Thomas Ehret

Assignment 4

CS519 SP23

Dr. Cao

Investigating Dimension Reduction

Testing Specifications

The IRIS and MNIST datasets were used to test the dimension reduction capabilities of Principal Component Analysis (PCA), Linear Discriminant Analysis (LDA), and Kernel Principal Component Analysis (KPCA).

Dimension reduction parameter values were tested 10 times and the results were averaged if there was any large variation in accuracy, time, and misclassification. Any large testing time outliers were discarded, and the test was rerun. The features were normalized with the *StandardScalar* function. All initial testing parameters were set to default except for the MNIST dataset wherein the training set total was decreased for the sake of memory usage: *train_test_split(train_size=10000)*. The reduced datasets were then tested with a Decision Tree classifier.

The performance of each dimension reduction algorithm was measured by timing the 'fit' process and the 'transform' process. The decision tree model performance was then measured by timing the 'fit' process, the overall accuracy as a percentage, and the total number of misclassified samples.

Defaults Parameters:

Iris Dataset: *train_test_split(X, y, test_size=0.2, random_state=42, stratify=y)*

MNIST Dataset: *train_test_split(X, y, train_size=10000 test_size=0.2, random_state=42, stratify=y)*

PCA(n_components=2)

LDA(n_components=None)

KPCA(n_components=2)

DecisionTreeClassifier(criterion='gini', splitter='best', max_depth=None,)

Iris Dataset

The stratification of the dataset is shown in Table 1. The initial performance of the dimension reduction algorithms is listed below in Table 2.

Instances	80:20 split		
y_train	40	40	40
y_test	10	10	10

Table 1

Metric	PCA	LDA	K PCA
Dimension Reduction Fit Time	1ms	1ms	13ms
Dimension Reduction Transform Time	<1ms	<1ms	<1ms
Model Fit Time	1ms	1ms	<1ms
Class Accuracy	87%	100%	90%
Misclassed	4	0	3

Table 2

The iris dataset is simple enough that the only notable differences in performance was the dimension reduction fit time of KCPA, and the classification accuracy of the decision tree model for PCA and KPCA (4 and 3) respectively.

The following parameters were changed during the second phase of testing. With only the changed parameters shown.

KPCA(kernel= 'poly', 'rbf', 'sigmoid', 'cosine', 'precomputed') kernel modes were tried with insignificant deviation from the original values. Modifying the Kernel gamma coefficient to greater than 3 resulted in an increase in Decision Tree misclassifications. Testing results are displayed in Table 3

Metric	PCA	LDA	K PCA
Dimension Reduction Fit Time	7ms	2ms	18ms
Dimension Reduction Transform Time	<1ms	<1ms	<1ms
Model Fit Time	1ms	1ms	<1ms
Class Accuracy	87%	100%	73%
Misclassed	4	0	8

Table 3

The *plot_tree* visualization was used to show the how the dimension reduction algorithms effect the decision tree model's choices. It seems that because LDA focuses on finding a feature subspace that maximizes the separability between the groups, that the decision tree classifier then needs fewer nodes before it converges. The images are embedded below with the originals in the accompanying zip file for ease of viewing.

Figure 1 (PCA -KPCA)

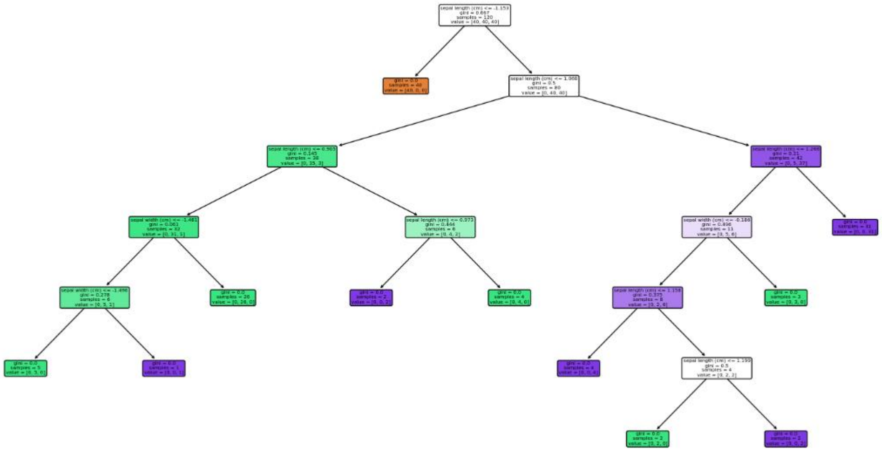
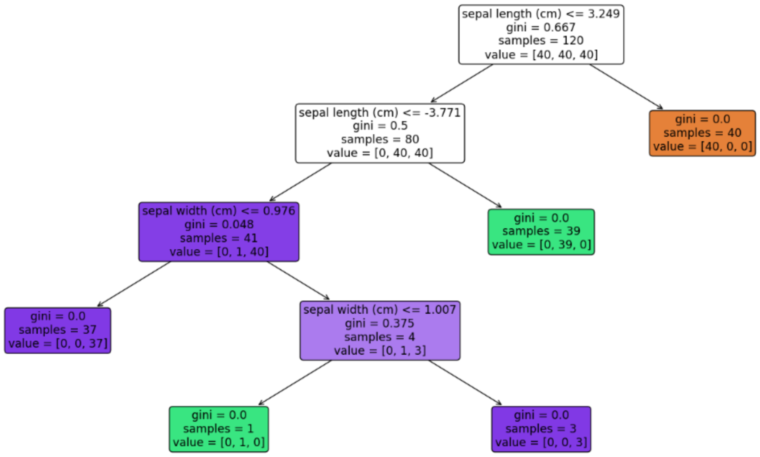


Figure 2 (LDA)



MNIST Dataset

The stratification of the dataset is shown in Table 4, the total instances in *y_train* are in accordance with the *train_size=10000* parameter . The initial performance of the dimension reduction algorithms is listed below in Table 5.

Instances	**80:20 split**									
y_train	986	1125	999	1020	975	902	982	1042	975	994
y_test	1381	1575	1398	1428	1365	1263	1375	1458	1365	1392

Table 4

Metric	**PCA**	**LDA**	**K PCA**
Dimension Reduction Fit Time	275ms	1972ms	3103ms
Dimension Reduction Transform Time	47ms	47ms	2903ms
Model Fit Time	61ms	57ms	49ms
Class Accuracy	28%	46%	28%
Misclassed	10104	7583	10103

Table 5

As evidenced by the metrics in table 5, the MNIST dataset proved a difficult classification task. Given the misclassification, it is also evident that limiting the number of components setting at 2 is not sufficient for a reliable classification. Therefore, by setting the number of components to 5 the following metrics were achieved. Approximately 200% increase in classification accuracy for PCA and KPCA, with a reduction of approximately 50% in misclassifications. LDA also showed an improvement with a similar reduction percentage in misclassifications, coupled with a 63% improvement for overall classification accuracy. The number of components were incrementally increased with LDA components being capped at 9. The cap is in accordance with the LDA algorithm that states the max number of components cannot be larger than the minimum value of *n_*features or *n_classes* – 1. and the with the resulting metric shown in Table 7.

Metric	**PCA**	**LDA**	**K PCA**
Dimension Reduction Fit Time	269ms	1932ms	3231ms
Dimension Reduction Transform Time	47ms	63ms	2783ms
Model Fit Time	88ms	89ms	91ms
Class Accuracy	64%	75%	65%
Misclassed	4978	3502	4911

Table 6

LDA n_components = 9, PCA and KPCA n_components 10

Metric	PCA	LDA	K PCA
Dimension Reduction Fit Time	292ms	2026ms	66869ms
Dimension Reduction Transform Time	36ms	55ms	2933ms
Model Fit Time	172ms	160ms	162ms
Class Accuracy	~77%	82%	~77%
Misclassed	3272	2578	3281

Table 7

AS evidenced by the results in Tables 8 and 9, there are diminishing returns by doubling the number of components past 10. Particularly when considering the increasing computational time cost for both dimension reduction and model fit.

PCA and KPCA n_components 20

Metric	PCA	K PCA
Dimension Reduction Fit Time	377ms	66671ms
Dimension Reduction Transform Time	63ms	3006ms
Model Fit Time	320ms	331ms
Class Accuracy	~78%	~78%
Misclassed	3058	3091

Table 8

PCA and KPCA n_components 40

Metric	PCA	K PCA
Dimension Reduction Fit Time	543ms	66821ms
Dimension Reduction Transform Time	78ms	3098ms
Model Fit Time	1643ms	646ms
Class Accuracy	~79%	~79%
Misclassed	2966	2977

Table 9

It seems that initially the LDA algorithm works best for a decision tree classifier, when taking into account the computational cost and diminishing returns of increasing the number of PCA components. However, it should be noted that even a PCA setting of 40 components is a significant decrease in the dimensions of the MNIST dataset.