To appear in Proc. IAPR International Conference on Pattern Recognition, Cambridge, UK, 2004

Landscape of Clustering Algorithms

Anil K. Jain, Alexander Topchy, Martin H.C. Law, and Joachim M. Buhmann[§]
Department of Computer Science and Engineering,
Michigan State University, East Lansing, MI, 48824, USA
[§] *Institute of Computational Science, ETH Zentrum, HRS F31*
Swiss Federal Institute of Technology ETHZ, CH-8092, Zurich, Switzerland
{jain, topchyal, lawhiu}@cse.msu.edu, jbuhmann@inf.ethz.ch

Abstract

Numerous clustering algorithms, their taxonomies and evaluation studies are available in the literature. Despite the diversity of different clustering algorithms, solutions delivered by these algorithms exhibit many commonalities. An analysis of the similarity and properties of clustering objective functions is necessary from the operational/user perspective. We revisit conventional categorization of clustering algorithms and attempt to relate them according to the partitions they produce. We empirically study the similarity of clustering solutions obtained by many traditional as well as relatively recent clustering algorithms on a number of real-world data sets. Sammon's mapping and a complete-link clustering of the inter-clustering dissimilarity values are performed to detect a meaningful grouping of the objective functions. We find that only a small number of clustering algorithms are sufficient to represent a large spectrum of clustering criteria. For example, interesting groups of clustering algorithms are centered around the graph partitioning, linkage-based and Gaussian mixture model based algorithms.

1. Introduction

The number of different data clustering algorithms reported or used in exploratory data analysis is overwhelming. Even a short list of well-known clustering algorithms can fit into several sensible taxonomies. Such taxonomies are usually built by considering: (i) the input data representation, e.g. pattern-matrix or similarity-matrix, or data type, e.g. numerical, categorical, or special data structures, such as rank data, strings, graphs, etc., (ii) the output representation, e.g. a partition or a hierarchy of partitions, (iii) probability model used (if any), (iv) core search (optimization) process, and (v) clustering direction, e.g. agglomerative or divisive. While many other dichotomies are also possible, we are more concerned with effective guidelines for a choice of clustering algorithms based on their objective functions [1]. It is the objective function that determines the output of the clustering procedure for a given data set.

Intuitively, most of the clustering algorithms have an underlying objective function that they try to optimize. The objective function is also referred to as a clustering criterion or cost function. The goal of this paper is a characterization of the landscape of the clustering algorithms in the space of their objective functions. However, different objective functions can take drastically different forms and it is very hard to compare them analytically. Also, some clustering algorithms do not have explicit objective functions. Examples include mean-shift clustering [13] and CURE [11]. However, there is still the notion of optimality in these algorithms and they possess their objective functions, albeit defined implicitly. We need a procedure to compare and categorize a variety of clustering algorithms from the viewpoint of their objective functions.

One possible approach for designing this landscape is to derive the underlying objective function of the known clustering algorithms and the corresponding general description of clustering solutions. For example, it was recently established [2,3] that classical agglomerative algorithms, including single-link (SL), average-link (AL) and complete-link (CL), have quite complex underlying probability models. The SL algorithm is represented by a mixture of branching random walks, while the AL algorithm is equivalent to finding the maximum likelihood estimate of the parameters of a stochastic process with Laplacian conditional probability densities. In most instances, the transformation of a heuristic-based algorithm to an optimization problem with a well-defined objective function (e.g. likelihood function) deserves a separate study. Unfortunately, given the variety of ad hoc rules and tricks used by many clustering algorithms, this approach is not feasible.

We propose an alternative characterization of the landscape of the clustering algorithms by a direct comparative analysis of the clusters they detect. The similarity between the objective functions can be estimated by the similarities of the clustering solutions they obtain. Of course, such an empirical view of the clustering landscape depends on the data sets used to compute the similarity of the solutions. We study two important scenarios: (i) average-case landscape of the variety of clustering algorithms over a number of real-world data sets, and (ii) a landscape over artificial data sets generated by mixtures of Gaussian components. In both cases multidimensional scaling [14] is employed to visualize the landscape. In the case of controlled artificial data sets, we also obtain a dynamic trace of the changes in the landscape caused by varying the density and isolation of

This research was supported by ONR contract # N00014-01-1-0266 and a Humboldt Research Award.

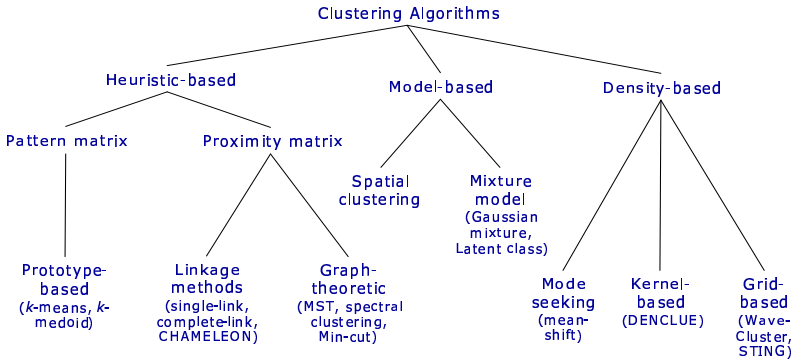

Figure 1. A possible taxonomy of clustering algorithms. Some representative algorithms in each category are named.

clusters. Unlike the previous study on this topic [1], we analyze a larger selection of clustering algorithms on many real data sets.

2. Landscape definition and computation

The number of potential clustering objective functions is arbitrarily large. Even if such functions come from a parameterized family of probability models, the exact nature of this family or the dimensionality of the parameter space is not known for many clustering algorithms. For example, the taxonomy shown in Fig. 1 cannot answer if the clustering criteria of any two selected clustering algorithms are similar. We adopt a practical viewpoint on the relationship between the clustering algorithms: distance $D(\cdot,\cdot)$ between the objective functions F_1 and F_2 on a data set X is estimated by the distance $d(\cdot,\cdot)$ between the respective data partitions $P_1(X)$ and $P_2(X)$ they produce:

$$D_X(F_1, F_2) = d(P_1(X), P_2(X))$$
$$P_i(X) = \arg\max_P F_i(P(X)).$$

Note that for some algorithms (like *k*-means), the partition that optimizes the objective function only locally is returned. Distance over multiple data sets $\{X^j\}$ is computed as:

$$\overline{D}(F_1, F_2) = \sum_j d(P_1(X^j), P_2(X^j)).$$

By performing multidimensional scaling on the *MxM* distance matrix $D_X(F_i, F_k)$ or $\overline{D}(F_i, F_k)$, *i*, *k* = 1…*M*, these clustering algorithms are represented as *M* points in a low-dimensional space, and thus can be easily visualized. We view this low-dimensional representation as the landscape of the clustering objective functions. Analysis of this landscape provides us with important clues about the clustering algorithms, since it indicates natural groupings of the algorithms by their outputs, as well as some unoccupied regions of the landscape. However, first we have to specify how the distance $d(\cdot,\cdot)$ between arbitrary partitions is computed.

While numerous definitions of distance $d(\cdot,\cdot)$ exist [4], we utilize the classical Rand's index [5] of partition similarity and Variation of Information (*VI*) distance which are both invariant w.r.t. permutations of the cluster indices. The Rand's index value is proportional to the number of pairs of objects that are assigned either to the same (n_{CC}) or different clusters ($n_{C\tilde{C}}$) in both the partitions:

$$rand = d(P_1, P_2) = \frac{n_{CC} + n_{C\tilde{C}}}{n_p},$$

where n_p is the total number of pairs of objects. The Rand's index is adjusted so that two random partitions have expected similarity of zero. It is converted to dissimilarity by subtracting from one. Performing classical scaling of the distances among all the partitions produces a visualization of the landscape. Alternatively, we compute the *VI* distance that measures the sum of "lost" and "gained" information between two clusterings. As rigorously proved in [4], the *VI* distance is a metric and it is scale-invariant (in contrast to Rand's index). Since the results using *VI* is similar to Rand's index, we omit the graphs for *VI* in this paper.

3. Selected clustering algorithms

We have analyzed 35 different clustering criteria. Only the key attributes of these criteria are listed below. The readers can refer to the original publications for more details on the individual algorithms (or objective functions). The algorithms are labeled by integer numbers in (1…35) to simplify the landscape in Fig. 2 and 3.

- Finite mixture model with Gaussian components, including four types of covariance matrix [6]: (i) Unconstrained arbitrary covariance. Different matrix for each mixture component (1), and same matrix for all the components (2). (ii) Diagonal covariance. Different matrix for each mixture component (3), same for all the components (4).
- The *k*-means algorithm (29), e.g. see [7].
- Two versions of spectral clustering algorithm [8,12] with two different parameters to select the re-scaling coefficients, resulting in four clustering criteria (31-34).
- Four linkage-based algorithms: SL (30), AL (5), CL (13) and Ward (35) distances [7].
- Seven objective functions using partitional algorithms, as implemented in CLUTO clustering program [9]:

$$\max I_1 = \sum_{i=1}^{k} \frac{S_i}{n_i} \quad (27), \quad \max I_2 = \sum_{i=1}^{k} \sqrt{S_i} \quad (28)$$

$$\min E_1 = \sum_{i=1}^{k} n_i \frac{R_i}{\sqrt{S_i}} \quad (18), \quad \min G_1 = \sum_{i=1}^{k} \frac{R_i}{S_i} \quad (19)$$

$$\min G_1' = \sum_{i=1}^{k} n_i^2 \frac{R_i}{S_i} \quad (20), \quad \max H_1 = \frac{I_1}{E_1} \quad (25), \quad \max H_2 = \frac{I_2}{E_1} \quad (26)$$

where n_i is the number of objects in cluster C_i and

$$S_i = \sum_{x,y \in C_i} sim(x,y), \quad R_i = \sum_j \sum_{x \in C_i, y \in C_j} sim(x,y).$$

- A family of clustering algorithms that combine the idea of Chameleon algorithm [10], with these seven objective functions. Chameleon algorithm uses two phases of clustering: divisive and agglomerative. Each phase can operate with an independent objective function. Here we use the *k*-means algorithm to generate a large number of small clusters and subsequently merge them to optimize one of the functions above. This corresponds to seven hybrid clustering criteria (6-12), where we keep the same order of objective functions (from Ch+I_1 to Ch+H_2).
- Four graph-based clustering criteria that rely upon min-cut partitioning procedure on the nearest-neighbor graphs [9]. Graph-based algorithms use four distance definitions that induce neighborhood graph structure: correlation coefficient (21), cosine function (22), Euclidean distance (23), and Jaccard coefficient (24).
- Four graph partitioning criteria similar to the CURE algorithm as described in [11], but with the above mentioned distance definitions (14-17).

4. Empirical study and discussion

The first part of our experiment uses real-world data sets from the UCI machine learning repository (table 1). We only consider data sets with a large number of continuous attributes. Attributes with missing values are discarded. Selected data sets include a wide range of class sizes and number of features. All the 35 clustering criteria were used to produce the corresponding partitions of the data sets. The number of clusters is set to be equal to the true number of classes in the data set. The known class labels were not in any way used during the clustering. We have considered several similarity measures to compare the partitions, though we only report the results based on the adjusted Rand's index. Sammon's mapping is applied to the average dissimilarity matrix to visualize different clustering algorithms in two-dimensional space. We have also applied classical scaling and INDSCAL scaling methods to the dissimilarity data with qualitatively similar results. Due to space limitation they are not shown.

Fig. 2(a) shows the results of Sammon's mapping performed on the 35x35 partition distance matrix averaged over the 12 real-world data sets. The stress value is 0.0587, suggesting a fairly good embedding of the algorithms into the 2D space. There are several interesting observations about Fig. 2(a). SL is significantly different from the other algorithms and is very sensitive to noise. A somewhat surprising observation is that AL is more similar to SL than one would expect, since it is also not robust enough against outliers. Chameleon type algorithm with G_1 objective function is also similar to single-link. The *k*-means algorithm is placed in the center of the landscape. This demonstrates that *k*-means can give reasonable clustering results that are not far away from other algorithms, and consistent with the general perception of the *k*-means approach. We can also detect some natural groupings in the landscape. Chameleon motivated algorithms with the objective functions (6, 8, 9, 10) are placed into the same group. This suggests that the objective function used to merge clusters during the agglomeration phase are not that important. Another tight group is formed by E_1 G_1', H_1 and H_2, showing that these four criteria, are, in fact, very similar. They also are close to the compact cluster of I_1, I_2, and Ch+I_1 outputs in the landscape. Ward's linkage clustering is similar to the *k*-means results. This is expected, as both of them are based on square error. The results of all the spectral clustering algorithms (31-34) are relatively close, hinting that different flavors of spectral clustering with reasonable parameters give similar partitions. All the mixture model based clusterings (1-4) are approximately placed within the same centrally located group of algorithms including the *k*-means and spectral clustering. Besides the single-link, the divisive-agglomerative hybrid algorithm Ch+I_2 as well as CL and AL algorithms produced the most "distinct" clusterings. We also produce a dendrogram of the clustering algorithms by performing complete-link on the dissimilarity matrix (Fig. 2(b)) and identify the major clusters in the plot of Fig. 2(a). Five algorithms are adequate to represent the spectrum of the 35 clustering algorithms considered here.

Table 1. The UCI ML data sets used in the experiments

Dermatology	Galaxy	Glass
Heart	Ionosphere	Iris
Letter recognition (A, B, C)	Segmentation	Texture
Letter recognition (X, Y, Z)	Wdbc	Wine

In another set of experiments, we generated 12 datasets with three 2-dimensional Gaussian clusters. The datasets differed in the degree of separation between clusters. Initially, the clusters were well separated and then gradually brought together until they substantially overlapped. Fig. 3(a) traces the changes in the clustering landscape as we move the clusters closer together (only a subset of the algorithms is shown in this landscape to avoid the clutter). Starting from the same point, some algorithms have dispersed on the landscape. Again, the *k*-means and certain spectral algorithms generated the most "typical" partitions in the center, while the SL and CL had the most unusual traces on the landscape. EM algorithms with diagonal and unconstrained covariance matrices, being close most of the time, diverge when cluster overlap became significant.

Analogous experiments were performed with 3 Gaussian clusters with variable density. We generated 12 data sets by gradually making two of the clusters sparse. Qualitatively, the algorithms behaved as before, except with a difference in starting points.

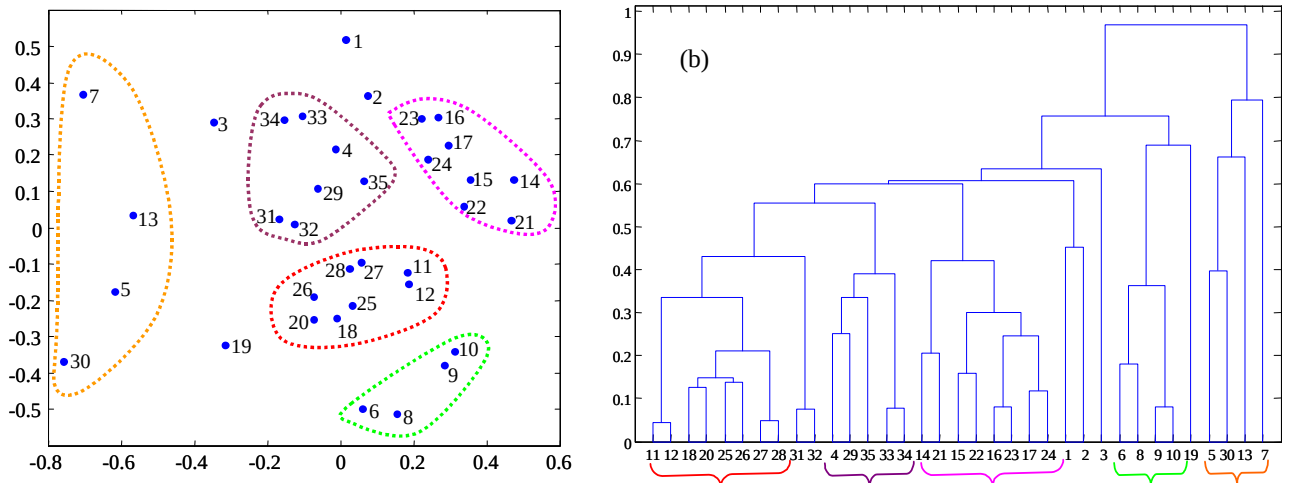

Figure 2. Landscape of clustering algorithms over real-world datasets: (a) classical scaling, (b) Complete link dendrogram.

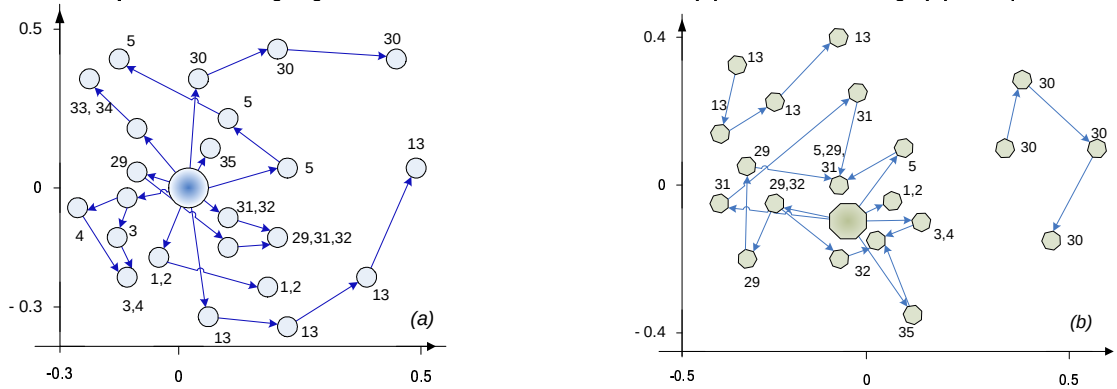

Figure 3. Landscape of clustering algorithms where paths correspond to the changes caused by: (a) gradually decreasing the separation distance between the three Gaussian clusters, (b) decreasing the density of clusters.

To summarize, we have empirically studied the landscape of some clustering algorithms by comparing the partitions generated for several data scenarios. While some algorithms like SL are clear "outliers", the majority of the clustering solutions have intrinsic aggregations. For example, Chameleon, Cure/graph partitioning, *k*-means/spectral/EM are representatives of the different groups. The parameters of the algorithms (other than the number of clusters) are of less importance. Hence, a practitioner willing to apply cluster analysis to new data sets, can begin by adopting only a few representative algorithms and examine their results. In particular, landscape visualization suggests a simple recipe that includes the *k*-means algorithm, graph-partitioning and linkage-based algorithms.

5. References

[1] R. Dubes and A.K. Jain, "Clustering Techniques: The User's Dilemma", *Pattern Recognition*, vol. 8, 1976, pp. 247-260.

[2] S.D. Kamvar, D.Klein, and C.D. Manning, "Interpreting and Extending Classical Agglomerative Clustering Algorithms using a Model-Based Approach", *Proc. of the 19th Intl. Conference on Machine Learning*, July 2002, pp. 283-290.

[3] C. Fraley and A.E. Raftery, *Model-based clustering, Discriminant Analysis, and Density Estimation*, Technical Report 380. Dept. of Statistics, Univ. of Washington, Seattle, WA.

[4] M. Meila, "Comparing Clusterings by the Variation of Information", *Proceedings of COLT 2003*, 2003, pp 173-187.

[5] W. M Rand, "Objective criteria for the evaluation of clustering methods", *J. of the Am. Stat. Association*, 66, 1971, pp. 846–850.

[6] A. K. Jain and R. C. Dubes. *Algorithms for Clustering Data*. Prentice Hall, 1988.

[7] R.O. Duda, P.E. Hart, and D.G. Stork, *Pattern Classification*, 2nd ed., John Wiley & Sons Inc., 2001

[8] A.Y. Ng, M.I. Jordan, and Y. Weiss, "On spectral clustering: Analysis and an algorithm", In T. G. Dietterich et al., eds., *Proc.of NIPS* 14, 2002, pp. 849-856.

[9] CLUTO 2.1.1 Software for Clustering High-Dimensional Datasets, available at http://www-users.cs.umn.edu/~karypis/cluto/

[10] G. Karypis, E.-H. Han, and V. Kumar: "CHAMELEON: A Hierarchical Clustering Algorithm Using Dynamic Modeling", *IEEE Computer*, 32 (8), 1999, pp. 68-75.

[11] S. Guha, R. Rastogi, and K. Shim. "CURE: An efficient clustering algorithm for large databases", *Proc.of ACM SIGMOD Conference*, 1998, pp. 73-84.

[12] J. Shi and J. Malik. "Normalized Cuts and Image Segmentation", *IEEE Trans. on PAMI*, 22 (8), 2000, pp. 888-905.

[13] D. Comaniciu and P. Meer. "Mean shift: A robust approach toward feature space analysis", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24 (5), 2002, pp. 603-619.

[14] T. Cox and M. Cox, *Multidimensional Scaling*, 2nd ed., Chapman & Hall/CRC, 2000.