Available online at www.isas.org.in/jisas
JOURNAL OF THE INDIAN SOCIETY OF AGRICULTURAL STATISTICS 64(1) 2010 33-44

Information Based Agglomerative Segmentation in Metric Spaces

Francesca Chiaromonte[1*] and James Taylor[2]
[1]*Department of Statistics and Center for Comparative Genomics and Bioinformatics, The Pennsylvania State University, University Park, PA 16802*
[2]*Department of Biology, and Department of Mathematics and Computer Science, Emory University, Atlanta, GA 30322*

SUMMARY

In this article, we introduce an approach to agglomerate points in a metric space into spatially contiguous groups which preserve both distance and frequency structure of the data. This is achieved using a traditional distance criterion to define candidate mergers, and then selecting among these candidates as to maximize the mutual information between pre- and post-merger partitions. Our information based agglomerative segmentation is particularly effective when grouping data that does not present spatially separated clusters, and can therefore be employed for reducing data complexity in a number of scientific applications. We illustrate the procedure using a simulated data structure and an application to the analysis of multi-species genomic alignment data.

Keywords : Agglomerative clustering, Mutual information and entropy, Data complexity reduction, Genomics.

1. INTRODUCTION

Many contemporary data analysis problems require the unsupervised grouping of large collections of elements that can be viewed as points in a metric space. This is a means to reduce data complexity while preserving critical information and, roughly speaking, can be thought of as a clustering exercise. However, because of their underlying logic, traditional clustering methods often provide poor solutions. Examples where grouping is needed for data reduction can be found in many scientific areas; to motivate the novel methodology proposed in this article, and create a practical context for it, we briefly introduce here two such examples from the field of genomics.

First, consider the problem of grouping yeast genes based on their transcription levels across a cell-cycle related time course, as reported in a pioneering microarray study by Spellman *et al.* (1998). Groups of genes displaying similar transcription patterns can be interpreted as being co-regulated or functionally related during the cell cycle. Preprocessing of the microarray data included:

(i) elimination of the terminal part of the time course, and imputation of missing values
(ii) standardization of transcription profiles
(iii) selection of genes whose profiles displayed distinct periodic behavior.

This led to 679 points (genes) mapped in a 12-dimensional Euclidean space (12-point time course covering approximately two cell cycles) which, in a way that is characteristic of many if not most microarray experiments, show little if any evidence for a natural partitioning. Fig. 1A shows a 2D view of the data obtained through a projection on the first principal components plane. While the points are unevenly distributed, and some concentration regions appear, the

[*]*Corresponding author* : Francesca Chiaromonte
E-mail address : chiaro@stat.psu.edu

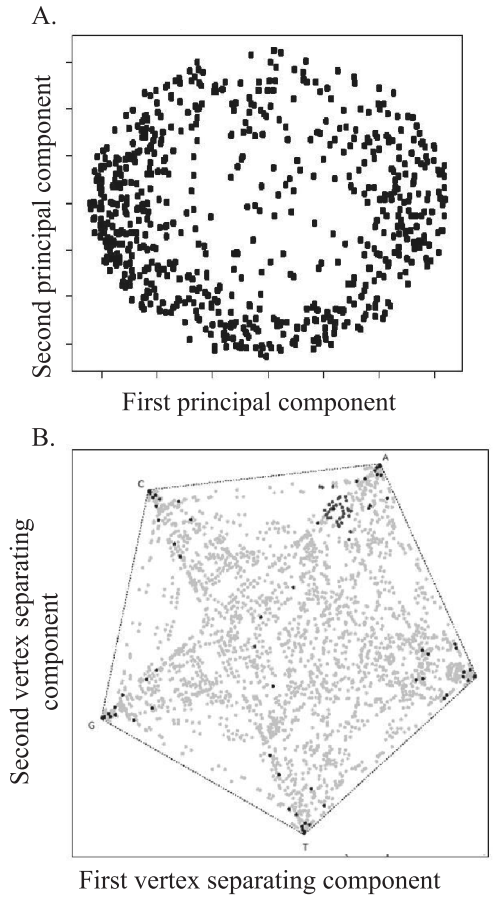

Fig. 1. **A** shows a 2D projection of yeast genes mapped into points in a 12-dimensional space, each coordinate representing transcription at a time point along the cell-cycle (data from Spellman *et al*. 1998). The projection plane is spanned by the first and second principal components, and the spherical shape is due to the normalization applied to transcription profiles. The points are unevenly distributed, and some concentration regions appear, but the data does not present isolated clusters. **B** shows a 2D projection of 7-way DNA alignment columns mapped into points in a 5-dimensional simplex, each coordinate representing the probability of an ancestral nucleotide (*A*, *C*, *G* or *T*) plus an artificial state corresponding to a gap (data from Taylor *et al*. 2006). The projection plane is the one maximizing separation among the 5 vertices of the simplex, and shades of gray represent frequencies of points after a preliminary clustering. Again we can observe concentration regions but no isolated clusters. In both examples, agglomerative clustering based on distance alone fails to capture the structure of the data.

data does not present spatially isolated clusters. A "low" cut of the dendrogram produced by a traditional agglomerative clustering algorithm (see for instance Hartigan 1975) would identify the concentration regions as clusters, but also locate an undue number of very small clusters (possibly singletons) in the less populated regions. On the other hand, a "high" cut of the dendrogram, providing a more substantial data reduction, would create a partition with poor fidelity to the structure of the data.

As a second example, consider the problem of grouping columns in nuclear DNA alignments of seven mammalian species (human, chimpanzee, macaque, mouse, rat, dog, and cow) based on their likely evolutionary history, as reported in a study of functional genomic loci by Taylor *et al.* (2006). A partition of the columns in groups capturing roughly equivalent evolutionary histories can be used as a reduced "alphabet" to encode aligned genomic regions – facilitating the identification of functional loci (more details will be given in Section 4).

A generic column has seven entries (the number of species aligned), each comprising one of the four nucleotides (*A*, *C*, *G* or *T*) or a gap (these are introduced when creating the alignments with specialized software; e.g. Blanchette *et al*. 2004). Thus, accounting for the fact that no alignment column can be entirely composed by gaps, there are a total of $(5^7-1) = 78{,}124$ columns to be grouped. Preprocessing of the alignment data included:

(i) a special handling of missing alignment entries, which in practice resulted into a much larger number of columns to be grouped (approximately 275,000 for 7 species)

(ii) associating each such column to a probability distribution for the corresponding base in an ancestral genome (this was done using an established phylogeny for the seven species, an HKY substitution model – see Hasegawa *et al*. 1985, and Felsenstein's algorithm – see Durbin *et al*. 1998, and Mayrose *et al*. 2004)

(iii) merging columns that occurred seldom or never in observed alignments with columns associated with very similar ancestral distributions, having a sizeable number of observed occurrences.

This led to 923 points (alignment columns) mapped in a 5-dimensional simplex (5 states; the 4 nucleotides and an artificial state introduced to accommodate gaps). Also here, no obvious partitioning structure exists. Fig. 1B shows a 2D projection of the data, and again we can observe concentration regions but no spatially isolated clusters. As in the yeast

cell-cycle gene transcription example (Fig. 1A), clusters obtained from a traditional agglomerative algorithm would provide an unsatisfactory representation of the data.

For both these examples, the situation can be only marginally improved by selecting different linkage functions, or using different clustering approaches based on distance alone (e.g. algorithms in the *k*-means family, or clustering through mixture models; see, among others, Hartigan 1975, McLachlan and Peel 2000, and Fraley and Raftery 2002).

In full generality, we consider the problem of partitioning n distinct points $x = \{x_i, i = 1 \ldots n\}$ in a feature space X endowed with a distance $d(\circ,\circ)$. Associated with these points is an $n \times n$ distance matrix

$$D(x) = \{d(x_i; x_{i'}), i, i' = 1 \ldots n\}$$

and a vector of n frequencies

$$f(x) = \{f(x_i), i = 1 \ldots n\}$$

In some settings each distinct point may appear only once, i.e. carry a frequency $f(x_i) = 1/n$ – this is the case in the yeast genes example. In other settings though, each distinct point may appear a different number of times, and thus carry a specific frequency, say $f(x_i) = n_i / (\Sigma_{i'=1\ldots n}\, n_{i'})$ – this is the case in the alignment columns example, because columns have already undergone some merging during data preprocessing.

Let $y = \{y_j , j = 1\ldots m\}$ indicate a partition of the points, or equivalently of the integers $\{1\ldots n\}$, in m classes, and $j[i]$ index the class to which x_i belongs. Given a linkage function, we can associate with the partition a new $m \times m$ matrix of distances between classes

$$D(y) = \{d_L(y_j; y_{j'}), j, j' = 1 \ldots m\}$$

Also, we can associate to the partition a new vector of m class frequencies

$$f(y) = \{f(y_j) = \sum_{i:j[i]=j} f(x_i), j = 1 \ldots m\}$$

Our aim is to identify a partition that captures as closely as possible both the distance and the frequency structure of the original data, in a way that will be made rigorous in the following.

We prefer to avoid the term clustering when referring to our approach, and talk instead of grouping, or *segmentation*. With this we try to convey the fact that, while we seek groups covering contiguous regions in the feature space (distance structure), our methodology does not rely on the existence of spatially separated clusters in the data.

Section 2 summarizes the theoretical underpinning of the information based agglomerative segmentation method we propose. Section 3 illustrates our method and its performance in comparison to traditional clustering approaches, using simulated data. In Section 4 we apply our method to the alignment columns example, as one step in the protocol used to compute so-called Regulatory Potential scores. Section 5 contains some concluding remarks.

2. INFORMATION BASED AGGLOMERATIVE SEGMENTATION

In traditional agglomerative clustering, at any given iteration h, a new partition $y^{[h+1]}$ is obtained merging the two classes in the current partition $y^{[h]}$ that correspond to the minimal off-diagonal entry of the *distance* matrix $D(y^{[h]})$. Ruling out for simplicity the case in which there is more than one minimal entry, at each iteration distance determines only one candidate merger, which is implemented regardless of whether the resulting $f(y^{[h+1]})$ resembles $f(y^{[h]})$ in any meaningful way.

Alternatively, we could consider all mergers as candidates regardless of distance, and select the one that maximizes the *mutual information* between $y^{[h+1]}$ and $y^{[h]}$. Since the two partitions are nested, we can easily derive a joint density as

$$f(y_j^{[h+1]}, y_i^{[h]}) = f(y_j^{[h+1]} \mid y_i^{[h]})\, f(y_i^{[h]})$$

$$= \begin{cases} 1\, f(y_i^{[h]}) = f(y_i^{[h]}) & \text{if } \; j[i] = j \\ 0\, f(y_i^{[h]}) = 0 & \text{if } \; j[i] \neq j \end{cases}$$

so the mutual information is given by

$$I(y^{[h+1]}, y^{[h]})$$

$$= \sum_{j,i} f(y_j^{[h+1]}, y_i^{[h]}) \log\left(\frac{f(y_j^{[h+1]}, y_i^{[h]})}{f(y_j^{[h+1]})\, f(y_i^{[h]})} \right)$$

$$= \sum_{j,i:j[i]=j} f(y_i^{[h]}) \log\left(\frac{1}{f(y_j^{[h+1]})}\right)$$

$$= -\sum_{j} \sum_{i:j[i]=j} f(y_i^{[h]}) \log\left(f(y_j^{[h+1]})\right)$$

$$= -\sum_{j} f(y_j^{[h+1]}) \log\left(f(y_j^{[h+1]})\right)$$

$$= H(y^{[h+1]})$$

which coincides with the *entropy* of $y^{[h+1]}$.

The merger of maximal entropy, which we can again assume to be unique for simplicity, is the one that allows $f(y^{[h+1]})$ to retain the most information about $f(y^{[h]})$. However, it may agglomerate classes that are not contiguous in space.

The core idea behind our approach is that, in many situations of practical interest, exploiting both the distance and the frequency structure of the data may lead to more satisfactory partitions. Thus, our aim is to create an agglomerative algorithm that operates somewhere in between the two extremes described above. Intuitively, the algorithm should approximate in a generic metric space the way a student in a beginner statistics course is taught to draw a histogram to summarize a univariate data set. She partitions the range of the data in m contiguous segments, each containing approximately the same number of points, and draws boxes on them with hights that guarantee approximately equal areas – such a histogram indeed corresponds to the partition of maximal entropy, and spatial contiguity of the classes is straightforward to implement on the real line.

Pursuing this logic, we relax the requirement on distance in a way that lets us consider several candidate mergers, say a set $C^{[h]}$ of pairs (i, i') that guarantee neighborship, i.e. proximity in the feature space, but are not restricted to the minimal off-diagonal entry of $D(y^{[h]})$. Among these candidates, we select the merger (pair) that results in the largest mutual information between $y^{[h+1]}$ and $y^{[h]}$, i.e. in the largest entropy $H(y^{[h+1]})$. After updating the distance matrix and frequency vector to $D(y^{[h+1]})$ and $f(y^{[h+1]})$, we proceed to the next iteration, et cetera. From now on, we will omit the iteration index h for notational simplicity.

Let m be the current number of classes, and $N(i, k)$ the set of k nearest neighbors of class i (excluding i itself), as obtained from $D(y)$. We define the set of candidate mergers $C(k, t)$ as:

$$C(k, t) = \{(i, i') : i' \in N(i, k) \text{ and } i \in N(i', t)\}$$

Thus, the set contains all pairs (i, i') such that $D_{i,i'}(y)$ is among the k smallest off-diagonal entries of row i, and the t smallest off-diagonal entries of column i'. This definition involves two integer parameters, k and t, to be selected in $\{0, 1\ldots (m-1)\}$. These determine the size of the candidate merger set, and thus the relative role of distance and entropy in the algorithm (see below). We chose, by convention, to restrict attention to $k \leq t$; this induces no loss of generality (and we could equivalently consider $t \leq k$) because the symmetry of $D(y)$ implies that $C(k, t) = C(t, k)$.

If we set by convention $C(0,0)$ to contain only the pair corresponding to the minimal off-diagonal entry of $D(y)$, taking $k = t = 0$ corresponds to traditional agglomerative clustering, since we maximize entropy on just this one minimum distance merger. $C(1, 1)$ comprises all pairs whose entries are off-diagonal minima by row and column simultaneously (i is the closest neighbor of i', and vice versa). At the opposite end of the spectrum $C(m-1, m-1)$ contains all $m(m-1)/2$ pairs. So taking $k = t = m-1$ we maximize entropy on all possible mergers, with distance playing no role.

It is easy to see that $C(0, 0)$ is contained in $C(k, t)$ for any k and t, and thus that all candidate sets are non-empty. It is also immediate from the definition that

$$C(k, t) \subseteq C(k', t'), \ \forall k' \geq k, t' \geq t$$

If we take $k = t$, the candidate set $C(k, k)$ is symmetric, in the sense that if (i, i') belongs to the set so does (i', i). This results in a *symmetric agglomeration*. Traditional agglomerative clustering based on distance alone, i.e. $C(0, 0)$, agglomeration that considers only mutually closest neighbors, i.e. $C(1, 1)$, and agglomeration based on entropy alone, i.e. $C(m-1, m-1)$, are all instances. For symmetric agglomerations, as k increases from 0 to $m-1$, the role of entropy increases and that of distance decreases. In fact, the containment $C(k, k) \subseteq C(k', k')$, $k' \geq k$ means that the candidate sets become larger, corresponding to a decreasing strength of the distance constraint.

An alternative way to decrease the role of distance is to break symmetry, exploiting the containment $C(k, k) \subseteq C(k, t)$, $t \geq k$. For a given k, $C(k, t)$ lets us consider for merger, among the k-neighbors of each $i = 1 \ldots m$, any class for which i is a t-neighbor – so neighborship can be less than mutual. This defines an *asymmetric agglomeration.* The weakest distance constrain in such an agglomeration is obtained setting $t = m - 1$; $C(k, m-1)$ contains all k-neighbors of each $i = 1 \ldots m$, regardless of whether neighborship is mutual to any degree. Relatedly, asymmetry has an interesting geometric interpretation; suppose the current partition comprises a number of high frequency classes very close to one another, together with some isolated points or low frequency classes at the periphery of the former. If we require mutual neighborship, the set of candidate mergers will likely contain only pairs of high frequency classes. Any such merger will result in a poor entropy value, but those are the only mergers under consideration. On the other hand, if we allow neighborship not to be mutual, mergers between low frequency classes and high frequency classes closest to them are likely to make it into the candidate set, and thus to be selected, as they result in better entropy values. Asymmetric agglomerations may therefore help achieve satisfactory segmentations by attributing points in low density areas to classes in nearby high density areas, while preventing the merger of classes in high density areas to one another.

Another interesting interpretation of asymmetry is that, in a way, it allows us to weaken the distance constraint more effectively, regardless of the specific structure of the data. Indicating by $\#(\circ)$ the number of elements is a set, we have that

$$\#(C(k, k)) + \#(C(k, m-1)) = km,\ k = 1 \ldots m-1$$

This equality can be easily proved as follows: scanning the rows of $D(y)$ and taking the k smallest off-diagonal entries for each row produces km pairs that, as a set, form $C(k, m-1)$. However, some of these pairs will be duplicates, i.e. pairs arising from two rows instead of one. To obtain $\#(C(k, m-1))$ we need to subtract the number of such duplicates from km. Suppose (i, i') arises from both the i-th and the i'-th row. Then i' must belong to $N(i, k)$ and i must belong to $N(i', k)$, i.e. (i, i') must belong to $C(k, k)$. Conversely, suppose (i, i') belongs to $C(k, k)$. Then by definition i' belongs to $N(i, k)$ and i belongs to $N(i', k)$, so that (i, i') arises from both the i-th and the i'-th row. In other words, the set of duplicates and the symmetric set $C(k, k)$ coincide. It follows that $\#(C(k, m-1)) = km - \#(C(k, k))$.

An immediate corollary of this, and the fact that $\#(C(k, k)) \leq \#(C(k, m-1))$ because $C(k, k)$ is contained in $C(k, m-1)$, is that

$$\#(C(k, k)) \leq \frac{km}{2} \leq \#(C(k, m-1)),\ k = 1 \ldots m-1$$

So, regardless of the structure of a specific data set, $km/2$ is an *upper bound* for the size of the symmetric candidate merger set, and a *lower bound* for the size of the asymmetric one. It follows that, as we increase k, the asymmetric sequence $C(k, m-1)$ grows faster than the symmetric sequence $C(k, k)$.

3. SIMULATED DATA

In this section, we introduce a simulated 2-dimensional data structure that allows us to illustrate the working of the proposed agglomeration, and to compare it with other clustering approaches. The data is produced drawing 20 points $x_i = (x_{i,1}, x_{i,2})$ at random from each of six bivariate spherical Gaussians (n = 180), whose locations and variances are selected as to provide the structure shown in Fig. 2A. A sparse background is given by a Gaussian located at $\mu_1 = (60, 60)$ with standard deviation $\sigma_1 = 30$, three Gaussians with standard deviation $\sigma_2 = \sigma_3 = \sigma_4 = 6$ are located in the lower left $\mu_2 = (20, 20)$, lower right $\mu_3 = (20, 100)$ and upper right $\mu_4 = (100, 100)$ corners of the data range, and two Gaussians with standard deviation $\sigma_5 = \sigma_6 = 3$ are located very close to one another with centers $\mu_5 = (30, 100)$ and $\mu_6 = (10, 100)$ in the upper left corner. Indicating with $\varphi_{\mu,\sigma}(\circ)$ a bivariate Gaussian density with mean vector μ and covariance $\sigma^2 I$, we can model this data with the density

$$\Phi(x) = \sum_{j=1\ldots6} \pi_j \varphi_{\mu_j,\sigma_j}(x),\ \pi_j = \frac{1}{6},\ j = 1 \ldots 6$$

and an ideal reduction for it would comprise six groups of approximately equal point content; one for the sparse background, and five for the concentration areas.

We use this data to investigate the performance of information based agglomerative segmentations (symmetric and asymmetric), and compare them to the extreme cases – i.e. traditional agglomerative clustering based on distance alone, $k = t = 0$, and agglomeration

based solely on entropy, $k = t = max$, where *max* indicates the number of classes minus 1, $m - 1$, to be used at each iteration. We include in the comparison also a *k*-means algorithm, which in principle should work effectively when partitioning data generated from spherical Gaussian mixtures. Whenever needed (i.e. in all agglomerations with $k < max$), distances among classes are defined by centroid linkage.

Here, we evaluate the performance of a generic partition *y* in, say, *m* classes (produced by any clustering algorithm) using an approach that relies on the known mixture structure of the simulated data. Namely, we measure the dissimilarity between the mixture density introduced above and the empirical mixture density associated with the partition; that is:

$$\hat{\Phi}_y(x) = \sum_{j=1...m} f(y_j)\varphi_{\bar{x}_j, s_j}(x)$$

where $f(y_j)$ is the frequency of class *j* in the partition, and the center and spread for each class are computed using the points in the class as

$$\bar{x}_j = \frac{1}{f(y_j)} \sum_{i: j[i]=j} f(x_i)x_i$$

$$s_j^2 = \frac{1}{f(y_j)} \sum_{i: j[i]=j} f(x_i)d^2(x_i, \bar{x}_j)$$

Note that for our simulated data all of the drawn points are distinct, i.e. $f(x_i) = 1/n$ for each $i = 1\ldots n$.

Concentrating on the available points, we compute dissimilarity as:

$$\Delta(y) = \sum_{i=1...n} \Phi(x_i)\log\left(\frac{\Phi(x_i)}{\hat{\Phi}_y(x_i)}\right)$$

and report results in Table 1. Clearly, for our simulated data structure, agglomerative segmentations that leverage *both* distance and entropy work better than those employing only distance, or only entropy. Also, while *k*-means does better than the latter, it does far worse than the former.

Interestingly, there appears to be an "optimal" combination of the two criteria; entropy improves performance up to a point, but when it becomes too dominant relative to distance (here, when *k* exceeds 2, and/or the asymmetry is maximized – recall both larger *k* and higher degree of asymmetry lead to a stronger role of entropy) performance deteriorates.

Table 1.

AGGLOMERATIVE	Distance $k = t = 0$	1.838049e-01			
	Distance and Entropy	Symmetric $t = k$	Asymmetric		
			$t = k + 1$	$t = k + 2$	$t = max$
	k=1	1.186953e-01	1.091739e-01	1.057796e-01	1.178201e-01
	k=2	1.098222e-01	**8.338563e-02**	**8.637635e-02**	1.178201e-01
	k=3	1.046846e-01	1.034067e-01	1.238444e-01	1.306283e-01
	Entropy $k = t = max$	2.270318e-01			
k-means clusters		1.794120e-01			

A performance comparison for: information based symmetric ($t = k$) and asymmetric ($t = k + 1$, $t = k + 2$ *and* $t = max$, where *max* is the number of classes minus 1 at each iteration) agglomerative segmentations at $k = 1, 2$ and 3 (decreasing strength of the distance constraint); traditional agglomerative clustering based on distance alone ($k = t = 0$); agglomeration based on entropy alone ($k = t = max$), and *k*-means clustering. The values reported are dissimilarities Δ (evaluated cumulatively at the observed points) between the six-component Gaussian mixture modeling the simulated data structure of Section 3, and the empirical Gaussian mixtures associated to the six-class partitions produced by each method. The best partitions are provided by information based asymmetric agglomerative segmentations with $k = 2$ and $t = 3$ or 4 (bold in Table; see Fig. 2B).

Finally, for our simulated data structure asymmetry clearly produces an advantage. When non-mutual neighborship is allowed, mergers between low frequency classes and high frequency classes closest to them are likely to make it into the candidate set, and thus to be selected, as they result in better entropy values than mergers between close high frequency classes. In general, this helps attributing points in low density areas to classes in nearby high density areas, while preventing the merger of classes in high density areas to one another. In the simulated data structure considered here, asymmetry allows us to successfully separate the two close-by, tight components in the upper left corner of the data range.

In summary, the best partitions are provided by information based asymmetric agglomerations with k = 2 and t = 3 or 4 (see Table 1). Fig. 2B shows the partition produced by the agglomeration with k = 2 and t = 3, which captures the simulated data structure with remarkable fidelity. The two panels of Fig. 3 show the partitions produced by traditional agglomerative clustering (distance only), and the k-means algorithm.

4. ALIGNMENT COLUMNS EXAMPLE AND REGULATORY POTENTIAL

Computational tools aiding in the identification of functional loci of a genome are an important and active area of research. Current algorithms for predicting the location of coding regions and genes are quite effective, because existing knowledge about the structure of these loci can be incorporated in explicit and relatively simple probabilistic models for their sequences (e.g. hidden Markov models; see for instance Siepel and Haussler 2004a and b, and Gross and Brent 2006). The problem becomes harder when considering other types of functional regions, such as those which regulate the transcription of genes. Knowledge about them is available; they often show a high degree of conservation across species, are located in the proximity of genes, and comprise modules of transcription factor binding sites. However, this knowledge is less conclusive than the one available for coding regions – for instance, regulatory regions can withstand a fair amount of divergence across species without losing function, may be quite distant from genes, and binding site motifs are well characterized only for a limited range of transcription factors (see Tompa *et al*. 2005 for a review, as well as Costas *et al*. 2003 and Dermitzakis and Clark 2002). For these reasons, when developing tools to help locate regulatory regions it is important to retain a data-driven perspective, allowing both expected and unexpected signals in observed alignments to contribute to prediction algorithms. This is the approach we took in Taylor *et al*. (2006), where we used 7-species alignments (human, chimpanzee, macaque, mouse, rat, dog, and cow) of known regulatory regions and non-functional sequences to train log-odds scores (called Regulatory Potential) based on variable order Markov models. This allowed us to exploit both conservation and composition patterns in the training alignments – without postulating *ex ante* which patterns may be relevant, or in what form they would contribute to prediction.

The main complication in developing Regulatory Potential scores arose from the very large number of possible alignment columns (see Introduction), and yet larger number of short motifs composed by them. Without strong *ex ante* assumptions, the available training data was extremely scarce relative to the size of the models to be fit to compute the scores. Thus, in order to extract meaningful predictive information from the known regulatory regions and non-functional sequences at our disposal, we had to collapse alignment columns into a much reduced "alphabet" which would allow us to parsimoniously encode alignments.

For this purpose, we designed a two-stage protocol. The first stage was unsupervised; we used our information based agglomerative segmentation approach to group alignment columns embodying roughly equivalent evolutionary histories. Once this moved us far enough along to dampen overfitting, we switched to a supervised stage, where alignment columns were further grouped based on classification performance (i.e. the ability to separate regulatory elements and non-functional sequences).

In more detail, we mapped each of the possible 7-way alignment columns (a number approximating 275,000 when accounting for special encodings of missing alignment entries) into a point in the 5-dimensional simplex representing an ancestral distribution (see Introduction). As a preprocessing step, we then merged ancestral distributions corresponding to alignment columns that occurred less than 10 times in the training data to the closest ones that occurred at least 10 times, resulting in 923 points. These points

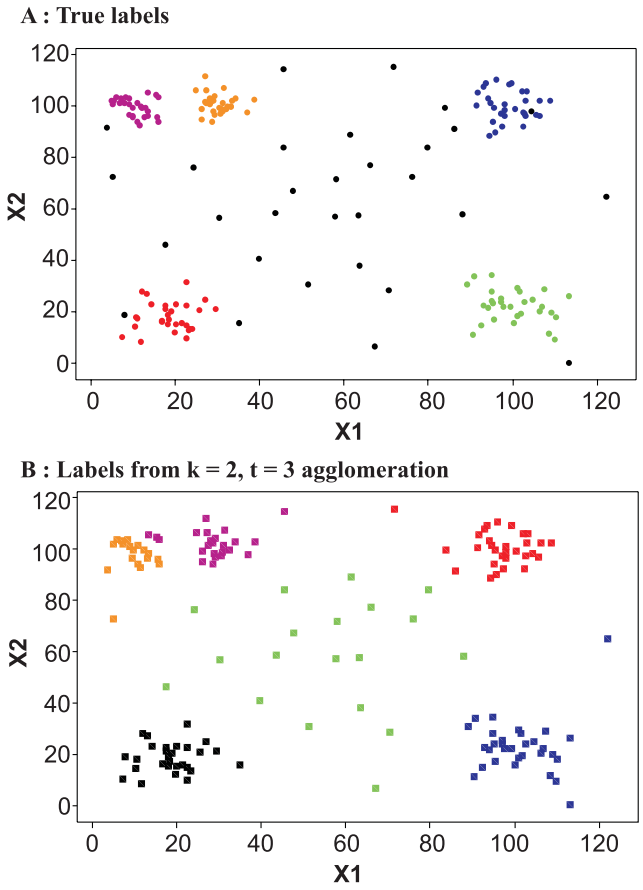

Fig. 2. **A** shows the 2-dimensional simulated data structure used in Section 3 to compare the performance of information based agglomerative segmentations (symmetric and asymmetric), traditional agglomerative clustering based on distance alone, agglomeration based on entropy alone, and *k*-means clustering (see Table 1). The data is produced through six Gaussian components – a very sparse one creating a background, and five more concentrated ones located at the corners of the data range. The two components very close to one another in the upper left corner illustrate a geometric advantage of asymmetric agglomeration: allowing non-mutual neighborship, mergers between low frequency classes and high frequency classes closest to them are likely to make it into the candidate set, and thus to be selected, as they result in better entropy values than mergers between close high frequency classes. This helps attributing points in low density areas to classes in nearby high density areas, while preventing the merger of classes in high density areas to one another. **B** shows the six-group segmentation produced by an information based asymmetric agglomeration with $k = 2$ and $t = 3$, which captures the simulated data structure with remarkable fidelity.

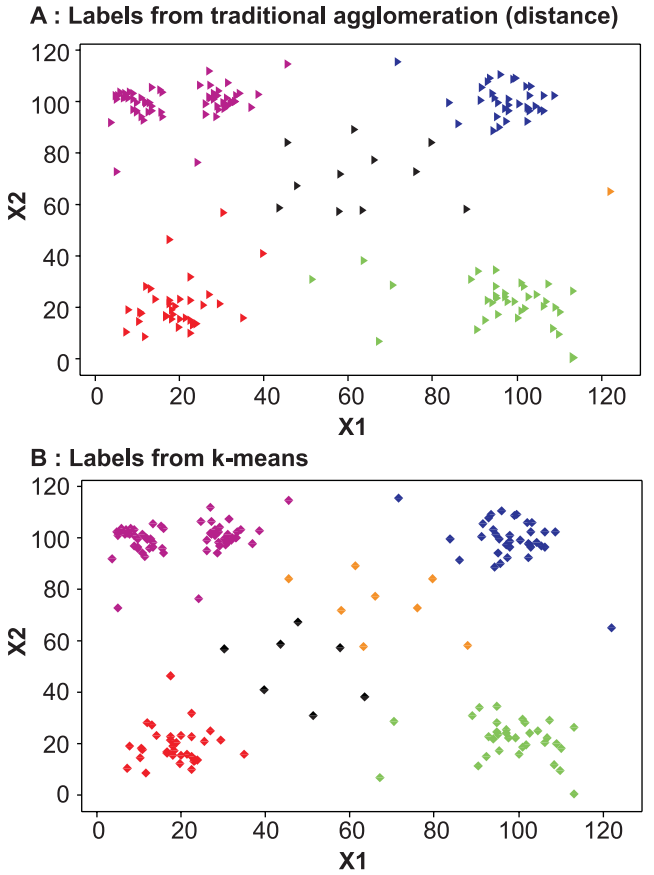

Fig. 3. The 2-dimensional simulated data from Section 3 partitioned in six groups using traditional agglomeration (based on distance only; **A**), and the *k*-means algorithm (**B**). True labels are shown in Fig. 2A. Both methods fail to separate the two components in the upper right corner, and attribute a number of background points to the more concentrated components. Moreover, traditional agglomeration "wastes" one group to render the isolated point on the extreme right, and k-means "wastes" one group splitting the central region (background) in two. Information based agglomerative segmentations perform much better on this data (see Fig. 2B and Table 1).

were grouped using Euclidean distance (other choices of distance did not produce qualitatively different results; data not shown), centroid linkage, and an information based asymmetric agglomeration with $k = 1$ and $t = max = m - 1$ (where m indicates the current number of classes in an iteration). Thus, in a generic iteration, two classes y_j and $y_{j'}$ were considered for merger if y_j was the closest to $y_{j'}$ (regardless of where $y_{j'}$ ranked as a neighbor of y_j), *or* $y_{j'}$ was the closest to y_j (regardless of where y_j ranked as a neighbor of $y_{j'}$); in symbols

$$d_L(y_i, y_{i'}) = \min_{\ell \neq i} d_L(y_i, y_\ell) \text{ or } \min_{\ell \neq i'} d_L(y_\ell, y_{i'})$$

The merger was then chosen by maximizing mutual information, i.e. the entropy of the resulting partition.

Agglomeration was conducted until 75 groups were reached, and these were used to initialize stage two – in which we used supervision to further group alignment columns. Again operating iteratively, we randomly generated candidate mergers and splits, and selected among them based on the ability of the resulting partitions to separate the two types of training units (alignments of known regulatory regions and non-functional sequences). This required fitting variable order Markov models, deriving log-odds scores, and computing classification success rates with cross-validation – which could meaningfully guide the search for a satisfactory partition only because stage one allowed us to bypass the worst effects of overfitting (see Taylor *et al*. 2006, for more details).

We terminated the process with a final alphabet comprising 17 groups of alignment columns. The Regulatory Potential scores derived from fitting variable order Markov models with this alphabet afforded a remarkable leave-one-out cross-validation success rate of ~94% on the training data – as well as excellent performance on largely independent test sets of experimentally confirmed regulatory sequences from the hemoglobin β gene cluster on human chromosome 11 (see again Taylor *et al*. 2006, for more details).

As with all tools relying on such a drastic reduction in data complexity, deciphering how different signals contribute to the predictive success of Regulatory Potential scores is a very difficult task. Fig. 4, which shows a Genome Browser (Karolchik *et al*. 2003) view of the hemoglobin β gene cluster, illustrates how Regulatory Potential manages to capture signals beyond traditional conservation. For more information on the interpretation and experimental validation of our scores we refer the reader to Taylor *et al*. (2006), Wang *et al*. (2006), King *et al*. (2005), and King *et al*. (2007).

5. CONCLUSIONS

In this article we presented an approach to agglomerate data points into spatially contiguous groups that preserve both their distance and frequency structure in a metric space. This approach is logically close to traditional agglomerative algorithms (e.g.

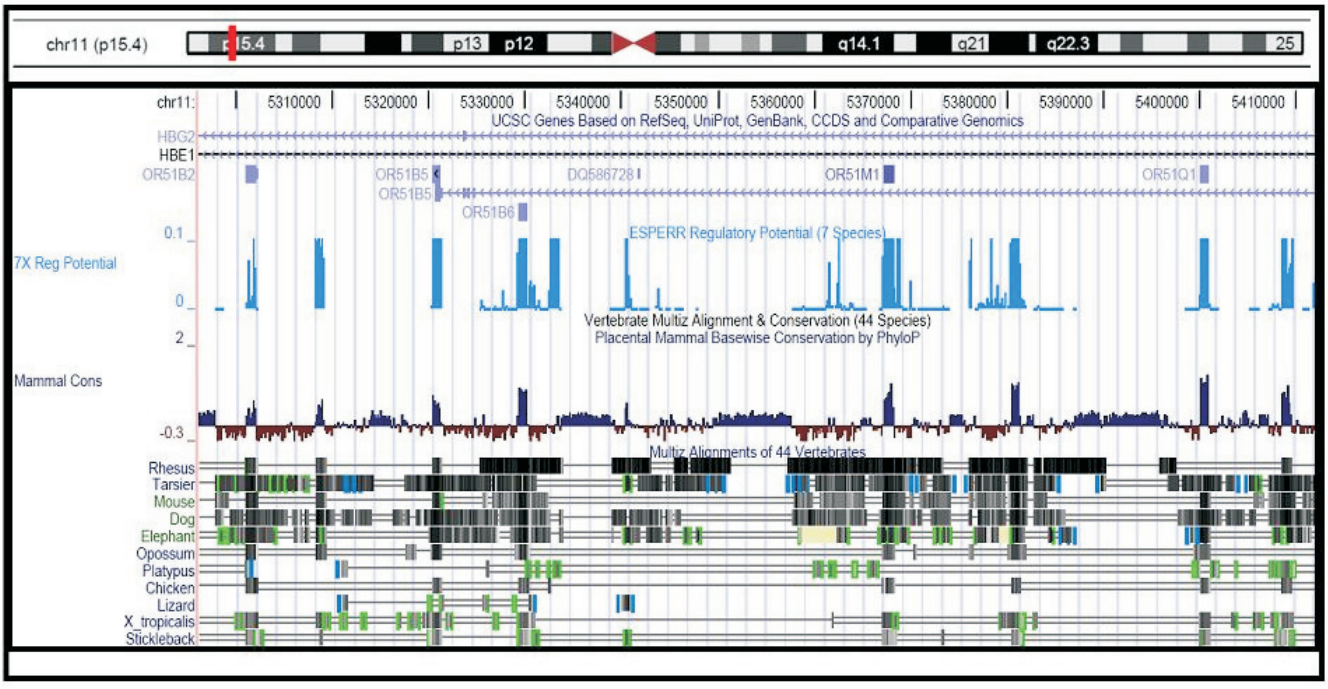

Fig. 4. A Genome Browser view of the hemoglobin β gene cluster locus on human chromosome 11. The blue track, which graphs the 7-species Regulatory Potential scores, is contrasted with information on conservation – in the form of mammalian conservation scores, and marked multispecies alignments from 44 vertebrates (for more information see track descriptions and references provided at http://genome.ucsc.edu/cgi-bin/hgGateway). While Regulatory Potential clearly captures a conservation signal, the existence of peaks specific to the former demonstrates that other signals extracted from the training data contribute to its predictions.

hierarchical clustering), as well as other segmentation techniques sometimes employed in statistics (e.g. Voronoi tassellations, which can be used for reweighting in several application contexts – see for instance Cook and Nachtsheim 1994 and Olive 2002). The approach also borrows from existing clustering techniques based on information measures, which are broadly used in machine learning (see Berjerano *et al*. 2004 and references therein in the domain of genomics applications, and Banerjee *et al*. 2005 and Dhillon *et al*. 2003 in other application domains). However, its ability to leverage *simultaneously* distance and entropy makes it more effective than other clustering approaches for reducing data sets that do not present spatially separated clusters. We list here a few concluding remarks that point towards avenues for future research.

Comparing partitions and data: In Section 4, we used our knowledge on the model underlying the simulated data to measure how accurately a given partition y captures its structure. Completely model-free measures can be designed though, and they have a more general appeal. In these, the original Gaussian mixture $\Phi(x_i)$ at the observed points would be replaced by the empirical distribution, i.e. $f(x_i)$, $i = 1\ldots n$, and the empirical Gaussian mixture $\hat{\Phi}_y(x_i)$ by a model-free empirical distribution associated to the partition, e.g.

$$\hat{f}_y(x_i) = \frac{1}{\#(y_{j[i]})} f(y_{j[i]}) \quad , \quad i = 1...n$$

where the frequency accrued by each class is spread uniformly on the points in the class. One could then compute dissimilarity as

$$\Delta(y) = \sum_{i=1...n} f(x_i) \log\left(\frac{f(x_i)}{\hat{f}_y(x_i)} \right)$$

However, this produces a non-trivial model-free evaluation only if the distinct data points present different frequencies, i.e. if the empirical distribution is *not* $f(x_i) = 1/n$, $i = 1\ldots n$. In fact, if the empirical distribution is uniform on the distinct points (as was the case in the simulated data structure of Section 4), we have that $\hat{f}_y(x_i) = 1/n$, $i = 1\ldots n$, too, regardless of the partition y. In other words, partitions with very different degrees of resemblance to the data will all result in the same distribution, and in dissimilarity equal to 0. This is not as paradoxical as it seems at first; when the empirical distribution is uniform on the distinct points, spreading the frequency accrued by each class uniformly on the points belonging to the class reproduces exactly the distribution of the data, no matter how uninformative the partition.

These observations support the practice of implementing a pre-merger of the data, prior to running an information based agglomerative segmentation – this was done on the alignment column data example of Section 4, and allows the evaluation of the final partition based on its resemblance to the pre-merger partition with which the agglomeration is initialized (since the latter is not uniform). Notably, the pre-merger could be obtained agglomerating with distance alone; that is, one could utilize distance as the sole criterion when grouping the data at a very fine scale, and then switch to a mix of distance and entropy for the grouping of larger scale segments.

Selecting the number of groups: Even though we did not explore it in this article, the issue of determining a satisfactory partition size is crucial for all kinds of clustering and segmentation methods, including ours. For methods comprising explicit models (e.g. mixture-based clustering) information criteria such as BIC or AIC can be used to optimize the number of clusters (see for instance Fraley and Raftery 1998). A large repertoire of heuristic diagnostics also exists, which can be used with several clustering methods. Diagnostics that do not rely directly on distance, such as those measuring stability or predictability of a partition (Ben-Hur *et al*. 2002, Dudoit and Fridlyand 2002), can be used straightforwardly with our method. Interestingly, diagnostics that do rely directly on distance, such as silhouettes or gaps (Hartigan 1975; Tibshirani *et al*. 2001), could also be used with our method *in conjunction with measurements of entropy performance* – in other words, since our approach combines two criteria, specialized "bivariate" diagnostics could and should be developed for it.

Applications to more general objects: Many application examples exist, in genomics as well as other fields, in which the elements one seeks to group for data reduction are not points in a Euclidean space, but more complex objects, such as (i) *graphs* (e.g. phylogenetic

trees for a set of species estimated on different multiple alignment data – from different regions of the nuclear genome, mitochondrial DNA, etc.), or (ii) *matrices* (e.g. substitution matrices between two species estimated on different 2-way alignment data), or even (iii) *functions* (e.g. parametric or non-parametric fits for an expression response sampled over a time or dose course for different transcribed loci).

Importantly, the methodology proposed in this article could be employed in all these cases, regardless of the complexity of the objects; all that is needed is a sensible distance on the feature space in which they are represented, and an empirical distribution (a frequency vector) on the elements if their occurrence is non-uniform.

AKNOWLEDGEMENTS

Both FC and JT were partially supported for this work by grant NIH NIDDK R01 DK065806-06. FC was also partially supported by NSF DMS-0704621.

REFERENCES

Banerjee, A., Merugu, S., Dhillon, I.S. and Ghosh, J. (2005). Clustering with Bregman Divergences. *J. Machine Learning Res.,* **6**, 1705-1749.

Bejerano, G., Haussler, D. and Blanchette, M. (2004). Into the heart of darkness: Large-scale clustering of human non-coding DNA. *Bioinformatics*, **20**, 140-148.

Ben-Hur, A., Elisseeff, A. and Guyon, I. (2002). A stability based method for discovering structure in clustered data. *Proc. of the Pacific Symposium on Biocomputing*, 6-17.

Blanchette, M., Kent, W.J., Riemer, C., Elnitski, L., Smit, A.F., Roskin, K.M., Baertsch, R., Rosenbloom, K., Clawson, H., Green, E.D., *et al*. (2004). Aligning multiple genomic sequences with the threaded blockset aligner. *Genome Res.,* **14**, 708-715.

Costas, J., Casares, F. and Vieira, J. (2003). Turnover of binding sites for transcription factors involved in early Drosophila development. *Gene*., **310**, 215-220.

Cook, R.D. and Nachtsheim, C.J. (1994). Reweighting to achieve elliptically contoured covariates in regression. *J. Amer. Statist. Assoc.,* **89**, 592-599.

Dermitzakis, E.T. and Clark, A.G. (2002). Evolution of transcription factor binding sites in Mammalian gene regulatory regions: Conservation and turnover. *Mole. Biol. Evol.,* **19**, 1114-1121.

Dhillon, I., Mallela, S. and Kumar, R. (2003). A divisive information-theoretic feature clustering algorithm for text classification. *J. Machine Learning Res.,* **3**, 1265-1287.

Dudoit, S. and Fridlyand, J. (2002). A prediction-based resampling method to estimate the number of clusters in a dataset. *Genome Biol.,* **3(7)**, 0036.1-21.

Durbin, R., Eddy, S., Krogh, A., and Mitchison, G. (1998). *Biological Sequence Analysis*. Cambridge University Press, Cambridge, UK.

Gross, S.S. and Brent, M.R. (2006). Using multiple alignments to improve gene prediction. *J. Comput. Biol.*, **13**, 379-393.

Fraley, C. and Raftery, A.E. (1998). How many clusters? Which clustering methods? Answers via model-based cluster analysis. *Computer J.,* **41**, 578-588.

Fraley, C. and Raftery, A.E. (2002). Model-based clustering, discriminant analysis, and density estimation. *J. Amer. Statist. Assoc.,* **97**, 611-631.

Hartigan, J.A. (1975). *Clustering Algorithms*. Wiley, New York.

Hasegawa, M., Kishino, H. and Yano, T. (1985). Dating of the human–ape splitting by a molecular clock of mitochondrial DNA. *Mole. Evol*., **22**, 160-174.

Karolchik, D., Baertsch, R., Diekhans, M., Furey, T.S., Hinrichs, A., Lu, Y.T., Roskin, K.M., Schwartz, M., Sugnet, C.W., Thomas, D.J., Weber, R.J., Haussler, D. and Kent, W.J. (2003). The UCSC Genome Browser Database. *Nucleic Acids Res.*, **31**, 51-54.

King, D.C., Taylor, J., Elnitski, L., Chiaromonte, F., Miller, W., and Hardison, R.C. (2005). Evaluation of regulatory potential and conservation scores for detecting cis-regulatory modules in aligned mammalian genome sequences. *Genome Res.,* **15**, 1051-1060.

King, D.C., Taylor, J., Zhang, Y., Cheng, Y., Lawson, H.A., Martin, J., ENCODE groups for Transcriptional Regulation and Multispecies Sequence Analysis, Chiaromonte, F., Miller, W. and Hardison, R.C. (2007). Finding cis-regulatory modules using comparative genomics: Some lessons from ENCODE data. *Genome Res.,* **17**, 775-786.

Mayrose, I., Graur, D., Ben-Tal, N. and Pupko, T. (2004). Comparison of site-specific rate-inference methods for protein sequences: Empirical Bayesian methods are superior. *Mole. Biol. Evol.*, **21**, 1781-1791.

McLachlan, G. and Peel, D. (2000). *Finite Mixture Models*. Wiley, New York.

Olive, D.J. (2002). Applications of robust distances for regression. *Technometrics*, **44(1)**, 64-71.

Siepel, A. and Haussler, D. (2004a). Computational identification of evolutionarily conserved exons. *Proc. of RECOMB*, San Diego, CA, 177-186.

Siepel, A. and Haussler, D. (2004b). Combining phylogenetic and hidden Markov models in biosequence analysis. *J. Comput. Biol.*, **11**, 413-428.

Spellman, P.T., Sherlock, G., Zhang, M.Q., Iyer, V.R., Anders, K., Eisen, M.B., Brown, P.O., Botstein, D. and Futcher, B. (1998). Comprehensive identification of cell cycle-regulated genes of the yeast *Saccharomyces cerevisiae* by microarray hybridization. *Mole. Biol. Cell*, **9(12)**, 3273-3297.

Taylor, J., Tyekucheva, S., King, D.C., Hardison, R., Miller W. and Chiaromonte, F. (2006). ESPERR: Learning strong and weak signals in genomic sequence alignments to identify functional elements. *Genome Res.*, **16**, 1596-1604.

Tibshirani, R., Walther, G. and Hastie, T. (2001). Estimating the number of clusters in a dataset via the gap statistic. *J. Roy. Statist. Soc.,* **B63**, 411-423.

Tompa, M., Li, N., Bailey, T.L., Church, G.M., De Moor, B., Eskin, E., Favorov, A.V., Frith, M.C., Fu, Y., Kent, W.J. *et al*. (2005). Assessing computational tools for the discovery of transcription factor binding sites. *Nature Biotech.*, **23**, 137-144.

Wang, H., Zhang, Y., Cheng, Y., Zhou, Y., King, D.C., Taylor, J., Chiaromonte, F., Kasturi, J., Petrykowska, H., Gibb, B. *et al*. (2006). Experimental validation of predicted mammalian erythroid cis-regulatory modules. *Genome Res.,* **16**, 1480-1492.