# Accelerating Clustering Methods through Fractal Based Analysis

Changhao Jiang+, Yiheng Li*, Minglong Shao+, and Peng Jia*

+Computer Science Department
*Center for Automated Learning and Discovery
Carnegie Mellon University
5000 Forbes Avenue, Pittsburgh, PA 15213, U.S.A.
{jiangch,yiheng,shaoml,pengj}@cs.cmu.edu

**Abstract.** Clustering on large high dimensional datasets is still a difficult challenge for practical data mining applications. Knowing intrinsic characteristics of dataset can help abstract the dataset and/or provide insightful hints in the clustering process, which potentially could improve the process dramatically. This paper presents a novel fractal analysis based preprocessing tool to accelerate clustering methods by sampling dataset into critical-sized subset which preserves original dataset's distribution patterns and by advising clustering methods of the dataset's intrinsic features through critical input parameters. Experiment result of applying this tool with BIRCH clustering method suggests its effectiveness and applicability.

## 1 Introduction

Though clustering methods with different strengths and weaknesses have been developed for many years, it is still a difficult challenge in practical data mining applications. Some methods, e.g. CLARAN [2], BIRCH [4], CURE [1], tried to address the problems of high-dimensionality and large datasets in data mining. However, they just generally cluster the dataset without taking advantage of information about the dataset's intrinsic distribution characteristics, which, if properly used, could provide promising performance and accuracy gain.

Fractal analysis is valuable for its capability to gain insight into dataset's intrinsic characteristics with practical overhead. "Self-Plot" [1][3] is among the most frequently used fractal analysis tools. It can provide information about dataset's *fractal-dimensionality, estimated intra-cluster and inter-cluster distance*, etc, in just linear times scan of the whole dataset. In this paper, we will present a novel tool to improve clustering methods by taking advantage of dataset's intrinsic information obtained through "Self-Plot".

[1] "Self-Plot" is also called "Correlational Integral Plot" in Fractal literature.

# 2 Method

## 2.1 Overview Description

The key idea in this paper is based on the assumption that: if a sampled subset has similar "Self-Plot" to that of original dataset, then it preserves original dataset's distribution pattern and can be used in place of original dataset for clustering.

With the assumption, we add a fractal-based sampling tool as a preprocessing step before clustering. A given large high dimensional dataset first goes through this sampling tool and is reduced to critical-sized[2] subset which preserves original data distribution pattern. Then the sampled subset is fed into a clustering method. The clustering result of sampled subset is reported as the clustering result of original one. (See figure 1)

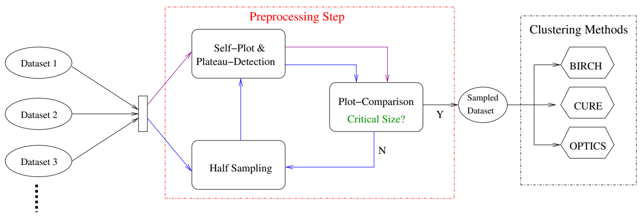



**Fig. 1.** The overview flow-chart of fractal-based sampling

The preprocessing box improves clustering methods in two aspects. First it reduces dataset's size; second it advises successive clustering method with some critical input parameters, which are obtained through "Self-Plot" analysis. The second aspect is dependent on specific clustering method. For example, BIRCH clustering method requires input parameter $T$ as the estimated intra-cluster distance. If wisely set, the rebuilding time of CF tree in BIRCH could be dramatically reduced. In later part, we will see the improvements made by advising $T$ to BIRCH.

## 2.2 Components and Algorithm

The preprocessing box has two main components. One is called "Self-Plot and Plateau Detection" module, ("Plateau" for short), the other is "Plot Comparison", ("Comparison" for short). They collaborate to find the critical-sized subset of original dataset.

[2] By "*critical-sized*", we mean the **smallest** subset that preserves original dataset's distribution pattern. Smallest is not absolute, but is relative to all iterations of operation.

**Plateau Module**

"Plateau" is a flat part within "Self-Plot". It yields much valuable information about the dataset's intrinsic characteristics, such as its inter-cluster and intra cluster distances. The starting distance of plateau is also a critical parameter for the later "Comparison module". Figure 2 shows a typical plateau found in "Self-Plot".

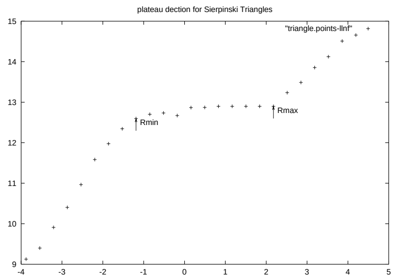



**Fig. 2.** Plateau detected in a Self-Plot

This plateau detection functionality is implemented based on slope verification. Segments with slopes under a predefined threshold value are detected as part of plateau. Finally all segments concatenate into a plateau.

**Comparison Module**

Comparison module receives input of two "Self-Plots", and starting distance of detected plateau from "Plateau" module to evaluate similarity between the two "Self-Plots". The module works under the statistically proved lemma: *the sampled subset is a good abstract of original dataset if its "Self-Plot" is parallel to that of original dataset after the plateau part.* (See figure 3.)

The intuition behind the lemma is that: *if observed from distance farther than intra-cluster distance, the sampled subset should have same fractal dimensionality as original dataset.* Because fractal dimensionality is reflected as slope of "Self-Plot", same fractal dimensionalities means same slopes in Self-Plots. Hence, the "Self-Plots" should be parallel after plateau part. In practice, we use correlation coefficients of two "Self-Plots", of first derivatives and of second derivatives to assess the parallelism of the later part of "Self-Plots".

## 3 Experiment with BIRCH method

BIRCH is one of the most cited clustering algorithms in data mining applications. It clusters large high-dimensional dataset into a hierarchical tree structure, called CF tree, with optimized I/O overhead and one time scan over the whole dataset.

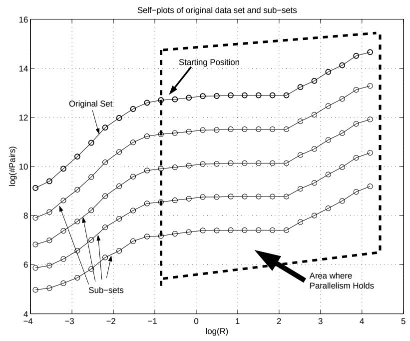
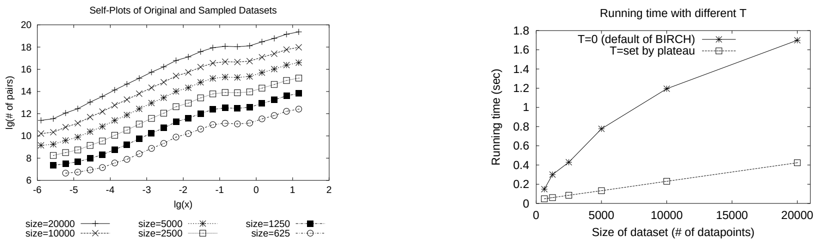



**Fig. 3.** Graphical Illustration of Parallelism Lemma of Self-plots of original and sampled datasets

In this experiment, we use the above method to improve BIRCH in two ways: reduce the size of dataset through fractal based sampling; advise BIRCH of critical input parameter, $T$, which is the estimated maximum distance within each cluster.

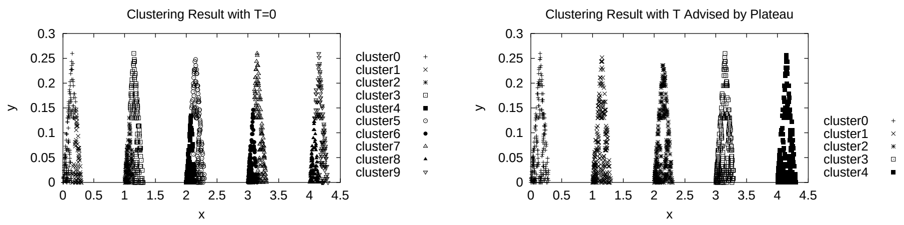



(a) Self-Plots of original and sampled datasets (b) Advised BIRCH vs. unadvised BIRCH

**Fig. 4.** Self-Plots of sampled datasets and running time of advised/unadvised BIRCH

In figure 4-(a), the fractal based sampling tool iteratively samples original dataset (20000 points) into smaller subsets until critical-sized subset (1250 points) is reached. Figure 4-(b) shows that using starting distance of plateau as the input parameter $T$ can greatly expedite the execution of BIRCH.

BIRCH requires several sensitive input parameters, like $T$ (maximum intra-cluster distance) and $K$ (number of clusters), which are often hard to estimate ahead of time. Figure 5 shows, as we use the estimated $T$ from plateau, the clustering result is improved. In (a), BIRCH grouped 10 clusters in the critical-sized subset (1250 points), while in (b), with properly advised $T$, BIRCH produced 5 clusters correctly.




(a) unadvised BIRCH (b) advised BIRCH

**Fig. 5.** cluster labels given by BIRCH

## 4 Conclusion

In this paper, we presented a new fractal-based sampling tool to accelerate prevailing clustering methods in two aspects: (1) as a preprocessing step to iteratively sample the original dataset into critical-sized subset with preserved distribution patterns, so as to reduce the size of dataset; (2) advising clustering methods with critical input parameters which is specific to different methods. The experiment result suggests the tool's effectiveness and applicability in both aspects. We also did experiments on OPTICS and CURE clustering methods, the results are also positive.


## 5 Acknowledgement

We are grateful to professor Christos Faloutsos for his invaluable guidance, and source code of "Self-Plot". We would like to thank Rande Shern for his previous work, and thank the authors of BIRCH for sharing the source code on web.


## References


1. Guha, S., Rastogi, R., and Shim, K. CURE: an efficient clustering algorithm for large databases. In *ACM SIGMOD International Conference on Management of Data* (Seattle, WA, USA, 1998), pp. 73–84.
2. Ng, R. T., and Han, J. Efficient and effective clustering methods for spatial data mining. In *20th International Conference on Very Large Data Bases, September 12–15, 1994, Santiago, Chile proceedings* (Los Altos, CA 94022, USA, 1994), J. Bocca, M. Jarke, and C. Zaniolo, Eds., Morgan Kaufmann Publishers, pp. 144–155.
3. Traina, A. J. M., Jr., C. T., Papadimitriou, S., and Faloutsos, C. Cross-cloud plots: Scalable tools for spatial and multidimensional data mining. In *Knowledge Discovery and Data Mining* (2001), pp. 184–193.
4. Zhang, T., Ramakrishnan, R., and Livny, M. BIRCH: an efficient data clustering method for very large databases. In *ACM SIGMOD International Conference on Management of Data* (Montreal, Canada, June 1996), pp. 103–114.