Q1. Eigenfaces

1. The total data set is randomly partitioned into 80% training data and 20% testing data. 80% training data is large enough to train our models. 20% test data is enough to evaluate the performance of different models with different hyperparameters. However, this random split does not take into account the label of each face, which means that theoretically it could happen that a person has no faces left in the training set. Therefore, a better option is to split data randomly into training and testing subset within each class, and then combine the partitioned subsets of each class into a larger training and testing set separately. From the results, there are no obvious differences in model performance between these two split methods since the special case almost never happens. Another reason is that PCA as an unsupervised learning maximizes total projected data variance and LDA maximizes the ratio of projected between-class and within-class scatter. Therefore, the performance of these two methods do not depend on the number of faces of a particular person to train, which means that even if a person has only one face left in the training set, it does not put the models at a disadvantage for recognizing that person in the testing set.

The number of eigenvalues (eigenvectors) is the same as the number of pixels per image D (2576) because the covariance matrix S is D×D. The number of eigenvectors with non-zero eigenvalues equals the number of data in the training set minus one (N - 1 = 415). In PCA optimization, the projected data variance along each eigenvector equals the corresponding eigenvalues and this number should be maximized to achieve larger data variance after projection. Hence the total number of eigenvectors used for PCA (Mpca) should not exceed the number of non-zero eigenvalues (N -1), otherwise the projected variance is zero along those eigenvectors with zero eigenvalues. The larger Mpca, the more information of original data space is kept after projection and the results of classification or reconstruction should be better, but more memory is used and it will take more time for classification or reconstruction.

1. The eigenvalues computed using low-dimensional method are identical to those computed without low-dimensional method. However, the corresponding eigenvectors are not identical, but have the following relationship (u = Av). The computation cost (memory and time) of low-dim method is much smaller than the normal method if D>>N, because AAT is D×D, but ATA is only N×N. Moreover, although AAT is D×D, rank (AAT) is always N – 1 which means that it has no more than N - 1 nonzero eigenvalues, hence the computation of zero eigenvalues and their corresponding eigenvectors is a waste of time and memory for PCA. This can be overcome by using low-dim method which only gives N eigenvalues and among them N-1 are non-zero, hence saving time and memory. However, the computed eigenvectors using low-dim method cannot be used as eigenfaces directly, but should be multiplied by A and then normalized, hence extra computation effort is needed. In summary, if D>>N, low-dim method is much preferable but if D < N, low-dim method is not useful. The above results always hold when N is changed.

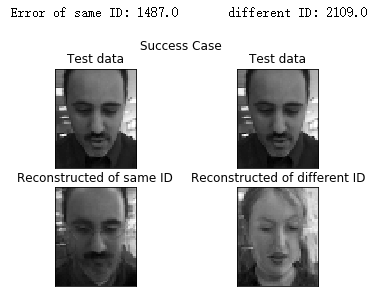
Q1. Applications of eigenfaces

1. The reconstruction error of each faces is computed, and the average of these error become smaller if Mpca is larger. This is reasonable since more PCA bases learnt implies higher dimension of PCA subspace, hence more information of original data space is kept after projection.
2. Comparison between NN-classification and alternative method.

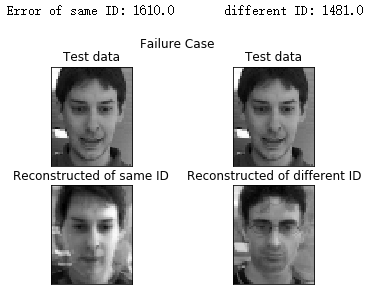
The accuracy of NN increases as Mpca increases at the beginning, but almost does not change after Mpca is around 50.

The accuracy of AM (or reconstruction error) does not change too much as the number of bases learnt per class changes. Even if AM chooses only one eigenvector to span each eigenspace (i.e. linear subspace), the accuracy of AM is still better than that of NN with the largest Mpca. The reason is that AM takes into account the labels of each face when training its model, making it more discriminative than NN. In other words, AM generates an eigenspace for each class, which will definitely capture the features of that particular person more accurately, than just project all training data from different identities onto the same eigenspace.

The success case below shows that AM recognizes the test data because it has smaller reconstruction error than the images reconstructed from all other different identities.



The failure case below shows that AM regards the test person as the one with different identity in the lower-right position because it has smaller reconstruction error than the image reconstructed from the same identity.



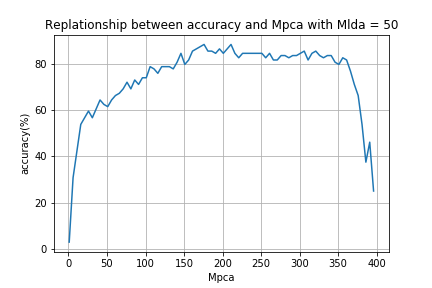
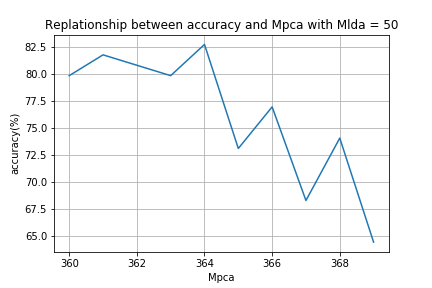
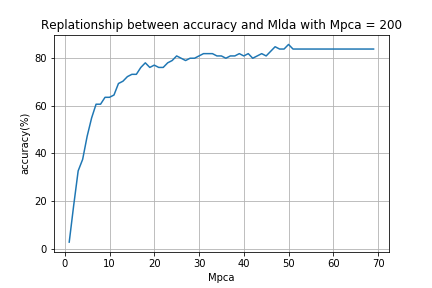
In summary, the reconstruction error can be used as a discriminative feature due to better accuracy than NN and reasonable computational cost. However, the performance of AM seems to be not related with its hyperparameter, which means that the best result is still not good enough compared with other discriminative methods like LDA. Moreover, since the number of images available to train a subspace is limited (no more than 10 in this coursework), the reconstruction error is relatively larger than NN, which can have smaller reconstruction error by increasing the number of bases learnt (up to N -1 = 415 bases available).

Q3. PCA-LDA

1. The accuracy of PCA-LDA increases as Mpca increases and does not change too much after around 170, but decreases very fast after around 364 as shown in the following two figures. The shape of this curve is very similar with that in Q1. However there are two differences.

One is that, when PCA is used to reduce dimension of original data space, the maximum number of eigenfaces available is N – 1 because other eigenvectors of covariance matrix are with zero eigenvalues. This is the result obtained in Q1. However, when doing PCA-LDA, the number of PCA bases learnt, i.e. the dimension of resulting eigenspace, should not exceed N – c (the rank of within scatter matrix Sw), otherwise the projected Sw on PCA subspace is singular. The major reason to do PCA first is to reduce dimension of origial data space so that doing LDA is possible (i.e. Sw after projection is nonsingular, hence invertable). Therefore, in this case, if Mpca is greater than N – c (364), the acuracy of PCA-LDA will drecease overall.

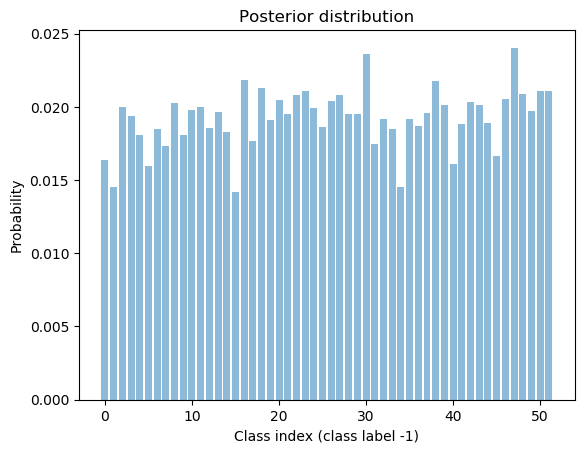
Another difference is that the maximum accuracy is achieved round Mpca=170, which is different from the value obtained in Q1. The reason is that when doing PCA-LDA, althogh the maximum number of meaningful components one can have for LDA is c -1 (will be discussed in b), then if Mpca is chosen as c – 1 (51), there are still too much information lost when reducing dimension via PCA. Hence, Mpca should be choosen in a way such that enough information is kept for doing LDA.

1. The accuracy of PCA-LDA increases as the number of Mlda increases but does not change after 51 (c -1) as shown in the following figure. The reason is that the LDA subspace is obtained by first choosing the generalized eigenvectors of with Mlda largest eigenvalues. Since there are only c -1 (rank of Sb) nonzero generalized eigenvalues, hence the meaningful components for LDA subspace learning is c – 1 (51 in this case).
2. The rank of between-class scatter matrix Sb is c – 1. The rank of within-class scatter matrix Sw is N – c.
3. The overall behaviour of PCA-LDA for face recognition is much better than the results obtained in Q1. The major reason is that LDA maximizes the ratio of between-class scatter and within-class scatter after projection, which makes it a discriminative model. But PCA just maximizes the total scatter (between-class scatter and within-class scatter are maximized at the same time), which is a generative model.

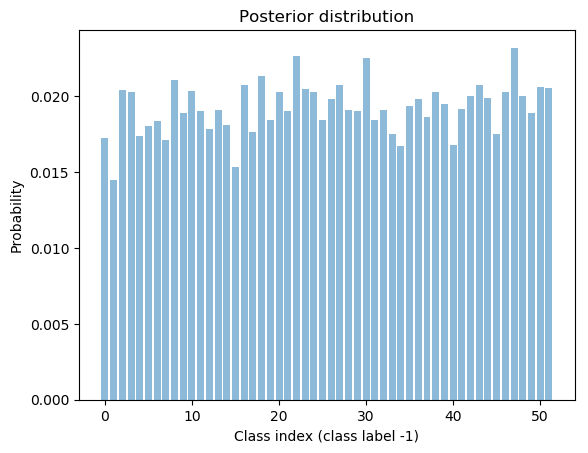
Q3. PCA-LDA Ensemble

1. Explanations on how to generate posterior distribution for PCA-LDA model. (fusion rules)

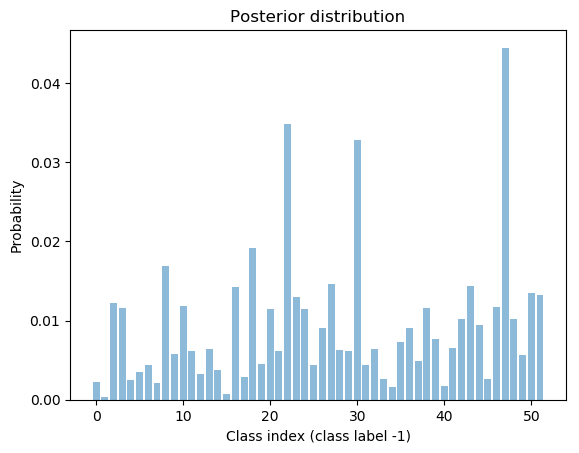
PCA-LDA model eventually will reduce the original data space into an LDA subspace. Each training data and new test data will then be projected into this LDA subspace. Normally, for NN-classification, the class has the minimum distance with the new test point in LDA subspace will be assigned. However, to generate a distribution, each class ci has a minimum distance with the test point denoted as di, and the probability of *not* choosing ci as the label of the test point *is*  because the label with larger distance has smaller chance to be the true label. For example, the figure below shows the posterior distribution of a PCA-LDA model. It can be concluded that label 16 has the lowest probability of *not* being the true label (i.e. highest probability of being the true label), which means that if NN classification is used, then label 16 is the output of this discriminative model.



If average fusion rule is used, i.e. , then the output of committee machine is the label with the smallest value of . The figure below shows posterior distribution by averaging 10 base models. In comparison with the previous distribution of a single model, the averaging effect makes label 2 has the highest possibility of being the true value. In fact, the true label is 2 indeed, which proves the success of averaging rule.



The figure below shows posterior distribution by using product rule of 10 base models i.e. . This figure also illustrates label 2 is successfully assigned.



1. Fixing all other conditions, just varying the number of base models.

For both two cases (bagging and randomization on feature space), the accuracy of the committee machine will increase as more base models are used but will not change after a certain value. The improvement of accuracy at the beginning is due to decorrelation between individual model predictions by randomization. However, as the number of base models increases after a certain value, this decorrelation achieves its limit, which means that the new base model added into the ensemble has common properties with the previous models. In other words, the amount of randomness will decrease. For example, we choose a random integer from 1 to 100, if we do the experiment 5 times, its very likely we have completely 5 different values, but if more experiments are done, then there must be common values occur, say we’ve done 10000 experiments, then it’s highly possible that we have all the integer between 1 to 100, then any new experiments done cannot add new information into our system.

1. Fixing all other conditions, just varying randomness parameter

In random feature case, the total number of eigenfaces are fixed (M0 + M1 is fixed). When the number of base models (T) is small (e.g. 5 base models), the accuracy increases as the ratio increases. The ratio increasing means that more eigenfaces with largest eigenvalues are used in PCA stage, making the model more optimized. The reason is that when there are not enough base models, the effect of ensemble is not obvious, which means that the accuracy will increase as the amount of randomization decreases (i.e. more optimized per model). When the number of base models is large enough (e.g. 50 base models), the accuracy increases as the ratio increases. The ratio increasing means that more random eigenfaces are used in PCA stage, making the model less optimized. The reason is that when there are enough base models, the effect of ensemble becomes obvious, which means that the accuracy will increase as the amount of randomization increases (i.e. less optimized per model).

In bagging case, similar results were observed by the number of samples per class chosen from the training set. The accuracy increases as this number increases when T is small because more training data are available for each model (more optimized, less randomized). When T is large, the accuracy increases as this number decreases which means that less samples per class (less optimized, more randomized).

1. Comparison between the error of the committee machine and the average error of individual model.

The error of the committee machine is becoming more and more smaller than the average error of individual models as the amount of randomization on bagging or model parameters is increasing. The reason is that more randomization means higher decorrelation between individual models, which makes the relationship tends to the following expression: .

1. Overall comparison between bagging and randomization on feature space.

Since the PCA-LDA model is constructed from scratch, most of the time and memory is used to compute the scatter matrices (each one is D×D). When doing bagging, each random training set for each model has a set of different scatter matrices which costs too much time to compute. For example, if a single PCA-LDA model needs 30s to compute, then 10 base models need 300s to compute if not using parallel computing and from the results, 10 models are still not enough to take the advantage of ensemble. In comparison, randomization on PCA feature space just need to compute scatter matrix once, hence running much faster than bagging. Moreover, among all different configurations, doing randomization on feature space with small number of base models together with more optimized hyperparameters gives the best result (overall accuracy and computation cost)