

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.



First 83 principal components explain at least 90% of the total variance. First 186 principal components explain at least 95% of the total variance. First 458 principal components explain at least 99% of the total variance. When the number of selected principal components are small, they explain little to the total variance. Total 784 components explain 100% of the total variance.

**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.
















Each image represents its main features at black and dark grey areas. First principal component mainly represent the feature of sleeves, as well as separating long sleeves and shoes since they share the largest variance on the first principal component plane(line). Likely, the second PC separates pants and shoes. However, when the id of PC becomes larger, the component seems to contain features of all classes which are mixed and undifferentiated, and it is explained by a small variance between each class.

**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.

Class	K = 5	K = 20	K = 50	K = 200
0	0.256	0.150	0.127	0.062
1	0.198	0.141	0.095	0.037
2	0.199	0.146	0.122	0.079
3	0.146	0.107	0.084	0.056
4	0.118	0.103	0.088	0.046
5	0.181	0.159	0.142	0.090
6	0.129	0.096	0.073	0.046
7	0.166	0.127	0.107	0.063
8	0.223	0.145	0.124	0.092
9	0.184	0.151	0.122	0.072

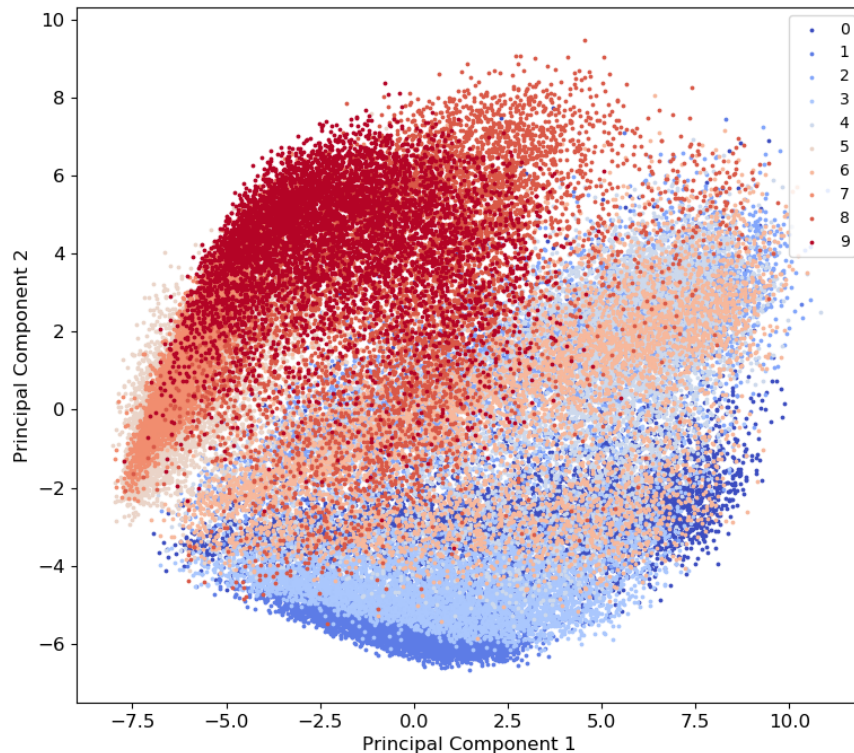
This answer is based on implementing PCA on the total data `Xtrn_nm` (which means that PCA fits total 60000 samples) and apply dimension reduction with PCA to the first sample in each class. We can find that when  $K$  is large, Root Mean Square Error is small, which means more principal components will represent and reconstruct data better.

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

			
class : 0, K = 5	class : 0, K = 20	class : 0, K = 50	class : 0, K = 200
			
class : 1, K = 5	class : 1, K = 20	class : 1, K = 50	class : 1, K = 200
			
class : 2, K = 5	class : 2, K = 20	class : 2, K = 50	class : 2, K = 200
			
class : 3, K = 5	class : 3, K = 20	class : 3, K = 50	class : 3, K = 200
			
class : 4, K = 5	class : 4, K = 20	class : 4, K = 50	class : 4, K = 200
			
class : 5, K = 5	class : 5, K = 20	class : 5, K = 50	class : 5, K = 200
			
class : 6, K = 5	class : 6, K = 20	class : 6, K = 50	class : 6, K = 200
			
class : 7, K = 5	class : 7, K = 20	class : 7, K = 50	class : 7, K = 200
			
class : 8, K = 5	class : 8, K = 20	class : 8, K = 50	class : 8, K = 200
			
class : 9, K = 5	class : 9, K = 20	class : 9, K = 50	class : 9, K = 200

The results show that when K value is small, the reconstruction effect is bad since it contains fewer patterns. When K value is large, the reconstruction is good. We can easily differentiate classes when K value is large.

**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.



Overall separation of classes are good as expected although a few classes are overlapped since they are similar in type of clothing and they are not dominant when speaking of first two pcs. Besides, the first and second images shown in 1.5 also told us that first two pcs will separate long sleeves, trousers and shoes, which is class 2,1 and 9 correspondingly. The color that represents these classes are deep blue and deep red and we can find that they have been separated well in the image.



## 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.

Classification accuracy of test set : 0.8401

Confusion matrix of test set:

	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

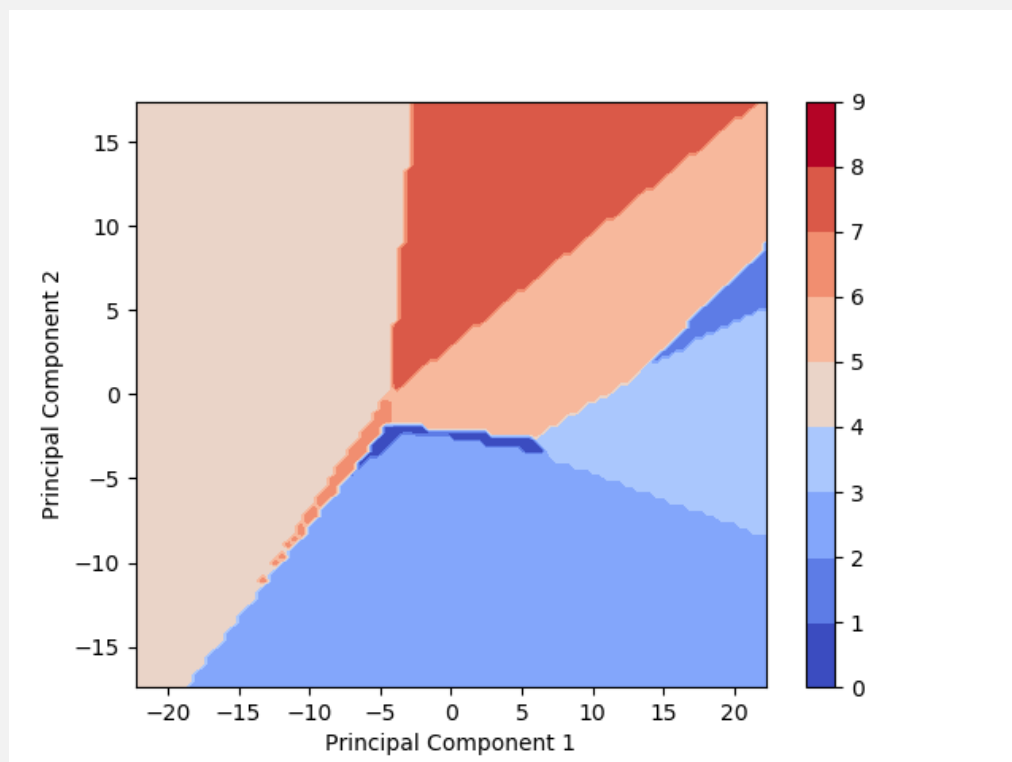
**2.2** (3 points) Carry out a classification experiment with **SVM classifiers**, and report the mean accuracy and confusion matrix (in numbers) for the test set.

Classification accuracy of test set : 0.8461

Confusion matrix of test set:

	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.

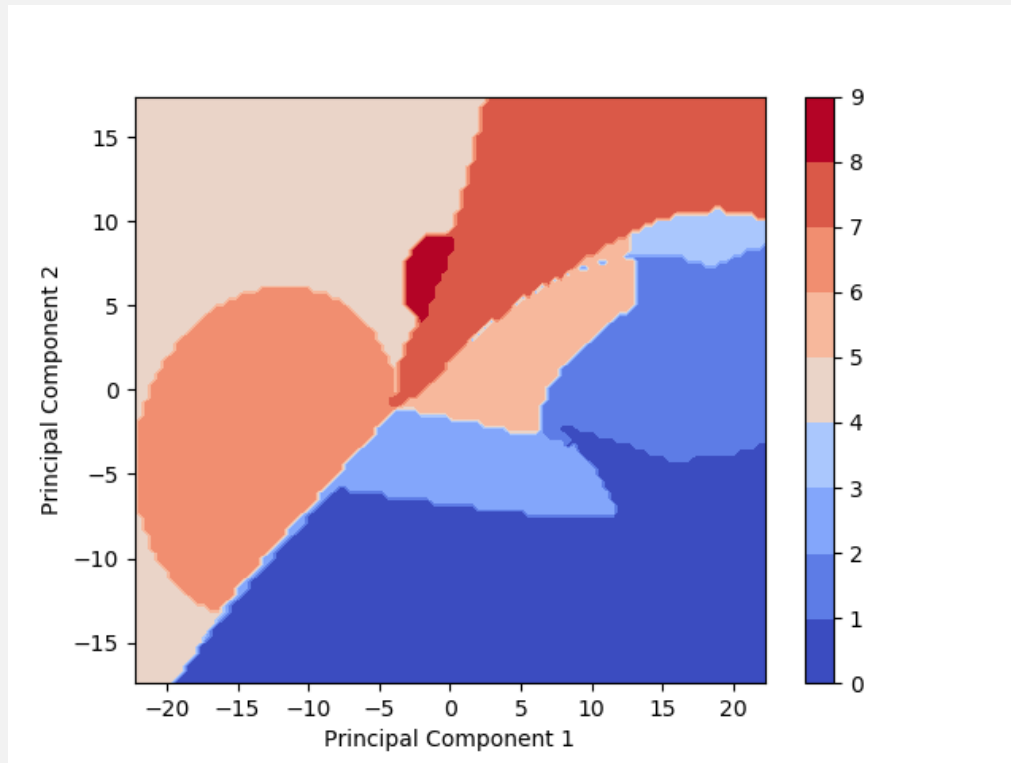


The numbers on colorbar range from 0 to 9, which represent 10 classes.

Finding 1: decision boundaries are straight.

Finding 2: the logistic regression classifier doesn't predict class 9. The classifier can only predict class 0-8.

**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.



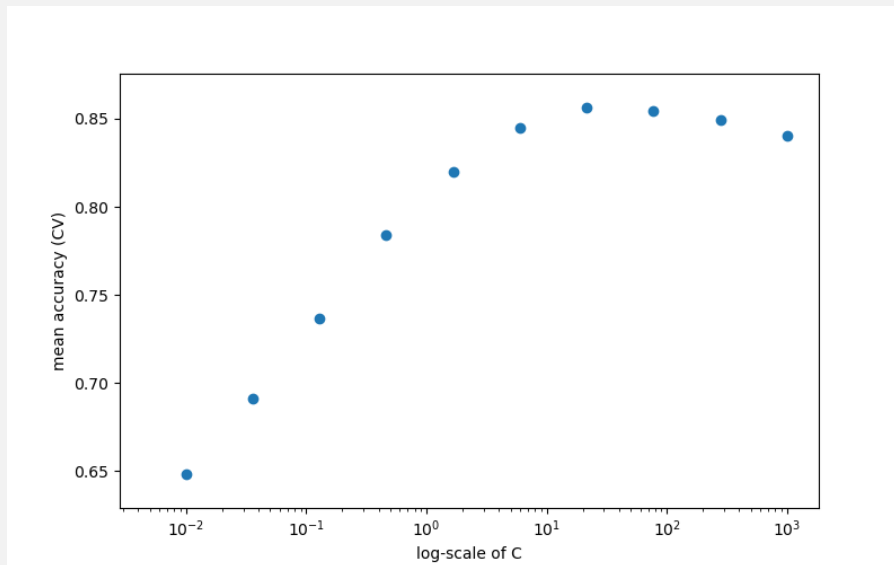
The numbers on colorbar range from 0 to 9, which represent 10 classes.

Finding 1: The decision boundaries are curved

Finding 2: One predicted class are surrounded by other predicted classes in the image.

Finding 3: Unlike logistic regression classifier, SVM classifier can predict 10 classes(from 0 to 9).

**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`.



highest mean accuracy score: 0.8565

Optimal C which yields the highest accuracy score: 21.54(or  $10^{\frac{4}{3}}$  equally)

**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](#).

Classification accuracy on training set: 0.9084

Classification accuracy on test set: 0.8765

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

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

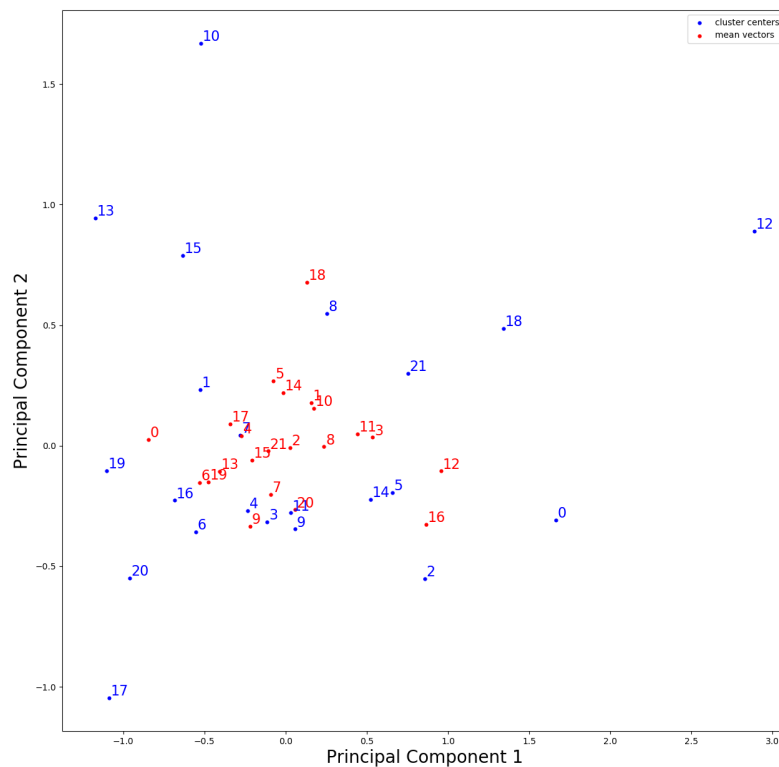
**3.1** (3 points) Apply k-means clustering on `Xtrn` for  $k = 22$ , where we use `sklearn.cluster.KMeans` with the parameters `n_clusters=22` and `random_state=1`. Report the sum of squared distances of samples to their closest cluster centre, and the number of samples for each cluster.

Sum of squared distance of samples to their closet cluster center: 38185.82

The number of samples for each cluster:

cluster id.	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

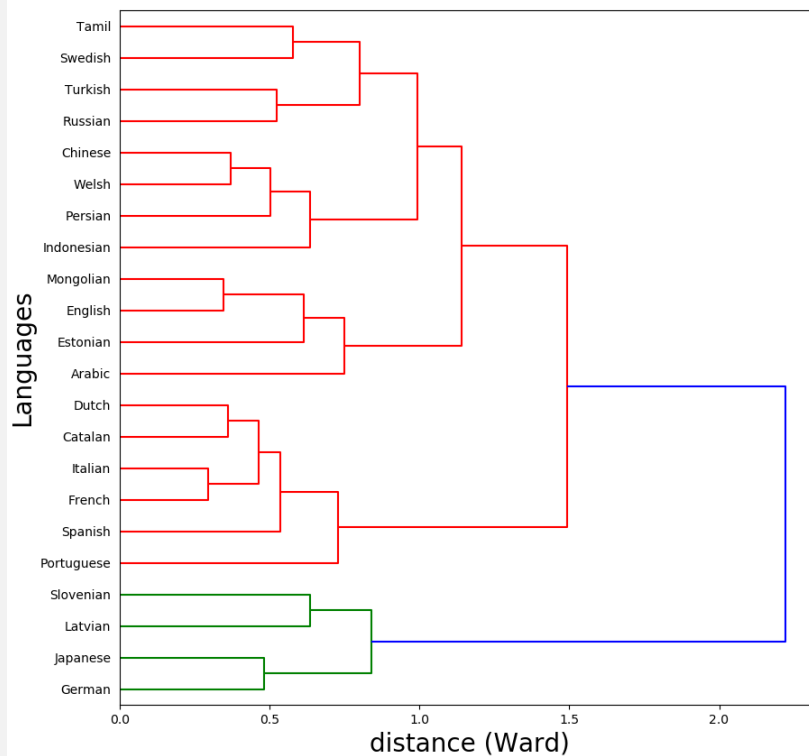
**3.2** (3 points) Using the training set only, calculate the mean vector for each language, and plot the mean vectors of all the 22 languages on a 2D-PCA plane, where you apply PCA on the set of 22 mean vectors without applying standardisation. On the same figure, plot the cluster centres obtained in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



Mean vectors and cluster centers are not similar. Mean vectors are more concentrated on PCA plane while cluster centers are dispersed widely on PCA plane.

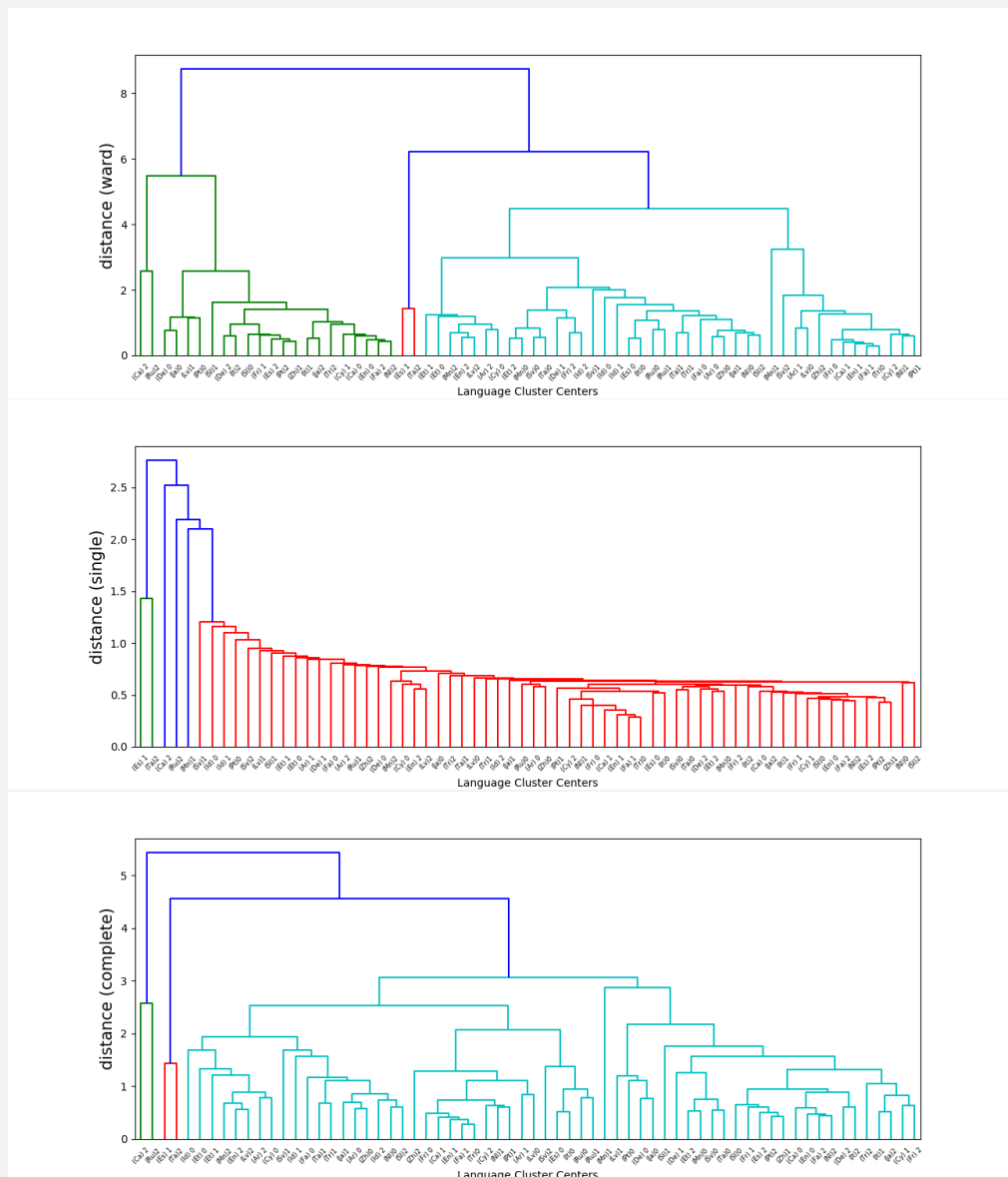


**3.3** (3 points) We now apply hierarchical clustering on the training data set to see if there are any structures in the spoken languages.



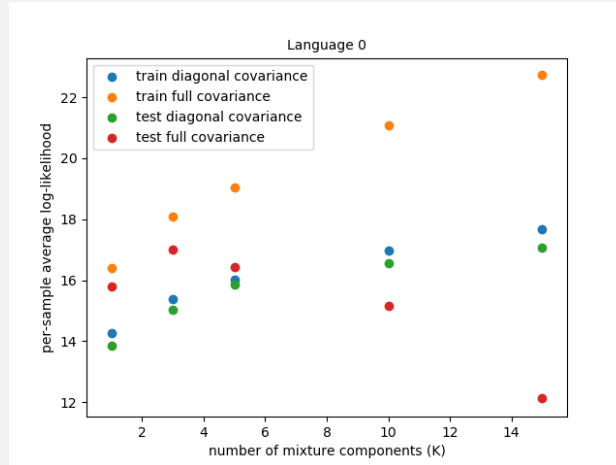
Two different colors represent leaf nodes' categories. Therefore, we can achieve two disparate language groups. Besides, similar languages have the same father node.

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



- 1: Trees in dendrogram based on ward algorithm and complete distance are more balanced than the one based on single algorithm.
- 2: We can find that according to the y-axis, which represents the distance, single distance metric achieves the lowest distance while ward distance metric reaches the largest distance.
- 3: Three cluster centers in one class is not close to each other. Commonly, three different tree node colors are shown in each algorithm. It means that each of them can separated all points into three groups.

**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.,



	K = 1	K = 3	K = 5	K = 10	K = 15
full train	16.39	18.10	19.15	21.14	22.92
diagonal train	14.28	15.40	16.01	16.92	17.61
full test	15.81	17.02	16.71	14.82	11.97
diagonal test	13.84	15.04	15.91	16.44	16.81

- 1: Within the training data, GMM model with 'full' covariance matrix has a higher log-likelihood score than the one with 'diagonal' covariance.
- 2: Within the test data, when the mixture components number becomes bigger, the log-likelihood score with full covariance becomes smaller. (worse generalization)