Indian Institute of Technology, Guwahati

EE657: Pattern Recognition and Machine Learning

Assignment 4 : GMM Based Clustering and Classification

Instructor : Dr. Suresh Sundaram

Name : Akshat Bordia
Roll No. : 11010848
Branch : Electronics and Electrical Engineering
Email : a.bordia@iitg.ernet.in

April 3, 2014

0.1 Contents

0.2 Problem Statement and Objective

Task 1

In this assignment, we have to implement a Gaussian Mixture Model (GMM) based clustering scheme on the following image 'ski image.jpg'.

Figure 1: Ski Image File

Task 2

In this task,we have to apply the Gaussian Mixture Model as a density estimation technique to the problem of classification. The description of the data to be used for training and testing is as follows:

- We have been provided training features, corresponding to two classes ω_1 and ω_2 in the Files Pattern1.mat and Pattern2.mat respectively. Each File contains 200 instances (training examples), of 120 feature dimensions.
- The features corresponding to 100 testing samples of ω_1 and ω_2 are contained in Test1.mat and Test2.mat respectively.

0.3 Gaussian Mixture Model for Clustering

Image Segmentation can be considered as boundary between **Image Processing** and **Pattern Recognition** in the sense that outputs in Image Segmentation are certain attributes of image which can be further used for information extraction, classification, template matching etc. Segmentation is process of dividing the image into various meaningful non overlapping regions or different objects in an image.

In this assignment, we use **Gaussian Mixture Models** and **Expectation Maximization** Algorithm to learn GMM for various pattern of data (different image segments). Similar kind of data can be assumed to come from a particular Gaussian Distribution and hence we can model complete image as Mixture of Gaussians. After estimating the parameters of these distributions satisfying convergence criterion, we can finally segment image into various parts.

Mathematical Formulation of GMM

Gaussian Mixture Models are linear superposition of Gaussian Component. Hence we can represent density function as :

$$P(X) = \sum_{k=1}^{K} \Pi_k N(X|\mu_k, \Sigma_k)$$

Where :

$$0 \leq \Pi_k \leq 1$$

and :

$$\sum_{k=1}^{K} \Pi_k = 1$$

Hence we see that to estimate a K-component GMM density, we need to estimate k separate mean vectors,covariance matrices and weights Π_k

0.3.1 Expectation Maximization Algorithm for GMM Parameters Estimation

It is well established to maximize **Likelihood** in order to estimate the densities. **Expectation Maximization** is well known method to find solutions of this maximum likelihood estimation. This method can be summarised as follows :

1. **Initialization Step :** Initialize the values of means μ_k, covariances Σ_k and mixing coefficients Π_k. We can calculated initial value of log likelihood.

2. **E Step :** Evaluate the responsibilities using current parameter values:

$$\gamma(z_{nk}) = \frac{\Pi_k N(X_n|\mu_k,\Sigma_k)}{\sum_{k=1}^{K} \Pi_k N(X_n|\mu_k,\Sigma_k)}$$

3. **M Step :** Now we re-estimate the parameters using current responsibilities :

$$\mu_k^{new} = \frac{\sum_{n=1}^{N} \gamma(z_n k) X_n}{N_k}$$

$$\mu_k^{new} = \frac{\sum_{n=1}^{N} \gamma(z_n k)(X_n - \mu_k^{new})(X_n - \mu_k^{new})^T}{N_k}$$

$$\Pi_k^{new} = \frac{N_k}{N}$$

   Where :

$$N_k = \sum_{n=1}^{N} \gamma(z_n k)$$

4. Now we can evaluate log likelihood :

$$lnp(X|\mu, \Sigma, \Pi) = \sum_{n=1}^{N} ln(\sum_{k=1}^{K} \Pi_k N(X_n|\mu_k, \Sigma_k))$$

   Now we need to check convergence of parameters or log likelihood. If the convergence criterion is satisfied, we can stop the algorithm and it it is not satisfied, we go back to step 2.

Thus we estimate the densities and cluster the image accordingly. Initialization and all other attributes related to problem are discussed in further sections.

0.4 Gaussian Mixture Model for Classification

For implementing **Baye's Classifier**, we often assume data to come from **Gaussian Distribution**.While it might not be true for many cases, another representation of data might be **A mixture of Gaussians**. It provides more degree of freedoms in density estimation and in some sense provide better learning and representation of data. Hence we can use **GMM** to model the data and then apply discriminant analysis to classify between two or more classes. Formulation and Estimation is same as previous section and Parameter are estimated using **Expectation Maximization Algorithm**.

0.5 Analysis of Problem and Mathematical Formulation

0.5.1 Task 1

- We assume that image comprises of 3 clusters i.e 3 Gaussian Components.
- We have RGB values of pixel intensities as feature vectors. We need to normalize these values before proceeding further.
- **Initialization of Means :** We initialize three means as :

$$\mu_1 = [0.47, 0.47, 0.47]^T$$

$$\mu_2 = [0.05, 0.05, 0.05]^T$$

$$\mu_3 = [0.7, 0.7, 0.7]^T$$

- We initialize three covariance matrix corresponding to each Gaussian as : $\Sigma_i = I$ for i=1,2,3 and I is Identity Matrix of dimensions 3X3 .
- Weights of each of Gaussian Components are initialized as $\frac{1}{3}$

$$\Pi_i = \frac{1}{3} \text{ i=1,2,3}$$

- Now we follow the **Expectation Maximization** as mentioned in earlier section. Hence we iterate for 3 Gaussian Components. To achieve proper convergence of Maximum Likelihood, we use **50 iterations**. We observe that for this many iteration, we achieve proper convergence of Log-Likelihood.
- Finally, using responsibilities values for each feature vector, we assign it to one of three clusters.
- **Note :** Image was resized to 0.5 of it's original dimensions for faster convergence.

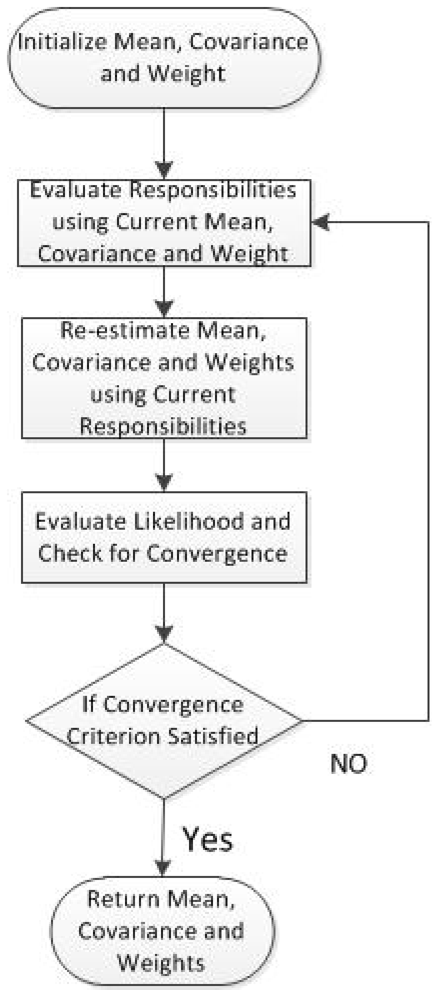
0.5.2 Task 2

- Here we need to learn two densities for two class of Data ω_1 andω_2.
- For each class, we assume its distribution to come from 2-component Gaussian Mixture.
- Means are initialized using **K-Means Clustering Algorithm** for two clusters.
- Covariance matrices are initialized as 0.2 times of Identity matrix of (120X120) dimension.
- We estimate the parameters using same method as in Task 1 ensuring proper convergence of Maximum Likelihood.
- After estimating distributions, for each test point, we calculate discriminant function and accordingly classify it to class 1 or class 2.
- Class specific accuracies and average accuracies are calculated accordingly.
- **Issues :** Here we have covariance matrix with dimension 120 X 120, but our data points are limited to 200. Hence to overcome the curse of dimensionality, we have to add a λI term in covariance matrix in order to avoid singularity. We choose $\lambda = 0.005$ for our implementation.

0.6 Implementation and Description of Code

Code for this assignment was implemented in **MATLAB R2013B** using **Image Processing Tool (IPT)Box**. Flow of Code Can be explained using flowcharts shown :

Figure 2: Flow Chart for GMM Estiamtion



After estimating density with proper convergence, image is segmented according to values of **responsibilities** and for **Task 2 Baye's Classifier** is implemented for 2-Component Gaussian Data.

To Run Code :

Task 1

- Run the MATLAB Code File for the Task 1.
- It will result give the following outputs :
    1. Segmented Image with three segments shown in different colour.
    2. Log Likelihood Plot showing convergence.
    3. Values of Means, Covariances and Weights.

Task 2

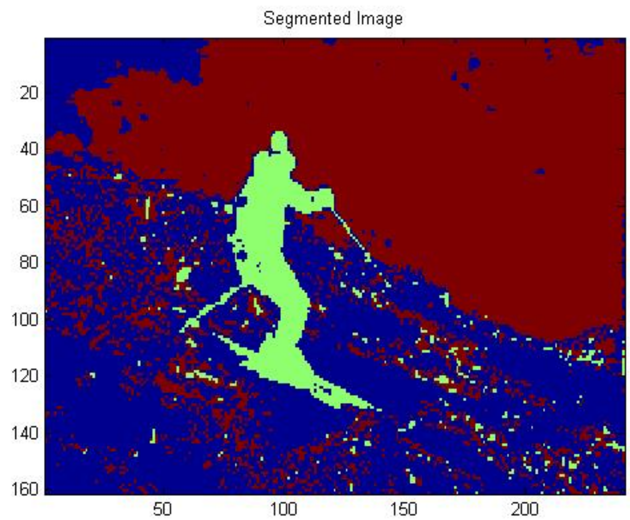
- Run the MATLAB Code File for the Task 2.
- It will result the accuracies for class 1 and class 2 as well as average accuracy.

0.7 Observation and Inferences

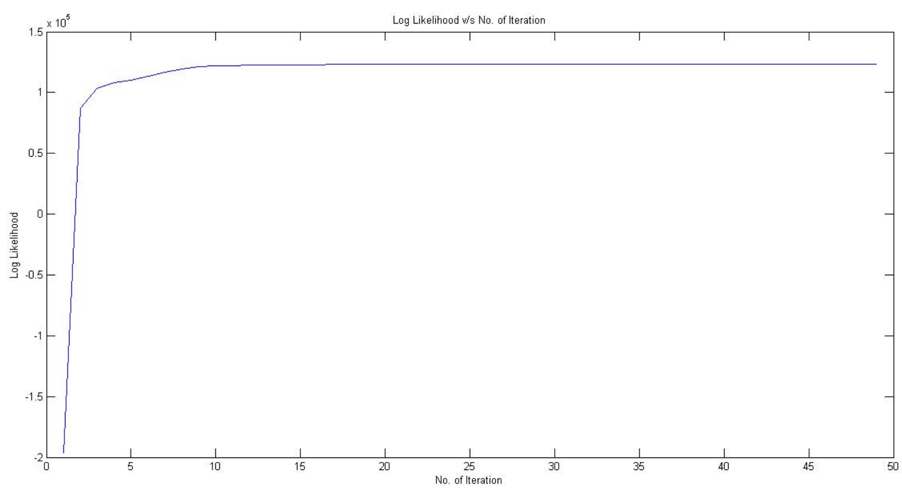
0.7.1 Task 1 : GMM Based Image Segmentation(Clustering)

As mentioned earlier, Image was resized to half of its original dimension in order to reduce time complexity of algorithm. Result obtained are as following:

1. Display the segmented output.



2. Display a graph depicting the convergence of the log likelihood values.



3. What are the final values of the means, prior weights and covariance matrices.

   - Weights :

$$\Pi_1 = 0.4598$$
$$\Pi_2 = 0.0766$$

$$\Pi_3 = 0.4635$$

- Means :

$$\mu_1 = [0.5924, 0.6733, 0.7607]^T$$
$$\mu_2 = [0.4465, 0.4754, 0.5144]^T$$
$$\mu_3 = [0.6954, 0.7892, 0.8473]^T$$

- Covariance Matrices :

$$\Sigma_1 = \begin{bmatrix} 0.0329 & 0.0284 & 0.0146 \\ 0.0284 & 0.0249 & 0.0132 \\ 0.0146 & 0.0132 & 0.0084 \end{bmatrix}$$

$$\Sigma_2 = \begin{bmatrix} 0.0970 & 0.0928 & 0.0849 \\ 0.0928 & 0.0924 & 0.0857 \\ 0.0849 & 0.0857 & 0.0824 \end{bmatrix}$$

$$\Sigma_3 = \begin{bmatrix} 0.0093 & 0.0069 & 0.0030 \\ 0.0069 & 0.0051 & 0.0023 \\ 0.0030 & 0.0023 & 0.0011 \end{bmatrix}$$

0.7.2 Task 2 : GMM Based Classifier

Proper convergence of EM Algorithm was ensured using log likelihood convergence criterion. Graphs ensuring convergence are shown on next page. Following accuracies were reported for various cases :

- **Accuracy for class 1 = 93 percentage**
- **Accuracy for class 2 = 95 percentage**
- **Average Accuracy = 94 percentage**

Figure 3: Pattern 1 : Log Likelihood

Codes/Pattern1.jpg

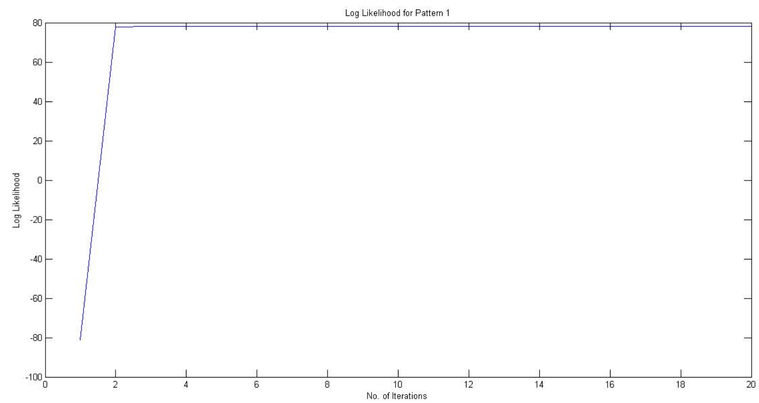
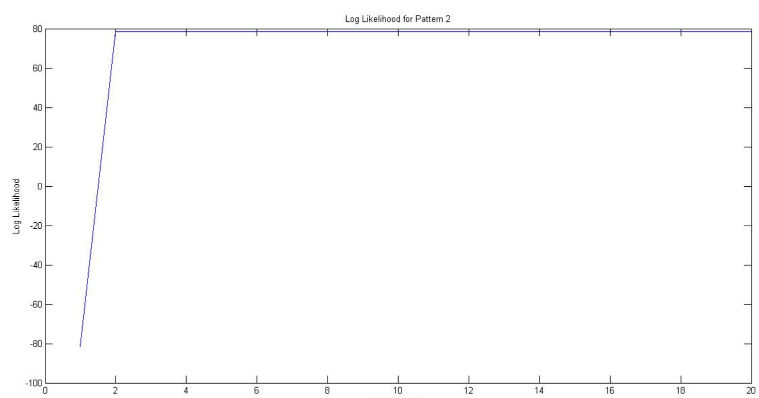


Figure 4: Pattern 1 : Log Likelihood

Codes/Pattern2.jpg



0.8 Bibliography

1. Permuter, Haim, Joseph Francos, and Ian Jermyn. "A study of Gaussian mixture models of color and texture features for image classification and segmentation." Pattern Recognition 39.4 (2006): 695-706.

2. Bishop, Christopher M. Pattern recognition and machine learning. Vol. 1. New York: springer, 2006.

3. Alpaydin, E. (2004). Introduction to machine learning. MIT press.

4. Duda, Richard O., Peter E. Hart, and David G. Stork. "Pattern Classification."