EVOLUTIONARY SPECTRAL CLUSTERING WITH ADAPTIVE FORGETTING FACTOR

Kevin S. Xu [*], *Mark Kliger* [†], *and Alfred O. Hero III* [*]

[*] University of Michigan, Ann Arbor, MI 48109 USA
[†] Medasense Biometrics Ltd., PO Box 633, Ofakim, 87516 Israel
[*] {xukevin,hero}@umich.edu, [†] mark@medasense.com

ABSTRACT

Many practical applications of clustering involve data collected over time. In these applications, evolutionary clustering can be applied to the data to track changes in clusters with time. In this paper, we consider an evolutionary version of spectral clustering that applies a forgetting factor to past affinities between data points and aggregates them with current affinities. We propose to use an adaptive forgetting factor and provide a method to automatically choose this forgetting factor at each time step. We evaluate the performance of the proposed method through experiments on synthetic and real data and find that, with an adaptive forgetting factor, we are able to obtain improved clustering performance compared to a fixed forgetting factor.

Index Terms— Clustering methods, temporal smoothing.

1. INTRODUCTION

In many practical applications, we wish to cluster data that have been collected at regular time intervals and obtain a clustering result at each time step. This situation arises in segmentation of a sequence of images of a dynamic scene, identifying changes in the community structure of a social network, and many other applications in finance, biomedical signal processing and bioinformatics. A naïve approach to this problem is to perform clustering at each time step using only the most recent data. This method is often referred to as incremental clustering and has two main disadvantages: it is extremely sensitive to noise, and it also produces clustering results that are unstable and inconsistent with clustering results from previous time steps.

Typically in these types of applications, the statistical properties of the data to be clustered evolve over time. The goal of evolutionary clustering is to separate this evolution from short-term variation in the data due to noisy samples. Ideally, the clustering results should be smooth over time yet still capture any drifts in the statistical properties of the data. In order to produce clustering results that are smooth over time, past data should be used in some manner.

Frameworks for evolutionary clustering have been proposed in previous studies [1, 2, 3]. We adopt an evolutionary extension for spectral clustering proposed in [2] that takes a convex combination of current and past affinities between data points as the input to the traditional spectral clustering algorithm. The weights in the convex combination act as a forgetting factor applied to past affinities. To the best of our knowledge, no methods have yet been proposed on how to choose the forgetting factor. A forgetting factor that is too large will lead to a clustering algorithm that is slow to detect evolutions in the data, while a forgetting factor that is too small will lead to unstable clustering results. Therefore, a good choice of forgetting factor is essential to obtain good clustering results.

In this paper, we propose to use an adaptive (time-varying) forgetting factor in the evolutionary spectral clustering procedure. We develop a method for estimating the optimal forgetting factor at each time step using a shrinkage approach. Our method is inspired by the Ledoit-Wolf shrinkage estimator for covariance matrices [4].

We evaluate the performance of our adaptive forgetting factor on synthetic and real data and find that it outperforms fixed forgetting factors as well as incremental clustering. In particular, with a fixed forgetting factor, there is a trade-off between smoothness of clustering results over time and lag in detecting changes in clusters. By allowing the forgetting factor to vary with time, we can achieve both objectives to obtain improved clustering performance.

2. BACKGROUND

2.1. Spectral clustering

Spectral clustering is a popular modern clustering technique inspired by spectral graph theory and often performs better than traditional clustering methods such as K-means. We provide a brief overview of spectral clustering and refer interested readers to [5] for a more detailed account.

The first step in spectral clustering is to create a similarity graph with vertices corresponding to the data points to be clustered and edges corresponding to the affinities between data points. This graph can be represented by an adjacency matrix W, also commonly referred to as an affinity matrix, where w_{ij} denotes the edge weight or affinity between vertices i and j. We represent the data by an $n \times p$ matrix X, with rows corresponding to data points and columns to features. The affinities w_{ij} are given by a positive semi-definite similarity function $s(\mathbf{x_i}, \mathbf{x_j})$, where $\mathbf{x_i}$ denotes the ith row of X. Two common choices for the similarity function are the dot product $s(\mathbf{x_i}, \mathbf{x_j}) = \mathbf{x_i}\mathbf{x_j}^T$ and the Gaussian similarity function $s(\mathbf{x_i}, \mathbf{x_j}) = \exp\left(-\|\mathbf{x_i} - \mathbf{x_j}\|_2^2/(2\sigma^2)\right)$ where σ is a positive scaling parameter. Define the degree matrix $D = \mathrm{diag}(W1_n)$ where $\mathrm{diag}(\cdot)$ creates a diagonal matrix from its vector argument, and 1_n is a vector of n ones. Spectral clustering aims to solve the following optimization problem over Y:

$$\text{maximize} \quad \mathrm{knassoc}(Y) = \frac{1}{k}\sum_{i=1}^{k} \frac{\mathbf{y_i}^T W \mathbf{y_i}}{\mathbf{y_i}^T D \mathbf{y_i}} \tag{1}$$

$$\text{subject to} \quad Y \in \{0,1\}^{n\times k} \tag{2}$$

$$Y1_k = 1_n, \tag{3}$$

where $\mathbf{y_i}$ denotes the ith column of Y, and k is the number of clusters to divide the data into.

This work was partially supported by NSF grant CCF 0830490 and ONR grant N00014-08-1-1065.

In short, the problem is one of finding an optimal graph partition which maximizes the ratio of the sum of edge weights between vertices in the same cluster C_i to the sum of edge weights between any two vertices where one vertex is in C_i. This is an NP-hard problem as noted in [6]. The spectral clustering solution involves first relaxing constraint (2), solving the resulting continuous optimization problem, and finally, discretizing the solution to obtain a near global-optimal graph partition [5]. We represent the partition by an $n \times k$ partition matrix Y where $y_{ij} = 1$ if vertex i is in cluster j and $y_{ij} = 0$ otherwise.

2.2. Related work

Evolutionary clustering is an area that has gained interest recently as more and more dynamic data sources become available. Sun et al. [3] proposed a method for clustering time-evolving graphs; however, their work was limited to unweighted graphs. Chakrabarti et al. [1] proposed evolutionary extensions of K-means and agglomerative hierarchical clustering. Chi et al. [2] proposed two evolutionary frameworks for spectral clustering, one of which we adopt in this paper. [1, 2] both make use of a fixed smoothing parameter to control the amount of weight to be applied to past data. However, a major shortcoming in both works is that the question of how to choose the smoothing parameter is not addressed. In this paper, we provide a method to estimate the optimal smoothing parameter, namely the forgetting factor, at each time step.

3. METHODOLOGY

We begin by stating our assumptions. We assume that the data are realizations from a mixture of random processes; that is, at each time step, the current data are realizations from a mixture of probability distributions. Furthermore, we assume that the random processes which form this mixture are approximately piecewise stationary and that the data are measured over short enough time intervals that the processes are approximately stationary over these intervals.

3.1. Evolutionary clustering framework

Let X^t denote the data matrix with rows $\mathbf{x_i^t}$ corresponding to the data points to be clustered. The superscript t denotes the time step. The goal of our approach is to accurately estimate the true affinity matrix at each time t. We define the true affinity matrix Ψ^t at time t to be the expected affinity matrix $\mathrm{E}\left[W^t\right]$, where the entries of W^t are given by $w_{ij}^t = s\left(\mathbf{x_i^t}, \mathbf{x_j^t}\right)$.

In incremental spectral clustering, W^t itself is used as an estimate for Ψ^t. The main disadvantage of this approach is that it suffers from high variance because the estimate uses only the most recent affinities. As a consequence, the obtained clustering results are unstable and inconsistent with clustering results from previous time steps.

We define the smoothed affinity matrix at time t to be

$$\bar{W}^t = \alpha \bar{W}^{t-1} + (1-\alpha) W^t, \tag{4}$$

for $t \geq 1$ and $\bar{W}^0 = W^0$. The forgetting factor α controls the amount of smoothing to be applied. $\bar{W}^t$ is another natural candidate for estimating Ψ^t.

Chi et al. [2] proposed to perform evolutionary spectral clustering by taking (4) as the input to the traditional spectral clustering algorithm. However, the question of how to select α was not considered. In a truly unsupervised scenario, we do not have any ground truth to compare to, so we cannot simply perform cross-validation to choose the optimal α. We propose a method to estimate the optimal α from the data itself.

The smoothed affinity matrix $\bar{W}^t$ incorporates past affinities so it has lower variance than W^t, but it may be biased since the past affinities may not be representative of the current ones. Thus the problem of estimating the optimal forgetting factor α may be considered as a bias-variance trade-off problem.

A similar bias-variance trade-off has been investigated in the problem of shrinkage estimation of covariance matrices [4, 7, 8], where an improved estimate of the covariance matrix is taken to be $\hat{\Sigma} = \alpha T + (1-\alpha) S$, a convex combination of a suitably chosen target matrix T and the sample covariance matrix S. Notice that this has the same form as the smoothed affinity matrix given by (4) where the smoothed affinity matrix at the previous time step $\bar{W}^{t-1}$ plays the role of the shrinkage target T and the current affinity matrix W^t plays the role of the sample covariance matrix S. We propose to estimate the optimal choice of α using an approach similar to the Ledoit-Wolf method of choosing α for shrinkage estimation of covariance matrices [4]. We describe our approach in the following section. α is re-estimated at each time step, and in this manner, we achieve an adaptive forgetting factor.

Similar to [4, 7, 8], we choose to optimize the squared Frobenius norm of the difference between the true affinity matrix and the estimated affinity matrix. That is, we take the loss function to be

$$L(\alpha) = \|\alpha \bar{W}^{t-1} + (1-\alpha) W^t - \Psi^t\|_F^2. \tag{5}$$

The risk function is then simply the expected loss. The risk function is differentiable and can be easily optimized.

3.2. Estimation of the optimal forgetting factor

First note that the risk function can be expressed as

$$\begin{aligned} R(\alpha) &= \mathrm{E}\left[L(\alpha)\right] && (6)\\ &= \sum_{i=1}^{n}\sum_{j=1}^{n} \mathrm{E}\left[\left(\alpha \bar{w}_{ij}^{t-1} + (1-\alpha) w_{ij}^t - \psi_{ij}^t\right)^2\right] && (7)\\ &= \sum_{i=1}^{n}\sum_{j=1}^{n} \Big\{ \mathrm{var}\left(\alpha \bar{w}_{ij}^{t-1} + (1-\alpha) w_{ij}^t - \psi_{ij}^t\right) + \\ &\qquad \mathrm{E}\left[\left(\alpha \bar{w}_{ij}^{t-1} + (1-\alpha) w_{ij}^t - \psi_{ij}^t\right)\right]^2 \Big\}. && (8) \end{aligned}$$

We treat $\bar{W}^{t-1}$ as a deterministic shrinkage target, so it has zero variance. Since $\mathrm{E}\left[w_{ij}^t\right] = \psi_{ij}^t$, (8) can be rewritten as

$$R(\alpha) = \sum_{i=1}^{n}\sum_{j=1}^{n} \left\{(1-\alpha)^2 \,\mathrm{var}(w_{ij}^t) + \alpha^2 (\bar{w}_{ij}^{t-1} - \psi_{ij}^t)^2\right\}. \tag{9}$$

From (9), the first derivative is easily seen to be

$$R'(\alpha) = 2\sum_{i=1}^{n}\sum_{j=1}^{n} \left\{(\alpha-1)\,\mathrm{var}(w_{ij}^t) + \alpha (\bar{w}_{ij}^{t-1} - \psi_{ij}^t)^2\right\}. \tag{10}$$

To determine the optimal forgetting factor α^* we set $R'(\alpha) = 0$. Rearranging to isolate α, we obtain

$$\alpha^* = \frac{\displaystyle\sum_{i=1}^{n}\sum_{j=1}^{n} \mathrm{var}(w_{ij}^t)}{\displaystyle\sum_{i=1}^{n}\sum_{j=1}^{n} \left\{(\bar{w}_{ij}^{t-1} - \psi_{ij}^t)^2 + \mathrm{var}(w_{ij}^t)\right\}}. \tag{11}$$

We confirm that α^* minimizes risk because $R''(\alpha) \geq 0$ for all α.

Notice that α^* is not implementable because it requires knowledge of the expected affinity matrix Ψ^t, which is what we are trying to estimate, as well as the variances of the entries of W^t. It was suggested in [7] to replace the unknowns with their sample equivalents. In our application, however, we cannot simply compute, say a sample mean, by summing over all of the samples because they are realizations from a mixture, and hence, not identically distributed. Instead we should sum over all of the samples that belong to a particular component in the mixture, but we don't know which samples belong to which components; in fact, this is what we are trying to discover by clustering!

To work around this problem, we estimate the component each sample belongs to (the component memberships) along with α^* in an iterative fashion. First we fix the component memberships by taking them to be the cluster memberships at the previous time step. Then we can sum over each cluster to estimate the entries of Ψ^t and the variances of the entries of W^t as detailed below, and substitute them into (11) to obtain an estimate $\hat{\alpha}^*$ of α^*. We then fix $\hat{\alpha}^*$ to obtain an updated estimate of the component memberships by substituting it into (4) and performing clustering on $\bar{W}^t$. This process is continued until $\hat{\alpha}^*$ converges to some value, which can be substituted into (11) to obtain the final smoothed affinity matrix $\bar{W}^t$. Unfortunately, $\hat{\alpha}^*$ does not always converge since cluster memberships are discrete, so the iteration should be stopped at some point if $\hat{\alpha}^*$ has not converged.

To estimate the entries of $\Psi^t = \mathrm{E}\left[W^t\right]$, we proceed as follows. For two distinct samples i and j both in cluster C_1, we can estimate ψ_{ij}^t using the sample mean

$$\hat{\mathrm{E}}[w_{ij}^t] = \frac{1}{|C_1|\left(|C_1|-1\right)} \sum_{k \in C_1} \sum_{\substack{l \in C_1 \\ l \neq k}} w_{kl}^t \tag{12}$$

where $|C_1|$ denotes the number of samples in cluster C_1. Similarly, we estimate ψ_{ii}^t by

$$\hat{\mathrm{E}}[w_{ii}^t] = \frac{1}{|C_1|} \sum_{k \in C_1} w_{kk}^t. \tag{13}$$

For distinct samples $i \in C_1$ and $j \in C_2$ with $C_1 \neq C_2$, we estimate ψ_{ij}^t by

$$\hat{\mathrm{E}}[w_{ij}^t] = \frac{1}{|C_1||C_2|} \sum_{k \in C_1} \sum_{l \in C_2} w_{kl}^t. \tag{14}$$

The variances of the entries of W^t can be estimated in a similar manner by taking the sample variances over the clusters.

4. EXPERIMENTS

4.1. Synthetic data

We begin by testing our proposed method on synthetic data. The objective of this experiment is to test the effectiveness of the adaptive forgetting factor when a cluster moves close enough to another cluster so that they have significant overlap. We also test the ability of our method to adapt to a change in cluster membership.

The setup for this experiment is shown in Fig. 1. We generate 40 samples from a mixture of two 2-D Gaussians, the first with mean $[20, 10]$ and the second with mean $[10, 20]$. Both components have the same covariance matrix, with variances equal to 2 and covariances equal to 1. The mixture proportion (the proportion of samples drawn from the first component) is initially chosen to be $1/2$, so that an equal number of samples is drawn from each component.

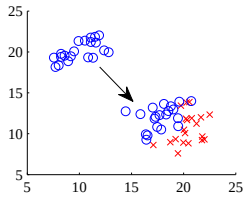
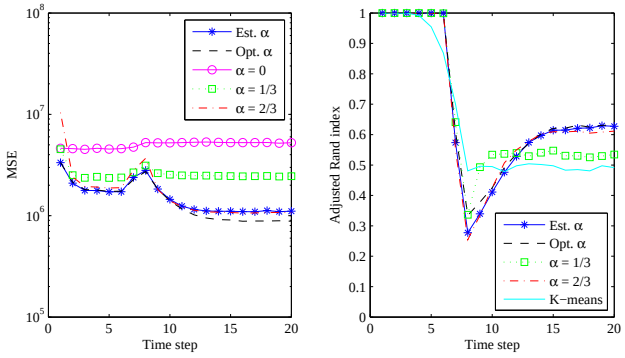


Fig. 1: Setup of experiment. One component is slowly moved towards the other until they overlap slightly. The mixture proportion is then altered to simulate a change in cluster membership.



(a) MSE (b) Adjusted Rand index

Fig. 2: Performance comparison for varying α.

From time steps 1 to 8, we move the mean of the second component towards the first one by $[d_1, d_2]$, where d_1 and d_2 are independent $N(1, 1)$ and $N(-1, 1)$, respectively. At time steps 7 and 8, we switch the mixture proportion to $3/8$ and $1/4$, respectively, to simulate points changing cluster. From time step 9 onwards, the mixture components are kept stationary. We use the dot product as the similarity function in this experiment.

We ran this experiment 500 times. In Fig. 2a we compare the mean squared error (MSE) between the true affinity matrix and the estimated affinity matrices for five different choices of α, including $\alpha = 0$, which corresponds to incremental spectral clustering. The error is taken to be the Frobenius norm of the difference between the true and estimated affinity matrices. It can be seen that the choice of α affects MSE significantly and that both the adaptive $\hat{\alpha}^*$ and fixed $\alpha = 2/3$ come close to achieving the optimal MSE. In Fig. 2b we compare the adjusted Rand index [9] between the clustering results and true component memberships for four different choices of α and an incremental version of the well-known K-means algorithm. For clarity, $\alpha = 0$ has been left out of the figure, but it performs roughly the same as incremental K-means. Again, $\hat{\alpha}^*$ and $\alpha = 2/3$ perform well. Notice that around time steps 8 and 9 when the true component memberships change, $\alpha = 1/3$ and incremental K-means temporarily perform better than the other choices of α. This represents a lag in detecting the change in mixture proportion and is a consequence of the temporal smoothing. However, after only three time steps, $\hat{\alpha}^*$ and $\alpha = 2/3$ catch up to and outperform $\alpha = 1/3$ and incremental K-means, so the lag is minimal. From this experiment, it can be seen that overall clustering performance is quite sensitive to the choice of α, so a method for identifying a good choice such as the one proposed in this paper is crucial for good performance.

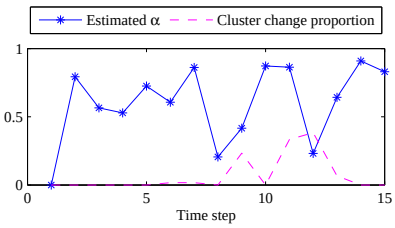


Fig. 3: $\hat{\alpha}^*$ and cluster change proportion for the MIT reality data set. Each time step corresponds to a two-week interval.

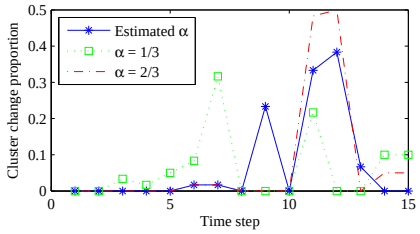


Fig. 4: Comparison of cluster change proportion for varying α.

4.2. Real data

We also test our proposed method on a publicly available data set: the MIT reality mining data set [10]. The data was collected by recording cell phone activity of one hundred students and faculty at MIT for over a year. In particular, we make use of the device span data, which recorded the times at which each cell phone was in proximity to another Bluetooth device. Our study focuses on the time period from September 9, 2004 to March 25, 2005 because the volume of logged activities was very low prior to the beginning of the school year and near the end of the study in May 2005. The data was split into 2-week intervals, resulting in 15 time steps of data. We remove users who were not in proximity to any others. Again, we use the dot product as the similarity function in this experiment.

In Fig. 3 we show the estimated forgetting factor $\hat{\alpha}^*$ at each time step as well as the cluster change proportion (proportion of users who changed cluster). As with most real data sets, we do not have true cluster memberships to compare the clustering results to, so we try to correlate cluster changes to real events. Notice that there are two sudden decreases in $\hat{\alpha}^*$. These correspond to the two-week intervals beginning on December 16, 2004 and February 10, 2005, respectively. From the MIT academic calendar [11], we see that these correlate with the end of the fall term and the beginning of the spring term. Around these time steps, the cluster change ratio increases significantly, indicating that social patterns changed at these times, which makes sense because they mark the start and end of the winter holidays.

In Fig. 4 we plot the cluster change proportion for both the adaptive $\hat{\alpha}^*$ and two fixed values of α. The adaptive $\hat{\alpha}^*$ provides both excellent smoothing during the school terms and is also able to detect both change periods, albeit with a slight lag. Fixing $\alpha = 2/3$ results in discovering only a single period of cluster change, which is a consequence of over-smoothing. On the other hand, fixing $\alpha = 1/3$ results in discovering both change periods but with a higher cluster change proportion during the school terms when the clusters should be relatively stable. This marks a clear drawback of choosing a fixed α, namely that one must trade off smoothing ability over periods where there is little to no change in the true cluster memberships with change detection ability when significant changes in the cluster memberships occur. With an adaptive forgetting factor, there is no such limitation. Hence our proposed method should be able to outperform fixed forgetting factors.

5. CONCLUSIONS

In this paper, we proposed a method for automatically selecting the forgetting factor applied to past affinities in evolutionary spectral clustering. Our proposed method produced an adaptive (time-varying) forgetting factor. Experiments on synthetic and real data indicate that our proposed method outperforms fixed forgetting factors and incremental clustering. By using an adaptive forgetting factor, we were able to obtain temporally smooth clustering results as well as detect sudden changes with minimal lag, which cannot be simultaneously achieved with a fixed forgetting factor.

Future research directions include a convergence analysis of our iterative method for estimating the optimal forgetting factor and investigating methods for dealing with new vertices being introduced into the similarity graph at some time step as well as existing vertices leaving the graph.

6. REFERENCES

[1] D. Chakrabarti, R. Kumar, and A. Tomkins, "Evolutionary clustering," in *Proc. ACM SIGKDD Conf. Knowledge Discovery and Data Mining*, 2006.

[2] Y. Chi, X. Song, D. Zhou, K. Hino, and B. L. Tseng, "Evolutionary spectral clustering by incorporating temporal smoothness," in *Proc. ACM SIGKDD Conf. Knowledge Discovery and Data Mining*, 2007.

[3] J. Sun, S. Papadimitriou, P. S. Yu, and C. Faloutsos, "Graphscope: Parameter-free mining of large time-evolving graphs," in *Proc. ACM SIGKDD Conf. Knowledge Discovery and Data Mining*, 2007.

[4] O. Ledoit and M. Wolf, "Improved estimation of the covariance matrix of stock returns with an application to portfolio selection," *Journal of Empirical Finance*, vol. 10, no. 5, pp. 603–621, 2003.

[5] S. Yu and J. Shi, "Multiclass spectral clustering," in *Proc. IEEE Int. Conf. Computer Vision*, 2003.

[6] J. Shi and J. Malik, "Normalized cuts and image segmentation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 22, pp. 888–905, 2000.

[7] J. Schäfer and K. Strimmer, "A shrinkage approach to large-scale covariance matrix estimation and implications for functional genomics," *Statistical Applications in Genetics and Molecular Biology*, vol. 4, no. 1, article 32, 2005.

[8] Y. Chen, A. Wiesel, and A. O. Hero III, "Shrinkage estimation of high-dimensional covariance matrices," in *Proc. IEEE Int. Conf. Acoustics, Speech, and Signal Processing*, 2009.

[9] L. Hubert and P. Arabie, "Comparing partitions," *J. Classification*, vol. 2, no. 1, pp. 193–218, 1985.

[10] N. Eagle and A. Pentland, "Reality mining: sensing complex social systems," *Personal and Ubiquitous Computing*, vol. 10, no. 4, pp. 255–268, 2006.

[11] "MIT academic calendar 2004-2005," `http://web.mit.edu/registrar/www/calendar0405.html`.