Brock University

Department of Computer Science

A Comparison of Knee Strategies
For
Hierarchical Spatial Clustering

Brian J. Ross

Brock University
Department of Computer Science
St. Catharines
Ontario Canada L2S 3A1

Technical Report # CS-18-01
February 2018

A Comparison of Knee Strategies for Hierarchical Spatial Clustering

Brian J. Ross

Brock University, Dept. of Computer Science,
500 Sir Isaac Brock Way,
St. Catharines, ON, Canada L2S 3A1
bross@brocku.ca
http://www.cosc.brocku.ca/∼bross/

Abstract. A comparative study of the performance of knee detection approaches for the hierarchical clustering of 2D spatial data is undertaken. Knee detection is usually performed on the dendogram generated during cluster generation. For many problems, the knee is a natural indication of the ideal or optimal number of clusters for the given problem. This research compares the performance of various knee strategies on different spatial datasets. Two hierarchical clustering algorithms, single linkage and group average, are considered. Besides determining knees using conventional cluster distances, we also explore alternative metrics such as average global medoid and centroid distances, and F score metrics. Results show that knee determination is difficult, and that efficacy of knee strategies is very much problem dependent. Furthermore, knee determination is often more effectively applied on alternative distance metrics and F scores. In summary, knee strategies are often a useful heuristic, but not a general solution, towards optimal cluster detection.

Keywords: Knee, Hierarchical Clustering, Spatial Clustering

1 Introduction

Clustering is a popular and much studied classification technique in which data is automatically grouped according to shared characteristics. Hierarchical clustering is a method in which a hierarchy of clusters is incrementally generated for a dataset. Unlike K-means clustering, which requires a target cluster size K to be supplied, hierarchical clustering algorithms require the determination of an appropriate number of clusters after the cluster hierarchy has been generated. Often, the clustering dendogram is used for making this decision. Each node in the dendogram denotes an incremental clustering step, in which two clusters are merged into one. The node is labeled with a distance (or height) between the two clusters being combined. Often, the knee (or elbow) is identified within the dendogram, using a plot of the distance measures. The knee has been defined as the point of maximal marginal rate of return [24]. Generally speaking, the knee (when it exists) is the node in the dendogram that corresponds to the optimal number of clusters. A knee can be visualized, and the term "knee" itself refers to the geometric bend on the dendogram distance plot that corresponds to the optimal cluster position.

Although knees can be effective for determining optimal clusterings, there is no canonical rule for specifying the knee point, nor prescribed distance metric on which to apply knee analyses [10]. The above definition of a knee presumes it resides on the plot of monotonically increasing distances that arises with a greedy clustering algorithm. Alternate metrics can be used, and different definitions of knees may be necessary. Therefore, knees are primarily a heuristic for determining an optimal cluster point within a given dendogram, and many different characterizations are found in the literature. Note that we are concerned with knee-oriented analyses in this paper, and we are not considering the wider body of techniques for cluster optimization (e.g. [7, 13]).

A selection of knee strategies are as follows. Tibshirani *et al.*'s gap statistic generates a metric curve, which then looks for a maximal gap value that may denote the optimal clustering [19]. Salvador and Chan iteratively split the dendogram plot in two, to find the point were two line segments fit to the data using linear regression show a minimal RMSE between them and the data points [17]. The knee resides at the point of intersection of these line segments. Zhao *et al.* use a Bayesian Information Criteria (BIC) metric plot, and apply an angle measurement to identify a knee point [25]. Breaban and Iftene define the knee as the level i in a dendogram exhibiting the maximum ratio of $distance_i/distance_{i-1}$ [1].

Although knee identification is a popular tool, it is not a general solution to the problem. Clustering is generally intractable, where determination of optimal clusters for even K=2 is NP-complete [5]. Optimized clusters for alternative distance measures are likewise NP-complete [18]. For many real-world datasets, the identification of "optimality" may be ill-defined or subjective. Furthermore, the dendogram's utility for finding an optimal clustering wholly depends upon the quality of the underlying dendogram.

This paper compares different knee strategies with respect to the clustering of 2D spatial data. Our experiments consider various knee selection approaches and spatial data sets refined from the literature. The spatial datasets used have different problem characteristics and complexities. We do not evaluate the performance of the underlying clustering algorithm, and we use both single-linkage and group average clustering in our trials. We also investigate alternative distance measurements upon which knee analyses can be applied. For example, the global average distance of cluster elements to medoids (and centroids), and the F scores of same measurements, were used. Our results will confirm that many problem cases do not exhibit identifiable knees, and hence knees can be a useful heuristic method, but not a solution, for cluster optimization.

The paper is organized as follows. Section 2 reviews the clustering algorithms and knee strategies used in the experiments. The spatial data used is described in Section 3. Details of experiment design are discussed in Section 4. Results are presented and discussed in Section 5. Conclusions and future directions for the research are in Section 6.

Table 1. Hierarchical Clustering Algorithm

```
Input: $S_i = (x_i, y_i), i = 1, ..., K$
Output: Dendogram.

Initialization:
For i=1,...,K
        Add new cluster $C_{\{i\}}$
For all $C_i$, $C_j$ $(i \neq j)$
        $Distance(C_i, C_j) \leftarrow DistanceMeasure(S_i, S_j)$

Cluster generation:
For i=1,...,K {
        Find $C_p$, $C_q$ with minimum $d_{min} \leftarrow Distance(C_p, C_q)$.
        $Dendogram.Dist_i \leftarrow d_{min}$
        $Dendogram.Clust_i \leftarrow (C_p, C_q)$
        Remove $C_p$ and $C_q$.
        Add new cluster $C_{p \cup q}$.
        Update $Distance$ table:
                Remove all distances referring to $C_p$ and $C_q$.
                For all clusters $C_w \neq C_{p \cup q}$
                        $Distance(C_w, C_{p \cup q}) \leftarrow DistanceMeasure(C_w, C_{p \cup q})$
}

Return Dendogram.
```

2 Background

2.1 Hierarchical Clustering

Table 1 shows pseudocode for hierarchical clustering algorithms. Hierarchical clustering generates a complete clustering of a dataset. Initially, all data points are considered individual clusters. A distance table *Distance* records the distances between all existing clusters. Using this table, the closest clusters are determined. They are joined together to form a new cluster, which replaces the two merged clusters. The distance table is updated to reflect this change. This iteratively continues until a single cluster has been created.

The incremetal clustering process is modeled by a dendogram. Each node denotes an incremental step, by recording which two clusters are joined. The node is also labelled with the distance between these joined clusters, which the clustering algorithm used in order to select them for merging.

The feature that defines different hierarchical clustering algorithms is the method which is used to measure distances between clusters. This is denoted in Table 1 by the function *DistanceMeasure*. We consider two methods. In single linkage clustering, a distance between a cluster C_w and new cluster $C_{p \cup q}$ is updated as:

$$Distance(C_w, C_{p \cup q)} = minimum(Distance(C_w, C_p),\ Distance(C_w, C_q))$$

The group average method uses:

$$Distance(C_w, C_{p \cup q)} = average(Distance(C_w, C_p),\ Distance(C_w, C_q))$$

2.2 Distance Measures

We apply knee detection to 3 distance measurements associated with dendogram nodes.

(i) *Standard distance (Std)*: This is the distance used by the clustering algorithm (Section 2.1). Each node of the dendogram indicates this distance, which is always the minimum between all the clusters at that point in the clustering.

(ii) *Global average medoid distance (Avg Med)*: Letting C_i be cluster i ($1 \leq i \leq K$), MD_i be the total distance between the medoid (the member that is on average closest to the other members) and other elements of C_i, and T be the total number of elements in all clusters. Then,

$$AvgMed = \frac{\Sigma_{i=1}^{K} MD_i}{T}$$

(iii) *Global average centroid distance (Avg Cent)*: Like (ii), but let CD_i be the total distance of all elements in C_i to the centroid (average coordinate of all elements):

$$AvgCent = \frac{\Sigma_{i=1}^{K} CD_i}{T}$$

2.3 Knee Determination

A knee is a point on a plot of dendogram measurements which identifies the optimal number of clusters. Conventionally, the measurement is the distance between the two clusters being merged at that node in the dendogram, where this distance is that used by the clustering algorithm.

There are a number of proposals for identifying knees in dendogram plots. Let n_i be node i in the dendogram ($1 \leq i \leq K$), and d_i be the distance measure associated with n_i. We assume that dendogram nodes are numbered in the *inverse* order they were created; the root of the dendogram is therefor n_1. The knee methods considered in this paper are as follows.

(i) **Magnitude**: Node with maximum magnitude $d_{i+1} - d_i$.

(ii) **Ratio**: Node with maximum ratio d_{i+1}/d_i.

(iii) **Second derivative**: Node with greatest second derivative [14].

(iv) **Minimum**: Node with minimal distance.

(v) **L-method, L-method D, L-method S**: This uses Salvador and Chan's L-method, which fits two lines segments to the plot using linear regression [17]. The first line starts at the root of the dendogram, and ends at node n. The second line starts at $n+1$ and continues to node m ($n < m$). The ideal line placement is one that minimizes the root mean square error (RMSE) between the regression lines and the distances in the plot. We apply this L-method iteratively to points on the plot being analyzed. For best results, Salvador and Chan recommend using a similar number of regression points for both line segments. They use divide-and-conquer reduction to the points being regressed, until an equal spread is obtained. Motivated by this recommendation, we implement two variations of the L-method:

- L-method D (distance): If line 1 uses nodes 1 to n, then line 2 uses nodes $n+1$ to $2n$.
- L-method S (sampling): If line 1 uses nodes 1 to n, then line 2 will sample n nodes between $n+1$ through k (last node). For example, if there are $k = 11$ nodes, and line 1 is fitted to nodes 1 to 4, then line 2 is fitted to nodes 5, 7, 9, and 11.

(vi) **F score A, F score B**: The F score is based on the F test of a one-way analysis of variance (ANOVA) [8]:

$$F\ score = \frac{Between\ group\ variance}{Within\ group\ variance}$$

$$Between\ group\ variance = \textstyle\sum_{i=1}^{K} (\bar{Y_i}. - \bar{Y})^2/(K-1)$$

$$Within\ group\ variance = \textstyle\sum_{i=1}^{K} \sum_{j=1}^{n_i} (Y_{ij} - \bar{Y_i}.)^2/(T-K)$$

where there are K clusters, T data samples ($K < T$), $\bar{Y_i}.$ denotes the sample mean in the i-th cluster, and Y_{ij} is the j^{th} data point in the i^{th} out of K clusters. The above shares some similarity to Tibshirani *et al.*'s gap statistic, which also uses the notion of within-cluster dispersion (variance) [19]. We plot the F score at all nodes in the dendogram. We then find the latest point (furthest from the root) that shows the first significant drop in F score, which suggests a decline in measurable cohesion throughout the entire clustering. We use two methods for finding this knee in the F score plot.

- F score A: This finds the highest i in which:

$$(f_{i+1} - f_i) > \delta^2_{1...i}$$

  where $\delta^2_{1..i}$ is the standard deviation of the plotted F scores 1 to i.
- F score B: The highest i in which:

$$(f_{i+1} - f_i) > \delta^2_{1...k}$$

  where $\delta^2_{1..k}$ is the standard deviation of all k F scores on the plot.

F score A looks for a drop (knee) that exceeds the standard deviation from the end of the dendogram to that point in the dendogram. F score B is similar, but uses the standard deviation of the entire dendogram, which usually defines a larger threshold value. F score B typically finds knees closer to the root, and hence smaller clusterings. Note that, although we do not assume that the data points follow a normal distribution, we still use the ANOVA statistic, since it provides suitable discriminatory power for knee determination. However, we refrain from assessing the F scores in terms of statistical significance.

3 Spatial Data

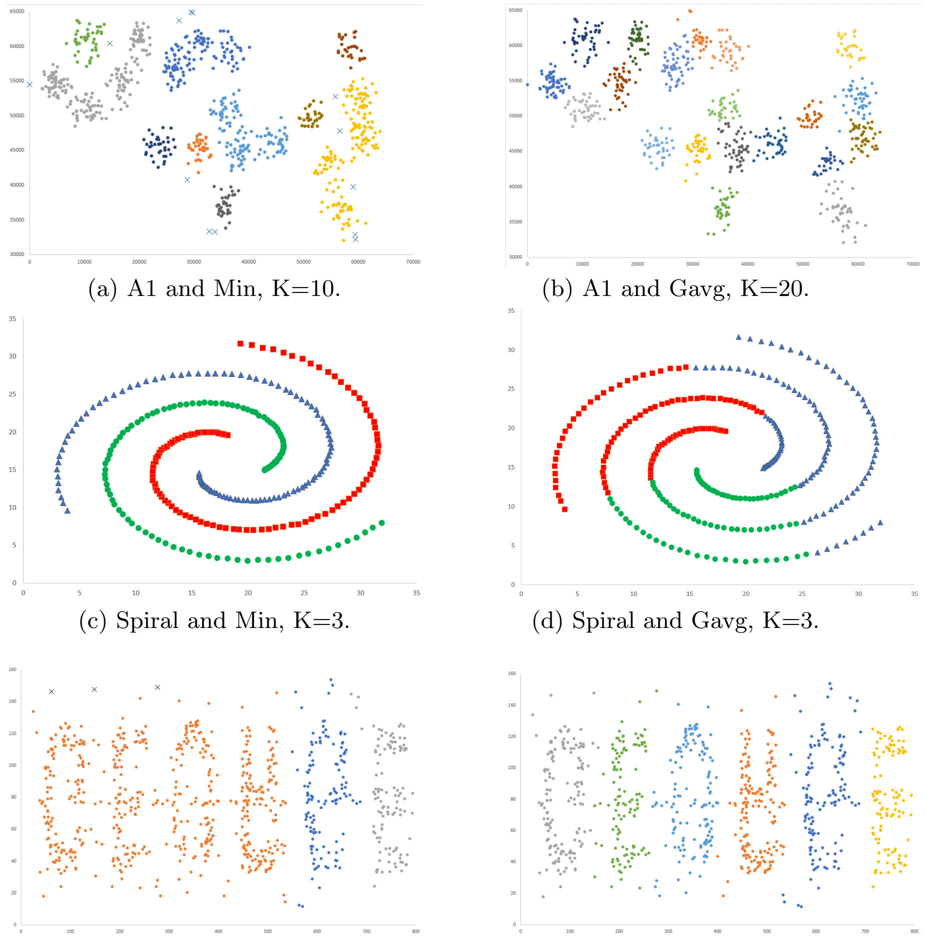
Table 2. Spatial Datasets. Boldface denotes a reasonable quality clustering.

		# Nodes		Target Cluster Size		
Name	Ref.	Orig.	Reduced	Orig.	Min	Gavg
a1	[11]	3000	800	20	10	**20**
Aggregation	[6]	788	788	7	5	**7**
Birch3	[23]	10000	800	100	47	**69**
Compound	[22]	399	399	6	3	**6**
D31	[20]	3100	800	31	19	**31**
Flame	[4]	240	240	2	-	**2**
Jain	[9]	373	373	2	**2**	**2**
Pathbased	[2]	300	300	3	-	**3**
R15	[20]	600	600	15	11	**15**
RRR	-	54	54	3	**3**	**3**
Spiral	[2]	312	312	3	**3**	3
t4.8k	[12]	8000	800	6	-	**6**
t5.8k	-	8000	800	6	3	**6**
t7.10k	[12]	10000	800	9	-	**9**
t8.8k	[12]	8000	800	8	2	**8**
Unbalance	[16]	6500	800	8	**7**	**7**

Sixteen spatial datasets with different data sizes, shapes, complexities, and target cluster sizes were selected (Table 2). Sources for the origin of most data are given, are most are available for download at [3] and [15]. (The RRR set is a trivial 3-cluster set used as a baseline.) Due to the computational overhead of hierarchical clustering, datasets with more than 800 points were randomly downsampled. Those with less than 800 points were left intact.

All the datasets have target cluster numbers associated with them (*Orig* column in Table 2). In some cases, these values are intuitively obvious when viewing scatter plots of the data. However, many datasets (e.g. the t4.8k family) contain noise, and clustering quality is more difficult to discern. In addition, downsampling can reduce the integrity of original spatial patterns. These factors make some datasets difficult to intuit obvious optimal clusterings. It is also not assured that single linkage or group average clustering will generate an optimal cluster set for the given data. Therefore, it is unwise to use the supplied target cluster numbers as "solution" targets for knee evaluation.

For these reasons, target cluster sizes for knee evaluation were obtained as follows. The datasets were each evaluated with single linkage (Min) and group average (Gavg) clustering at the supplied clustering target values. The resulting clusters were examined, and any clusters of size ≤ 3 were discarded. The remaining clusters were taken to denote the target clustering for that dataset and cluster algorithm. In some cases with single linkage clustering, only a sin-



(a) A1 and Min, K=10.

(b) A1 and Gavg, K=20.

(c) Spiral and Min, K=3.

(d) Spiral and Gavg, K=3.

(e) t5.8k and Min, K=3.

(f) t5.8k and Gavg, K=6.

Fig. 1. Example target clusterings for single linkage and group average. Points in (a) and (e) belonging to clusters of size ≤ 3 are not used for target cluster size, and are plotted with ”X”.

gle cluster resulted; these cases were not studied, and are left blank in Table 2. Throughout, we do not consider the actual quality of clusterings when assessing knee performance (other than ignoring cases with tiny clusters).

Example scatter plots of a few datasets with their colour-coded target clusterings are shown in Fig. 1. In plots (a) and (e) of single linkage clusterings, the target K is less than the original target value, due to discarded small clusters. Although both clusterings for Spiral satisfy the original target of 3 clusters, the group average result is obviously not the intended one. Nonetheless, when analyzing knee performance, we do not ascribe value judgements on the underlying quality of clusters.

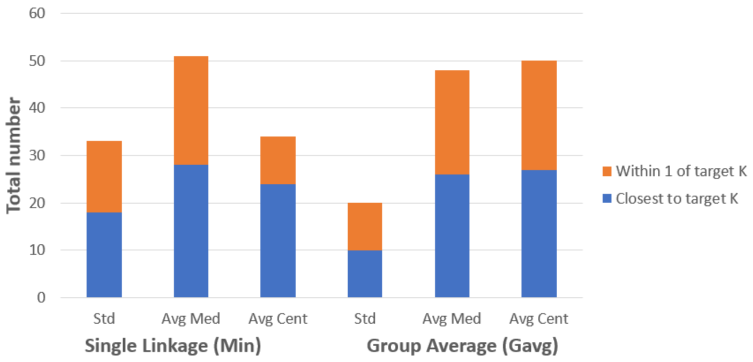


Fig. 2. Summary of knee performance with respect to clustering algorithm and distance metric. *Within 1 of target* tallies when a knee strategy generated a clustering size at, or within 1, of the target K. *Closest to target* tallies when a knee strategy generated a clustering size closest to the target K value. A single result can be tallied as both within 1 of target, and closest to target.

4 Experiment Design

The single linkage (Min) and group average (Gavg) hierarchical clusterings from Section 2.1 are applied to the datasets in Section 3.

Before knee analysis, the size of each dendogram is reduced to its final 100 nodes (which denotes clusterings of sizes between 1 to 100). Knee solutions will therefore never exceed 100 clusters. The RRR dataset has 54 points, and is not reduced.

The L methods require a minimal number of points for linear regression. For all datasets with target cluster numbers above 5, we set this minimum to 5, in order to infer suitable quality regression lines. Since such a minimum will preclude finding clusterings of size less than 5, we set the minimum to 2 for smaller target datasets.

5 Results

Figure 2 compares the quality of knee evaluations with respect to the distance measurement used by the clustering algorithm. The summary shows that alternative distance metrics such as average global medoid and centroid distance are effective for knee identification.

Fig. 3 shows how frequently difference knee strategies found the nearest value to the target K associated with the dataset and clustering algorithm. The chart shows separate tallies for single linkage and group average clusterings, and the tallies sum the results of all datasets and distance measures. The Mag, Ratio, and 2nd Deriv knee strategies were the most successful, followed by the F score athe Min strategies. The L method was not successful on the datasets we studied, and the L method variants were completely unsuccessful.

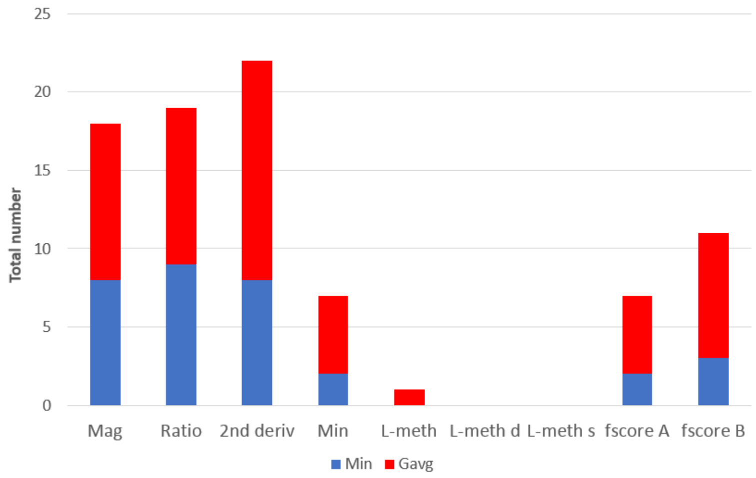


Fig. 3. Frequency that knee strategies were closest to target cluster size. The nearest for a dataset is selected from results of all knee strategies and distance measurements.

Table 3 show the performance of knee strategies with respect to the distance metric and clustering algorithm. The table tallies the number of times when knee strategies produced the closest clustering to the target size for that dataset and clustering algorithm. Standard distances were only useful for the Mag, Ratio and 2nd deriv knee methods; however, these methods were also successful on the other distance metrics.

Fig. 4 shows two dendogram distance plots (standard distance, and F score of global average medoid distance) from single linkage clustering on the Aggregation dataset. The plots show how knee detection can differ for different strategies. Magnitude, Ratio, and 2nd Derivative strategies found the knee for the standard distance plot (orange). F score A, however, found the knee on the average medoid distance plot. This knee was flagged by the drop (moving right-to-left) between nodes at # clusters 5 and 4, which signalled a decrease in cluster coherence. The drop between # clusters 7 and 6 after this point was not flagged, as the drop was within the standard deviation threshold.

Fig. 5 shows the knees for the Aggregation dataset, but this time for the group average clustering and the global average medoid and centroid distances. Both F score strategies found the knees in both curves, again signaled by the first drop (moving right-to-left) outside their respective standard deviation thresholds. Fig. 6 deconstructs the F score of the global average medoid curve (blue plot) of Fig. 5). The knee is primarily determined by the curve shape, which is in turn defined by the *Between group variance* (see Section 2.3). The *Within-group variance* plays the role of a scaling factor for the final F score.

Table 3. Breakdown of performance of knee strategies with respect to clustering algorithm and distance metric. Values denote the number of times each knee strategy produced clusterings that were the closest to the target cluster size for a given dataset and clustering algorithm (ties are possible).

	Std		Avg Med		Avg Cent		
Knee	Min	Gavg	Min	Gavg	Min	Gavg	Total
Mag	5	4	3	1	0	4	18
Ratio	5	2	3	3	1	5	19
2nd deriv	5	4	3	5	0	5	22
Min	0	0	1	2	1	3	7
L-meth	0	0	0	1	0	0	1
L-meth D	0	0	0	0	0	0	0
L-meth S	0	0	0	0	0	0	0
F score A	-	-	1	2	1	3	7
F score B	-	-	2	5	1	3	11

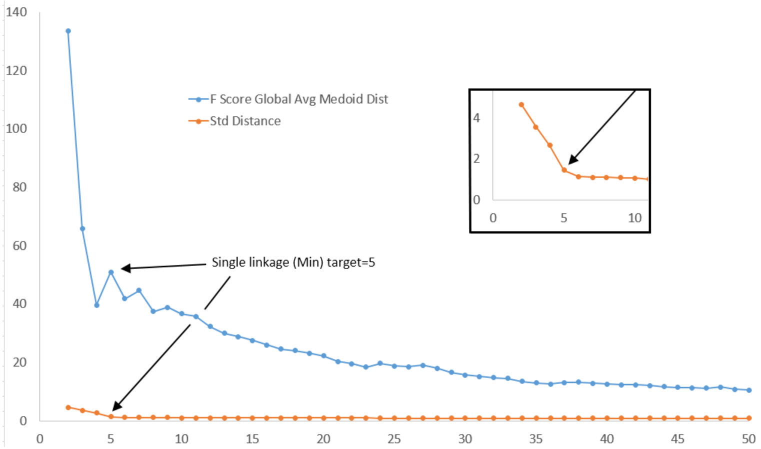


Fig. 4. Comparison of different distance curves. The Aggregation dataset is used, and single linkage (Min) clustering. The inset is a rescaled magnification of standard distance curve. Magnitude, Ratio, and 2nd Derivative strategies found the knee for the standard distance plot (orange). F score A found the knee on the global average medoid distance plot (blue).

6 Conclusion

This paper compared a variety of knee detection strategies, and the effect of variables such as spatial datasets, hierarchical clustering algorithms, and distance metrics. The results confirm that knee strategies are heuristics whose success depends on a combination of many factors, the most basic of which is the spatial data being processed. We show that alternative distance metrics are often worth considering, and that a serendipitous match of dataset, clustering algo-

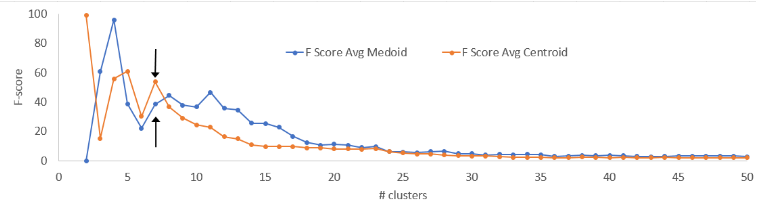


Fig. 5. Aggregation dataset, Group average clustering, F score (last 50 clusters) of global average medoid and centroid distances. Arrows point to flagged knee via both F score A and B variants.

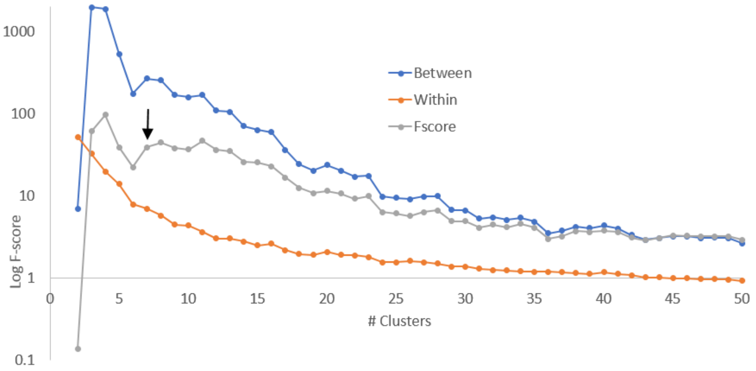


Fig. 6. Decomposition of global average medoid curve of Fig. 5.

rithm, distance metric, and knee strategy can result in high quality hierarchical clusterings. However, results must be evaluated in light of the computational limitations of clustering algorithms in general, and knee detection in particular, as in many instances dendograms may not exhibit identifiable knees.

There are many directions for future work. Other data besides 2D spatial datasets should be explored, as well as other hierarchical clustering algorithms, knee strategies, and distance metrics. However, we expect that the same conclusions as above will apply. More generally, a study of various mathematical transformations of distance plots, with a goal towards effective knee detection, should be investigated. Given the many variables that determine the success of knee detection, another avenue of research is to use machine learning (e.g. deep learning) to determine more principled means for matching datasets with appropriate clustering and knee detection parameters (e.g. see [21] for an interesting application).

Acknowledgements: This research was supported by NSERC Discovery Grant RGPIN-2016-03653.

References

1. Breaban, M., Iftene, A.: Dynamic objective sampling in many-objective optimization. Procedia Comptuer Science 60, 178–187 (2015)
2. Chang, H., Yeung, D.: Robust path-based spectral clustering. Pattern Recognition 41(1), 191–203 (2008)
3. Franti, P.: Clustering datasets (2015), http://cs.uef.fi/sipu/datasets/, last accessed October 31, 2017
4. Fu, L., Medico, E.: Flame: A novel fuzzy clustering method for the analysis of dna microarray data. BMC Bioinformatics 8(1) (2007)
5. Garey, M., Johnson, D.: Computers and Intractability. W.H. Freeman, New York (1979)
6. Gionis, A., Mannila, H., Tsaparas, P.: Clustering aggregation. ACM Trans. Knowledge Discovery from Data 1(1), 1–30 (2007)
7. Gordon, A.: Classification (2e). Chapman and Hall-CRC (1999)
8. Hayter, A.: Probability and Statistics for Engineers and Scientists. Duxbury (2007)
9. Jain, A., Law, M.: Data clustering: A user's dilemma. In: Intl Conf. Pattern Recognition and Machine Intelligence. pp. 1–10 (2005)
10. Jr, D.K., Shook, C.: The application of cluster analysis in strategic management research: an analysis and critique. Strategic Management Journal 17, 441–458 (1996)
11. Kärkkäinen, I., Fränti, P.: Dynamic local search algorithm for the clustering problem. Tech. Rep. A-2002-6, Department of Computer Science, University of Joensuu, Joensuu, Finland (2002)
12. Karypis, G., Han, E., Kumar, V.: Chameleon: A hierarchical 765 clustering algorithm using dynamic modeling. IEEE Trans. Computers 32(8), 68–75 (1999)
13. Milligan, G., Cooper, M.: An examination of procedures for determiningthe number of clustersin a data set. Psychometrika 50, 159–179 (1985)
14. online: A numerical second derivative from three points (2013), https://mathformeremortals.wordpress.com/2013/01/12/a-numerical-second-derivative-from-three-points/, last accessed February 6, 2018.
15. online: (2017), https://github.com/deric/clustering-benchmark, last accessed October 31, 2017
16. Rezaei, M., Tranti, P.: Set-matching methods for external cluster validity. IEEE Trans. Knowledge and Data Engineering 28(8), 2173–2186 (August 2016)
17. Salvador, S., Chan, P.: Determining the number of clusters/segments in hierarchical clustering/segmentation algorithms. In: Proc. IEEE Intl Conf on Tools with Artificial Intelligence (ICTAI). pp. 576–584. IEEE (2004)
18. Sima, J., Schaeffer, S.: On the np-completeness of some graph cluster measures. In: Proc. 32nd Intl Conf on Current Trends in Theory and Practice of Computer Science. pp. 530–537. Springer-Verlag (2006), lNCS 3831
19. Tibshirani, R., Walther, G., Hastie, T.: Estimating the number of clusters in a data set via the gap statistic. J. R. Statistic. Soc. B 63 (Part 2), 411–423 (2001)
20. Veenman, C., Reinders, M., Backer, E.: A maximum variance cluster algorithm. IEEE Trans. Pattern Analysis and Machine Intelligence 24(9), 1273–1280 (2002)
21. Xie, J., Girshick, R., Farhadi, A.: Unsupervised deep embedding for cluster analysis. In: Proc 33rd Intl Conf on Machine Learning. JMLR:W&CP (2008)
22. Zahn, C.: Graph-theoretical methods for detecting and describing gestalt clusters. IEEE Trans. Computers 100(1), 68–86 (1971)

23. Zhang, T., Ramakrishnan, R., Livny, M.: Birch: A new data clustering algorithm and its applications. Data Mining and Knowledge Discovery 1(2), 141–182 (1997)
24. Zhang, Y., Zhang, X., Tang, J., Luo, B.: Decision-making strategies for multi-objective community detection in complex networks. In: Proc 9th Intl Conf on Bio-inspired Computing: Theories and Applications (BIC-TA). pp. 621–628. Springer (2014)
25. Zhao, Q., Hautamaki, V., Franti, P.: Knee point detection in bic for detecting the number of clusters. In: Proc AVICS 2008. pp. 664–673. Springer-Verlag (2008), lNCS 5259