**CS6690- Pattern Recognition**

Report on

Programming Assignment 3

Submitted By

Group IV

Divya (ED11S002)

Swarna (EE11S019)

Archana (AM11S027)

****

Gaussian Mixture Models(GMM)

**Introduction**

The likelihood function is maximized with respect to the parameters – mean, covariance,mixing coefficients. Given a Gaussian mixture model, the goal is to maximize the likelihood function with respect to the parameters (comprising the means and covariances of the components and the mixing coefficients).

|  |  |  |  |
| --- | --- | --- | --- |
|  | Mixing Proportion | Mean of the component | Covariance matrix of the component |
| Component I | **0.610247** | **610.8251 973.6656** | **1.0e+004 \***  **1.7377 1.8390**  **1.8390 2.2714** |
| Component II | **0.334577** | **1.0e+003 \***  **0.3051 2.2829** | **1.0e+003 \***  **1.0642 -0.7433**  **-0.7433 6.0880** |
| Component III | **0.055176** | **1.0e+003 \***  **0.4706 1.1206** | **1.0e+005 \***  **1.2426 0.9676**  **0.9676 1.9286** |



Figure 1: Gaussian mixture distribution with 3 components in 2 dimensions

1.3-Means Clustering is done on pooled data to obtain initial estimates of mixin proportions , means and covariance matrices.

2.Expectation maximisation algorithm given below is then employed to improve the estimates until the coverence condition is met

    

Hidden markov models

A random sequence has the Markov property if its distribution is determined solely by its current state. Any random process having this property is called a Markov random process. If the state sequence can be determined from the data then it is called markov chain model. If the states are non observable then it is called Hidden Markov model.

In a Hidden Markov model, the state is not directly visible, but output, dependent on the state, is visible. Each state has a probability distribution over the possible output symbols. Therefore the sequence of symbols generated by an HMM gives some information about the sequence of states. Hidden Markov models are especially known for their application in temporal pattern recognition such as speech, handwriting, gesture recognition etc.

1. Tools for modelling DHMM using Data is given.

2. 3-Means clustering is employed over entire range of data and the given data is converted into a 39 length sequence containing patterns composed of {1,2,3}

3. The encoded representation is given as input to the DHMM code given and the HMM is modelled using training data.

4.Initial estimates for the model(i.e aij and bjk’s):

5.Final estimates after training:

states: 3

symbols: 5

**0.545654 0.000001 0.037958 0.000001 0.962027 0.000016**

**0.454346 0.000001 0.917419 0.000001 0.082195 0.000386**

**0.899305 0.000001 0.573247 0.000001 0.426752 0.000001**

**0.100695 0.000001 0.857465 0.000001 0.142534 0.000001**

**1.000000 0.004756 0.459215 0.017896 0.511260 0.006872**

**0.000001 0.000001 0.000001 0.000001 0.000001 0.000001**

**states: 3**

**symbols: 5**

**0.901606 0.000001 0.413716 0.000001 0.586283 0.000001**

**0.098394 0.000664 0.859506 0.000001 0.139829 0.000001**

**0.757078 0.000001 0.826153 0.000001 0.173845 0.000001**

**0.242922 0.000001 0.873134 0.000001 0.126865 0.000001**

**1.000000 0.008061 0.465447 0.018005 0.507932 0.000556**

**0.000001 0.000001 0.000001 0.000001 0.000001 0.000001**

**states: 3**

**symbols: 5**

**0.902391 0.000025 0.455988 0.000001 0.543987 0.000001**

**0.097609 0.000001 0.847716 0.000001 0.152283 0.000001**

**0.828186 0.000001 0.688630 0.000001 0.311369 0.000001**

**0.171814 0.000001 0.695207 0.000001 0.304791 0.000001**

**1.000000 0.013592 0.448994 0.019795 0.516210 0.001409**

**0.000001 0.000001 0.000001 0.000001 0.000001 0.000001**

The above training is done with the data generated by the tool since the encoded version of training data was not working.

**Support Vector Machine(SVM)**

The support vector machine is fundamentally a two-class classifier.The determination of the model parameters corresponds to a convex optimization problem and thus any local solution is even a global optimum.The SVM is a decision machine and so does not provide posterior probabilities. One commonly used approach is to construct *K* separate SVMs, in which the *k*th model *yk*(x) is trained using the data from class Ckas the positive examples and the data from the remaining *K −* 1 classes as the negative examples. This is known as the *one-versus-the-rest* approach.

a)For Traning SVM Classifier for two classes

*SVMStruct* = svmtrain(*Training*, *Group*)

*Training*Matrix of training data, where each row corresponds to an observation or replicate, and each column corresponds to a feature or variable. *Group*Column vector, character array, or cell array of strings for classifying data in *Training* into two groups. It has the same number of elements as there are rows in *Training*. Each element specifies the group to which the corresponding row in *Training* belongs.

b)For Tesing given data

*Group* = svmclassify(*SVMStruct*, *Sample*)

*Group* = svmclassify(*SVMStruct*, *Sample*) classifies each row of the data in Sample using the information in a support vector machine classifier structure SVMStruct, created using the [svmtrain](jar:file:///C:/Program%20Files%20%28x86%29/MATLAB/R2010a/help/toolbox/bioinfo/help.jar%21/ref/svmtrain.html) function. Sample must have the same number of columns as the data used to train the classifier in svmtrain. Group indicates the group to which each row of Sample has been assigned.

2.SVM classifiers are constructed for classes 1 and 2, 2 and 3 and 3 and 1 and the three decions are used to make final conclusion.

..

**Observation:**

Accuracy with 3 folds is 89.68% and with 9 folds is 92.08%