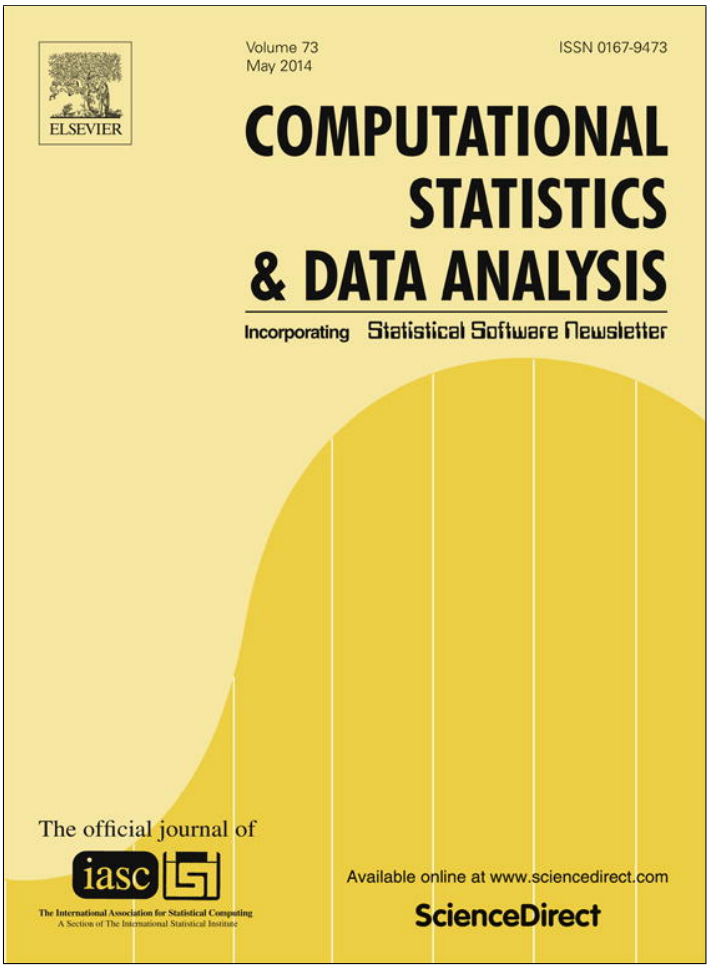

Computational Statistics and Data Analysis 73 (2014) 27–39

Contents lists available at ScienceDirect

Computational Statistics and Data Analysis

journal homepage: www.elsevier.com/locate/csda

A non-parametric method to estimate the number of clusters

André Fujita [a,*], Daniel Y. Takahashi [b], Alexandre G. Patriota [c]

[a] Department of Computer Science, Institute of Mathematics and Statistics, University of São Paulo, Brazil
[b] Department of Psychology and Neuroscience Institute, Green Hall, Princeton University, USA
[c] Department of Statistics, Institute of Mathematics and Statistics, University of São Paulo, Brazil

ARTICLE INFO

Article history:
Received 6 February 2013
Received in revised form 19 November 2013
Accepted 20 November 2013
Available online 4 December 2013

Keywords:
Clustering
Silhouette method
k-means
Spectral clustering

ABSTRACT

An important and yet unsolved problem in unsupervised data clustering is how to determine the number of clusters. The proposed slope statistic is a non-parametric and data driven approach for estimating the number of clusters in a dataset. This technique uses the output of any clustering algorithm and identifies the maximum number of groups that breaks down the structure of the dataset. Intensive Monte Carlo simulation studies show that the slope statistic outperforms (for the considered examples) some popular methods that have been proposed in the literature. Applications in graph clustering, in iris and breast cancer datasets are shown.

1. Introduction

Cluster analyses are methods of classifying "similar" elements into clusters or groups. They are applied in a wide range of areas such as machine learning, pattern recognition, image analysis and bioinformatics. Several clustering methods have been proposed, namely, *k*-means, hierarchical clustering, expectation–maximization clustering, spectral clustering, and many others. By using clustering techniques, one important task is to estimate the proper number of clusters in actual datasets. For example, in cancer data analysis, the grade of tumorigenesis is determined by geometrical parameters such as the cell's shape, density, etc., and the estimation of the number of clusters using these characteristics is important to correctly classify the patients that will receive different treatments depending on the grade of the tumor. In neuroscience, functional magnetic resonance imaging (fMRI) data is clustered and the number of clusters is estimated in order to identify the cortical areas that are activated in a determined cognitive task (Sato et al., 2007). In machine learning and pattern recognition, the estimation of the number of clusters is important in image segmentation in order to identify different objects (Xiang and Gong, 2008), in real-time monitoring network to recognize emerging behavior of a physical system (Zang and Chen, 2010) and in the detection of the number of distinct facial poses under varying illuminations (He et al., 2010).

Although there are several proposals to determine the number of clusters, it is yet an unsolved and difficult problem due to the absence of a clear definition of cluster and especially because it is dependent on both the adopted clustering method and the characteristics of the data distribution (shape and scale, for instance). One difficulty for the majority of the methods is to correctly classify the dataset when the data points inside the same cluster are correlated or are not Gaussian, in high dimensional situations or when there is a dominant cluster (Sugar and James, 2003; Yin et al., 2008). In this paper we propose the slope statistic, a non-parametric and data-driven method for determining the number of clusters in a dataset. The slope statistic is free of reference distributions, has an intuitive interpretation and does not require

* Correspondence to: Rua do Matão, 1010 - Building C, Cidade Universitária São Paulo, SP, CEP 05508-090, Brazil. Tel.: +55 11 3091 5177.
E-mail address: andrefujita@gmail.com (A. Fujita).

0167-9473/$ – see front matter © 2013 Elsevier B.V. All rights reserved.
http://dx.doi.org/10.1016/j.csda.2013.11.012

intensive computations. Furthermore, it can handle situations when the dataset is not a mixture of Gaussian distributions, when there exists a dominant cluster and correlation in the dataset, and when the number of parameters is large. Our proposal is an extension of the silhouette method introduced by Rousseeuw (1987).

In intensive Monte Carlo simulations for determining the number of clusters on artificial datasets, we compare the proposed slope statistic to other seven methods: (a) Bayesian Information Criterion (BIC) (Celeux and Govaert, 1992) for a mixture of Gaussian distributions, (b) the Calinski and Harabasz (CH) index Calinski and Harabasz (1974), (c) the Krzanowski and Lai (KL) index Krzanowski and Lai (1985), (d) the silhouette method (Rousseeuw, 1987), (e) the gap statistic (Tibshirani et al., 2001), (f) the prediction strength (Tibshirani and Walther, 2005), and (g) the jump method (Sugar and James, 2003). In this article, we show that the slope heuristic performs significantly better than these seven methods when the data points inside the same cluster are correlated, non-Gaussian, in high dimensional situations, or when there is a dominant cluster. We also apply the slope statistic in graph clustering and actual biological datasets. We obtain results consistent with prior knowledge of the empirical datasets.

The paper unfolds as follows. Section 2 presents basic notations. Section 3 introduces the slope statistic. Section 4 provides a brief review of other common methods. Some simulations in items generated by different probability distributions and graph clustering are provided in Section 5 and finally, applications in actual datasets in Section 6.

2. Basic notation

Let $\mathcal{X} = \{x_1, \ldots, x_n\}$ be the data with n elements and let $d(x_i, x_j)$ denote the distance between x_i and x_j. The Euclidean distance is the most common choice but other metrics can also be considered. Suppose that we must classify each element of the data $\mathcal{X}$ in one of the following k clusters $C^1, C^2, \ldots, C^k$. The first difficulty is that the number of clusters k is usually unknown *a priori*, thus we have to estimate this value. Furthermore, we note that the definition of a cluster depends on the application and it is not always clear what should be the optimal number of clusters for a given problem even in theoretical grounds. The usual approach to solve this problem is to define a parametric model for the shape of clusters or to use a two-step procedure where a clustering algorithm is applied and then goodness of the classification determines the number of clusters. We follow the latter approach and use the silhouette statistic proposed by Rousseeuw (1987) as the goodness of classification measure. For the sake of completeness, we present a brief review of this measure in what follows.

Define

$$d(x_i, B) = \frac{1}{\#B} \sum_{x \in B} d(x_i, x), \tag{1}$$

as the average dissimilarity of x_i to all elements of cluster B, where $\#B$ is the number of elements of B. Denote by A the cluster to which x_i has been assigned by the clustering algorithm and by C any other cluster different of A. Define

$$a_i = d(x_i, A) \quad \text{and} \quad b_i = \min_{C \neq A} d(x_i, C).$$

The quantities a_i and b_i are the "within" dissimilarity and the smallest "between" dissimilarity, respectively. Then a proposal to measure how well object x_i has been clustered is given by Rousseeuw (1987)

$$s_i = \begin{cases} \dfrac{b_i - a_i}{\max\{b_i, a_i\}}, & \text{if } \#A > 1, \\ 0, & \text{if } \#A = 1. \end{cases} \tag{2}$$

Now, for each number of clusters $k = 2, 3, \ldots, n$ compute the silhouette statistic as

$$s(k) = \frac{1}{n} \sum_{i=1}^{n} s_i.$$

The choice of the silhouette statistic ($s(k)$) is interesting due to its interpretations. Notice that $-1 \leq s_i \leq 1$, therefore, there are three possible situations that must be analyzed. The first one is when $s_i \approx 1$. This implies that the "within" dissimilarity is much smaller than the smallest "between" dissimilarity ($a_i \ll b_i$). In other words, the object x_i has been assigned to an appropriate cluster since the second-best choice cluster is not nearly as close as the cluster the object is assigned. The second situation occurs when $s_i \approx 0$. Then $a_i \approx b_i$, and hence it is not clear whether i should have been assigned to the cluster the object is assigned or to the second-best choice cluster because object x_i lies equally far away from both. The third situation takes place when $s_i \approx -1$. Then $a_i \gg b_i$, so object x_i lies much closer to the second-best choice cluster than to the cluster the object is assigned. Therefore it is more natural to assign object x_i to the second-best choice cluster instead of the cluster the object is assigned because this object x_i has been "misclassified". Usually, the clustering algorithms (the k-means algorithm, for instance) find at least a local optimum solution, therefore this case where $s_i \approx -1$ rarely occurs. To conclude, s_i measures how well object x_i has been classified. Consequently, the silhouette statistic $s(k)$ ($-1 \leq s(k) \leq 1$) measures how well all the objects x_i for $i = 1, \ldots, n$ have been classified on average (Rousseeuw, 1987).

In Rousseeuw's original proposal, it is suggested to select the k such that $s(k)$ is maximum ($\hat{k} = \arg\max_{k \in \{2,\ldots,n\}} s(k)$). This procedure proceeds well if all clusters are homogeneous, i.e., they have approximately the same inner variability.

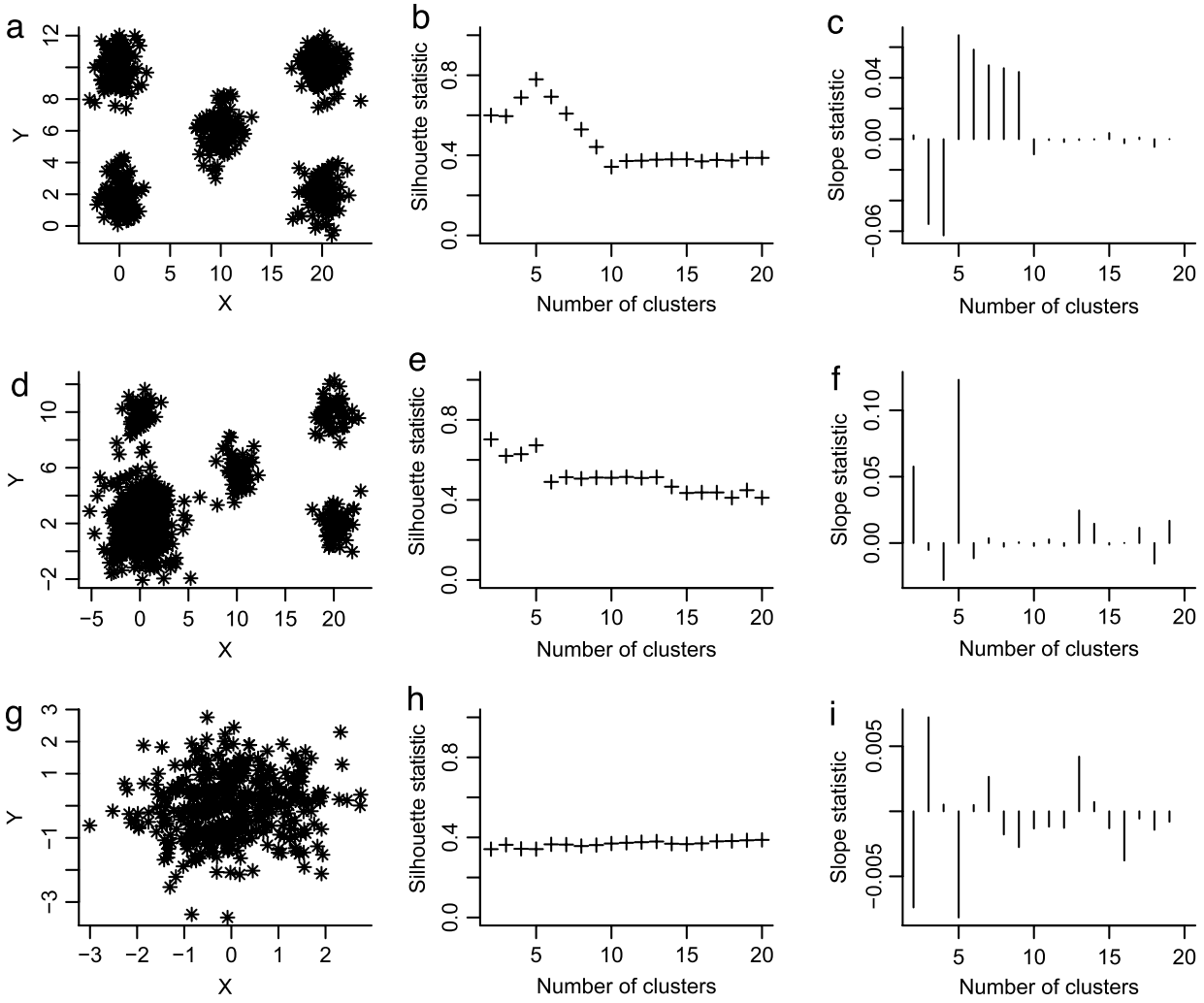

Fig. 1. Illustration of the behavior of the silhouette statistic in two different scenarios. (a) Five clusters with similar sizes and variances. (b) Plot of the number of clusters versus the silhouette statistic. Notice that for $k = 5$, the silhouette statistic presents the maximum value. (c) Plot of the number clusters versus the slope statistic. Notice that for $k = 5$, the slope statistic presents the maximum value. (d) Five clusters with a dominant cluster in terms of size and variance. (e) Plot of the number of clusters versus the silhouette statistic. Notice that, between $k = 5$ and 6, there is a breakdown in the silhouette statistic. (f) Plot of the number clusters versus the slope statistic. Notice that for $k = 5$, the slope statistic presents the maximum value while the silhouette statistic presents the maximum value on for $k = 2$. (g) One cluster. (h) Plot of the number of clusters versus the silhouette statistic. Notice that there is no breakdown. (i) Plot of the number clusters versus the slope statistic.

However, as we shall see in our simulation results in Section 5, when there is a cluster with a dominant inner variability, this procedure becomes unreliable. The main reason is that when there is a cluster with larger inner variability, $s(k)$ can assume large values even when the number of clusters is very small. In the next section we introduce a new procedure, which is far more consistent than the original version.

3. The slope statistic

Here we introduce a new procedure, called slope statistic, that takes into account the above remarks made on $s(k)$. We show in our simulation studies that our procedure works well for a variety of situations, in particular when there is a dominant cluster in the data, in (low and) high dimensions and also for non-normal data.

Heuristically, we would like to choose the number of clusters (k) that not only has the highest silhouette ($\hat{k} = \arg\max_{k\in\{2,\ldots,n\}} s(k)$) but also clusters that cannot be partitioned into smaller clusters, such that increasing the number of clusters would result in a much smaller silhouette, i.e., a *gap* in the silhouette value would be observed. Fig. 1 is an illustration of this idea. In the case that clusters have similar inner variability (Fig. 1(a)), by increasing the number of clusters from $k = 2, \ldots, k^*$ (where k^* is the "correct" number of clusters), the silhouette statistic increases, since the clustering algorithm partitions data into correct clusters, reaching its maximum value when $k = k^*$ (Fig. 1(b)). But, we observe also that when the number of cluster k is larger than k^*, there is a significant decrease in the silhouette. In another situation, when there is a dominant cluster (Fig. 1(d)), the value of $s(k)$ is more influenced by the dominant cluster. In other words, the contribution of s_i of small clusters to $s(k)$ is low if compared to the dominant cluster. Thus, there is no relevant change

in $s(k)$ from $k = 2, \ldots, k^*$ (Fig. 1(e)). On the other hand, when $k = k^* + 1$, the value of $s(k)$ decreases abruptly, since one or more clusters are partitioned incorrectly. Therefore, we observe that in both cases considered above, the optimal number of clusters k^* is characterized by two factors: a large silhouette value when the number of clusters is k^* and a significant decrease in the silhouette value when the number of clusters is larger than k^*. This suggests the following estimator $\hat{k}$ for k^*

$$\hat{k} = \arg \max_{k \in \{2,\ldots,n-1\}} -[s(k+1) - s(k)]s(k)^p, \tag{3}$$

where p is a positive integer value that can be tuned to interpolate between a criterion where the *gap*, $s(k+1) - s(k)$, is more important (small p) and a criterion where the silhouette value has more weight (large p). We observe that $[s(k+1) - s(k)]s(k)^p$ is the discrete version of the derivative of $h_p(k) = \frac{1}{p+1}s(k)^{p+1}$ for $p \geq 0$, which motivates the name *slope statistic* for the criterion (3). Figs. 1(c) and 1(f) illustrate the slope statistic when clusters present similar sizes or when there is a dominant cluster, respectively.

We consider the silhouette approach in the construction of the slope statistic because it takes into account the information about both the inner and inter cluster variabilities for each observation.

In the case the dataset is composed of only one cluster, one may verify the Pearson's correlation between $s(k)$ and the number of clusters $k = 2, \ldots, n-1$. Notice that if the optimal number of clusters for a given dataset is greater than one, the slope method identifies a fast decrease in the silhouette statistic ($s(k)$) right after the "true" number of clusters (k^*), i.e., at $k^* + 1$. However, if the optimal number of clusters for the dataset is one ($k^* = 1$) (Fig. 1(g)), there is no decrease in the silhouette statistic (Fig. 1(h)). In other words, if the Pearson's correlation between the number of clusters and the silhouette statistic is lower than zero, the data is composed of more than one cluster and the slope statistic can be used to estimate k^*, otherwise the dataset is composed of only one cluster.

4. Other approaches

Many methods have been proposed for estimating the number of clusters. For a good review, refer to Milligan and Cooper (1985).

In the next subsections, we describe some of the most used methods, including classical ones such as the CH index Calinski and Harabasz (1974) and the KL index Krzanowski and Lai (1985), one parametric method namely Bayesian Information Criterion—BIC (Celeux and Govaert, 1992) and the recently reported gap statistic (Tibshirani et al., 2001), prediction strength (Tibshirani and Walther, 2005), and jump method (Sugar and James, 2003). Then, we compare the slope method to them in order to evaluate the performance of the proposed method.

4.0.1. Calinski and Harabasz index

The CH index Calinski and Harabasz (1974) is computed as

$$\mathrm{CH}(k) = \frac{B_k/(k-1)}{W_k/(n-k)}, \tag{4}$$

where n is the total number of data points, k is the number of clusters, $B_k = \sum_{l=1}^{k} \#C^l(\bar{x}^l - \bar{x})(\bar{x}^l - \bar{x})'$ is the between cluster sum of squares and $W_k = \sum_{l=1}^{k}\sum_{i=1}^{\#C^l-1}\sum_{j=i+1}^{\#C^l}(x_i^l - x_j^l)(x_i^l - x_j^l)'$ is the within cluster sum of squares, x_i^l and x_j^l are the ith and jth items, respectively, of cluster C^l, $\bar{x}^l$ is the centroid of the lth cluster and $\bar{x}$ is the centroid of the entire dataset. The choice of the number of clusters k is given by the argument that maximizes the CH index. Note that $\mathrm{CH}(k)$ is not defined for $k = 1$ and hence cannot be used to verify whether the data points are composed of one or more clusters.

4.0.2. Krzanowski and Lai index

Based on the proposal by Marriot (1971), who suggested to use $k^2|W_k|$, Krzanowski and Lai (1985) defined

$$\mathrm{DIFF}(k) = (k-1)^{2/m}W_{k-1} - k^{2/m}W_k, \tag{5}$$

where m is the number of features, and uses as a criterion, the number of clusters k that maximizes the quantity

$$\mathrm{KL}(k) = \left|\frac{\mathrm{DIFF}(k)}{\mathrm{DIFF}(k+1)}\right|. \tag{6}$$

Note that similar to CH index, $\mathrm{KL}(k)$ is also not defined for $k = 1$.

4.0.3. Bayesian information criterion

Clusters can be roughly defined as objects belonging to the same distribution of finite variance. A nice property of this definition is that artificial datasets can be generated by randomly sampling objects from a distribution. Therefore, based on this idea that objects are generated by distributions, one may develop an expectation–maximization clustering algorithm (Celeux and Govaert, 1992, 1995). In this method, the dataset is usually modeled with a fixed number of distributions,

Gaussian distributions for instance, that are initialized randomly and whose parameters are iteratively optimized to fit the dataset. Since this approach is a model-based algorithm, a BIC can be derived. The general BIC is given by

$$\mathrm{BIC}(k) = -2\ln(L_k) + q\ln(n), \tag{7}$$

where L_k is the maximized value of the likelihood function for the estimated model and q is the number of free parameters to be estimated. The choice of the number of clusters k is given by the argument that minimizes the BIC. For details of the BIC for mixture Gaussian models, refer to Celeux and Govaert (1992). For our simulations, we used the *mclust* package (version 4.0) implemented in R (R Development Core Team, 2010).

4.0.4. Gap statistic

The idea of this criterion (Tibshirani et al., 2001) is to standardize the graph of $\log(W_k)$ by comparing it with its expectation under an appropriate full reference distribution of the data. The estimate of the optimal number of clusters is then the value of k for which $\log(W_k)$ falls the farthest below this reference curve. Thus, they define

$$\mathrm{Gap}_n(k) = E_n^* \log(W_k) - \log(W_k), \tag{8}$$

where E_n^* denotes the expectation under a sample of size n from the reference distribution. The estimated $\hat{k}$ will be the value maximizing $\mathrm{Gap}_n(k)$ after taking the sampling distribution into account. For our simulations, we used the *clusterSim* package (version 0.40–6) implemented in R (R Development Core Team, 2010).

4.0.5. Prediction strength

For a candidate number of clusters k, let $C^{k1}, C^{k2}, \ldots, C^{kk}$ be the indices of the test observations in test clusters $1, 2, \ldots, k$. Let $\#C^{k1}, \#C^{k2}, \ldots, \#C^{kk}$ be the number of observations in these clusters. The prediction strength of the clustering $\mathrm{Cl}(., k)$ is defined by Tibshirani and Walther (2005)

$$ps(k) = \min_{1\le l\le k} \frac{1}{\#C^{kl}(\#C^{kl}-1)} \sum_{i\neq j\in A_{kl}} D[\mathrm{Cl}(\mathcal{X}_{\mathrm{tr}}, k)\mathcal{X}_{\mathrm{te}}]_{ij}, \tag{9}$$

where $\mathcal{X}_{\mathrm{tr}}$ and $\mathcal{X}_{\mathrm{te}}$ are the training and independent test samples, respectively, and $D[\mathrm{Cl}(\ldots), \mathcal{X}_{\mathrm{te}}]_{ij} = 1$ if items i and j fall into the same cluster, and zero otherwise. For each test cluster, the proportion of observation pairs in that cluster that are also assigned to the same cluster by the training set centroids are computed. The prediction strength is the minimum of this quantity over the k clusters. For our simulations, we used the *fpc* package (version 2.1–4) implemented in R (R Development Core Team, 2010).

4.0.6. Jump method

This procedure is based on distortion, which is a measure of within-cluster dispersion. Formally, let x_i be a m-dimensional random variable with a mixture distribution of G components, each with covariance τ; let $c_1, c_2, \ldots, c_k$ be a set of candidate cluster centers; and let c_{x_i} be the one closest to x_i. Then the minimum achievable distortion associated with fitting k centers to the data is (Sugar and James, 2003)

$$d_k = \frac{1}{m} \min_{c_1,\ldots,c_k} E[(x_i - c_{x_i})^T \tau^{-1} (x_i - c_{x_i})], \tag{10}$$

which is simply the average Mahalanobis distance, per dimension, between x_i and c_{x_i}. Note that in the case where τ is the identity matrix, distortion is simply mean squared error. For our simulations, we used the R code available at http://www-bcf.usc.edu/~gareth/research/jump.

5. Simulations

5.1. Toy data

We constructed datasets in seven different scenarios. We generated items from several distributions assuming that items from the same distribution belong to the same cluster. The seven scenarios are described as follows:

1. *Five clusters with equal number of data points and variances*—100 data points are generated for each cluster by bivariate Gaussian distributions with unit variance centered at (0, 2), (20, 2), (20, 10), (0, 10) and (10, 6).
2. *Five clusters with different number of data points and different variances*—the clusters are centered at (0, 2), (20, 2), (20, 10), (0, 10) and (10, 6). The number of data points for each cluster is 250, 50, 25, 25, 25 and the variances are 3, 1.5, 1, 1, 1, respectively. There is no correlation between the data points.

3. *Five clusters with different number of data points, different variances and with correlation among data points*—this scenario is similar to scenario 2, but the data points belonging to the same cluster (generated by the same distribution) are correlated. The covariance between two data points of the same cluster was set to 0.5.
4. *Three clusters composed of different distributions*—the clusters are uniform distributions in the interval [0, 1] in the x-axis and Gaussian distributions with means 0, 10 and 20 with unit variance in the y-axis. The data points generated for clusters with means 0, 10 and 20 are 250, 100 and 50, respectively.
5. *Five clusters composed of different distributions*—the clusters are t distributions with seven degrees of freedom in the x-axis and exponential distributions in the y-axis. The clusters are centered at (0, 2), (20, 2), (20, 10), (0, 10), (10, 5), and the number of items is 250, 50, 25, 25, and 25, respectively.
6. *Four clusters in high dimensional data*—the clusters are Gaussian distributions with 75 dimensions. The number of data points for each cluster is 350, 50, 25 and 25 and the variances are 3, 2, 1 and 1, respectively. The correlation between items of the same cluster is 0.3. All coordinates are set to 0, 5, 10, and 15 for clusters with sizes 350, 50, 25, and 25, respectively.
7. *One cluster*—500 data points were generated by a Gaussian distribution with mean zero and unit variance.

We applied the k-means clustering algorithm with Euclidean distance and the eight different methods for estimating the number of clusters namely, the slope, silhouette, BIC, CH, KL, gap, prediction strength, and jump. The reason to choose both the k-means and the Euclidean distance is because they are simple and well known by the scientific community. However, any other clustering method or metric could replace them. In order to evaluate the BIC, we replaced the k-means algorithm by the clustering expectation–maximization algorithm assuming Gaussian distributions since BIC is a parametric method, i.e., it requires a model. One hundred realizations were generated from each setting.

5.2. Overlapping data

The simulation studies on Section 5.1 illustrated clusters that are well-separated and the number of clusters is clearly defined. However, when data are overlapped, the concept of what is a cluster can be distinct for different methods, and consequently, the number of clusters too.

In this section, we carried out an experiment to assess how the identification of one or more clusters in the slope method responds to overlapped data. Each simulated dataset consists of 100 observations derived by three bivariate normal populations, with means $(-\Delta, 0)$, $(0, 0)$ and $(\Delta, 0)$, and identity covariance matrix. In other words, 100 items are generated by one bivariate normal distribution with mean $(-\Delta, 0)$, 100 items are generated by a bivariate normal distribution with mean $(0, 0)$ and another 100 items are generated by another bivariate normal distribution with mean $(0, \Delta)$, all of them with identity covariance matrices. As a result, there are "three clusters", where the distances among the means of the three clusters vary depending on the values of Δ. We use $\Delta = 0, 0.5, 1.0, 1.5, \ldots, 7.0$. For each of the 15 values of Δ, 300 repetitions were carried out, and we counted how many times the slope method estimated the number of clusters as three.

5.3. Applications in graph clustering

With the advances of high throughput technologies in molecular biology and neuroscience, such as microarrays, fMRI and EEG (electroencephalogram), large amounts of data are generated. By using these data, researchers often model functional networks in order to characterize the cell or brain state by the network topology. These networks are usually large, with hundreds to thousands of nodes, complicating the analysis. Therefore, in the topological analysis, firstly pass through a clustering step, in order to partition the graph (network) into smaller graphs that are easier to analyze. The problem of partitioning a graph with n nodes into k sub-graphs can be overcome by using the spectral clustering technique that is described as follows (Luxburg, 2007):

Input: The adjacency matrix $\mathbf{H}$ of the graph, where $\mathbf{H} = (h_{i,j})_{i,j=1,\ldots,n}$ (consider $h_{i,j} = 1$ if vertices v_i and v_j are connected by an edge and $h_{i,j} = 0$, otherwise), and the number of clusters k.

1. Compute the Laplacian matrix $\mathbf{Q} = \mathbf{D} - \mathbf{H}$, where $\mathbf{D}$ is the degree matrix defined as the diagonal matrix with the degrees $d_1, \ldots, d_n$ on the diagonal ($d_i = \sum_{j=1}^{n} h_{i,j}$).
2. Compute the k eigenvectors $v_1, \ldots, v_k$ of $\mathbf{Q}$ corresponding to the k largest eigenvalues.
3. Let $\mathbf{V} \in \mathbb{R}^{n \times k}$ be the matrix containing the vectors $v_1, \ldots, v_k$ as columns.
4. For $i = 1, \ldots, n$, let $y_i \in \mathbb{R}^k$ be the vector corresponding to the i-th row of $\mathbf{V}$.
5. Cluster the points $(y_i)_{i=1,\ldots,n}$ with the k-means algorithm into clusters $C^1, \ldots, C^k$.

Output: Clusters $C^1, \ldots, C^k$.

Although the spectral clustering is widely applied in the graph partitioning problem, one must set the number of clusters k. When the sub-graphs are well separated, the number of clusters k is usually set by counting the number of eigenvalues of the matrix $\mathbf{Q}$ that are zero. However, notice that the eigenvalues are not exactly zero, but they fluctuate close to zero due to the variance of the data. Thus, determining a threshold to set which eigenvalue is zero is another problem.

Here, we constructed two scenarios where we compare the proposed slope statistic to the eigenvalue equals-to-zero counting method.

Table 1
Results of the simulation studies.

Number of clusters	1	2	3	4	5	6	7	8	9	10
Scenario 1										
Equal number of points and variances										
Slope ($p = 0$)	0	0	0	0	**7**	9	22	36	12	10
Slope ($p = 1$)	0	0	0	0	**88**	9	0	1	2	0
Slope ($p = 2$)	0	0	0	0	**99**	0	0	1	0	0
Slope ($p = 3$)	0	0	0	0	**99**	0	0	1	0	0
Slope ($p = 4$)	0	0	0	0	**99**	0	0	1	0	0
Silhouette	0	0	0	0	**100**	0	0	0	0	0
BIC	0	0	0	0	**100**	0	0	0	0	0
CH	0	0	0	0	**100**	0	0	0	0	0
KL	0	0	0	0	**100**	0	0	0	0	0
Gap	0	0	0	0	**100**	0	0	0	0	0
Prediction strength	20	0	2	44	**28**	0	0	0	0	0
Jump	0	0	0	0	**100**	0	0	0	0	0
Scenario 2										
Different number of data points and variances										
Slope ($p = 0$)	0	1	0	0	**99**	0	0	0	0	0
Slope ($p = 1$)	0	2	0	0	**98**	0	0	0	0	0
Slope ($p = 2$)	0	12	0	0	**88**	0	0	0	0	0
Slope ($p = 3$)	0	21	0	0	**79**	0	0	0	0	0
Slope ($p = 4$)	0	46	0	0	**54**	0	0	0	0	0
Silhouette	0	100	0	0	**0**	0	0	0	0	0
BIC	0	0	0	0	**23**	53	24	0	0	0
CH	0	0	0	0	**0**	0	19	60	12	4
KL	0	0	0	0	**0**	0	1	13	15	15
Gap	0	14	0	0	**0**	1	60	23	2	0
Prediction strength	0	100	0	0	**0**	0	0	0	0	0
Jump	0	0	0	0	**0**	5	20	0	0	0
Scenario 3										
Gaussian and uniform										
Slope ($p = 0$)	0	0	**85**	0	4	11	0	0	0	0
Slope ($p = 1$)	0	0	**91**	0	2	7	0	0	0	0
Slope ($p = 2$)	0	0	**96**	0	1	3	0	0	0	0
Slope ($p = 3$)	0	0	**98**	0	1	1	0	0	0	0
Slope ($p = 4$)	0	0	**98**	0	1	1	0	0	0	0
Silhouette	0	1	**99**	0	0	0	0	0	0	0
BIC	0	0	**39**	13	13	7	21	6	1	0
CH	0	0	**0**	0	0	0	55	15	26	1
KL	0	0	**0**	0	0	0	3	3	17	2
Gap	0	0	**0**	0	56	6	38	0	0	0
Prediction strength	0	89	**11**	0	0	0	0	0	0	0
Jump	0	0	**0**	0	0	0	8	0	0	0
Scenario 4										
Correlated items										
Slope ($p = 0$)	0	1	0	0	**99**	0	0	0	0	0
Slope ($p = 1$)	0	16	0	0	**84**	0	0	0	0	0
Slope ($p = 2$)	0	27	0	0	**73**	0	0	0	0	0
Slope ($p = 3$)	0	36	0	0	**64**	0	0	0	0	0
Slope ($p = 4$)	0	56	0	0	**44**	0	0	0	0	0
Silhouette	0	100	0	0	**0**	0	0	0	0	0
BIC	0	0	0	0	**18**	45	28	9	0	0
CH	0	0	0	0	**0**	0	13	68	6	3
KL	0	0	0	0	**0**	0	0	6	2	2
Gap	0	12	0	0	**0**	10	60	17	1	0
Prediction strength	0	100	0	0	**0**	0	0	0	0	0
Jump	0	0	0	0	**0**	4	2	0	0	0
Scenario 5										
t and exponential										
Slope ($p = 0$)	0	0	0	0	**100**	0	0	0	0	0
Slope ($p = 1$)	0	0	0	0	**100**	0	0	0	0	0
Slope ($p = 2$)	0	0	0	0	**100**	0	0	0	0	0
Slope ($p = 3$)	0	0	0	0	**100**	0	0	0	0	0
Slope ($p = 4$)	0	0	0	0	**100**	0	0	0	0	0
Silhouette	0	59	0	0	**41**	0	0	0	0	0
BIC	0	0	0	0	**0**	34	39	17	10	0
CH	0	0	0	0	**0**	23	69	8	0	0
KL	0	0	0	0	**0**	5	24	9	4	0
Gap	0	20	0	0	**34**	38	8	0	0	0
Prediction strength	0	100	0	0	**0**	0	0	0	0	0
Jump	0	0	0	0	**12**	0	0	0	0	0

(*continued on next page*)

Table 1 (*continued*)

Number of clusters	1	2	3	4	5	6	7	8	9	10
Scenario 6 *High dimensional data* (75 *dimensions*)										
Slope ($p = 0$)	0	0	6	**93**	1	0	0	0	0	0
Slope ($p = 1$)	0	0	7	**93**	0	0	0	0	0	0
Slope ($p = 2$)	0	0	7	**93**	0	0	0	0	0	0
Slope ($p = 3$)	0	18	3	**79**	0	0	0	0	0	0
Slope ($p = 4$)	0	81	1	**18**	0	0	0	0	0	0
Silhouette	0	100	0	**0**	0	0	0	0	0	0
BIC	0	0	0	**1**	17	47	28	7	0	0
CH	0	23	77	**0**	0	0	0	0	0	0
KL	0	0	0	**0**	0	0	0	0	0	0
Gap	0	0	0	**0**	0	28	30	22	13	7
Prediction strength	1	96	1	**1**	1	0	0	0	0	0
Jump	0	0	0	**0**	0	0	0	0	0	0
Scenario 7 *One cluster*										
Slope ($p = 0$)	**98**	0	1	1	0	0	0	0	0	0
Slope ($p = 1$)	**98**	0	1	1	0	0	0	0	0	0
Slope ($p = 2$)	**98**	0	1	1	0	0	0	0	0	0
Slope ($p = 3$)	**98**	0	1	1	0	0	0	0	0	0
Slope ($p = 4$)	**98**	0	1	1	0	0	0	0	0	0
Silhouette	**0**	0	3	0	0	0	1	0	0	0
BIC	**98**	2	0	0	0	0	0	0	0	0
CH	**0**	0	0	0	0	0	0	0	0	0
KL	**0**	0	0	0	0	0	0	0	0	0
Gap	**97**	3	0	0	0	0	0	0	0	0
Prediction strength	**99**	1	0	0	0	0	0	0	0	0
Jump	**0**	0	0	0	0	0	0	0	0	0

The two toy scenarios are:

1. *Four independent sub-graphs*—four cliques (all vertices of the sub-graph are connected by an edge to all other vertices of the sub-graph) without edges connecting one clique to other clique are constructed. Each sub-graph is composed of 50 vertices.
2. *Four sub-graphs connected by edges*—similar to scenario 1, but now, the four cliques are connected uniformly by 1200 edges. In other words, each pair of cliques is connected on average by 200 edges.

The distance between a pair of vertices (i, j) is defined as the number of edges belonging to the shortest path between vertices (i, j). The shortest path between vertices (i, j) is given by the breadth-first-search algorithm. For both scenarios, we set the threshold to consider an eigenvalue as zero as one. In other words, eigenvalues less than one were considered as zero.

5.4. Results and discussions

The slope statistic was compared in distinct scenarios with methods comprising classical techniques (silhouette, CH index and KL index), one method that is model-based and assumes a mixture of Gaussian distributions (BIC), and three recently published and well diffused in the literature, namely the gap statistic, prediction strength, and jump method.

The simulated scenarios illustrate different configurations that may exist in the real world, such as mixture of Gaussian distributions, items generated by non-Gaussian distributions, one dominant cluster with a few small clusters, correlated items and a scenario that there is only one cluster. The results of the number of clusters identified by each method in a total of 100 repetitions are given in Table 1. Some rows do not add up to 100 because the number of estimated clusters was greater than 10. In bold are the correct number of clusters.

It is important to point out that, for scenarios 1–6, although some items were generated by distributions with infinite support, such as normal and exponential distributions, all items were correctly clustered by k-means in all experiments when the correct number of clusters was used. In other words, clusters are totally independent, without overlap; hence, both clusters and number of clusters are well defined.

The simulation results show that, in practice, the slope method with $p = 1$ provides good results in any configuration. However, depending on the case, higher values of p can be used to give a higher weight to lower number of clusters if one would like to be more conservative (underestimate the number of clusters rather than overestimate).

All the compared methods do quite well when the clusters are a mixture of Gaussian distributions with identity covariance matrix and with the same density. However, when there is a dominant cluster with higher variance, i.e., when one of the clusters is very large, they fail. On the other hand, the slope method provided good results under non-Gaussian distributions or Gaussian with non-identity covariance matrix (correlated items) and also under high dimensionality.

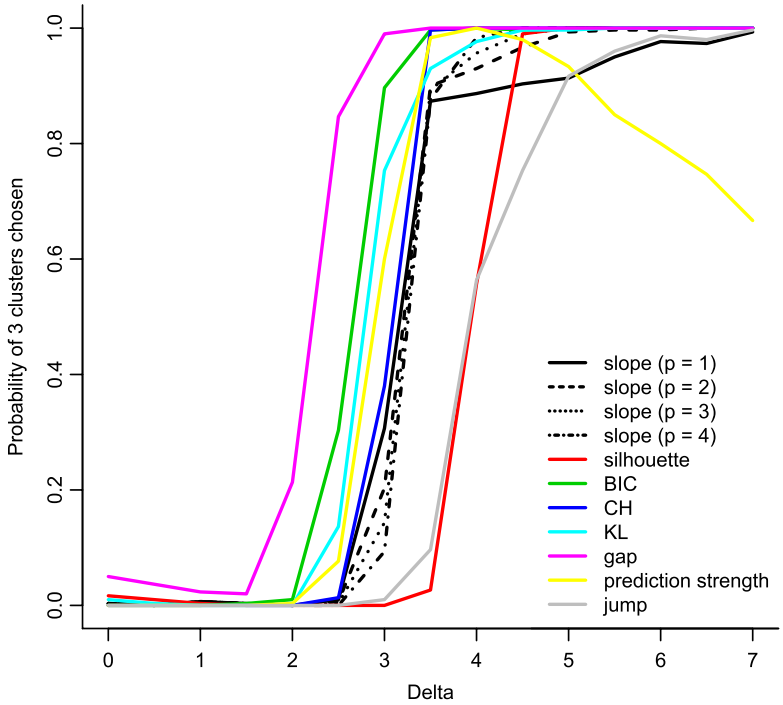

Fig. 2. Three clusters having different portions of overlapping area in each dataset. The higher the Δ, the farther are the means of the three Gaussian distributions with unit variance. The line represents the probability of identifying three clusters.

By comparing the slope method that is non-parametric to BIC which is parametric, it is necessary to point out that the slope method was equivalent or better than BIC even in situations that the data points were generated by multi-Gaussian distributions (scenarios 1, 2, 3 and 6).

The slope, BIC, gap, and prediction strength presented good performance when there was only one cluster. For other methods, the performance was not good because they are not defined for $k = 1$.

The tendency of the gap statistic to overestimate the number of clusters as observed on scenarios 2–5 was also reported by Dudoit and Fridlyand (2002). It is also known that the gap statistic may not work correctly in cases where data are derived by exponential distributions such as on scenario 5 (Sugar and James, 2003). Yin et al. (2008) pointed out that in situations where a dataset contains clusters of different densities the gap statistic might fail (scenarios 2–5).

Regarding the case that the data points are overlapped (Section 5.2), Fig. 2 shows the probability of finding three as the number of clusters for each value of Δ by the methods. The closer the centers of the clusters, the more difficult is for the methods to identify three clusters. The performance obtained by the slope method was comparable to the results obtained by other methods. Interestingly, prediction strength seems to not consistently identify three clusters when the distances among the centers of the normal distributions become farther. As the centers are further away the prediction strength overestimates the number of clusters to four.

The slope statistic was also applied in the graph's clustering problem. The most common criterion to select the number of clusters in graphs is to count the number of eigenvalues of the matrix $\mathbf{Q}$ that are equal to zero. In practice, the eigenvalues are rarely zero due to numerical fluctuations. Thus, usually (in machine learning) a threshold is set and eigenvalues lower than this threshold are considered as zero. Two scenarios, one with four independent cliques and another with four cliques connected by a few edges were studied. By analyzing Table 2, we verify that when the sub-graphs are clearly separated (scenario 1), both methods estimated correctly the number of clusters. However, when there are edges among the cliques (scenario 2), the eigenvalue equals-to-zero counting method identified only one cluster. In order to verify what was happening to the eigenvalue equals-to-zero method, we plot the eigenvalues of one of the simulations as an illustration. In Fig. 3(f) it is possible to see that there is only one eigenvalue close to zero (eigenvalue less than one) and three other eigenvalues that are clearly not zero. However, one can verify that there is a gap between the three eigenvalues that are not zero to the rest of the eigenvalues. Thus, in this case, the eigenvalue equals-to-zero counting method suggests that the graph is composed of one or four clusters, depending on the assumed threshold. On the other hand, the slope statistic and other methods identified correctly the number of cliques (partitions of the graph). Both BIC and jump method were not applied to graph clustering problem because they need as input the distribution of the dataset. However, in graph clustering, the distances among vertices are given by their connectivity.

6. Applications in actual data

In this section, we illustrate the slope method in two real world datasets. They are the classical iris and Fisher (1936) breast cancer (Wolberg and Mangasarian, 1990) datasets. The k-means algorithm with the Euclidean distance were used as a clustering algorithm and a metric, respectively.

Table 2
Results of the simulation study on graphs.

Number of clusters	1	2	3	4	5	6	7	8	9	10
Scenario 1										
Disjoint cliques										
Slope ($p = 0$)	0	0	0	**44**	56	0	0	0	0	0
Slope ($p = 1$)	0	0	0	**89**	11	0	0	0	0	0
Slope ($p = 2$)	0	0	0	**97**	3	0	0	0	0	0
Slope ($p = 3$)	0	0	0	**100**	0	0	0	0	0	0
Slope ($p = 4$)	0	0	0	**100**	0	0	0	0	0	0
Silhouette	0	0	0	**100**	0	0	0	0	0	0
CH	0	0	0	**100**	0	0	0	0	0	0
KL	0	0	0	**100**	0	0	0	0	0	0
Gap	0	0	0	**100**	0	0	0	0	0	0
Prediction strength	100	0	0	**0**	0	0	0	0	0	0
Eigenvalue	0	0	0	**100**	0	0	0	0	0	0
Scenario 2										
Interconnected cliques										
Slope ($p = 0$)	0	0	0	**65**	18	11	3	3	0	0
Slope ($p = 1$)	0	0	0	**80**	15	3	0	1	1	0
Slope ($p = 2$)	0	0	0	**87**	13	0	0	0	0	0
Slope ($p = 3$)	0	0	0	**91**	9	0	0	0	0	0
Slope ($p = 4$)	0	0	0	**94**	6	0	0	0	0	0
Silhouette	0	0	0	**100**	0	0	0	0	0	0
CH	0	4	0	**96**	0	0	0	0	0	0
KL	0	4	0	**96**	0	0	0	0	0	0
Gap	0	0	0	**100**	0	0	0	0	0	0
Prediction strength	100	0	0	**0**	0	0	0	0	0	0
Eigenvalue	100	0	0	**0**	0	0	0	0	0	0

The iris data, studied by Fisher (1936) in linear discriminant analysis, contain 150 items measured on four variables (sepal length, sepal width, petal length and petal width). The objects are categorized by three species (iris setosa, iris versicolor and iris virginica). Each species is represented by 50 items.

The Wisconsin breast cancer data (Wolberg and Mangasarian, 1990) contain measurements on nine variables (clump thickness, uniformity of cell size, uniformity of cell shape, marginal adhesion, single epithelial cell size, bare nuclei, bland chromatin, normal nucleoli and mitosis) recorded for the biopsy specimens of 683 patients. These data consist of at least two distinct clusters that correspond to one group of the 444 benign specimens and the other group containing the remaining 239 malignant specimens.

In real world data applications, the slope method was used to identify the number of clusters in classic iris and breast cancer datasets.

In the iris dataset, it has been found that iris setosa could be well separated from the other two species, whereas iris versicolor and iris virginica are somewhat overlapping. Fig. 4(a) is the plot after multidimensional scaling procedure, i.e., re-scaling the four dimensional data to two dimensions. The colors represent the two different clusters identified by k-means. The characters represent the three types of iris (the triangles, crosses and squares represent the setosa, versicolor and virginica iris, respectively). Notice that it is reasonable to conclude that the data contain either two or three clusters since iris versicolor and iris virginica are overlapping. In this example, the slope statistic correctly identified two clusters (followed by the suggestions of five or three) as can be observed by the slope statistic values in Fig. 4(c). The suggestion of five clusters is probably due to the outliers in the virginica iris (represented by squares) and the presence of a gap at the center of the virginica dataset, dividing the set into two sub-groups.

In the breast cancer data, it is known that there are at least two clusters: the benign and the malign groups. Fig. 4(d) is the plot after multidimensional scaling the nine dimension data to two dimensions. The crosses represent the patients with malign cancer and the triangles represent the patients with benign types of cancer. The slope method suggests four clusters that are represented by four different colors. The clusters represented by green and black colors are overlapped due to multidimensional scaling. Notice that the benign cluster is compact while the malign cluster was split into three sub-groups suggested by the slope method. This is reasonable since it is known that the malign cancer is very heterogeneous and can be sub-classified into at least two sub-groups (Anderson and Matsuno, 2006) while the benign type is more homogeneous (Fig. 4(d)). Notice that the second number of clusters suggested by the slope method is two, clustering the items into benign and malign groups.

7. Final remarks

As observed in the applications of the slope method in real world data, one may also analyze the k related to the second largest slope, the third largest slope and so on and so forth in order to extract more information from the dataset.

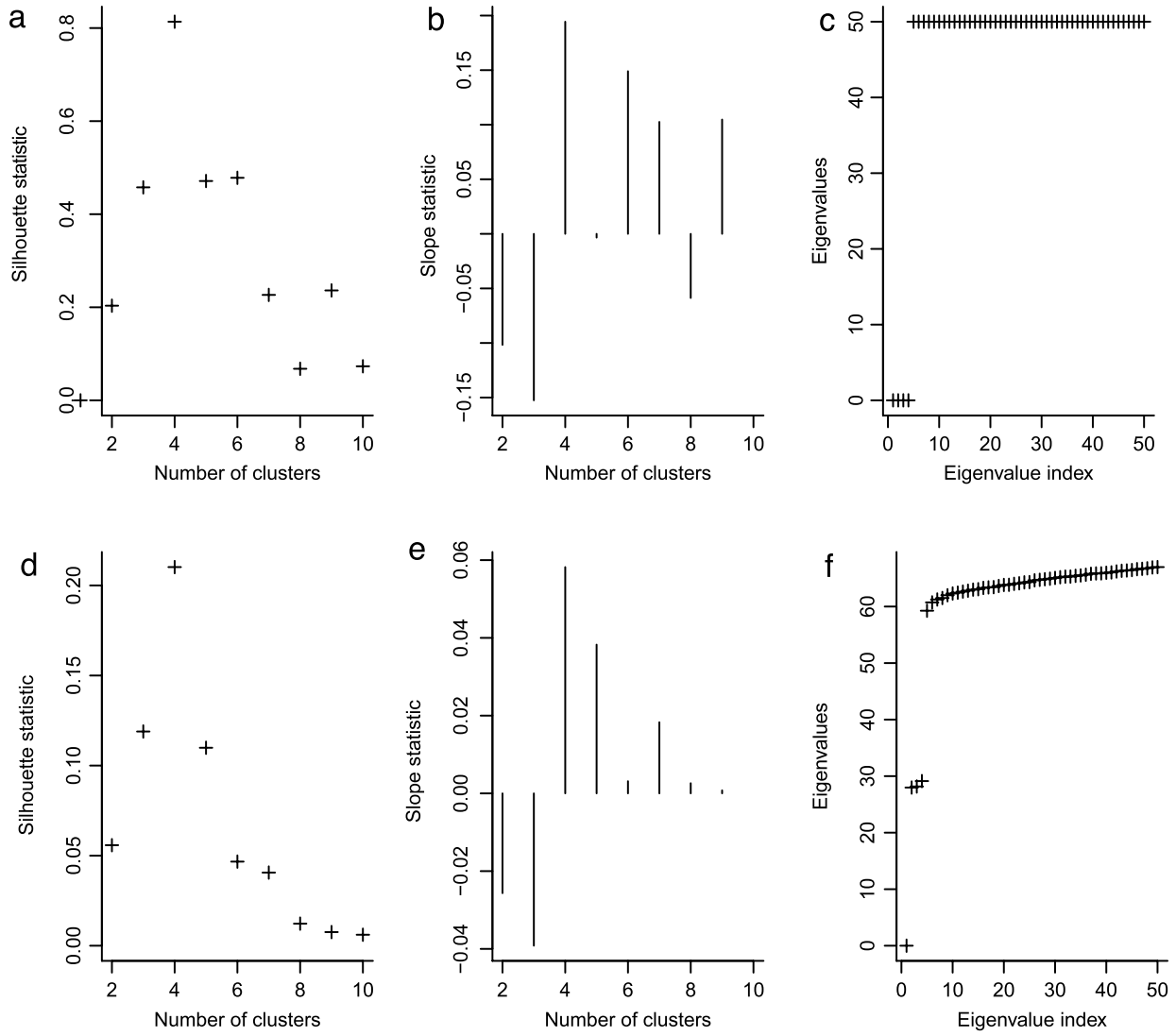

Fig. 3. Panels (a)–(c) represent the plots for the cluster value index, the slope statistic for $p = 1$ and the eigenvalue method, respectively, for scenario 1—*four independent sub-graphs*. Panels (d)–(f) represent the plots for the cluster value index, the slope statistic for $p = 1$ and the eigenvalue method, respectively, for scenario 2—*four sub-graphs connected by edges*.

One may argue how to choose the k when the slope statistics for different numbers of clusters are similar. In this case, we suggest to use a bootstrap approach and select the k with the highest frequency.

In addition to the methods studied here, there are also other methods based on the concept of stability of a cluster, such as (Fowlkes and Mallows, 1983; Gnanadesikan, 1997; Ben-Hur et al., 2002; Lange et al., 2004; Ben-David et al., 2006; Fang and Wang, 2012) that are interesting for further future studies.

Another point for discussion is how to choose the tuning variable p. As a heuristic, if there is any *a priori* reason for the clusters to be homogeneous between them, a larger p would be recommended, whereas a smaller p would be more adequate if a dominant cluster is expected. Our simulations show that, although the optimum value can vary slightly, the choice of $p = 1$ performed well in all examples considered in this article. Therefore, with the lack of any good reason, we recommend $p = 1$.

In this article, we have studied how the slope method combined with the silhouette statistic can increase the efficiency of correctly identifying the number of clusters in items generated by different probability distributions and in the graph partition problem. Illustrations in standard biological examples are provided.

Acknowledgments

AF was partially supported by FAPESP (11/07762-8) and CNPq (306319/2010-1). DYT was partially supported by Pew Latin American Fellowship.

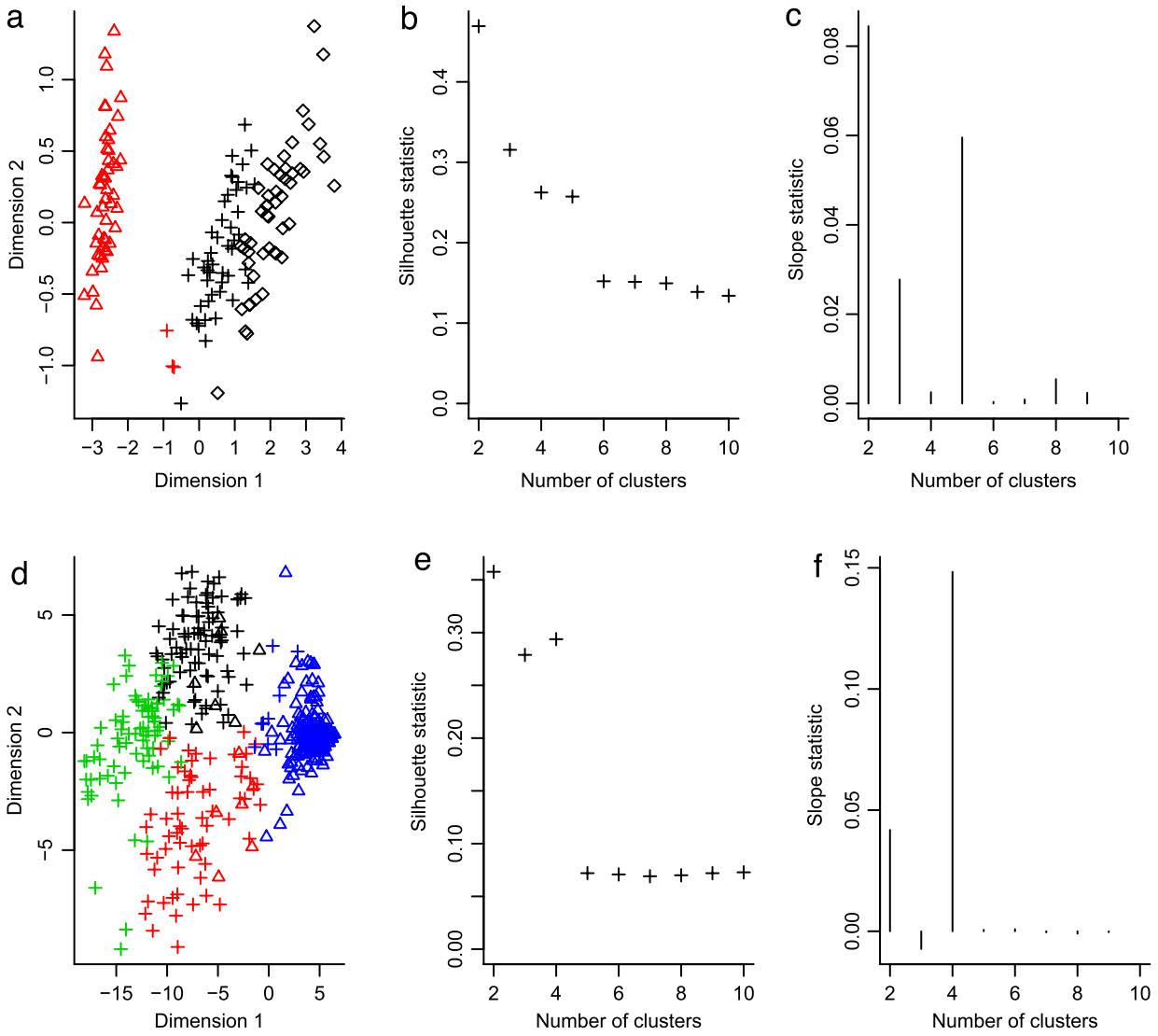

Fig. 4. k-means clustering results. The colors in panels (a) and (d) represent the number of clusters and the respective clusters identified by the slope method with $p = 1$ and the k-means clustering, respectively. The characters (triangles, crosses and squares) represent the original data classes. Panels (a)–(c) represent the 2D plot of the iris data, the silhouette statistic ($s(k)$) and the slope statistic for the iris data. Panels (d)–(f) represent the 2D plot of the breast data, the silhouette statistic and the slope statistic for the breast data. In panel (a), the triangles, crosses and squares represent the setosa, versicolor and virginica iris, respectively. In panel (d), triangles and crosses represent the benign and malign breast data, respectively.

References

Anderson, W.F., Matsuno, R., 2006. Breast cancer heterogeneity: a mixture of at least two main types? J. Nat. Cancer Inst. 98, 948–951.
Ben-David, S., von Luxburg, U., Pal, D., 2006. A sober look at stability of clustering. In: 19th Annual Conference on Learning Theory, COLT.
Ben-Hur, A., Elisseeff, A., Guyon, I., 2002. A stability based method for discovering structure in clustered data. In: Pacific Symposium on Biocomputing, Vol. 7. pp. 6–17.
Calinski, R.B., Harabasz, J., 1974. A dendrite method for cluster analysis. Commun. Stat. 3, 1–27.
Celeux, G., Govaert, G., 1992. A classification EM algorithm for clustering and two stochastic versions. Comput. Statist. Data Anal. 14, 315–332.
Celeux, G., Govaert, G., 1995. Gaussian parsimonious clustering models. Pattern Recognit. 28, 781–793.
Dudoit, S., Fridlyand, J., 2002. A prediction-based resampling method for estimating the number of clusters in a dataset. Genome Biol. 3, 7.
Fang, Y., Wang, J., 2012. Selection of the number of clusters via the bootstrap method. Comput. Statist. Data Anal. 56, 468–477.
Fisher, R.A., 1936. Multiple measurements in taxonomic problems. Ann. Eugenics VII, 179–188.
Fowlkes, E.B., Mallows, C.L., 1983. A method for comparing two hierarchical clusterings. J. Amer. Statist. Assoc. 78, 553–569.
Gnanadesikan, R., 1997. Methods for Statistical Data Analysis of Multivariate Observations, second ed. John Wiley & Sons, Inc., New York.
He, Z., Cichocki, A., Xie, S., Choi, K., 2010. Detecting the number of clusters in n-way probabilistic clustering. IEEE Trans. Pattern Anal. Mach. Intell. 21, 2006–2021.
Krzanowski, W.J., Lai, Y.T., 1985. A criterion for determining the number of clusters in a data set. Biometrics 44, 23–34.
Lange, T., Roth, V., Braun, M., Buhmann, J., 2004. Stability-based validation of clustering solutions. Neural Comput. 16, 1299–1323.
Luxburg, U., 2007. A tutorial on spectral clustering. Stat. Comput. 17, 395–416.
Marriot, F.H.C., 1971. Practical problems in a method of cluster analysis. Biometrics 27, 501–514.
Milligan, G.W., Cooper, M.C., 1985. An examination of procedures for determining the number of clusters in a data set. Psychometrika 50, 159–179.
R Development Core Team, 2010. R: a language and environment for statistical computing. Vienna, Austria. ISBN: 3-900051-07-0. http://www.R-project.org.
Rousseeuw, P., 1987. Sihouettes: a graphical aid to the interpretation and validation of cluster analysis. J. Comput. Appl. Math. 20, 53–65.
Sato, J.R., Fujita, A., Amaro Jr., E., Mourão-Miranda, J., Morettin, P.A., Brammer, M.J., 2007. DWT-CEM: an algorithm for scale-temporal clustering in fMRI. Biol. Cybernet. 97, 33–45.

Sugar, C.A., James, G.M., 2003. Finding the number of clusters in a data set—an information theoretic approach. J. Amer. Statist. Assoc. 98, 750–763.
Tibshirani, R., Walther, G., 2005. Cluster validation by prediction strength. J. Comput. Graph. Statist. 14, 511–528.
Tibshirani, R., Walther, G., Hastie, T., 2001. Estimating the number of clusters in a data set via the gap statistic. J. R. Stat. Soc. Ser. B Stat. Methodol. 63, 411–423.
Wolberg, W.H., Mangasarian, O.L., 1990. Multisurface method of pattern separation for medical diagnosis applied to breast cytology. Proc. Natl. Acad. Sci. 87, 9193–9196.
Xiang, T., Gong, S., 2008. Spectral clustering with eigenvector selection. Pattern Recognit. 41, 1012–1029.
Yin, Z., Zhou, X., Bakal, C., Li, F., Sun, Y., Perrimon, N., Wong, S., 2008. Using iterative cluster merging with improved gap statistics to perform online phenotype discovery in the context of high-throughput RNAi screens. BMC Bioinformatics 9, 264.
Zang, C., Chen, B., 2010. Automatic estimation the number of clusters in hierarchical data clustering. In: Mechatronics and Embedded Systems and Applications, MESA. pp. 269–274.