

This report discusses the performance of MATLAB implementations of the machine learning algorithms Bayes rule, K-nearest-neighbor, Principle component analysis, and Fisher's linear discriminant analysis. There are three main datasets that were tested on a variety of different combinations of algorithms. Namely, they are data.mat (called face dataset), pose.mat (called pose dataset), and illumination.mat (called illumination dataset).

In the following experiments that were conducted, there was no significant pre-processing of the data. The input data treated each pixel as a feature and although this was inefficient, it makes the code simpler to read and understand. Incidentally, that means that the overall performance of the classifiers was adversely affected. In each of the experiments, the amount of training/testing data is tuned across various values and the performance is noted.

RESULTS:

Experiment 1: Classifying on Face Dataset without preprocessing:

Training samples per class	Bayes' rule	1-nearest-neighbor	2-nearest-neighbor
1	60.5% accuracy	60.5% accuracy	36.25% accuracy
2	60% accuracy	59.5% accuracy	46.5% accuracy

Experiment 2: Classifying on Pose Dataset without preprocessing

Training samples per class	Bayes' rule	1-nearest-neighbor	2-nearest-neighbor
7	63.73% accuracy	75.49% accuracy	52.70% accuracy
8	62.94% accuracy	73.24% accuracy	51.18% accuracy
9	53.68% accuracy	69.85% accuracy	47.79% accuracy
10	39.22% accuracy	61.76% accuracy	36.27% accuracy
11	38.24% accuracy	45.59% accuracy	27.21% accuracy
12	60.29% accuracy	67.65% accuracy	36.76% accuracy

Experiment 3: Classifying on Illumination Dataset without preprocessing

Training samples per class	Bayes' rule	1-nearest-neighbor	2-nearest-neighbor
11	32.79% accuracy	42.50% accuracy	36.03% accuracy
12	35.46% accuracy	78.76% accuracy	55.56% accuracy
13	39.34% accuracy	92.28% accuracy	73.53% accuracy
14	43.70% accuracy	95.38% accuracy	83.19% accuracy
17	67.65% accuracy	100% accuracy	99.26% accuracy
19	33.82% accuracy	95.38% accuracy	83.19% accuracy

The first three experiments provide a good benchmark performance for the dimensionality reduction algorithms which will be discussed in the next part. For the most part, the raw classifiers work as expected on the face and the illumination datasets. We see that as the size of the training set increases, we see an increase in performance. However, the pose dataset proved to be problematic. We see a strange behavior in the accuracy of the classifier on the pose dataset. Where in some cases, less training data led to more performance. Although this is unexpected, it could be caused by the way that

we are defining the feature space which is each pixel is one dimension. This method of feature extraction can be very sensitive to changes in pose, position, and various other variations of the image data. This is likely the cause of the strange behavior that the classifiers exhibit with the pose dataset.

Another anomaly that is commonly found is that the k-nearest-neighbor classifier seems to outperform the Bayes' classifier in many cases. Again, the probable root cause of this issue is likely the method of feature extraction which produces extremely high dimensional features with few samples in comparison. The high dimensionality of the feature space could cause computational errors in the Bayes model. The classifier also assumes a Gaussian distribution of all pixels which again, could be inaccurate.

Another note regarding the k-nearest-neighbor classifier is that it as the value of k increased, we see a drop off in performance. This is expected as the sample space for each class is extremely low in comparison to the feature vector dimension.

The two classifiers perform well without any preprocessing. However, since the dimensionality of the feature vectors are so high, the time to compute the predicted labels of each test sample is very high. The next section discusses the performance impact of the dimensionality reduction algorithms PCA and LDA.

Experiment 4: PCA on Face Dataset (66% of data used for training)

Target Dimension	Bayes' rule	1-nearest-neighbor	2-nearest-neighbor
10	23.50% accuracy	24.0% accuracy	21.5% accuracy
50	50.50% accuracy	50.5% accuracy	39.0% accuracy
100	57.50% accuracy	58.0% accuracy	41.0% accuracy
200	60% accuracy	58.5% accuracy	45.0% accuracy
Benchmark	60% accuracy	59.5% accuracy	46.5% accuracy

Experiment 5: PCA on Pose Dataset (50% training data used)

Target Dimension	Bayes' rule	1-nearest-neighbor	2-nearest-neighbor
10	34.80% accuracy	47.06% accuracy	37.75% accuracy
50	50.49% accuracy	73.28% accuracy	48.77% accuracy
100	53.68% accuracy	76.72% accuracy	51.23% accuracy
200	54.17% accuracy	76.72% accuracy	53.68% accuracy
Benchmark	63.73% accuracy	75.49% accuracy	52.70% accuracy

Experiment 6: PCA on Illumination Dataset (80% training data)

Target Dimension	Bayes' rule	1-nearest-neighbor	2-nearest-neighbor
10	9.93% accuracy	84.19% accuracy	63.24% accuracy
50	22.43% accuracy	99.63% accuracy	95.22% accuracy
100	29.41% accuracy	100.0% accuracy	98.53% accuracy
200	30.15% accuracy	100.0% accuracy	99.26% accuracy
Benchmark	67.65% accuracy	100.0% accuracy	99.26% accuracy

Experiments 4 to 6 with PCA worked as expected. More specifically, we saw that PCA did not significantly destroy performance while reducing the classification time significantly. I found that when the target dimension of PCA is set to around the number of classes, we see that the accuracy is very close to the benchmark performance while maintaining the reduced classification time. We also see as expected that the performance increases as the dimensionality of the dataset increases towards the optimal Bayes' rate. Moreover, we find that when the target dimensionality approaches c or the number of classes, there seems to be a high accuracy as well as a very fast classification time. PCA had significantly reduced the time to classify and preserved much of the information given by the original dataset. The following sections show how LDA performs under various circumstances.

Experiment 7: LDA on Face Dataset (66% training data)

Classifier	Accuracy	Benchmark
Bayes	23.50% accuracy	60.00% accuracy
1-nearest neighbor	58.00% accuracy	59.50% accuracy
2-nearest neighbor	42.00% accuracy	46.50% accuracy

Experiment 8: LDA on Pose Dataset (50% training data)

Classifier	Accuracy	Benchmark
Bayes	35.78% accuracy	35.29% accuracy
1-nearest neighbor	65.20% accuracy	79.49% accuracy
2-nearest neighbor	40.93% accuracy	52.70% accuracy

Experiment 9 LDA on Illumination Dataset (80% training data)

Classifier	Accuracy	Benchmark
Bayes	23.50% accuracy	67.65% accuracy
1-nearest neighbor	91.54% accuracy	100% accuracy
2-nearest neighbor	99.26% accuracy	99.26% accuracy

Based on the results from experiments 6 – 9, we see that LDA also has some nice properties. However, it seems like LDA destroys more discriminative information than PCA does at the same dimensionality. We do see that it performs fairly well. However, there are some cases where LDA does not perform well. For example, applying the LDA on the illumination dataset and classifying that data with the Bayes' classifier seems to significantly reduce classification accuracy.

CONCLUSIONS:

The various implementations of the four machine learning algorithms seem to be capable of recognizing faces to a measurable degree of accuracy as shown by the results of the 9 experiments that were carried out. A few things to note is that the method of feature extraction seems to be a serious limitation in terms of the behavior of the algorithms. Throughout the experiments, we have seen that the high dimensionality of the feature vector can adversely affect the speed and performance of the Bayes' classifier and performance on some datasets. This can especially be seen in the pose dataset where minor variations in location and angle can hurt performance and accuracy. It would be more efficient and accurate if we were to use something like a local binary pattern for feature extraction. However, the usage of a single pixel as a dimension seems to perform fairly well for face classification.