

Detecting the number of clusters during Expectation-Maximization clustering using Information Criterion

Ujjwal Das Gupta

Department of Computer Engineering
Delhi College of Engineering
India
ujjwal.das.gupta@coe.dce.edu

Vinay Menon

Department of Computer Engineering
Delhi College of Engineering
India
vinay.menon@coe.dce.edu

Uday Babbar

Department of Computer Engineering
Delhi College of Engineering
India
uday.babbar@coe.dce.edu

Abstract—This paper presents an algorithm to automatically determine the number of clusters in a given input data set, under a mixture of Gaussians assumption. Our algorithm extends the Expectation-Maximization clustering approach by starting with a single cluster assumption for the data, and recursively splitting one of the clusters in order to find a tighter fit. An Information Criterion parameter is used to make a selection between the current and previous model after each split. We build this approach upon prior work done on both the K-Means and Expectation-Maximization algorithms. We also present a novel idea for intelligent cluster splitting which minimizes convergence time and substantially improves accuracy.

Keywords—clustering, unsupervised learning, expectation-maximization, mixture of gaussians

I. INTRODUCTION

The clustering problem is defined as follows : given a set of n -dimensional input vectors $\{x_1, \dots, x_m\}$, we want to group them into an appropriate number of clusters such that points in the same cluster are positionally coherent. Such algorithms are useful for image compression, bioinformatics, pattern matching, data mining, astrophysics and many other fields. A common approach for the clustering problem is to assume a Gaussian Mixture Model. In this model, the input data is assumed to have been generated by selecting any one of k Gaussian distributions, and drawing the input vector from the chosen distribution. Each cluster is thus represented by a single distribution. The Expectation-Maximization algorithm [3] is a well known method to estimate the set of parameters for such a mixture corresponding to maximum likelihood, however, it requires pre-knowledge about the number of clusters in the data (k).

Determining this value is a fundamental problem in data clustering, and has been attempted using Information Theoretic [5], Silhouette based [7], [8] and Goodness-of-fit methods [4], [6]. The X-Means algorithm [10] is an Information Criteria based approach to this problem developed for use with the K-Means algorithm. X-Means works by alternatively applying two operations – The K-Means algorithm (Improve-params) to optimally detect the clusters for a chosen value of k , and cluster splitting

(Improve-structure) to optimize the value of k according to Information Criterion.

One of the major problems with X-means is that it assumes an identical spherical Gaussian of the data. Because of this, it tends to overfit data in elliptical clusters [6], or in an input set with data of varying cluster size. The G-Means and PG-Means algorithms try to solve this problem by projecting the data onto one dimension, and running a statistical goodness-of-fit test. This approach leads to better performance for non-spherical distributions, however, projections may not work optimally for all data sets. A projection can collapse the data from many clusters together, neglecting the difference in density. This requires multiple projections for accuracy [4].

Our algorithm uses an Information Criterion test similar to X-Means, however, it differs by considering each cluster to be generated by a general multivariate Gaussian distribution. This allows each distribution to take a non-spherical shapes, and permits accurate computation of the likelihood of the model. We use Expectation-Maximization instead of K-Means for greater accuracy in detection of the parameters. Further, relying on multivariate Gaussian distributions allow us to use the directional properties of the distributions as an aid for intelligent splitting. As we show in our test results, not only does this result in reduced convergence time, it remarkably improves the accuracy of our algorithm.

II. CONCEPTS AND DEFINITIONS

A. Mixture of Gaussians and Expectation-Maximization

In the Gaussian Mixture model the input data set $I = \{x_1, \dots, x_m\}$ where $x_i \in \mathbb{R}^n$ is assumed to be sampled from a set of distributions $L = \{A_1, \dots, A_k\}$ such that the probability density is given by

$$p(x_i) = \sum_{j=1}^k \phi_j G(x_i | A_j)$$

Where A_j denotes a gaussian distribution characterized by mean μ_j and variance matrix Σ_j . Φ_j denotes the normalized

weight of the j th distribution, and k is the number of distributions.

The likelihood of the data set is given by

$$L = \prod_{i=1}^m p(x_i)$$

The Expectation-Maximization algorithm EM(I,L) maximizes the likelihood for the given data set by repeating the following two steps for all $i \in \{1, \dots, m\}$ and $j \in \{1, \dots, k\}$ until convergence [9].

1. Expectation Step

$$w_{ij} \leftarrow P(A_j | x_i)$$

Equivalently,

$$w_{ij} \leftarrow \frac{\phi_j}{(2\pi)^{n/2} |\Sigma_j|^{1/2}} e^{-\frac{1}{2}(x_i - \mu_j)^T \Sigma_j^{-1} (x_i - \mu_j)}$$

2. Maximization Step

$$\phi_j \leftarrow \frac{1}{m} \sum_{i=1}^m w_{ij}$$

$$\mu_k \leftarrow \frac{\sum_{i=1}^m w_{ij} x_i}{\sum_{i=1}^m w_{ij}}$$

$$\Sigma_j \leftarrow \frac{\sum_{i=1}^m w_{ij} (x_i - \mu_j)(x_i - \mu_j)^T}{\sum_{i=1}^m w_{ij}}$$

B. Maximum Likelihood and Information Criterion

Increasing the number of clusters in the mixture model results in an increase in the dimensionality of the model, causing a monotonous increase in its likelihood. If we were to focus on finding the maximum likelihood model with any number of clusters, we would ultimately end up with a model in which every data point is the sole member of its own cluster. Obviously, we wish to avoid such a construction, and hence we must choose some criteria that does not depend solely on likelihood.

An Information Criteria parameter is used for selection among models with different number of parameters. It seeks to balance the increase in likelihood due to additional parameters by introducing a penalty term for each parameter. Of the various Information Criterion available, Bayesian Information Criterion (BIC) or Schwarz Criterion [11] has been shown to work the best with a mixture of Gaussians [12].

$$BIC = 2 \log(L) - f \log(|I|)$$

Where f is the number of free parameters. For a mixture of k Gaussians, f is given by

$$f = (k-1) + kd + k \frac{d(d-1)}{2}$$

Other Information Criterion measures like Akaike's Information Criterion [1] or Integrated Completed Likelihood [2] may also be used.

III. ALGORITHM

The algorithm functions by alternating Expectation-Maximization and Split operations, as depicted in Illustration 1. In the first figure, the EM algorithm is applied to a single cluster resulting in (a). The obtained distribution is split in (b), and the parameters are maximized once again to get (c). This is repeated to get the final set of distributions in (e). A formal version of the algorithm is given in Figure 3. The following lines describe the algorithm:

I is the set of input vectors

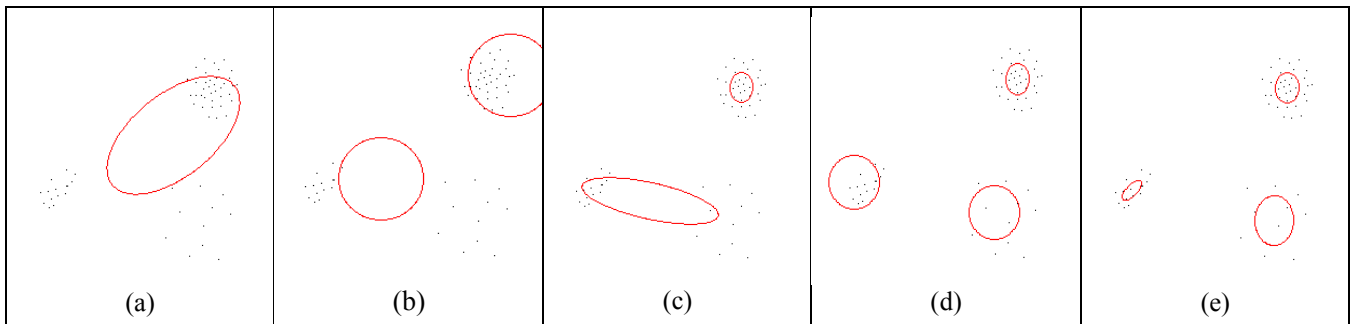


Figure 1. Sequence of EM and split operations in the algorithm

```

S ← {k}
SBackup ← S
PrevNode ← k
x ← -∞
B ← ∅
while (S ≠ B) do
  S ← EM(S, I)
  if BIC(S, I) < x then
    S ← SBackup
    B ← B ∪ {PrevNode}
  else
    x ← BIC(S, I)
    SBackup ← S
    for any d ∈ S - B do
      PrevNode ← d
      S ← (S - {d}) ∪ SPLIT(d)
    end for
  end if
end while

```

Figure 2. Algorithm

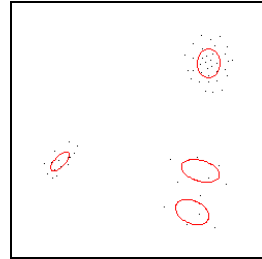
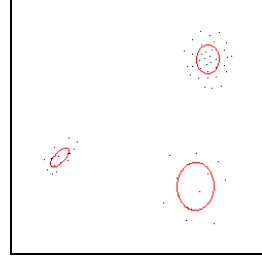


Figure 3. Backtracking

S is a set of normal distributions, initially consisting of a single member k , a distribution with random parameters

B is a set of marked or backtracked distributions

$PrevNode$ is the distribution which had been split in the previous iteration.

x is the highest encountered value of BIC

S_{Backup} is the set of distributions corresponding to a BIC value of x .

Repeat the following steps until all members of S are marked

1. Apply Expectation-Maximization on S
2. Compute the Bayesian Information Criterion for L . If it is less than x , backtrack to S_{backup} and mark the previously split distribution by adding it to B .
3. Remove an unmarked distribution from L , and apply the SPLIT procedure and add the two new distributions to L .

An instance of Backtracking can be seen in Figure 2. The algorithm splits one of the clusters in (a) and applies the EM algorithm to get (b). However, the BIC in (b) is lesser than in (a), and hence the algorithm backtracks to the three cluster model. The algorithm will try splitting each of the three clusters, and after backtracking from each approach, it will exit with (a) being the final output.

There are two ways to handle B , the list of marked distributions. One way is to clear the list every time a distribution has been split (A backup copy, as in the case of S , needs to be maintained). This is a very accurate method, however, performance is lost because a single cluster may be split multiple times, even when it has converged to its final position. Another approach is to maintain a common list

which is continuously appended, as in Figure 3. Although this may give rise to inaccuracy in the case when a previously backtracked distribution is moved, it can improve the convergence time of the algorithm considerably. The splitting procedure, described in the following section, ensures that such cases are encountered rarely.

A. The SPLIT Procedure

The above algorithm does not specify how a chosen cluster must be split. A simple solution would be to randomly introduce two new distributions with a mean vector close to that of the split distribution, and to allow the EM algorithm to correctly position the new clusters. However, there are a few problems with this approach. Not only does this increase convergence time in case the new clusters are poorly chosen, the chosen points may be near some other cluster, resulting in its displacement. Several correctly chosen clusters may be displaced during the runtime of the algorithm, leading to inefficiency.

A properly chosen split method should result in a cluster hierarchy similar to Divisive Hierarchical Clustering [7]. For this, the new clusters should largely account for those points which were already accounted for by the split cluster. Also, the direction of the split should be chosen such that the new centres are close to their positions after convergence. We suggest that the line for splitting should be chosen along the major axis representing the contour of constant density of the distribution (which is a hyper-ellipsoid of n -dimensions).

In the following steps we shall derive the parameters under a two-dimensional assumption in which the contour is given by an ellipse. We also assume $C = 1$ (The actual value of C affects the size and not the orientation of the ellipse, and as such does not make a difference in calculations).

We assume the following forms for the matrices:

$$\Sigma_j = \begin{bmatrix} \sigma_{11} & \sigma_{12} \\ \sigma_{21} & \sigma_{22} \end{bmatrix}$$

$$\Sigma_j^{-1} = \frac{1}{\sigma_{11}\sigma_{22} - \sigma_{12}\sigma_{21}} \begin{bmatrix} \sigma_{22} & -\sigma_{21} \\ -\sigma_{12} & \sigma_{11} \end{bmatrix}$$

$$x_i - \mu_j = \begin{bmatrix} x \\ y \end{bmatrix}$$

The contour of constant density for the distribution is given by the locus of points such that

$$(x_i - \mu_j)^T \Sigma_j^{-1} (x_i - \mu_j) = C$$

Solving the equation, we get:

$$\sigma_{22}x^2 - (\sigma_{21} + \sigma_{12})xy + \sigma_{11}y^2 = \sigma_{11}\sigma_{22} - \sigma_{12}\sigma_{21}$$

The angle made by the major axis of the ellipse with the x axis is given by:

$$\phi = \tan^{-1} \left(\frac{\sigma_{12} + \sigma_{21}}{\sigma_{11} - \sigma_{22}} \right)$$

The centers of the new distributions can be computed as follows:

$$new_1 = \mu_j + \begin{bmatrix} D \cos \phi \\ D \sin \phi \end{bmatrix}$$

$$new_2 = \mu_j - \begin{bmatrix} D \cos \phi \\ D \sin \phi \end{bmatrix}$$

Where D is proportional to the length of the major axis.

IV. RESULTS

We tested our algorithm on a set of test two-dimensional images generated according to the mixture of Gaussians assumption. The results are shown in Figures 4 and 5. The number of iterations required are roughly linear with respect to the number of clusters. The error in detection was measured as the root mean squared difference between the actual and detected number of clusters in the data. The error was very low, especially when the SPLIT procedure was used (under 2%). The intelligent splitting algorithm consistently outperforms the random split algorithm in both accuracy and performance. Its usage improves the average accuracy of the algorithm by more than 90%.

V. CONCLUSION

In conclusion, our algorithm presents a general approach to use any Information Criterion to automatically detect the number of clusters during Expectation-Maximization cluster analysis. Based on results of prior research, we chose the Schwarz criterion in our algorithm, however it may be applied to any criterion which may provide better performance for the data under consideration.

Interestingly, not only did our SPLIT algorithm help in the performance of the algorithm, it was observed that it improved its accuracy to a remarkable extent. Randomly splitting the clusters often tends to under-fit the data in the case when the algorithm gets stuck in local minima. Our split method reduces this problem by placing the new clusters in the vicinity of their final positions. Although we have derived it under two dimensions, it can easily be generalized to multi-dimensional problems.

We believe that our algorithm is a useful method to automatically detect the number of clusters in a Gaussian mixture data, with particular significance when non-spherical distributions may be present and multi-dimensional data projection is not desired.

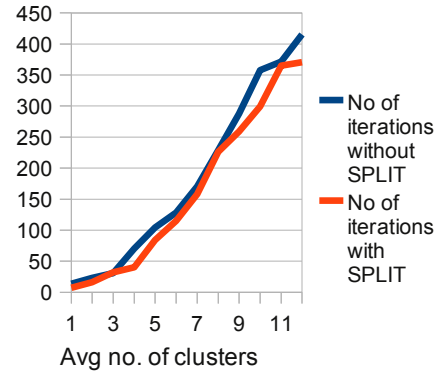


Figure 4. Performance of the algorithm

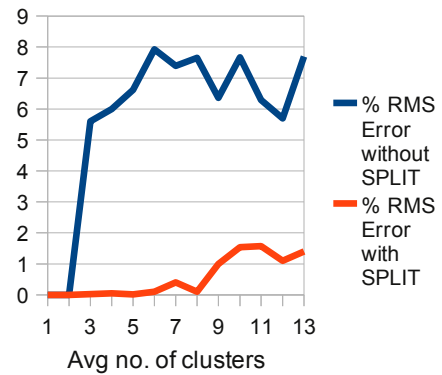


Figure 5. Accuracy of the algorithm

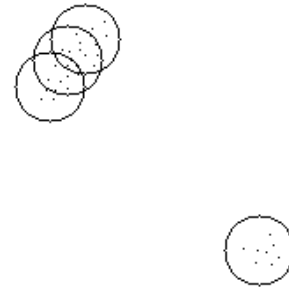


Figure 6. Overfitting in X-Means



Figure 7. Image with our algorithm

VI. ACKNOWLEDGEMENT

We would like to thank Prof. Asok Bhattacharyya, and Dr. Daya Gupta from Delhi Technological University for their invaluable comments and suggestions to help us in writing our paper.

We would also like to thank Dan Pelleg and the members of the Auton Lab at Carnegie Mellon University for providing us with access to the X-Means code.

VII. REFERENCES

- [1] H. Akaike, "A new look at the statistical model identification," *IEEE Transactions on Automatic Control*, vol.19, no.6, pp. 716-723, Dec 1974
- [2] C. Biernacki, G. Celeux and G. Govaert, "Assessing a mixture model for clustering with the integrated completed likelihood," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol.22, no.7, pp.719-725, Jul 2000
- [3] A.P. Dempster, N.M. Laird and D.B. Rubin, "Maximum likelihood from incomplete data via the em algorithm," *Journal of the Royal Statistical Society. Series B (Methodological)*, vol. 39, no. 1, pp. 1-38, 1977
- [4] Y. Feng, G. Hamerly and C. Elkan, "PG-means: learning the number of clusters in data." *The 12-th Annual Conference on Neural Information Processing Systems (NIPS)*, 2006
- [5] E. Gokcay, and J.C. Principe, "Information theoretic clustering," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol.24, no.2, pp.158-171, Feb 2002
- [6] G. Hamerly and C. Elkan, "Learning the k in k-means," *Advances in Neural Information Processing Systems*, vol. 16, 2003.
- [7] S. Lamrous and M. Taïeb, "Divisive Hierarchical K-Means," *International Conference on Computational Intelligence for Modelling, Control and Automation, 2006 and International Conference on Intelligent Agents, Web Technologies and Internet Commerce*, vol., no., pp.18-18, Nov. 28 2006-Dec. 1 2006
- [8] R. Lletí, M. C. Ortiz, L. A. Sarabia, and M. S. Sánchez, "Selecting variables for k-means cluster analysis by using a genetic algorithm that optimises the silhouettes," *Analytica Chimica Acta*, vol. 515, no. 1, pp. 87-100, July 2004.
- [9] A. Ng, "Mixtures of Gaussians and the EM algorithm" [Online] Available: <http://see.stanford.edu/materials/aimlcs229/cs229-notes7b.pdf> [Accessed: Oct 19, 2009]
- [10] D. Pelleg, and A. Moore, "X-means: Extending k-means with efficient estimation of the number of clusters," in *In Proceedings of the 17th International Conf. on Machine Learning*, 2000, pp. 727-734.
- [11] G. Schwarz, "Estimating the dimension of a model," *The Annals of Statistics*, vol. 6, no. 2, pp. 461-464, 1978.
- [12] R. Steele and A. Raftery, "Performance of Bayesian Model Selection Criteria for Gaussian Mixture Models", *Technical Report no. 559*, Department of Statistics, University of Washington, Sep 2009