CLUSTERING AND FINDING THE NUMBER OF CLUSTERS BY UNSUPERVISED LEARNING OF MIXTURE MODELS USING VECTOR QUANTIZATION

Sangho Yoon and Robert M. Gray

Information Systems Lab., Electrical Engineering
Stanford University
{holyoon,rmgray}@stanford.edu

ABSTRACT

A new Lagrangian formulation with entropy and codebook size was proposed to extend the Lagrangian formulation of variable-rate vector quantization. We use the new Lagrangian formulation to perform clustering and to find the number of clusters by fitting mixture models to data using vector quantization. Experimental results show that the entropy and memory constrained vector quantization outperforms the state-of-the art model selection algorithms in the examples considered.

Index Terms— Clustering, Vector Quantization, Mixture Models

1. INTRODUCTION

In pattern recognition, it is important to find the underlying distribution of given data. Cluster analysis is often used to estimate the underlying distribution, and selecting the number of clusters is a crucial part in clustering. Estimating the number of clusters has been actively studied over the years and many algorithms have been suggested, including an EM-based approach [12] with complexity penalties [17][14], a Bayesian approach [18], and an information theoretic approach [19]. Estimating the number of clusters is not only a difficult problem in unsupervised learning where the number of classes is unknown in advance, but also is an important issue in supervised learning when the number of components to fit a mixture model to each class must be estimated.

Vector quantization (VQ) design [5] can be used to cluster data because in VQ an input vector is represented by one of a predefined set of patterns on the basis of which pattern is closest to the given input vector. VQ has been used successfully in pattern recognition, including speech and image processing [1][10][7]. VQ design can be also viewed as fitting a model when partition cells are represented by their conditional probability density functions and prior probabilities are weights. In particular, we are interested in fitting Gauss mixture models (GMM) to data in VQ design (Gauss mixture VQ or GMVQ) [8][7], but our algorithm can be generalized to any type of model-based clustering. See usefulness of mixture models in [17].

In [2][3][4], an iterative model selection algorithm using a Lagrangian formulation with entropy and codebook size constraints was proposed and an optimization step for pruning an encoder partition of unneeded cells was added to the generalized Lloyd algorithm [5]. The algorithm in [2][3][4] prunes codewords (or clusters) iteratively if so doing decreases a Lagrangian distortion. We follow this idea of pruning codewords in fitting GMMs.

The rest of paper is organized as follows: In Section 2, we briefly review VQ. In Section 3, we present our realization of the Lagrangian formulation with combined entropy and codebook size constraints in [2][3][4] for fitting a GMM. In Section 4, we show experimental results in finding the number of clusters on both synthetic and real world data sets. Finally we conclude in Section 5.

2. BACKGROUND

A vector quantizer of dimension p (the number of features is p) and size K (the number of clusters is K) is made up of an *encoder* α, a *decoder* β, and a *length function* l. An *encoder* α is a mapping of an input vector x in p-dimensional Euclidean space, $\mathcal{R}^p$, into an index $i \in \mathcal{I} = \{1, 2, \ldots, K\}$. The encoder is described by a partition $S = \{S_i : i = 1, 2, \ldots, K\}$ such that $S_i = \{x : \alpha(x) = i\}$. A *decoder* β converts the index into a source reproduction $\hat{x}$, and β is associated with a reproduction codebook $C = \{\beta(i) : i \in \mathcal{I}\}$. Finally, a *length function* l measures the cost or instantaneous rate of an index i, and it is admissible if $\sum_{i \in I} e^{-l(i)} \leq 1$. Both l and the requirement of admissibility will be seen to be closely related to the "prior probability" of the cluster indexed by i. For a *fixed-rate* quantizer, $l(i)$ is fixed at $\ln K$ for all i. Otherwise, a quantizer is said to be *variable-rate*. The symbol q denotes the overall mapping $q(x) = \beta(\alpha(x))$.

This work was supported by the National Science Foundation under NSF Grants CCR-0309701.

The performance of a quantizer can be measured by the quality of its reproduction and the average rate of the reproduction index. We assume that X is a random vector described by density f. The reproduction quality can be measured by average distortion between input X and reproduction $\hat{X}$ (=$\beta(\alpha(X))$):

$$D_f(q) \quad = \quad E_f d(X, \hat{X}) = E_f d(X, \beta(\alpha(X))).$$

The average index cost is measured by an average rate incorporating both an average length and a codebook size penalty [2][3][4]:

$$R_f(q) \quad = \quad (1-\eta) E_f(l(\alpha(X))) + \eta \ln K(q) \qquad (1)$$

where $K(q)$ is the number of codewords (or clusters) of q and $\eta \in [0,1]$. When the length function is optimized over all admissible length functions ($l(i)$ = -lnPr$(\alpha(X) = i)$), $E_f(l(\alpha(X)))$ becomes the Shannon entropy of encoder output ($H_f(q)$) [9][2][3][4]. The codebook size constraint can be interpreted as a constraint on memory [2][3][4]. Given a Lagrangian multiplier $\lambda > 0$, define the Lagrangian distortion

$$\rho(\lambda, x, i) \quad = \quad d(x, \beta(i)) + \lambda[(1-\eta) l(i) + \eta \ln K(q)] \qquad (2)$$

where $\beta(i) = \beta(\alpha(x))$. Then the expected Lagrangian distortion is

$$\begin{aligned} \rho(f, \lambda, \eta, q) \quad &= \quad D_f(q) + \lambda R_f(q) \\ &= \quad E_f d(X, \beta(\alpha(X))) \\ &\quad + \lambda[(1-\eta) E_f(l(\alpha(X))) + \eta \ln K(q)] \end{aligned} \qquad (3)$$

We are interested particularly in GMVQ, where we fit a Gauss mixture model (GMM) to data using the Lloyd algorithm with a suitable distortion measure [7]. The EM algorithm [12] is the most popular approach to fitting a GMM to data, but the Lloyd algorithm provides an alternative. The main difference between the Lloyd and the EM algorithms is that the EM fits a GMM to each observed vector, whereas the Lloyd fits a single component of a GMM to each observed vector. This "hard" assignment of components to observed data is based on the information theoretic property of the Gaussian density being a "worst case" for designing robust compression/source coding systems [7][9]. In GMVQ, each cluster is represented by its prior probability w_i ($w_i \geq 0$ and $\sum_{i=1}^{K} w_i = 1$) and a cluster conditional pdf $g_i(x)$, which is a multivariate Gaussian:

$$\begin{aligned} g_i(x) \quad &= \quad g(x|\alpha(x) = i) \\ &= \quad \frac{\exp\left(-\frac{(x-\mu_i)^t \Sigma_i^{-1} (x-\mu_i)}{2}\right)}{(2\pi)^{p/2} |\Sigma_i|^{1/2}} \end{aligned} \qquad (4)$$

where g is a fitted GMM, and μ_i and Σ_i are the mean vector and covariance matrix of cluster i, respectively, and we assume Σ_i to be non-singular.

The quantizer mismatch (QM) distortion was used to design GMVQ in [7], and it can be interpreted as the maximum a posteriori (MAP) selection of a Gaussian component from a collection of Gauss models g_i with a probability mass function w_i if the unknown source f is in fact a GMM. In [7], to have flexible control over the number of components in a GMM, the QM distortion is modified to have a general multiplier λ for a log probability term $\ln w_i$:

$$d_{\mathrm{QM},\lambda}(x, i) \quad = \quad d_{GMVQ}(x, i) - \lambda \ln w_i, \qquad (5)$$

where $d_{\mathrm{GMVQ}}(x, i) = \frac{1}{2}(x-\mu_i)^t \Sigma_i^{-1} (x-\mu_i) + \frac{1}{2} \ln |\Sigma_i|$.

3. VECTOR QUANTIZATION FOR UNSUPERVISED FITTING OF GAUSS MIXTURE MODELS

The new rate term in (1) is an explicit function of the number of clusters. Thus it will allow us to have better control over the number of components than just using the entropy term $H_f(q)$ by penalizing complex models with large number of clusters. Incorporating this new rate term into the QM distortion in (5), we have the following distortion function

$$\begin{aligned} d_{\mathrm{QM},\lambda}(x, i) \quad &= \quad d_{\mathrm{GMVQ}}(x, i) - \lambda \left[(1-\eta) \ln w_i \right. \\ &\quad \left. + \eta \ln K(q)\right]. \end{aligned} \qquad (6)$$

We use the Lloyd algorithm to fit a GMM and estimate the number of components simultaneously by minimizing the average distortion, $E_f d_{\mathrm{QM},\lambda}(X, \alpha(X))$, iteratively. When the true distribution f is unknown, minimization of the sample average, $\sum_{n=1}^{N} d_{\mathrm{QM},\lambda}(x_n, \alpha(x_n))$, is used for a training set $\{x_1, x_2, .., x_N\}$.

Using the Lloyd optimality conditions for encoder, decoder, length function and partition in [4], for $\eta \in [0, 1)$, the complete optimization steps of our algorithm for the modified QM distortion in (6) are

- For a given decoder β, length function l, and the number of clusters K, the optimal encoder is

  $$\alpha(x) \quad = \quad \mathrm{argmin}_i (d_{\mathrm{GMVQ}}(x, i) + \lambda(1-\eta) l(i)).$$

- For a given encoder α, length function l, and the number of clusters K, the optimal decoder is

  $$\beta(i) = \mathrm{argmin}_y E(d_{\mathrm{GMVQ}}(X, y) | \alpha(X) = i) = \mathcal{N}(\mu_i, \Sigma_i)$$

  where $\mu_i = E(X|\alpha(X) = i)$ and $\Sigma_i = E((X - \mu_i)(X - \mu_i)^t | \alpha(X) = i)$.

- For a given encoder α, decoder β, and the number of clusters K, the optimal length function is

  $$l(i) = -\ln(\Pr(\alpha(X) = i)),$$

  and $\Pr(\alpha(X) = i) \neq 0$ for $i \in \mathcal{I}$.

- A necessary condition for a partition $\mathcal{S}$ to be optimal is that there is no subpartition $\mathcal{S}^{'} \subset \mathcal{S}$ for which

$$d_{\text{GMVQ}}(f, q') + \lambda((1-\eta)H_f(q') + \eta \ln K') \leq d_{\text{GMVQ}}(f, q) + \lambda((1-\eta)H_f(q) + \eta \ln K)$$

  where K' is the number of clusters of a subpartition $\mathcal{S}^{'}$, and q' and q are optimal quantizers for $\mathcal{S}^{'}$ and $\mathcal{S}$, respectively.

Note that the $\ln K$ term plays a role only in optimizing the number of clusters. The subpartition $\mathcal{S}'$ has either cells of $\mathcal{S}$ or unions of cells of $\mathcal{S}$. The number of subpartitions of $\mathcal{S}$ ($|\mathcal{S}| = K$) is ${}_K C_2 + {}_K C_3 + ... + {}_K C_K$, where ${}_K C_i = K!/i!(K-i)!$. To reduce searching complexity of candidate subpartitions, we use the pairwise nearest neighbor algorithm (PNN) [6]. In using PNN, we merge clusters which maximally decrease the average distortion $\sum_{n=1}^{N} d_{\text{QM},\lambda}(x_n, \alpha(x_n))$.

The proposed algorithm starts with large number of clusters and we use a tree-structured VQ [5] to obtain the initial clustering and corresponding parameters (μ_i, Σ_i, and w_i). We grow a tree until the number of leaf nodes reaches at $K_{\max}$. Then we iteratively prune clusters as we optimize encoder, decoder, length function and the number of clusters. We repeat these four optimization steps until convergence is reached.

To avoid singular covariance estimates, we use the regularization technique in [15]. We first regularize the sample-based estimates by the pooled covariance and we further regularize by a multiple of the identity matrix, which effectively decreases the larger eigenvalues and increases the smaller eigenvalues.

4. EXPERIMENTAL RESULTS

In this section, we test our algorithm on synthetic and real world data sets. We compare our algorithm with two other model selection algorithms that are fitting mixture models by the EM algorithm. Figueiredo et al. proposed a model selection algorithm using the minimum message length criterion (MML) [17] (termed EM-MML hereafter), and Dy et al. proposed a model selection algorithm using the Bayesian information criterion (BIC) [14] (termed EM-BIC hereafter). They both use the EM algorithm to fit GMMs, but they differ in estimating the number of clusters. EM-MML prunes a cluster with minimum weight in each iteration, and then runs the EM to fit a GMM. It finally selects the best model based on the MML criterion. However, EM-BIC merges clusters based on the method in [16], and selects the best model based on the BIC. Note that Dy et al. also proposed to perform feature selection along with estimating the number of clusters, but we only run their algorithm without feature selection for fair comparison. GMMs are fitted by all three model selection algorithms, and they are compared in terms of estimating the number of clusters and classification performance as described in [13]. We preprocess all data sets so that each feature in each data set has zero mean and unit variance. As stated in [4], we are interested in the behavior of codebook size in (3) for small values of η. Thus we set η to 0.2 in (6). The Lagrangian parameter λ in (6) is set to 1 and 3 for synthetic data set and real world data sets, respectively. Anecdotal evidence suggests that as the dimension of feature vector increases or data become sparse, merging or pruning clusters become harder. In this paper, we do not attempt to find optimal values of η and λ. The values are chosen based on our intuition and some trial and errors.

We run the model selection algorithms on three synthetic data sets in an unsupervised manner. We fit a GMM to each data set according to each model selection algorithm and record the number of clusters of a GMM. The first synthetic data set has 1000 samples generated from two Gaussian clusters

$$\mu_1 = \begin{bmatrix} 0 \\ 0 \end{bmatrix}, \Sigma_1 = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}, w_1 = 0.8;$$
$$\mu_2 = \begin{bmatrix} 2 \\ 2 \end{bmatrix}, \Sigma_2 = \begin{bmatrix} 1 & 1 \\ 1 & 1.5 \end{bmatrix}, w_2 = 0.2.$$

The second synthetic data set has 900 samples from three equiprobable Gaussian clusters with means at (0,-2), (0,0) and (0,2), and equal covariance matrices diag(2,0.2). The third synthetic data set has 800 samples from four equiprobable Gaussian clusters with means at (0,3), (1,9), (6,4) and (7,10) and equal covariance matrices diag(1,1). For each data set, we randomly generate data 50 times and $K_{\max}$ is set to 30.

Table 1 shows unsupervised learning results on the three synthetic data sets. As can be seen from Table 1, Lloyd and EM-BIC perfectly identify the number of clusters, whereas EM-MML makes mistakes in the two- and four- cluster data sets.

We also test the model selection algorithms on two real world data sets from the UCI learning repository: the image data set and the handwritten digit data set with 47 zernike moments. The image data set is the image segmentation data set, and contains 2,320 samples. The image data set has seven classes: brickface, sky, foliage, cement, window, path, and grass, and 18 features are extracted from a 3×3 region. The zernike data set is the handwritten digit recognition data set, and has 200 images for each digit (i.e., 2000 images in total).

For the real world data sets, we randomly divide each data set into two sets of equal size: a training set and a test set. We first perform unsupervised learning for the three algorithms on training data. We record the estimated number of clusters. We do not use any class label information in the training stage. After unsupervised learning, we label each cluster by majority vote using the class labels provided, and we evaluate three algorithms by classifying test samples. We repeat this random division of half and testing twenty times. Table 2 and 3 shows results on the two real world data sets. $K_{\max}$ is also set to 30. In the image data set, both Lloyd and EM-MML

estimate about two clusters per class, but Lloyd shows better classification performance than EM-MML. In the zernike data set, although Lloyd and EM-MML estimate about one cluster per class, Lloyd outperforms EM-MML. According to [13], EM-MML uses the number of classes as the lower bound for estimating the number of clusters. EM-BIC shows the worst classification performance for both the image and zernike data sets. This can be attributed to the underfitting of EM-BIC. In particular, it uses only about two clusters in the zernike data set although there are ten classes.

Table 1. Average number of clusters. Results on three synthetic data sets. Numbers in parenthesis are standard deviations.

	Lloyd	EM-MML	EM-BIC
Two-cluster data set	2(0)	2.06(0.24)	2(0)
Three-cluster data set	3(0)	3(0)	3(0)
Four-cluster data set	4(0)	4.06(0.42)	4(0)

Table 2. Image data set: seven classes.

	Lloyd	EM-MML	EM-BIC
Avg # of clusters	14.6(1.32)	13.8(1.94)	11.8(1.22)
Avg misclassification (%)	21.54(4.9)	32.84(5.1)	34.39(4.47)

Table 3. Zernike data set: ten classes

	Lloyd	EM-MML	EM-BIC
Avg # of clusters	9.65(1.77)	10(0)	1.8(0.4)
Avg misclassification (%)	34.87(7.1)	56.42(3.62)	84.03(3.67)

5. CONCLUSIONS

Our algorithm fits GMMs for clustering and attempts to find an optimal number of clusters by pruning clusters iteratively. Experimental results on both synthetic and real world data sets show that the entropy and memory constrained VQ is superior to the state of the art model selection algorithms on these data sets.

6. REFERENCES

[1] P.A. Chou, T. Lookabaugh, R.M. Gray, "Entropy-constrained vector quantization," IEEE Trans. on ASSP Vol. 37, No. 1, pp.31-42, January 1989.

[2] R.M. Gray, J.T. Gill, III, "A Lagrangian formulation of fixed rate and entropy/memory constrained quantization," DCC 2005, March 2005.

[3] R.M. Gray, J.T. Gill, III, "Quantization with Joint Entropy/Memory Constraints," DCC 2006, March 2006.

[4] R.M. Gray, T. Linder, J. Gill, III, "Lagrangian Vector Quantization with Combined Engropy and Codebook Size Constraints," submitted to *IEEE Trans. on Info. Theory*.

[5] A. Gersho, R.M. Gray, Vector Quantization and Signal Compression, Kluwer Academic Press, 1992.

[6] W.H. Equitz, "A New Vector Quantization Clustering Algorithm," IEEE Trans. ASSP, Oct., 1989.

[7] A. Ayer, K. Pyun, Y. Huang, D. O'Brien, R.M. Gray, "Lloyd Clustering of Gauss Mixture Models for Image Compression and Classification," Signal Processing: Image Communication, Vol. 20, June 2005, pp. 459-485

[8] P. Hedelin, J. Skoglund, "Vector Quantization based on Gauss Mixture Models," *IEEE Trans. Speech Audio Process 8(4)*, July, 2004, 385-401.

[9] R.M. Gray, T. Linder, "Mismatch in high rate entropy constrained vector quantization," Vol. 49, pp. 1204-1217, IEEE Trans. Inform.Theory, May, 2003.

[10] S. Yoon, R.M. Gray, "Feature Selection based on Maximizing Separability in Gauss Mixture Model and its Application to Image Classification," *Proc. of ICIP 2005*, Sep, 2005, Genoa, Italy.

[11] S. Lloyd, "Least square quantization in PCM," IEEE Trans. on Infor. Theory, 1957. Bell Labs Tech. Note.

[12] A. Dempster, N. Laird, D. Rubin, "Maximum likelihood from incomplete data via the EM algorithm (with discussion)," Journ. of the Royal Statis. Soc., Series B, Vol. 39, 1977, p. 1-38.

[13] M. Law, M. Figueiredo, A. Jain,"Simultaneous Feature Selection and Clustering Using Mixture Models," IEEE Trans. PAMI, 26(9): 1154-1166, 2004

[14] J. Dy, C. Brodley , "Feature Selection for Unsupervised Learning," Journal of Mach. Learn. Res, Aug., 2004.

[15] J.H. Friedman, "Regularized Discriminant Analysis," *J. Am. Statistical Assoc.,* vol. 84, pp. 165-175. March 1989.

[16] C.A. Bouman, M. Shapiro, G.W. Cook, C.B. Atkins, H. Cheng, "Cluster: An unsupervised algorithm for modeling Gaussian mixtures," in http://cobweb.ecn.purdue.edu/∼bouman/software/cluster/

[17] M. Figueiredo, A.K. Jain, "Unsupervised learning of finite mixture models", IEEE Trans. PAMI, vol. 24, no. 3, pp. 381-396, Mar., 2002.

[18] C. Fraley, A. Raftery, "How many clusters? Which clustering methods? Answers via model-based cluster analysis," *Computer Journal,* 41, 578-588. 1998.

[19] C. Sugar, G. James, "Finding the Number of Clusters in a Data Set : An Information Theoretic Approach," *Journal of the American Stat. Assoc.* 98, 750-763, 2003.