CS6690: Pattern Recognition Assignment #3

Group 3: Akshay (ME10B124) & Suthirth (ED10B059)

October 30, 2014

1 Gaussian Mixture Models

Gaussian Mixture Models are a type of classification that sets soft boundaries to data. It estimates data as a probabilistic mixture of its features. Each mixture is represented by by a gaussian density function in a gaussian mixture model.

$$p(\bar{x}) = \sum_{i=1}^{K} \pi_i N(\bar{x}|\mu_i, \Sigma_i)$$

K is the total number of classes, π the weights.

Gaussian Mixture Models are solved using the Expectation maximization Algorithm (EM). This consists of calculating the posterior probability γ which is calculated by using.

$$\gamma_{nk} = \frac{\pi_k N(\bar{x}|\mu_k, \Sigma_k)}{\sum\limits_{i=1}^{K} \pi_i N(\bar{x}|\mu_i, \Sigma_i)}$$

This posterior for each class gives us the contribution of class k to the mixture, using this, we can maximize the other parameters.

$$\pi_k = \frac{\sum\limits_{i=1}^{K} \gamma_{nk}}{N}$$

$$\mu_k = \frac{1}{N_k} \sum_{n=1}^{N} \gamma_{nk} \bar{x}_n$$

$$\Sigma_k = \frac{1}{N_k} \sum_{n=1}^{N} \gamma_{nk} (\bar{x}_n - \mu_k)(\bar{x}_n - \mu_k)^T$$

These three update equations are used to maximize and convergence is checked. The initial point for convergence is usually the k-means algorithm.

2 Dynamic Time Warping

Dynamic Time Warping is a template matching algorithm that works by using the priciples of dynamic programming. It is a template matching that is capable of matching uneven templates as well. DTW aligns two sequence of feature vectors by warping the time axis iteratively until an optimal match between the two sequences is found. Using this, a distance measure can be computed, and can be used to match to a template.

Since we're dealing with continuous sequences in our dataset, the algorithm can be written as:

D[i,j] = $|a[i] - b[j]|$ + min(D(i-1,j), D(i-1,j-1), D(i,j-1))

3 Hidden Markov Models

A Markov Model is one in which the current state of a system depends only on the present and previous states of the system. The order of the system tells us about the dependency of the nth previous value on the conditional probability density of the model.

In a Hidden Markov Model, the state sequence is hidden as is the transition probabilities. We use the HMMs to characterize images and speech data. The performance of the HMM is also benchmarked to that of a GMM.

4 Metrics

The metrics used to evaluate the performance of a classifier are as follows

1. Precision
2. Recall
3. ROC Curves

5 Image dataset

5.1 Data Exploration

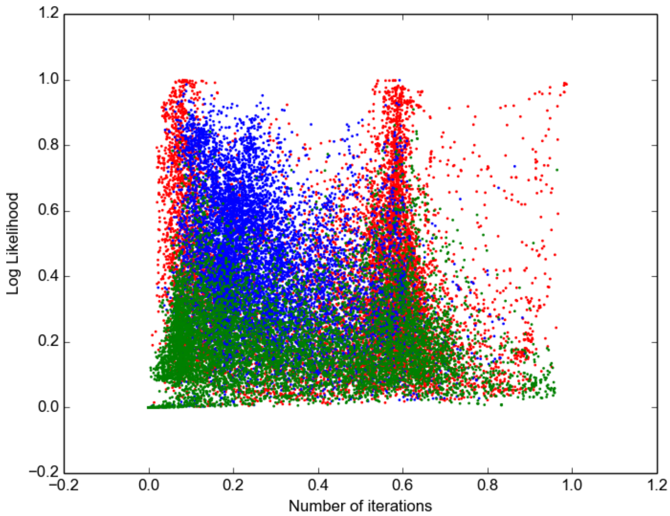
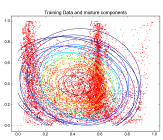
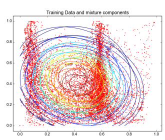
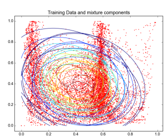
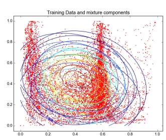


Figure 1: Scatter plot of the image dataset

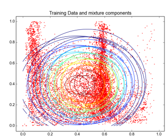
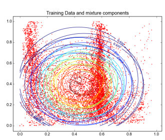
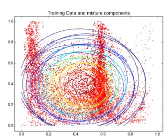
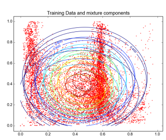
Following images show mixture components for iterations 0, 5, 15 and 30.

5.2 Coast

Full Scatterplot:

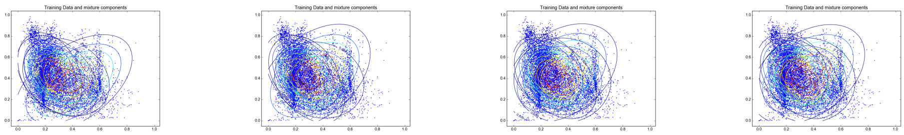
   

Diagonal Scatterplot:

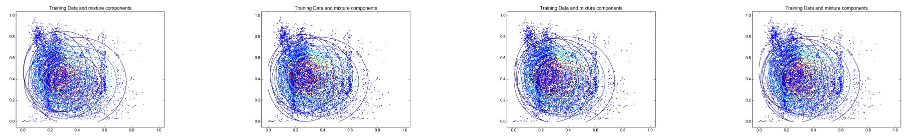
   

5.3 Forest

Full Scatterplot:

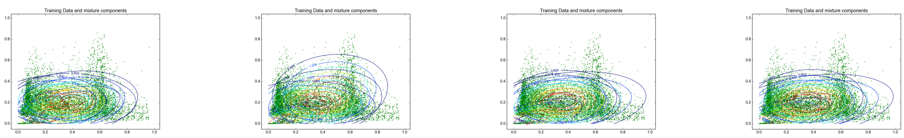


Diagonal Scatterplot:

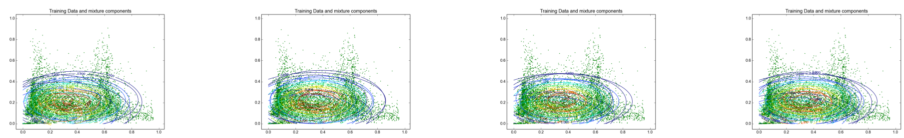


5.4 Street

Full Scatterplot:



Diagonal Scatterplot:



5.5 Results

5.6 Log Likelihood

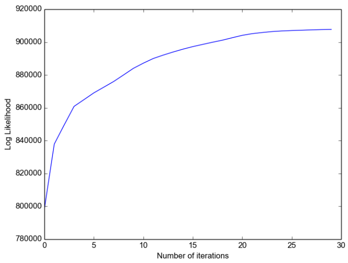


Figure 2: Confusion Matrix for full covariance

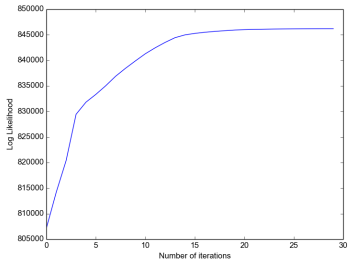


Figure 3: Confusion Matrix for Diagonal Covariance

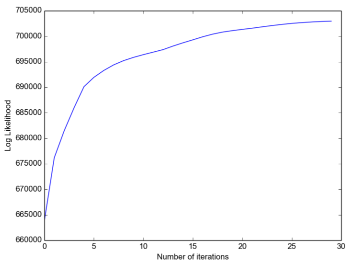


Figure 4: Confusion Matrix for Diagonal Covariance

5.7 Confusion Matrices

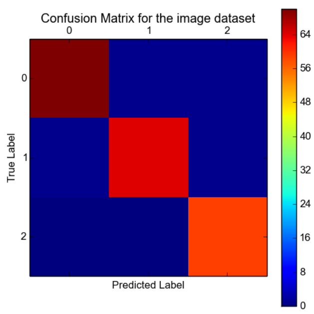


Figure 5: Confusion Matrix for full covariance

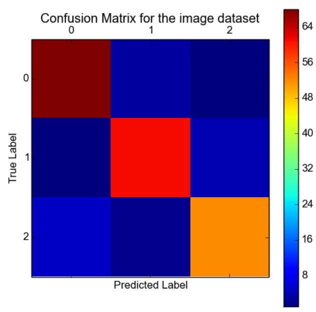


Figure 6: Confusion Matrix for Diagonal Covariance

5.8 Precision and Recall

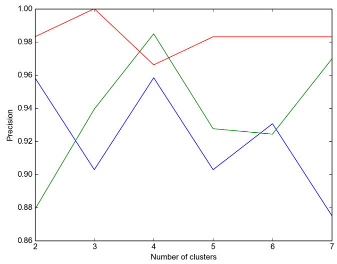


Figure 7: Precision

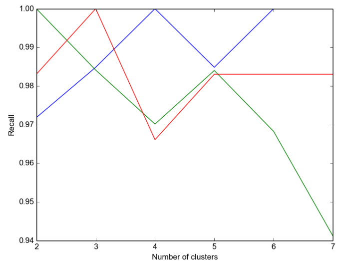
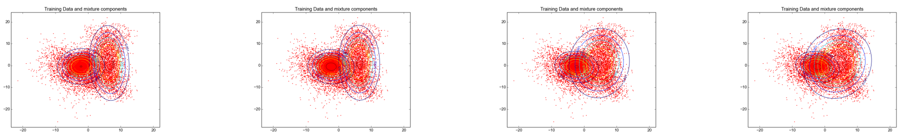


Figure 8: Recall

6 Digit dataset

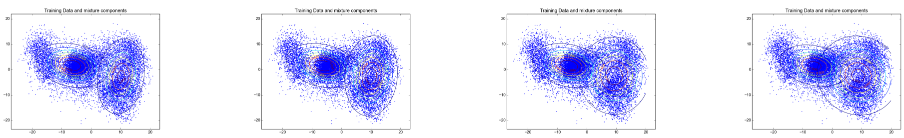
Number of clusters per class: 2
Following figures show mixture components for iterations 0, 5, 15 and 20.

6.1 Eight



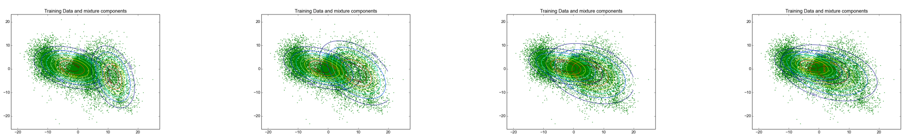
6.2 Seven

Full Scatterplot:



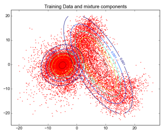
6.3 Six

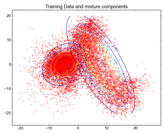
Full Scatterplot:

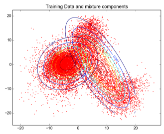


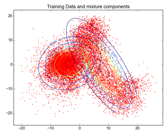
6.4 Three

Full Scatterplot:



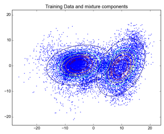


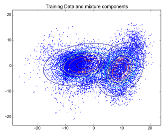


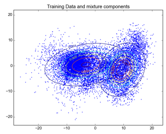


6.5 Two

Full Scatterplot:







6.6 Results

6.7 Log Likelihood

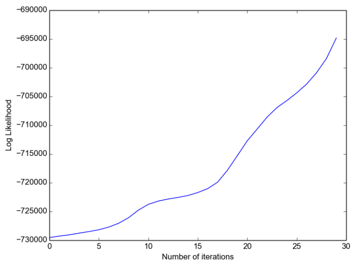


Figure 9: Confusion Matrix for full covariance

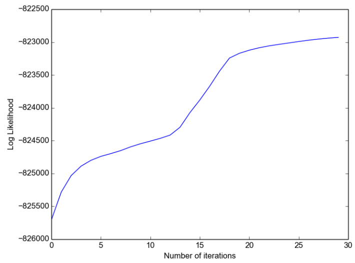


Figure 10: Confusion Matrix for Diagonal Covariance

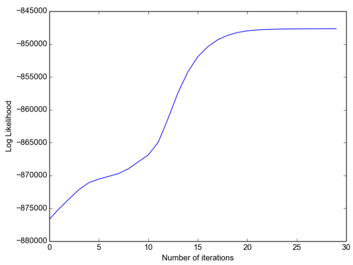


Figure 11: Confusion Matrix for Diagonal Covariance

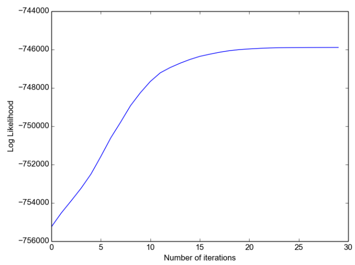


Figure 12: Confusion Matrix for Diagonal Covariance

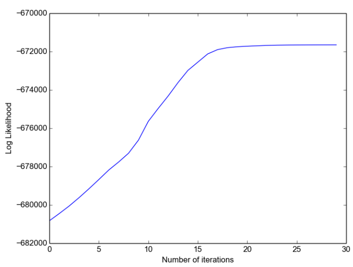


Figure 13: Confusion Matrix for Diagonal Covariance

6.8 Confusion Matrices

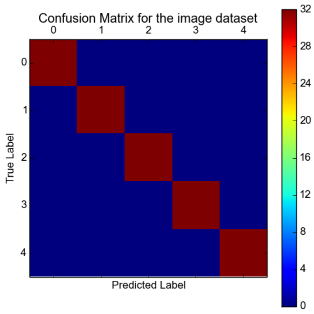


Figure 14: Confusion Matrix for Digit

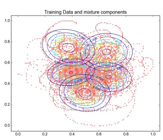
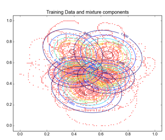
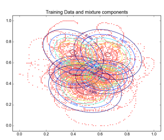
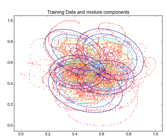
6.9 Precision and Recall

Precision	Recall
1.0	1.0
1.0	1.0
1.0	1.0
1.0	1.0
1.0	1.0

7 Handwriting dataset

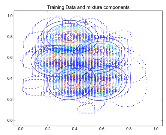
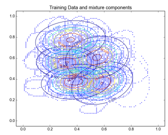
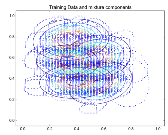
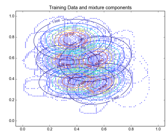
Number of clusters per class: 5
Following figures show mixture components for iterations 0, 5, 10 and 15.

7.1 a

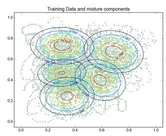
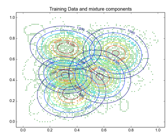
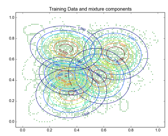
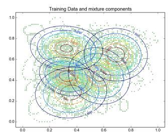
7.2 ai

Full Scatterplot:

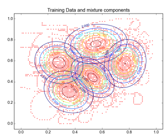
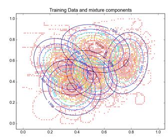
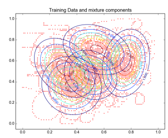
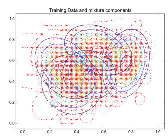
7.3 bA

Full Scatterplot:

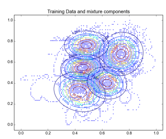
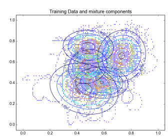
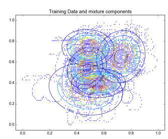
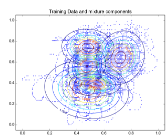
7.4 chA

Full Scatterplot:

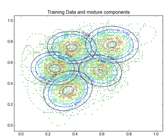
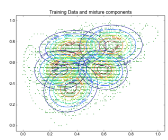
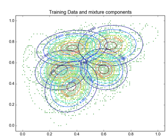
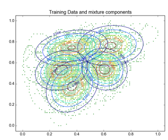
7.5 dA

Full Scatterplot:

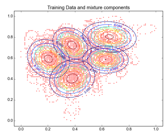
7.6 lA

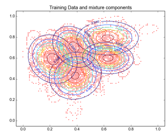
Full Scatterplot:

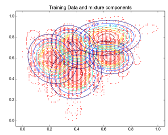
   

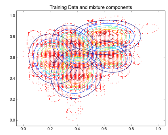
7.7 LA

Full Scatterplot:



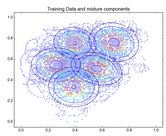


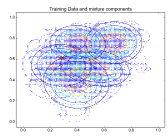


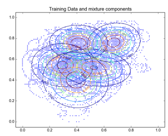


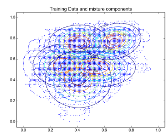
7.8 tA

Full Scatterplot:









7.9 Results

7.10 Log Likelihood

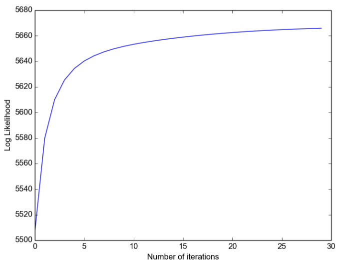


Figure 15: Confusion Matrix for full covariance

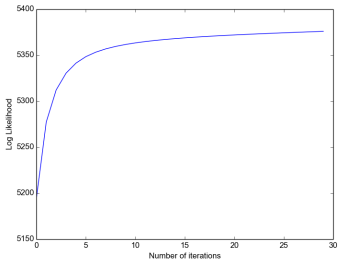


Figure 16: Confusion Matrix for Diagonal Covariance

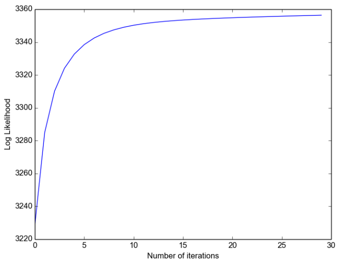


Figure 17: Confusion Matrix for Diagonal Covariance

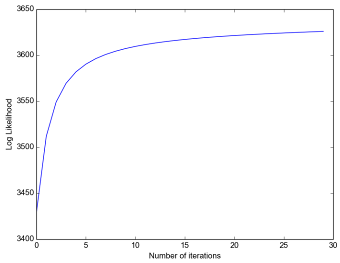


Figure 18: Confusion Matrix for Diagonal Covariance

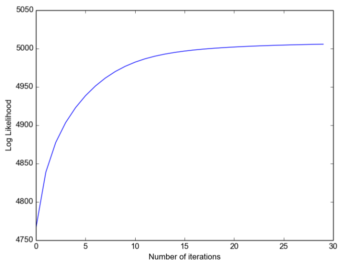


Figure 19: Confusion Matrix for Diagonal Covariance

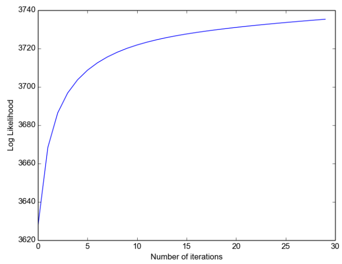


Figure 20: Confusion Matrix for Diagonal Covariance

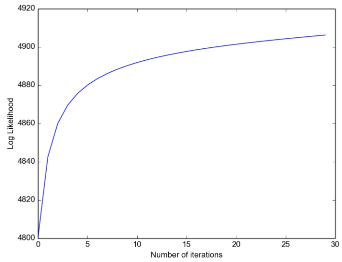


Figure 21: Confusion Matrix for Diagonal Covariance

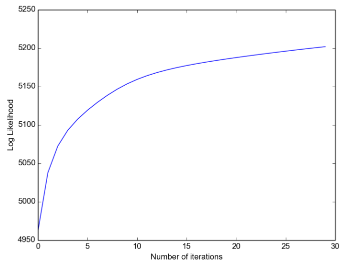


Figure 22: Confusion Matrix for Diagonal Covariance

7.11 Confusion Matrices

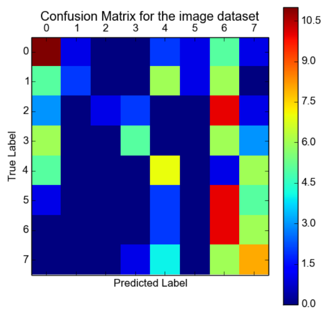


Figure 23: Confusion Matrix for Digit

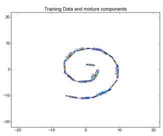
7.12 Precision and Recall

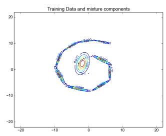
Precision	**Recall**
0.5280	0.3548
0.1	0.6666
0.0588	1.0
0.25	0.625
0.3684	0.3043
0.0	0.0
0.5555	0.1851
0.4210	0.2666

8 Spiral dataset

Number of clusters per class: 10,15
Following figures show mixture components

8.1 Mixture Components





8.2 Results

8.3 Log Likelihood

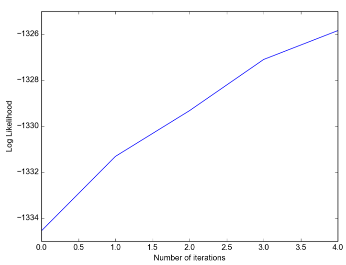


Figure 24: Confusion Matrix for full covariance

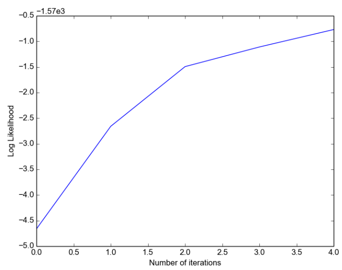


Figure 25: Confusion Matrix for Diagonal Covariance

8.4 Precision and Recall

Precision	Recall
1.0	1.0
1.0	1.0

9 Hidden Markov Models

9.1 Solution Methodology

The feature vectors were first quantized using a K-means clustering with number of clusters varying from 8 to 16. From this, a new quantized feature set was developed which was used as input to the HMM. This was then tested with the test set, from the given data which was clustered with the given means, to form the quantized test data. We train a HMM for each class, and the class with maximum emission probability is assigned to the test data. The results are as follows

9.2 Image Dataset

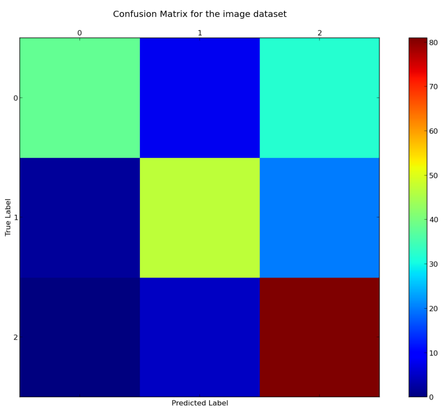


Figure 26: Confusion matrix for HMM

Precision	Recall
0.487179487179	0.95
0.68115942029	0.783333333333
0.941860465116	0.609022556391

9.3 Digits Dataset

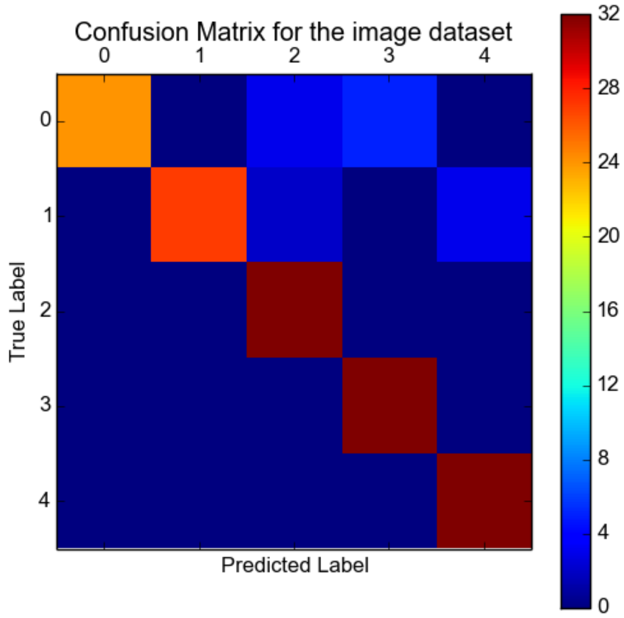


Figure 27: Confusion matrix for HMM Digits

Precision	0.75	0.84	1.0	1.0	1.0
Recall	1.0	1.0	0.86	0.86	0.91

10 Conclusion

From the exercise, it is clear that GMMs and HMMs are good at different classification tasks. HMMs outperform GMM for sequential data, while GMMs are far superior when it comes to data with spatial features. HMMs work well with temporal features.