Project on Automatic Learning (Phase 2, Part 2)

Zhuo Liu (0983311)

February 25, 2014

1 Introduction

In this report, we will first use the data obtained by PCA from Part 1, and linear discriminant analysis (LDA) to build a linear classifier on the feature space. We will test the classifier by checking the accuracy on both training and test set. Then, we will apply kernel PCA to the same data set again, to obtain a KPC space, and also apply LDA on this KPC space to build a classifier. Since kernel PCA is nonlinear, the classifier we build will be also a nonlinear classifier. In particular, the kernel we use in KPCA will be Gaussian Kernel, i.e.

$$k(x,y) = exp(-\frac{||x-y||^2}{2\sigma^2}). \tag{1}$$

Still, the accuracy on both training and test set will be provided, and we can compare the result to the one by first linear classifier.

2 Database: the Large One

2.1 Linear Classifier

First we check the accuracy of the classifier build from LDA on the principle components space. The requirement of the PC space is that 90 percent of energy will be kept, i.e. we will keep k largest eigenvalues of covariance matrix, s.t.

$$\frac{\sum_{i=1}^{k}\lambda_i}{\sum_{i=1}^{256}\lambda_i} > 0.9 \tag{2}$$

In order to see how the size of training set affects on the accuracy, we randomly choose training sets 9 times, whose sizes are 10%,20%,...,90% of whole data set, and the remaining data compose the test set. The following table shows the accuracy of the linear classifier (Here, when the size of training set

is 10%, the result is not available. The reason is there will be more descriptors than obeservations, so PCA cannot be applied):

Size of Traing Set	Number of PCs	Acc. on Training Set	Acc. on Test Set
10%	N/A	N/A	N/A
20%	82	0.9842767	0.8541176
30%	92	0.9769392	0.8772401
40%	98	0.9670330	0.8891213
50%	102	0.9585427	0.8946048
60%	104	0.9591623	0.8855799
70%	106	0.9551570	0.9058577
80%	108	0.9466248	0.9122257
90%	109	0.9469644	0.8937500

Figure 1,2,3 show the data projected on the first two, second two and third two LD components space:

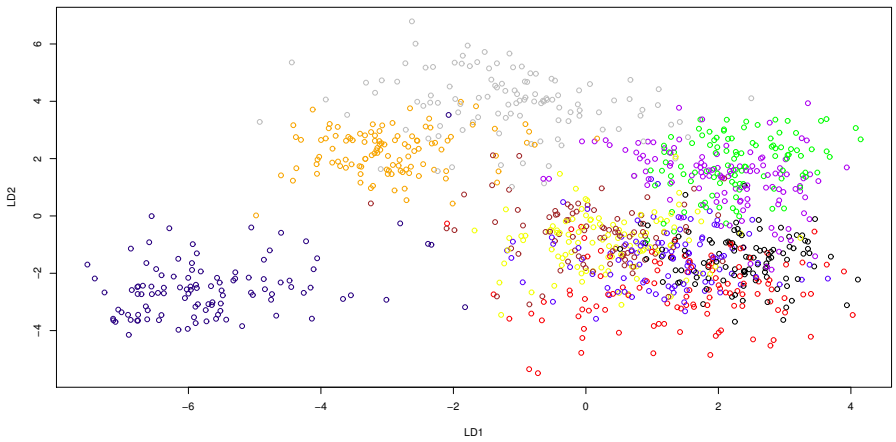

Figure 1: *Projection of PC space onto first two LD componets space (large data set). Here, the colors represent different classes: navy–0, green–1, blue–2, black–3, grey–4, brown–5, orange–6, purple–7, yellow–8, red–9.*

It is not difficult to find that obeservations in different classes are not well seperated, so it is impossible to build a very good linear classifier.

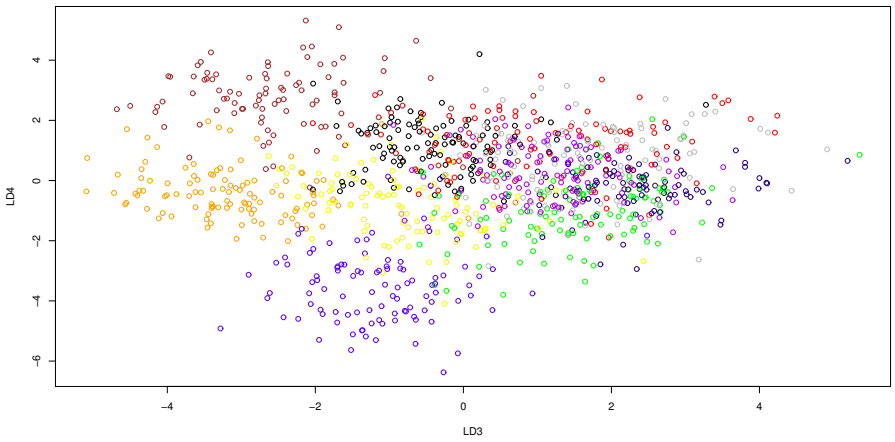

Figure 2: *Projection of PC space onto second two LD components space (large data set). Here, the colors represent different classes: navy–0, green–1, blue–2, black–3, grey–4, brown–5, orange–6, purple–7, yellow–8, red–9.*

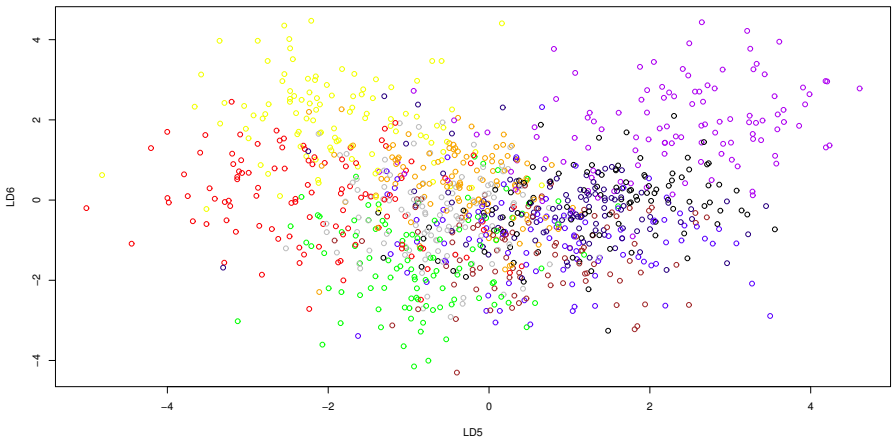

Figure 3: *Projection of PC space onto third two LD components space (large data set). Here, the colors represent different classes: navy–0, green–1, blue–2, black–3, grey–4, brown–5, orange–6, purple–7, yellow–8, red–9.*

2.2 Nonlinear Classifier

Next, we will also use LDA to build a classifier, but now, on the kernel PC space. It will be a nonlinear classifier on the original feature space. By several experiment, the optimal parameter σ in Gaussion kernel function is $\sigma = 6.455$. The following table shows the result:

Size of Traing Set	Number of KPCs	Acc. on Training Set	Acc. on Test Set
10%	118	1	0.8026499
20%	222	1	0.8847059
30%	320	1	0.9094982
40%	416	1	0.9476987
50%	505	1	0.9560853
60%	588	1	0.9514107
70%	667	1	0.9623431
80%	726	1	0.9655172
90%	759	1	0.9625000

Figure 4,5 show the data projected on the first two and second two LD components space:

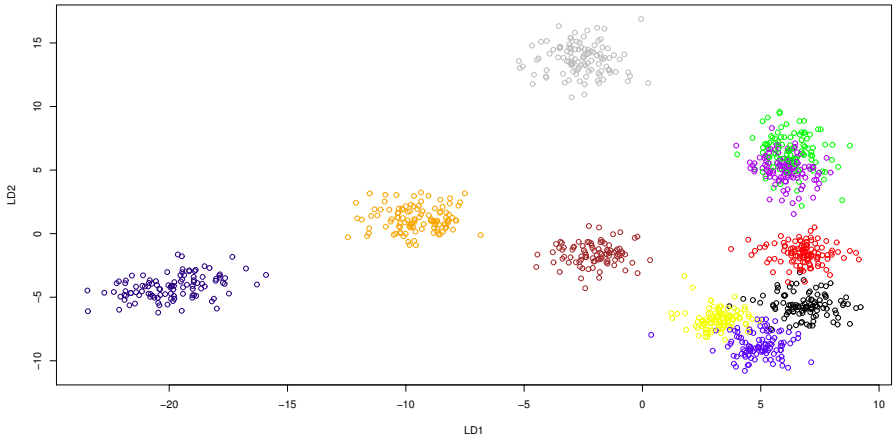

Figure 4: *Projection of KPC space onto first two LD components space (large data set). Here, the colors represent different classes: navy–0, green–1, blue–2, black–3, grey–4, brown–5, orange–6, purple–7, yellow–8, red–9.*

By doing kernel trick, the obeservations in different classes are well separated, and we can easily build a linear classifier on this space. This explains why the result for this nonlinear classifier performs much better than the linear classifier build on PC space.

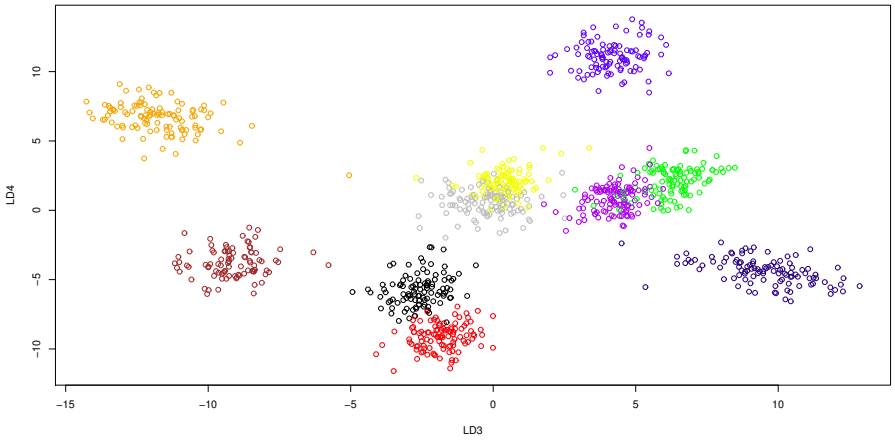

Figure 5: *Projection of KPC space onto second two LD components space (large data set). Here, the colors represent different classes: navy–0, green–1, blue–2, black–3, grey–4, brown–5, orange–6, purple–7, yellow–8, red–9.*

3 Database: the Small One

3.1 Linear Classifier

Similar to the procedure for large data set, first we build a linear classifier from the PC space which keeps 90% of energy, the following table shows the accuracy:

Size of Traing Set	Number of PCs	Acc. on Training Set	Acc. on Test Set
10%	3	0.7878788	0.7638288
20%	4	0.7748918	0.7754329
30%	4	0.7878788	0.7835498
40%	4	0.7846320	0.7886003
50%	4	0.7844156	0.7913420
60%	4	0.7813853	0.7889610
70%	4	0.7903525	0.7676768
80%	4	0.7911255	0.7597403
90%	4	0.7883598	0.7532468

Fiugre 6,7 shows the data projected on the first two and second two LD components space. From these two figures, we can find that it is even more difficult to seperate the data in this PC space than those in large database, so the accuracy is very low.

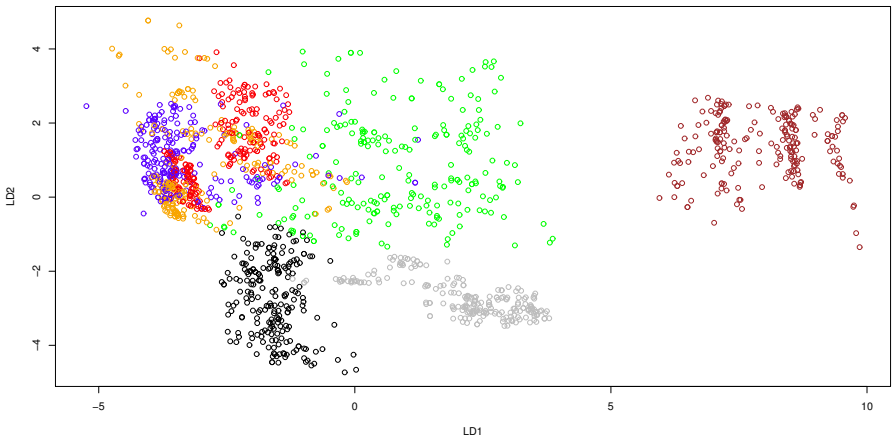

Figure 6: *Projection of PC space onto first two LD componets space (small data set). Here, the colors represent different classes: red–brickface, brown–sky, blue–foliage, green–cement, orange–window, grey–path, black–grass.*

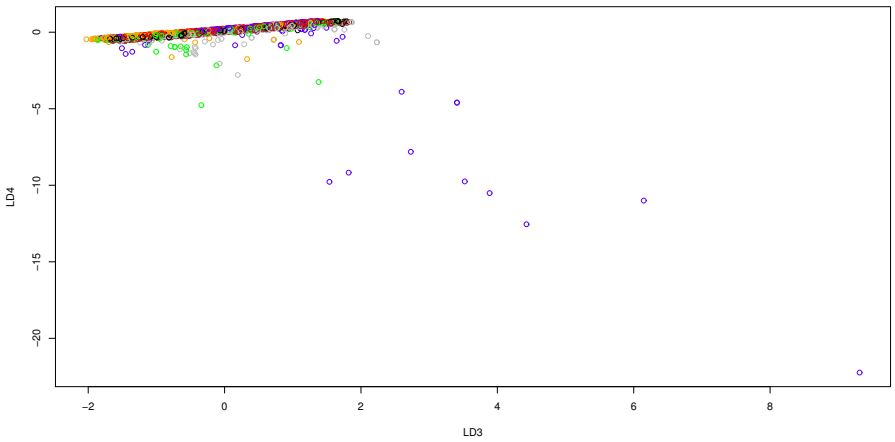

Figure 7: *Projection of PC space onto second two LD components space (small data set). Here, the colors represent different classes: red–brickface, brown–sky, blue–foliage, green–cement, orange–window, grey–path, black–grass.*

3.2 Nonlinear Classifier

Now, it's time for kernel PCA. By several experiments, the optimal parameter σ in Gaussion kernel function is $\sigma = 35.355$. The following table shows the result for the classifier:

Size of Traing Set	Number of KPCs	Acc. on Training Set	Acc. on Test Set
10%	105	0.9740260	0.7994228
20%	146	0.9805195	0.8809524
30%	180	0.9711400	0.8998145
40%	199	0.9696970	0.9163059
50%	205	0.9653680	0.9238095
60%	214	0.9639250	0.9274892
70%	223	0.9635127	0.9379509
80%	227	0.9577922	0.9458874
90%	230	0.9610390	0.9393939

Figure 8,9 shows the data projected on the first two and second two LD components space. As expected, the data processed by kernel PCA become much more separated, as a result, building a classifier on this space becomes much easier, and the accuracy is much better.

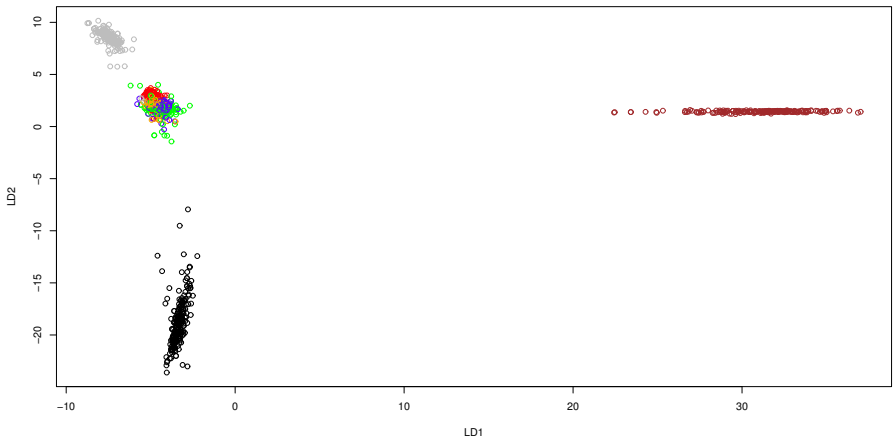

Figure 8: *Projection of KPC space onto first two LD components space (small data set). Here, the colors represent different classes: red–brickface, brown–sky, blue–foliage, green–cement, orange–window, grey–path, black–grass.*

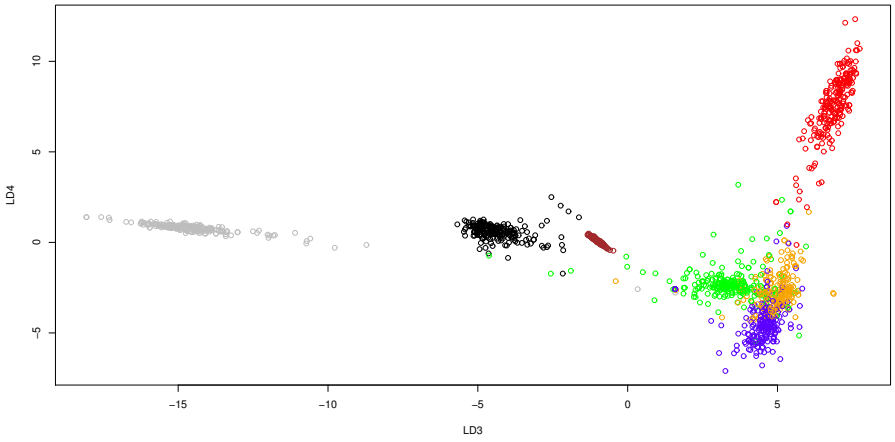

Figure 9: *Projection of KPC space onto second two LD components space (small data set). Here, the colors represent different classes: red–brickface, brown–sky, blue–foliage, green–cement, orange–window, grey–path, black–grass.*