Probabilistic Model-Based Clustering of Multivariate and Sequential Data

Padhraic Smyth*
Department of Information and Computer Science
University of California, Irvine,
CA 92697-3425
smyth@ics.uci.edu

Abstract

Probabilistic model-based clustering, based on finite mixtures of multivariate models, is a useful framework for clustering data in a statistical context. This general framework can be directly extended to clustering of sequential data, based on finite mixtures of sequential models. In this paper we consider the problem of fitting mixture models where both multivariate and sequential observations are present. A general EM algorithm is discussed and experimental results demonstrated on simulated data. The problem is motivated by the practical problem of clustering individuals into groups based on both their static characteristics and their dynamic behavior.

1 Introduction and Motivation

Consider the following problem. We have a set of individuals (a random sample from a larger population) whom we would like to cluster into groups based on observational data. For each individual we can measure characteristics which are relatively *static* (e.g., their height, weight, income, age, sex, etc). Probabilistic model-based clustering in this context usually takes the form of a finite mixture model, where each component in the mixture is a multivariate probability density function (or distribution function) for a particular group. This approach has been found to be a useful general technique across a variety of applications for extracting hidden structure from multivariate data (Symons, 1981; McLachlan and Basford, 1988; Banfield and Raftery, 1993; Celeux and Govaert, 1995; Cheeseman and Stutz, 1996; Thiesson et al, 1997).

Consider instead where we have *dynamic* observations (rather than *static* multivariate measurements) for each individual (such as time series data). For example, in a medical context we might have EEG traces for a set of patients, or in a user-modeling context we might have traces of user actions within a digital environment such as Unix, or at a particular Website, etc. It is straightforward to extend the standard multivariate model-based clustering methods to such sequential data by using a finite mixture of probability density functions on sequences (e.g., a Markov model for discrete-valued observations or an autoregressive model on real-valued data; see Rabiner et al. (1989), Krogh et al (1994), or Smyth (1997)). Issues such as defining distances between sequences, clustering sequences of different lengths, and finding the best number of clusters from the data, can all be handled in a principled manner. This is a distinct advantage of the probabilistic framework compared to alternatives (such as defining pairwise distances between sequences using some form of edit-distance, for example), but comes of course at the cost of imposing a specific model structure on the data.

The more general case of course is when we have both *static* and *dynamic* observations on each individual, e.g., a person's age and income (static) combined with traces of Web page accesses (dynamic). Such combinations of heterogeneous data types are increasingly common across applications in medicine, business, science, and engineering as data becomes easier to collect and store.

Clustering in this context is non-trivial. The problem is how to combine the static and dynamic data in a meaningful manner; for example, how should the dynamic data be weighted relative to the static data? One practical option is to reduce the dynamic observations to multivariate form (e.g., a histogram of which Web pages were visited) and, thus, perform multivariate clustering. However, while this methodology is often useful in practice, it may ignore useful dynamic

*Also with the Jet Propulsion Laboratory 525-3660, California Institute of Technology, Pasadena, CA 91109.

information. For example, one could have the same marginal distributions for two different sets of dynamic behavior.

In this paper we show that the mixture model clustering framework can be extended relatively easily to clustering of individuals based on static and dynamic observations. For the particular case when the static characteristics and dynamic behavior are assumed independent given the hidden cluster variable, the Expectation-Maximization (EM) procedure (for estimating the parameters of the model) has a particularly intuitive interpretation. In this paper we outline the motivation for the model, the EM learning algorithm, and illustrate the method on simulated data.

2 Notation

2.1 Multivariate Model-Based Clustering

Let X be a multivariate random variable taking values $\underline{x}$ and let C be a discrete-valued cluster variable taking values $c_k, 1 \leq k \leq K$. We will refer to the $\underline{x}$ as the *static characteristics* of an individual on which measurements are made. The probability of individual observation $\underline{x}$, assuming a finite mixture model with K components, is defined as:

$$f(\underline{x}) = \sum_{k=1}^{K} f_k(\underline{x}|c_k)p(c_k) \qquad (1)$$

where $p(c_k)$ is the marginal probability of the kth cluster, and $f_k(\underline{x}|c_k)$ is the multivariate density model for the kth cluster. (The dependence on the parameters of the mixture model and the model itself (K) has been suppressed here for simplicity). This mixture model is the basis for probabilistic model-based clustering, where EM can be used to fit the parameters given observed data $D_x = \{\underline{x}_1, \ldots, \underline{x}_n\}$ and the resulting K fitted component models are interpreted as individual clusters.

2.2 Model-Based Clustering of Sequences

Now let S be a "sequential" random variable taking values $\underline{s} = \{s_1, \ldots, s_t, \ldots, s_T\}$. (For convenience of presentation we will assume that the s_t are univariate and discrete-valued, although this is not necessary). We can think of the process generating the s_t as being a stochastic finite state machine, i.e., a model which can generate a string of observations $\underline{s}$, where the length T can vary, according to some probabilistic model $p(S = \underline{s})$. We will assume each individual behaves according to some such probability model on S, and we will refer to the observed $\underline{s}$ as the observed *dynamic behavior* of an individual. Thus, we can have a random sample of individuals for whom we have sequential measurements $D_s = \{\underline{s}_1, \ldots, \underline{s}_N\}$. For simplicity we can imagine a 1-1 mapping from randomly chosen individuals and the $\underline{s}_i$, although this is not strictly necessary since multiple observations for a given individual can easily be handled.

Clustering in this context means modeling the dynamic behavior of the population of individuals as a finite group of K behaviors in the form of a mixture model:

$$f(\underline{s}) = \sum_{k=1}^{K} p_k(\underline{s}|c_k)p(c_k) \qquad (2)$$

where $p(c_k)$ is the marginal ("prior") probability of component model k and $p_k(\underline{s}|c_k)$ is the generative probabilistic sequence model for the kth group. For example, each $p_k(\underline{s}|c_k)$ could be a first-order Markov model with a different transition matrix A_k. Learning in this context is directly analogous to the standard EM algorithm for learning with multivariate mixture models (not surprising since Equations (1) and (2) have the same general form). Note that the model above is not equivalent to the standard hidden Markov model: there the cluster variable C itself has Markov behavior whereas here it is static.

This general idea of sequence clustering can be generalized to a variety of models where the cluster variable itself has Markov behavior, some of which lead to the standard hidden Markov models, and some of which are quite different in flavor (see Smyth (1998) for examples). We note in passing that all of these models have natural interpretations as *directed graphical models* (aka belief networks) (Smyth, Heckerman, and Jordan; 1997).

3 Clustering based on Static Characteristics and Dynamic Behavior

In this paper, we restrict our attention specifically to the following situation (since it is a well-motivated one from a practical viewpoint as described earlier):

- A static cluster variable C
- Observed static characteristics $\underline{x}$
- Observed dynamic behavior $\underline{s}$

Thus, our observed data are of the form $D = \{(\underline{x}_1, \underline{s}_1), \ldots, (\underline{x}_N, \underline{s}_N)\}$. We assume that there is a hidden variable C (the cluster identity) whose value is unobserved and that the data are being generated by

a finite mixture model of the form

$$f(\underline{x}, \underline{s}) = \sum_{k=1}^{K} f_k(\underline{x}, \underline{s}|c_k)p(c_k) \quad (3)$$

where the $p(c_k)$ are the marginal probabilities for cluster k. The $f_k(\underline{x}, \underline{s}|c_k)$ are the component models for each cluster, and can be factorized as either (1) $p_k(\underline{s}|\underline{x}, c_k)f_k(\underline{x}|c_k)$ or (2) as $f_k(\underline{x}|\underline{s}, c_k)p_k(\underline{s}|c_k)$.

From a causal viewpoint we can interpret each factorization. The first factorization tells us that the dynamic behavior $\underline{s}$ is dependent on the static characteristics $\underline{x}$. This type of dependence is quite plausible in many practical situations, e.g., the EEG of a patient may depend on the patient's age or condition, or the Web surfing behavior of an individual may depend on an individual's educational background, age, or related factors. The second factorization, where the static characteristics $\underline{x}$ are dependent on the dynamic behavior $\underline{s}$, seems less likely to be a common causal mechanism for typical applications.

Parametrization of these dependencies depends on the precise nature of the sequential and multivariate probability models being used. For example, if the sequential observations s_t are discrete-valued and modeled by a finite-order Markov model, $p_k(\underline{s}|\underline{x}, c_k)$ can represent a model where the Markov transition probabilities are parametrized as a logistic function of the multivariate observation $\underline{x}$ (Hughes, 1993; MacDonald and Zucchini, 1997). For a binary first-order Markov chain we have

$$p_k(s_i|s_j, \underline{x}, c_k) = \frac{e^{f_{ijk}(\underline{x})}}{1 + e^{f_{ijk}(\underline{x})}}, \quad (4)$$

or, equivalently,

$$f_{ijk}(\underline{x}) = \log \frac{p_k(s_i|s_j, \underline{x}, c_k)}{1 - p_k(s_i|s_j, \underline{x}, c_k)} \quad (5)$$

where $f_{ijk}(\underline{x})$ is a parametrized function of $\underline{x}$ for the log-odds ratio of the Markov transition probability from state j to state i for class k. If $f_{ijk}(\underline{x})$ is assumed to be a linear function of $\underline{x}$ then one is in effect modeling the transition probability as a threshold function in $\underline{x}$-space, i.e., $p_k(s_i|s_j, \underline{x}, c_k) = 0.5$ for $f_{ijk}(\underline{x}) = 0$ and goes to 0 or 1 as one travels away from the line defined by this solution. Such a threshold effect might be useful (for example) in modeling the development of motor skills (the dynamic variable) as a threshold function of infant age (a 1-dimensional static variable). Equally well, one could model $f_{ijk}(\underline{x})$ as a symmetric unimodal "bump" function centred around a mean $\underline{\mu}_k$ in $\underline{x}$-space, where here the transition probability is maximized for $\underline{x}$ values close to $\underline{\mu}_k$ and falls off monotonically as $\underline{x}$ gets further from $\underline{\mu}_k$. In this case $\underline{\mu}_k$ could (for example) represent a specific 2-dimensional cluster in *age*, *education* − *level* space (the static variables) with particular Web-surfing characteristics (the dynamic variable).

For the purpose of clustering, a potentially simplifying assumption is that the static and dynamic variables are conditionally independent given the cluster variable:

$$f_k(\underline{x}, \underline{s}|c_k) = f_k(\underline{x}|c_k)p_k(\underline{s}|c_k). \quad (6)$$

This leads to a particularly simple model structure in that we do not need to parametrize the coupling of the static and dynamic models explicitly. It also leads to an intuitive interpretation in terms of the associated EM algorithm for learning, which we will discuss in the next section. This assumption may be especially useful in practice for the following general reasons:

- For clustering purpose we are interested in the *differences* between classes. To detect such differences we do not necessarily need to fully model all dependencies.
- Learning with hidden variables in a Markov context is known to be difficult since typically there are many local maxima of the likelihood surface in parameter space. Thus, the simpler the model, the better chance there is of fitting it in a reliable fashion.
- In the absence of problem-specific prior knowledge for a particular application, it may be difficult to choose an appropriate parametrization for the dependence of $\underline{s}$ on $\underline{x}$ and to reliably learn this parametrization from a finite amount of data. For example, the issue of identifiability of such models in a general mixture context is somewhat open.

4 EM Learning of the Cluster Model

The EM algorithm can be used to search for the unknown parameters that maximize the likelihood of the observed data (or find the mode of the posterior density from a *maximum a posteriori* viewpoint) under the model assumptions stated earlier. To make the discussion more specific, we could assume that the $\underline{x}$ are d-dimensional real-valued variables with each multivariate cluster component $f_k(\underline{x}|c_k)$ being modeled by a Gaussian with parameters Σ_k and mean $\underline{\mu}_k$. We could also assume that the sequential observations s_t are univariate and discrete-valued (taking m values), and each cluster is modeled as a first-order Markov model with transition matrix A_k and initial distribution $\pi(s_1)$. We will use these models for illustration in the next section, but in the general discussion below we can think of much broader class of model structures,

e.g., general graphical models for the multivariate components and linear models (such as ARMA models) for real-valued sequential data.

The $p(c_k)$, the marginal probabilities of each cluster, are also typically unknown and must be learned from the data. Let Φ_K denote the overall set of unknown parameters. In this paper we will assume K is fixed: more generally, one can use penalized likelihood, cross-validation, or Bayesian techniques to find the best value for K as well.

We can express $f(\underline{x}, \underline{s}|\Phi_K)$ as

$$f(\underline{x}, \underline{s}|\Phi_K) = \sum_{k=1}^{K} f_k(\underline{x}, \underline{s}|c_k, \Phi_K)p(c_k) \qquad (7)$$

If we assume that the Φ_K are known (fixed to tentative values) we can calculate the probability that a particular observation $(\underline{x}, \underline{s})$ was generated by a specific component as:

$$p(c_k|\underline{x}, \underline{s}) = \frac{f_k(\underline{x}, \underline{s}|c_k, \Phi_K)p(c_k)}{\sum_{j=1}^{K} f_j(\underline{x}, \underline{s}|c_j, \Phi_K)p(c_j)} \qquad (8)$$

These "membership probabilities" are then used in the M-step of the EM algorithm find new parameters Φ_K^{new}, such that new likelihood is at least as great as the original likelihood, ensuring convergence (under fairly general conditions) to at least a local maximum of the likelihood function.

It is interesting to look at the exact form of the membership probabilities if we make the conditional independence assumption, namely that $f(\underline{x}, \underline{s}|c_k, \Phi_K) = f(\underline{x}|c_k, \Phi_K)p_k(\underline{s}|c_k, \Phi_K) = f_k(\underline{x})p_k(\underline{x})$ for short. The membership probabilities are then

$$p(c_k|\underline{x}, \underline{s}) = \frac{f_k(\underline{x})p_k(\underline{s})p(c_k)}{\sum_{j=1}^{K} f_j(\underline{x})p_j(\underline{s})p(c_j)} \qquad (9)$$

Here we can see the relative roles of the static and dynamic information in terms of clustering and how this information is implicitly combined and weighted. Consider what happens when, for example, $p_k(\underline{s}) = 1/K, 1 \leq k \leq K$, i.e., a sequence $\underline{s}$ is equally likely to belong to any of the K components. In this case, the $p_k(\underline{s})$ terms drop out of the expression above and we are left with the standard weights for multivariate mixture modeling. Thus, if the dynamic sequence information does not provide any discriminative power between clusters, the membership weights are solely a function of the static multivariate information (and vice-versa). More generally the membership information provided by each of the dynamic and static models are combined by Equation 9 in such a way that they are weighted relative to their discriminative power relative to the K clusters.

Note in particular that although the static and dynamic models are assumed conditionally independent, the models are implicitly coupled during learning by Equation 9. Thus, the parameters of the joint model can in principle be quite different to the model which would be obtained by clustering each separately and then combining the models.

5 Experimental Results on Simulated Data

We present here a simple experiment on "toy" simulated data to demonstrate the approach. We generated data from a two-component mixture of 2-dimensional Gaussians coupled to a first-order Markov chain on a binary alphabet. The Gaussians each had identity covariance matrices and had means at (0,0) and (2,2). The Markov transition matrix for the first cluster had self transition probabilities of 0.45 for each state, with a 0.45 probability of transiting to the other state, and a 0.1 probability of terminating the string. The other cluster had (for each state) a self-transition probability of 0.8, a cross-transition probability of 0.1, and a termination probability of 0.1. 20 observations from each cluster were simulated using the conditional independence model.

EM was judged to have adequately converged when the average difference in cluster membership probabilities (across observations) from one iteration to the next was less than 10^{-6}. The Gaussian mixtures were initialized using the k-means algorithm, and the Markov transition matrices were initialized randomly. We ran EM on the Gaussian data alone (Gaussian mixture model), the sequence data alone (Markov mixture model), and joint data set (the coupled mixture model), in each case assuming that the functional form of the true model structure is known.

Table 1 summarizes the results from the 3 different EM runs. There is one observation per row, sorted by true cluster label `C`. The columns `pM, pG, pJ` indicate the posterior probabilities for cluster 1 for the Markov, Gaussian, and joint clusterings, respectively. `x1, x2` are the static multivariate 2d data, and `s` is the dynamic sequential data.

Looking at the rightmost 3 columns provides some idea of the nature of the clustering problem (imagine that the rows are randomized rather than ordered). One can clearly see the separation between the two Gaussian clusters and one can also clearly see (in the longer sequences) some distinction between the short runs of the same symbol in cluster 1 data and the longer runs in data from cluster 2.

Clearly the most "noisy" information comes from clus-

Table 1: Table of posterior cluster probabilities under different models (`pM`: Markov, `pG`: Gaussian, and `pJ`: Joint) for a subset of the training data samples. `(x1, x2)` are the 2-dimensional multivariate measurements and `s` is the associated sequence data.

#	C	pM	pG	pJ	x1	x2	s
1	1	0.71	1.00	0.99	-0.98	1.08	babbaabbbbbbb
2	1	0.97	1.00	0.99	-0.69	2.37	baaaba
3	1	0.45	0.00	0.51	1.34	0.23	a
4	1	0.78	1.00	1.00	-0.91	-0.27	baa
5	1	0.48	1.00	0.94	-0.41	0.70	b
6	1	0.89	1.00	1.00	-0.51	-0.49	ba
7	1	0.71	0.00	0.10	1.62	1.86	bbba
8	1	1.00	0.99	1.00	0.08	1.11	aababab
9	1	0.30	1.00	1.00	-1.00	-1.23	aabb
10	1	1.00	1.00	1.00	-1.12	-0.67	abababbabbaababababaaabaabbbabbababaabbbaaa
.....							
21	2	0.00	0.00	0.00	2.84	1.35	bbbbbaaaaaaaabbbaaaaa
22	2	0.02	0.00	0.00	1.28	0.92	bbbbbbbb
23	2	0.45	0.00	0.09	1.28	1.95	a
24	2	0.02	0.00	0.00	1.80	2.38	bbbbbbbb
25	2	0.48	0.00	0.02	1.98	1.67	b
26	2	0.12	0.00	0.00	2.28	1.50	aabbbb
27	2	0.01	0.00	0.00	3.06	1.96	aaaaaaa
28	2	0.02	0.00	0.00	2.62	1.83	aaaaaa
29	2	0.00	0.96	0.01	0.25	1.04	aaaaaaaaa
30	2	0.05	0.00	0.00	2.70	3.29	aaaabbb
.....							

tering based on the sequence data alone (column `pM`). This is partly due to the fact that the cluster identities of the shorter sequences (e.g., numbers 3, 5, 6, 23, 25) will be completely ambiguous even given the true parameters. The Gaussian and joint clusterings are better able to separate the clusters (as evidenced by columns `pG` and `pJ`). But the joint clustering is the more accurate of the two virtue of the fact that it can leverage the extra information present in the sequential data. For example, in terms of classification of the data, the joint method makes 2 classification errors compared to the 7 made by the Gaussian clustering.

Of course it is not surprising that the model with the more parameters (the joint model) should fit better on the data used for parameter estimation. However, the joint model also models the true structure better than either single model. The average absolute distance (in units of standard deviation) between the estimated mean parameters from the joint clustering was 0.128, compared to 0.255 from clustering on the multivariate data alone. The average absolute distance between the estimated Markov transition probabilities from the joint clustering was 0.038, compared to 0.044 from clustering on the sequential data alone. Thus, the joint clustering is not just combining two separate static and dynamic models for prediction, but learns a more accurate model by coupling the two sources of information during the EM learning iterations. On other performance metrics, such as the log-probability score on out-of-sample data, the joint model also outperformed the single models.

The main point of this experiment is not to show that the coupled model is better. After all, since it matches the true structure of the data generating mechanism we know it will be better, at least once given enough training data. Instead the point is simply to demonstrate that coupled clustering of static and dynamic data is quite feasible.

6 Related Work

For modeling of single sequences, there has been a variety of work on modeling heterogeneity within a sequence (rather than across sequences). For example, the "mixture of experts" model, applied to sequences, provides a very general and flexible framework for identifying multiple regimes in sequential data (e.g., Jordan and Jacobs (1992), Weigend et al (1995), Cacciatore and Nowlan (1995), Zeevi et al (1997)).

The focus of all of this work, however, is on improving predictive accuracy, rather than developing inter-

pretable clustering models per se (although clearly these models could be used for the purposes of clustering). Indeed, if one can write down a generative mixture model for whatever data structure one has (be it multivariate, sequential, spatial, longitudinal, functional, and so forth), then the general framework for model-based clustering follows very naturally. In this context, the contribution of this paper should be viewed as noting that the earlier model-based clustering work of the likes of Banfield and Raftery (1993) and Cheeseman and Stutz (1996) applies to a much broader framework than simply for multivariate data.

7 Conclusion

We looked at the problem of modeling cluster structure across a set of individuals, based on measurements of static characteristics (multivariate data) coupled to dynamic behavior (sequential data). Probabilistic clustering (based on finite mixture models) provides a principled and coherent methodology for this problem. In particular, issues such as how to "weight" the relative contributions of the multivariate and sequential data measurements are handled in a natural and coherent manner. Obvious topics for further lines of investigation include determining the practicality of this approach on real-world data sets as well as more systematic investigation of model selection issues in this general context.

References

Banfield, J. D., and A. E. Raftery, 1983, Model-based Gaussian and non-Gaussian clustering, *Biometrics*, 49, 803–821.

Cacciatore T., and S. Nowlan, 1994, 'Mixtures of controllers for switching processes,' in *Advances in Neural Information Processing 6*, 1994.

Cheeseman, P. and Stutz. J., 1996, 'Bayesian classification (AutoClass): theory and results,' in *Advances in Knowledge Discovery and Data Mining*, U. M. Fayyad, G. Piatetsky-Shapiro, P. Smyth, R. Uthurusamy (eds.), Cambridge, MA: AAAI/MIT Press, pp. 153–180.

Celeux, G., and Govaert, G., 1995, 'Gaussian parsimonious clustering models,' *Pattern Recognition*, 28(5), 781–793.

Hughes, J. P., 1993, *A Class of Stochastic Models for Relating Synoptic Atmospheric Patterns to Local Hydrologic Phenomena*, Ph.D. dissertation, Dept. of Statistics, University of Washington, Seattle, WA.

Jordan, M. I., and R. A. Jacobs, 1992, 'Hierarchical mixtures of experts and the EM algorithm,' *Neural Computation*, 6, 181-214.

Krogh, A. et al., 1994, 'Hidden Markov models in computational biology: applications to protein modeling,' *J. Mol. Bio.*, 235:1501–1531.

McLachlan, G. J. and K. E. Basford, 1988, *Mixture Models: Inference and Applications to Clustering*, New York: Marcel Dekker.

MacDonald I., and W. Zucchini, *Hidden Markov Models and Other Models for Discrete-Valued Time Series*, Chapman and Hall, 1997.

Rabiner, L. R., C. H. Lee, B. H. Juang, and J. G. Wilpon, 1989, 'HMM clustering for connected word recognition,' *Proc. Int. Conf. Ac. Speech. Sig. Proc*, IEEE Press, 405–408.

Smyth, P., 1997, 'Clustering sequences using hidden Markov models,' in *Advances in Neural Information Processing 9*, M. C. Mozer, M. I. Jordan and T. Petsche (eds.), Cambridge, MA: MIT Press, 648–654.

Smyth, P., D. Heckerman, M. Jordan, 1997, 'Probabilistic independence networks for hidden Markov probability models,' *Neural Computation*, 9(2), 227–269.

Smyth, P., 'Probabilistic model-based clustering of sets of sequences,' Technical Report TR-ICS-98-38, Information and Computer Science, UC Irvine, 1988.

Symons, M., 1981, 'Clustering criteria and multivariate normal mixtures,' *Biometrics*, 37, 35–43.

Thiesson, B., Meek, C., Chickering, D. M., Heckerman, D., 'Learning mixtures of Bayesian networks,' 1997, *Microsoft Research Technical Report, MSR-97-TR-30.*

Weigend, A. S., M. Mangeas, and A. N. Srivastava, 1995, 'Nonlinear gated experts for time series: discovering regimes and avoiding overfitting,' *Int. J. Neural Sys.*, 6(4), 373–399.

Zeevi, A. J., Meir, R., Adler, R., 1997, 'Time series prediction using mixtures of experts,' in *Advances in Neural Information Processing 9*, MIT Press.