

Study of Overlapping Clustering Algorithms Based on Kmeans through FBcubed Metric

Argenis A. Aroche-Villarruel¹, J.A. Carrasco-Ochoa¹, José Fco. Martínez-Trinidad¹,
J. Arturo Olvera-López², and Airl Pérez-Suárez³

¹ Computer Science Department, Instituto Nacional de Astrofísica, Óptica y Electrónica,
Luis Enrique Erro No. 1, Sta. María Tonantzintla, Puebla, CP: 72840, México
{argenis, ariel, fmartine}@inaoep.mx

² Faculty of Computer Sciences, Benemérita Universidad Autónoma de Puebla,
Av. San Claudio y 14 sur. Ciudad Universitaria. Puebla, Pue., México
aolvera@cs.buap.mx

³ Advanced Technologies Application Center, 7ma A #21406 e/ 214 y 216, Rpto. Siboney,
Playa. C.P. 12200. La Habana, Cuba
asuarez@cenatav.co.cu

Abstract. In this paper we present a study of the overlapping clustering algorithms OKM, WOKM and OKMED, which are extensions to the overlapping case of the well known Kmeans algorithm proposed for building partitions. Different to other previously reported comparisons, in our study we compare these algorithms using the external evaluation metric FBcubed which takes into account the overlapping among clusters and we contrast our results against those obtained by F-measure, a metric that does not take into account the overlapping among clusters and that has been previously used in another reported comparison.

Keywords: Clustering, Overlapping Clustering, Clustering Validation.

1 Introduction

Among the clustering algorithms, those building overlapping clusters are useful in different applications, where it is common that objects belong to more than one cluster. Some examples of this kind of applications are information retrieval [1], social network analysis [2], text segmentation [3], among others. In the literature, several algorithms have been proposed for overlapping clustering [4, 5, 6, 7, 8, 9, 10, 11], which are different according to their mathematical basis and clustering strategies, as well as the type of datasets they can process.

Due to their simplicity, the Kmeans algorithm [12] together with its variants are clustering algorithms that have been widely used in several applications. However, since these algorithms do not produce overlapping clusterings therefore, they could be non suitable for applications needing this kind of clustering. The algorithms OKM [8], WOKM [9] and OKMED [9], have been proposed as extensions to the overlapping clustering case of the Kmeans algorithm [12], weighting Kmeans [13], and k-medoids (Partitioning Around Medoids) algorithms [14], respectively.

In [9] an experimental evaluation of these algorithms is reported; however, in this work the efficacy of the algorithms was assessed using F-measure that is not defined for evaluating overlapping clustering. Taking into account that some metrics have been recently proposed for evaluating overlapping clusterings, we perform an experimental evaluation of the above commented algorithms employing these metrics. In this way, we will be able to know how good really are these algorithms for overlapping clustering problems.

This paper is organized as follows: Section 2 briefly describes the algorithms OKM, WOKM and OKMED. Section 3 reports the experimental study in which we compare these algorithms using some standard overlapping datasets. Finally, in section 4 some conclusions are presented.

2 OKM, WOKM and OKMED Algorithms

The problem of clustering a set of objects $X = \{x_1, x_2, \dots, x_n\}$ in k clusters (a priori parameter), using the Kmeans algorithm [12] is formulated as an optimization problem, where the objective function (1) is minimized.

$$Q(\pi) = \sum_{j=1}^k \sum_{x_i \in \pi_j} d^2(x_i, z_j) \quad (1)$$

being $\pi = \{\pi_1, \dots, \pi_k\}$ is a set of k clusters, $\pi_i \cap \pi_j = \emptyset$, for $i \neq j$; $Z = \{z_1, \dots, z_k\}$ is a set such that z_i is the centroid of π_i , for $i = 1, \dots, k$; and $d(x_i, z_j)$ is the Euclidean distance between objects x_i and z_j .

2.1 OKM

The OKM algorithm [8], extends the objective function used in Kmeans [12], to consider the possibility of overlapping clusters. In this algorithm, the objective function is defined as in (2).

$$Q'(\pi) = \sum_{i=1}^n d^2(x_i, \phi(x_i)) \quad (2)$$

where π, Z and d are as in (1), removing the condition of $\pi_i \cap \pi_j = \emptyset$, for $i \neq j$; $\phi(x_i)$ is the “image” of x_i , which is defined as a combination of the centroids (z_j) of the clusters π_j where x_i belongs to, computed as in (3).

$$\phi(x_i) = (\phi_1(x_i), \dots, \phi_p(x_i)) \text{ with } \phi_v(x_i) = \frac{\sum_{z_j \in A_i} z_{j,v}}{|A_i|} \quad (3)$$

where $A_i = \{z_j | x_i \in \pi_j\}$

There are two important differences between Kmeans and OKM. The first one is in the way the objects are assigned to one or more clusters and, the second, is in the way the centroid of each cluster is computed. The assignment step for an object x_i , consists in scrolling through the list of centroids, from the nearest to the farthest, and assigning x_i to the respective cluster while $d(x_i, \phi(x_i))$ decreases. The new assignment list is stored only if it is better than the previous one.

The centroid z_j for the cluster π_j is updated as follows:

$$z_{j,v} = \frac{1}{\sum_{x_i \in \pi_j} \frac{1}{\delta_i^2}} \sum_{x_i \in \pi_j} \frac{1}{\delta_i^2} \cdot \tau_{i,v}^j \quad (4)$$

where: $z_{j,v}$ denotes the v -th feature of the centroid z_j , δ_i is the number of clusters to which x_i belongs to ($\delta_i = |A_i|$) and $\tau_{i,v}^j$ is computed as follows:

$$\tau_{i,v}^j = \delta_i \times x_{i,v} - \sum_{z_j \in A_i / \{z_i\}} z_{j,v} \quad (5)$$

where A_i is as in (3).

2.2 WOKM

In WOKM [9] the objective function of weighting-Kmeans [13] is extended to take into account feature weights into each cluster; therefore, it is necessary to redefine the concept of "image". In this algorithm the image for x_i is defined through a weighted average of the cluster centroids for x_i as follows:

$$\phi(x_i) = (\phi_1(x_i), \dots, \phi_p(x_i)) \text{ with } \phi_v(x_i) = \frac{\sum_{z_j \in A_i} \lambda_{j,v}^\beta z_{j,v}}{\sum_{z_j \in A_i} \lambda_{j,v}^\beta} \quad (6)$$

where $z_{j,v}$ is defined as in (4); $\lambda_{j,v} \in [0, 1]$ denotes the weight associated to the feature v in the cluster j (initially this value is $1/p$); β is a parameter ($\beta > 1$) that regulates the influence of the weights in the algorithm.

The vector of weights γ_i for the images $\phi(x_i)$ is defined as follows:

$$\gamma_{i,v} = \frac{\sum_{z_j \in A_i} \lambda_{j,v}}{|A_i|} \quad (7)$$

From this definition the objective function for the algorithm WOKM is given by:

$$Q''(\pi) = \sum_{x_i \in X} \sum_{v=1}^p \gamma_{i,v}^\beta |x_{i,v} - \phi_v(x_i)|^2 \quad (8)$$

where π and Z are as in (1).

The assignment step is similar to the corresponding step in the OKM algorithm, i.e., an object is assigned to its nearest clusters while $\sum_{v=1}^p \gamma_{i,v}^\beta |x_{i,v} - \phi_v(x_i)|^2$ decreases.

The new centroid z_j^* for the cluster π_j is obtained from the set $\{(\tau_i^j, w_i) | x_i \in \pi_j\}$; τ_i^j like in OKM, allows those objects that belong to more clusters to have less impact on the position of the new centroid; w_i denotes the weight vector and it is defined as follows:

$$w_{i,v} = \frac{\gamma_{i,v}^\beta}{(\sum_{z_l \in A_i} \lambda_{l,v}^\beta)^2} \quad (9)$$

For computing the weights WOKM introduces a heuristic based in the one proposed in [15]; the heuristic for each class consists in:

- Computing a new weight $\lambda_{j,v}$ for the cluster π_j by estimating on each feature the variance of the objects that belong only to π_j :

$$\lambda_{j,v} = \frac{(\sum_{\{x_i \in \pi_j \mid |A_i|=1\}} (x_{i,v} - z_{j,v})^2)^{1/(1-\beta)}}{\sum_{u=1}^p (\sum_{\{x_i \in \pi_j \mid |A_i|=1\}} (x_{i,v} - z_{j,v})^2)^{1/(1-\beta)}} \quad (10)$$

- Store the weight only if it improves the objective function (8).

2.3 OKMED

OKMED [9] is based on the k-medoids algorithm, which uses medoids instead of centroids. With this purpose, the objective function is extended from the OKM algorithm, so that we can use any dissimilarity function between objects. Assuming that $d(\cdot)$ is a dissimilarity function from $X \times X \rightarrow \mathbb{R}^+$, the objective function for OKMED is given by (2).

The notion of image was redefined using cluster medoids instead of centroids. The image $\phi(x_i)$ of x_i in the cluster π_j is then defined as the object from X that minimizes the sum of the dissimilarities with all the medoids of the clusters where x_i belongs to:

$$\phi(x_i) = \arg \min_{x_j \in X} \sum_{z_l \in A_i} d^2(x_j, z_l) \quad (11)$$

Notice that, in this new definition, the computation of an image requires to test all the objects in the collection.

The assignment of an object to one or more clusters is done in the same way as in OKM, but using a medoid for each cluster instead of a centroid.

For updating the medoid OKMED tests each object x_i in the cluster π_j , until finding the first object that improves the objective function with respect to the current medoid, that object will be the new medoid for π_j .

3 Experimental Analysis

For our study we propose to use the FBcubed validation metric [16], since this, unlike most external metrics reported in the literature, allows to evaluate overlapping clustering algorithms. Additionally, we compare our results against those results reported in [9]. The clustering algorithms were programmed in ANSI C.

F-measure combines Precision and Recall taking into account a labeled dataset and the result of clustering the same dataset. Let N_C be the set of pairs of objects belonging to the same class (same label) and N_π the set of pairs of objects belonging to the same cluster, the Precision, Recall and F-measure metrics are defined as:

$$Precision = \frac{|N_{\pi} \cap N_C|}{|N_{\pi}|} \quad (12)$$

$$Recall = \frac{|N_{\pi} \cap N_C|}{|N_C|} \quad (13)$$

$$F - measure = \frac{2 \times Precision \times Recall}{Precision + Recall} \quad (14)$$

FBcubed is calculated using the Bcubed Precision and Bcubed Recall metrics as proposed in [16]. The Bcubed Precision and Bcubed Recall are based on the Multiplicity Precision and Multiplicity Recall metrics respectively; which are defined as:

$$Multiplicity\ Precision(x_i, x_j) = \frac{Min(|\pi(x_i) \cap \pi(x_j)|, |C(x_i) \cap C(x_j)|)}{|\pi(x_i) \cap \pi(x_j)|} \quad (15)$$

$$Multiplicity\ Recall(x_i, x_j) = \frac{Min(|\pi(x_i) \cap \pi(x_j)|, |C(x_i) \cap C(x_j)|)}{|C(x_i) \cap C(x_j)|} \quad (16)$$

where x_i and x_j are two objects, $C(x_i)$ are the classes associated to x_i , $\pi(x_i)$ are the clusters associated to x_i . These formulas are only defined when x_i and x_j share at least one cluster (15) and when they share at least one class (16).

Let $D(x_i)$ be the set of objects that share at least one cluster with x_i including x_i . The Bcubed Precision metric of x_i is defined as:

$$Bcubed_{Precision}(x_i) = \frac{\sum_{x_j \in D(x_i)} Multiplicity\ Precision(x_i, x_j)}{|D(x_i)|} \quad (17)$$

Let $H(x_i)$ be the set of objects that share at least one class with x_i including x_i . The Bcubed Recall metric of x_i is defined as:

$$Bcubed_{Recall}(x_i) = \frac{\sum_{x_j \in H(x_i)} Multiplicity\ Recall(x_i, x_j)}{|H(x_i)|} \quad (18)$$

Finally, the FBcubed metric is defined as follows:

$$FBcubed = \frac{2 \left(\frac{1}{n} \sum_i^n Bcubed_{Precision}(x_i) \right) \left(\frac{1}{n} \sum_i^n Bcubed_{Recall}(x_i) \right)}{\left(\frac{1}{n} \sum_i^n Bcubed_{Precision}(x_i) \right) + \left(\frac{1}{n} \sum_i^n Bcubed_{Recall}(x_i) \right)} \quad (19)$$

Where n is the number of objects in the dataset.

In [16] the authors present an analysis of different external metrics for clustering evaluation, and based on this analysis they propose some constraints, that these metrics should satisfy. They conclude that FBcubed satisfies all constraints, but F-measure does not, therefore, F-measure is unable to distinguish certain undesirable situations at evaluating overlapping clusters:

In order to show an example of these situations, suppose a dataset $X = \{x_1, x_2, x_3, x_4, x_5, x_6, x_7, x_8, x_9, x_{10}\}$, which is divided in three overlapping classes

$C_1 = \{x_1, x_2, x_3, x_4, x_5, x_6\}$, $C_2 = \{x_5, x_7, x_8\}$, and $C_3 = \{x_6, x_9, x_{10}\}$. Now suppose that two clustering algorithms obtained the following clusters:

$$\begin{aligned} \pi_{1,1} &= \{x_1, x_2, x_3, x_4, x_5, x_6\}, & \pi_{1,2} &= \{x_6, x_7, x_8\}, & \pi_{1,3} &= \{x_5, x_9, x_{10}\} \\ \pi_{2,1} &= \{x_1, x_2, x_3, x_4, x_5, x_6\}, & \pi_{2,2} &= \{x_5, x_7, x_{10}\}, & \pi_{2,3} &= \{x_6, x_9, x_8\} \end{aligned}$$

Both results have two objects in a wrong cluster but in $\pi_{1,2}$ and $\pi_{1,3}$ the error is in the objects x_5, x_6 , which belong to the overlapping with $\pi_{1,1}$. While in $\pi_{2,2}$ and $\pi_{2,3}$ the error is in the objects x_8, x_{10} , which belong to only one cluster, clearly both errors are different. However, when the assessment is done through F-measure in both cases we get the same result (0.8095), while if we do the assessment through Fbcubed it allows to distinguish these different errors, obtaining 0.8166 and 0.7750; indicating that the first clustering algorithm obtains a better result than the second one. Based on the situations above commented it is noteworthy that F-measure may not be adequate for evaluating overlapping clusters; as it will be analyzed in the following experiments.

For our experiments, we used three datasets taken from the MULAN repository¹ (see table 1). These data sets were chosen because they are the same used in [9] where the clustering algorithms OKM, WOKM, and OKMED were compared through F-measure.

Table 1. Description of datasets used in the experiments

<i>Name</i>	<i>Domain</i>	<i>#Objects</i>	<i>#Features</i>	<i>#Labels</i>	<i>Overlapping</i>
<i>Emotions[17]</i>	<i>Music</i>	593	72	6	1.869
<i>Scene[18]</i>	<i>Image</i>	2307	294	6	1.074
<i>Yeast[19]</i>	<i>Biology</i>	2417	103	14	4.237

Since the WOKM algorithm uses the β parameter to regulate the influence of feature weights. In order to study the behaviour of β , as first experiment, several values, 1.1, 1.5, 1.9, 2, 2.1, 2.5 and 3, for β were tested, these values were used in [13]. Table 2 shows the results of F-measure (FM), Fbcubed (FBC), and the relative overlapping² (RO) of these experiments, the best result for each column is boldfaced.

From Table 2, it can be seen that a higher value of the parameter β does not necessarily mean a better result. According to F-measure, the values of β that produced the best results were 3 for Emotions and Scene, and 2.5 for Yeast. But, according to Fbcubed the best values are 3, 1.5 and 2.5, respectively.

¹ <http://mulan.sourceforge.net/datasets.html>

² Relative overlapping is computed dividing the overlapping of the clustering result by the overlapping of the original dataset. Values close to 1 mean that the overlapping built by the clustering algorithm is close to the original overlapping, values greater than 1 mean that the original dataset has more overlapping than the overlapping built by the clustering algorithm and values lesser than 1 represent the opposite.

Table 2. F-measure (FM) and FBcubed (FBC) results of the algorithm WOKM for different values of the parameter β , the relative overlapping (RO) is also reported

β	<i>Emotions</i>			<i>Scene</i>			<i>Yeast</i>		
	<i>FM</i>	<i>FBC</i>	<i>RO</i>	<i>FM</i>	<i>FBC</i>	<i>RO</i>	<i>FM</i>	<i>FBC</i>	<i>RO</i>
1.1	0.5575	0.4957	1.127	0.3528	0.2701	2.385	0.8150	0.6698	1.077
1.5	0.4549	0.4227	0.651	0.3342	0.2935	1.826	0.8149	0.6698	1.078
1.9	0.5423	0.5146	0.688	0.3545	0.2727	2.383	0.8146	0.6694	1.077
2	0.5624	0.5265	0.833	0.3545	0.2727	2.383	0.8150	0.6698	1.077
2.1	0.5897	0.5607	0.658	0.3548	0.2743	2.373	0.8157	0.6706	1.079
2.5	0.5803	0.5557	0.610	0.3558	0.2754	2.376	0.8167	0.6718	1.082
3	0.5929	0.5653	0.610	0.3574	0.2786	2.370	0.8150	0.6700	1.078

Analyzing with more detail the results in Table 2, for the Scene dataset, we can observe that $\beta = 1.5$ produces the worst clustering result according to F-measure, while for the same value of β , according to FBcubed, it obtains the best result. In order to better understand what is happening, notice that the clustering result evaluated through F-measure for $\beta = 1.1$ is 0.3528, while for $\beta = 1.5$ is 0.3342, and the relative overlapping obtained for $\beta = 1.1$ is 2.385 and for $\beta = 1.5$ is 1.826, i.e. F-measure gives a better evaluation in the case where more overlapping is obtained. This is a clear example of what happens when the clustering built by an algorithm has a higher rate of overlapping with respect to the overlapping in the original classes. If there is a high overlapping in the clusters, it means objects belong to more than one cluster; increasing the intersection of object pairs in the same cluster and class, and consequently, it makes the Recall increases. Precision is affected only by the number of pairs of objects belonging to the same cluster³. Conversely, if the overlapping is not high, Recall will have a lower value and Precision will only be slightly higher compared to the high overlapping scenario. It explains why F-measure gets a higher values (better results) for $\beta = 1.1$ in comparison to $\beta = 1.5$. Contrarily, FBcubed obtains better results for $\beta = 1.5$ in comparison to $\beta = 1.1$, since it is formulated for considering the amount of classes and clusters to which each object belongs instead of only taking into account if they are in the same cluster or class.

We present another experiment, reported in table 3, for comparing WOKM, OKM and OKMED algorithms using F-measure (FM) and Fbcubed (FBC). For WOKM we report the best result obtained in the previous experiment after testing different values of β , for OKMED we use as dissimilarity function the Euclidean distance. For the three algorithms the parameter k was set as the number of classes in the dataset. In order to have a fair comparison, the same seeds were used in the initialization of each algorithm. In Table 3, also the relative overlapping (RO) obtained by each algorithm is reported.

³ The same happens when the classes have a higher rate of overlapping with respect to the clusters built by an algorithm, Precision will be higher and Recall will be lower.

Table 3. Values obtained for the OKM, WOKM, and OKMED algorithms on all datasets with the F-measure (FM) and FBCubed (FBC) metrics. The relative overlapping (RO) is reported also.

<i>Dataset</i>	<i>OKM</i>			<i>WOKM</i>			<i>OKMED</i>		
	<i>FM</i>	<i>FBC</i>	<i>RO</i>	<i>FM</i>	<i>FBC</i>	<i>RO</i>	<i>FM</i>	<i>FBC</i>	<i>RO</i>
<i>Emotions</i>	0.5575	0.4957	1.1268	0.5929	0.5653	0.6096	0.5013	0.4588	0.9770
<i>Scene</i>	0.3552	0.2744	2.3766	0.3574	0.2935	1.8264	0.3749	0.3689	1.6855
<i>Yeast</i>	0.8146	0.6693	1.0764	0.8167	0.6718	1.0817	0.2029	0.1102	0.2679
<i>Average</i>	0.5758	0.4798	1.5266	0.5890	0.5102	1.1726	0.3597	0.3126	0.9768

In Table 3, we can see that the only dataset where OKMED obtained the best result compared to OKM and WOKM was Scene dataset, contrarily to the results reported in [9] where the author reports that both OKMED and OKM obtained a similar result for this dataset, since he used F-measure and in terms of this metric the F-measure results didn't show a great difference. However in our experiments evaluating using FBCubed a different result is obtained, and clearly OKMED outperforms OKM and WOKM. Notice that Scene is a dataset with little overlapping (1.074) and OKMED obtains less overlapping than the other clustering algorithms. From our experiments we can see that OKMED builds clusterings with low overlapping therefore it is a good algorithm at those datasets where we expect a clustering with low overlapping. From this experiment we also can see that WOKM algorithm obtained the best results in average. From this, we can deduce that the use of weights helps to get better results in most cases, the problem with WOKM algorithm lies in finding an adequate value for β that allows getting a good result, and if it is not possible to determine what value of β is the best, possibly WOKM is not the best choice. Finally the OKM algorithm is a good choice since it only requires, as input, the number of clusters, moreover, in terms of quality OKM results are very close to the results obtained by WOKM and both algorithms are good at datasets where we expect a clustering with high overlapping.

4 Conclusions

This paper presents a study of the overlapping clustering algorithms OKM, WOKM and OKMED, which are based on the Kmeans algorithm. Different to other previously reported comparisons, in our study we compare these algorithms using the external evaluation metric FBCubed which takes into account the overlapping among clusters. From our experiments we can conclude that in general WOKM algorithm obtains the best results in comparison to OKM and OKMED however it is not easy to finding out an adequate value for β parameter, while OKM results are very close to the results obtained by WOKM and it only requires, as input, the number of clusters, moreover in terms of overlapping both algorithms produce similar clusterings with high

overlapping. On the other hand, OKMED builds clusterings with low overlapping but unlike OKM and WOKM it has the characteristic that can use any dissimilarity function.

Finally, and the most important, we can conclude that for evaluating overlapping clustering algorithms a metric that takes into account the overlapping must be used, since the use of other metrics as F-measure cannot correctly evaluate the results obtained by this kind of algorithms.

Acknowledgment. This work was partly supported by the National Council of Science and Technology of Mexico (CONACyT) through the project grants CB2008-106443 and CB2008-106366; and the scholarship grant 362371.

References

1. Aslam, J., Pelehov, E., Rus, D.: The star clustering algorithm for static and dynamic information organization. *Journal of Graph Algorithms and Applications* 8(1), 95–129 (2004)
2. Davis, G., Carley, K.: Clearing the FOG: Fuzzy, overlapping groups for social networks. *Social Networks* 30(3), 201–212 (2008)
3. Abella-Pérez, R., Medina-Pagola, J.E.: An incremental text segmentation by clustering cohesion. In: *Proceedings of HaCDAIS 2010*, pp. 65–72 (2010)
4. Pons-Porrata, A., Ruiz-Shulcloper, J., Berlanga-Llavorí, R., Santiesteban-Alganza, Y.: Un algoritmo incremental para la obtención de cubrimientos con datos mezclados. In: *Proceedings of CIARP 2002*, pp. 405–416 (2002)
5. Zamir, O., Etziony, O.: Web document clustering: A feasibility demonstration. In: *Proceedings of the 21st Annual International ACM SIGIR Conference*, pp. 46–54 (1998)
6. Aslam, J., Pelehov, E., Rus, D.: Static and dynamic information organization with star clusters. In: *Proceedings of the Seventh International Conference on Information and Knowledge Management*, pp. 208–217 (1998)
7. Gil-García, R.J., Badía-Contelles, J.M., Pons-Porrata, A.: Extended star clustering algorithm. In: Sanfeliu, A., Ruiz-Shulcloper, J. (eds.) *CIARP 2003*. LNCS, vol. 2905, pp. 480–487. Springer, Heidelberg (2003)
8. Cleuziou, G.: An Extended Version of the k-Means Method for Overlapping Clustering. In: *19th ICPR Conference, Tampa, Florida, USA*, pp. 1–4 (2008)
9. Cleuziou, G.: Two Variants of the OKM for Overlapping Clustering. In: Guillet, F., Ritschard, G., Zighed, D.A., Briand, H. (eds.) *Advances in Knowledge Discovery and Management*. SCI, vol. 292, pp. 149–166. Springer, Heidelberg (2010)
10. Gil-García, R.J., Badía-Contelles, J.M., Pons-Porrata, A.: Parallel algorithm for extended star clustering. In: Sanfeliu, A., Martínez Trinidad, J.F., Carrasco Ochoa, J.A. (eds.) *CIARP 2004*. LNCS, vol. 3287, pp. 402–409. Springer, Heidelberg (2004)
11. Suárez, A.P., Pagola, J.E.M.: A clustering algorithm based on generalized stars. In: Perner, P. (ed.) *MLDM 2007*. LNCS (LNAI), vol. 4571, pp. 248–262. Springer, Heidelberg (2007)
12. MacQueen, J.B.: Some Methods for classification and Analysis of Multivariate Observations. In: *Proceedings of 5th Berkeley Symposium on Mathematical Statistics and Probability*, pp. 281–297. University of California Press (1967)
13. Chan, E.Y., Ching, W.K., Ng, M.K., Huang, J.Z.: An optimization algorithm for clustering using weighted dissimilarity measures. *Pattern Recognition* 37(5), 943–952 (2004)

14. Kaufman, L., Rousseeuw, P.J.: Clustering by means of medoids. In: Dodge, Y. (ed.) *Statistical Data Analysis based on the L1 Norm*, pp. 405–416 (1987)
15. Bezdek, J.C.: *Pattern Recognition with Fuzzy Objective Function Algorithms*. Plenum Press, New York (1981)
16. Amigó, E., Gonzalo, J., Artiles, J., Verdejo, F.: A comparison of extrinsic clustering evaluation metrics based on formal constraints. *Information Retrieval*, 461–486 (2009)
17. Trohidis, K., Tsoumakas, G., Kalliris, G., Vlahavas, I.: Multilabel Classification of Music into Emotions. In: *Proc. 2008 International Conference on Music Information Retrieval (ISMIR 2008)*, pp. 325–330 (2008)
18. Boutell, M.R., Luo, J., Shen, X., Brown, C.M.: Learning multi-label scene classification. *Pattern Recognition* 37(9), 1757–1771 (2004)
19. Elisseeff, A., Weston, J.: A kernel method for multi-labelled classification. In: *Advances in Neural Information Processing Systems*, vol. 14 (2002)