Towards a Compact and Effective Representation for Datasets with Inhomogeneous Clusters

Haimei Zhao, Zhuo Chen, Qiuhui Tong, and Bo Yuan

Intelligent Computing Lab, Division of Informatics, Graduate School at Shenzhen,
Tsinghua University, Shenzhen 518055, People's Republic of China
z-chen17@mails.tsinghua.edu.cn, zhaohm17@mails.tsinghua.edu.cn
tongqh@126.com, yuanb@sz.tsinghua.edu.cn

Abstract. Due to the restriction of computing resources, it is often inconvenient to directly conduct analysis on massive datasets. Instead, a set of representatives can be extracted to approximate the spatial distribution of data objects. Standard data mining algorithms are then performed on these selected points only, which typically account for a small fraction of the original data, reducing the computational time significantly. In practice, the boundary points of data clusters can be regarded as a compact and effective representation of the original data, with great potential in clustering, outlier or anomaly detection and classification. As a result, given a complex dataset, how to reliably identify a set of effective boundary points creates a new challenge in data mining. In this paper, we present a boundary extraction technique similar to the method in SCUBI (Scalable Clustering Using Boundary Information). The key difference is that our technique exploits the clustering information in a feedback loop to further refine the boundary. Experimental results show that our technique is more robust and can produce more representative boundary points than SCUBI, especially on complex datasets with large inhomogeneity in terms of cluster density.

Keywords: Boundary · Extraction · Clustering · SCUBI

1 Introduction

In face of the ever increasing volume of datasets, efficiency is becoming a critical concern in data analytics. For example, the computational cost of many clustering algorithms [1, 2] such as K-means, DBSCAN, Affinity Propagation and Hierarchical Clustering is greatly affected by the number of data points (n) in the dataset. Among them, the time complexity of K-means is $O(tkn)$ where t is the iteration number [4]; the time complexity of DBSCAN is $O(n^{4/3})$ in the best situation and $O(n^2)$ on some datasets [5]; the time complexity of Affinity Propagation is $O(n^2T)$ [6] and Hierarchical Clustering features $O(n^2\log n)$ time complexity [7].

Consequently, it may be impractical to directly apply existing clustering algorithms on large datasets, due to the potentially intolerable computing time. Although it is possible to accelerate these algorithms using parallel or distributed platforms such as Hadoop, an interesting idea is to tackle this challenge from a different perspective: whether

it is necessary to involve all available data into the clustering process? Under some general assumptions (e.g., each cluster is a solid object without holes), the shape of each cluster can be determined solely by the data points on its surface while all interior points impose no influence on the clustering results. This observation can be significant as it implies that only data points on the surface of each cluster need to be clustered, which only account for a small fraction of the entire dataset.

SCUBI (Scalable Clustering Using Boundary Information) is a latest clustering scheme that exploits the idea of boundary information to achieve good scalability [8]. Firstly, it extracts boundary points from the original dataset. Next, boundary points are clustered using existing clustering algorithms. Finally, the boundary information and the clustering results are used to assign interior points to appropriate clusters. SCUBI is a general scheme and a variety of clustering methods such as DBSCAN, Affinity Propagation and Spectral Clustering can be plugged into the framework. Typically, less than 5% data points are extracted as boundary points and experimental results show that SCUBI can significantly improve the efficiency of existing clustering algorithms on massive datasets with little impact on the quality of clusters.

In this paper, we investigate the boundary information from a more general perspective: it can be regarded as a compact and generic representation of the dataset, which potentially plays a key role in many data mining tasks such as stream clustering, outlier/anomaly detection and classification. The major contribution of our work is an effective approach to the extraction of boundary points, which follows the general principle of boundary extraction in SCUBI but alleviates the issues of SCUBI on complex datasets with large inhomogeneity in terms of cluster density. The core idea is a strategy with an extra feedback loop: i) select a relatively large set of boundary points initially to ensure robustness; ii) conduct clustering on boundary points; iii) use the cluster information as feedback to further refine the boundary set.

In Section 2, we give a brief review of existing boundary extraction techniques based on different principles. The details of our improved boundary extraction method are presented in Section 3. Comprehensive experimental results on 2D and higher dimensional datasets are shown in Section 4. This paper is concluded in Section 5 with some directions for future work.

2 Review of Boundary Extraction

The importance of boundary points comes from the fact that they inexplicitly specify the distribution of data points. For example, once the boundary of each cluster is known, all data points can be easily assigned to a certain cluster based on their relative locations. Similarly, in classification tasks, if two classes can be perfectly classified, it is likely that the decision boundary is only determined by those boundary points, in analogy to support vectors in SVM, instead of any other interior points.

Existing approaches to boundary extraction can be divided into four categories: concave theory based, diagram based, information entropy based and density based. The methods based on concave theory sequentially build the boundary from one point to another. Some typical methods include alpha shape [9], conformal alpha shape [10] and

concave hull [11]. They tend to perform well on 2D datasets with evenly distributed data points and are widely used in image processing and machine vision [12]. However, they cannot be directly extended to high-dimensional datasets.

The methods based on diagram establish a Delaunay diagram of the dataset. Since the Delaunay diagram is unique for a fixed dataset, the boundary points that are found are also fixed. Furthermore, these methods do not require user defined parameters and can therefore be used when little *priori* knowledge of the original dataset is available. However, the complexity of building a Delaunay diagram is $O(n^2)$ and the computational cost is strongly influenced by the dimension of datasets. As a result, they are only used to process 2-D or 3-D data [13, 14, 15].

The methods based on information entropy are mainly used in the domain of point clouds [16]. They are useful for simplifying 3D models but are insufficient for processing high-dimensional datasets. Also, they suffer from high time complexity on datasets with multiple clusters, as the internal model needs to converge several times.

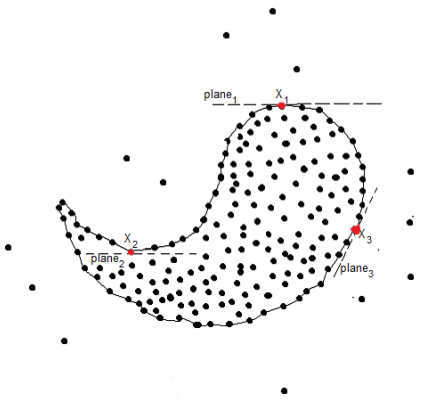


Fig. 1. Boundary point and tangent planes

The core idea of density-based approaches is that a boundary point should be the point where the density difference on both sides of its tangent plane is large [17, 18] as shown in Fig. 1. However, in practice, it is not easy to calculate such a tangent plane directly. Instead, the normal vector, which is directly related to the tangent plane, is much easier to calculate [19, 20, 21]. The normal vector (direction of density gradient) of a point is conceptually defined as the average of the vectors pointing to its *k* nearest neighbors [22], as shown in Fig. 2. Note that as the direction instead of the length of the normal vector is important, in Eq. 1, we do not need the value of ρ.

Next, according to Eq. 2 with normalized *NV*, a score is calculated for each data point based on the cosine of the angle between each vector to its nearest neighbors and *NV*. The higher the score of a point, the greater the density gradient at that point, which means that it is more likely to be a boundary point (Algorithm 1).

$$NV(x) = \rho \cdot \frac{1}{k} \sum_{i=1}^{k} u_i \tag{1}$$

$$u_i(x) = x_i - x, where\ x_i \in kNN(x)$$

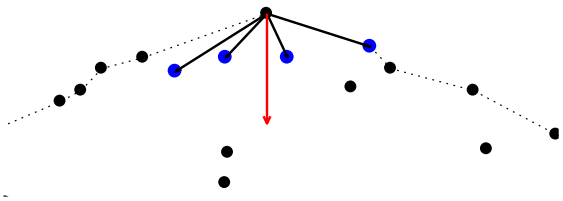


Fig. 2. The normal vector of a point is the average of the vectors pointing form the point itself to its *k* nearest neighbors. The red arrow shows the direction of the normal vector.

$$\text{scores}(x) = \sum_{i=1}^{n} cos\left(u_i,\ NV(x)\right) \tag{2}$$

$$\text{where } NV(x) = \sum_{i=1}^{k} \frac{(x_i - x)}{|x_i - x|}$$

Algorithm 1 The Boundary Extraction Method used in SCUBI

Input: $D, k, \alpha_{noise}, \alpha_{boundary}$
Output: $Boundary, NV$

$scores \leftarrow -\infty$
for *each* $x \in D$ **do**
  $Find\ kNN(x)$
  $kNN_{average}(x) \leftarrow Average\ distance\ of\ kNN(x)\ to\ x$
end for
$Sort\ kNN_{average}\ in\ descending\ order, Mark\ first\ \alpha_{noise}\ as\ noises$
for *each* $x \in D$ **do**
  if x *is not the noise* **then**
    $Exclude\ the\ noises\ in\ the\ kNN(x)$
    $Find\ the\ NV(x)\ of\ x$
    $Calculate\ cos(NV(x), u_i(x))$
    $scores(x) \leftarrow \sum_{i=1} cos(NV(x), u_i(x))$
  end if
end for
$Sort\ scores\ in\ descending\ order$
$Select\ first\ \alpha_{boundary}\ as\ Boundary$
$Return\ Boudanry, NV\ of\ boudnary\ points$

After boundary extraction, SCUBI uses clustering algorithms such as DBSCAN and Affinity Propagation to group boundary points into clusters. Finally, interior points are assigned to the same cluster as its nearest boundary point (Algorithm 2).

Algorithm 2 SCUBI

(1) *Extract boundary points using the Normal Vector Method*
(2) *Cluster the boundary points using Dbscan*
(3) *Cluster the non − boundary points*
for *each* $x \in D$ **do**
  if x is marked "noise" **then**
    $C_{id}(x) = -1$
  else
    $Find\ the\ C_{id}(boundary)\ of\ nearest\ boundary\ for\ x$
    $Assign\ the\ C_{id}(boundary)\ to\ x$
  end if
end for

3 Robust Boundary Extraction

In practice, the quality of the boundary extracted can directly determine the results of subsequent data mining operations. For example, if the boundary points from the same cluster are sparse with large gaps, they may be grouped into multiple clusters incorrectly. Furthermore, in the dynamic situation, one can use the boundary of the origin data as a reference to group new coming data points or detect possible outliers from the data stream. However, the poor quality of the boundary extracted from the origin data may hamper the accuracy of data mining work greatly.

Fig. 3 (left) shows a set of high quality boundary points, which is effective for detecting outliers as it gives a reasonably good approximation of the data distribution. For example, the new data point shown as a cyan triangle will be regarded as an interior point but the data point indicated by a red square will be recognized as an outlier, according to their relative locations to the boundary. By contrast, Fig. 3 (right) shows an imperfect boundary with a large gap in the top-right region. In this case, it is possible that an outlier detection algorithm based on boundary information may incorrectly regard the triangle point as an outlier.

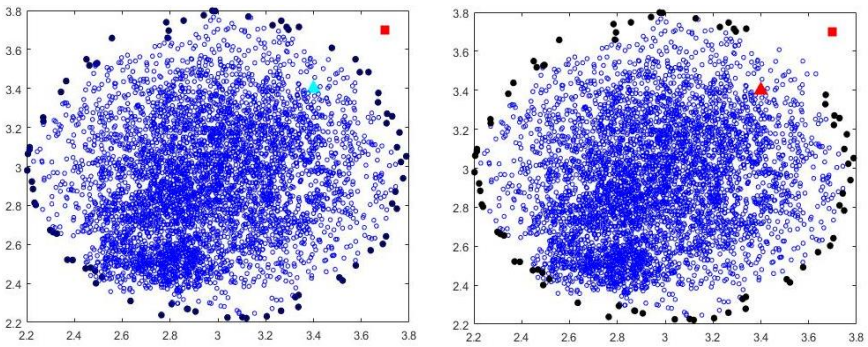


Fig. 3. An effective boundary (left) and an imperfect boundary (right)

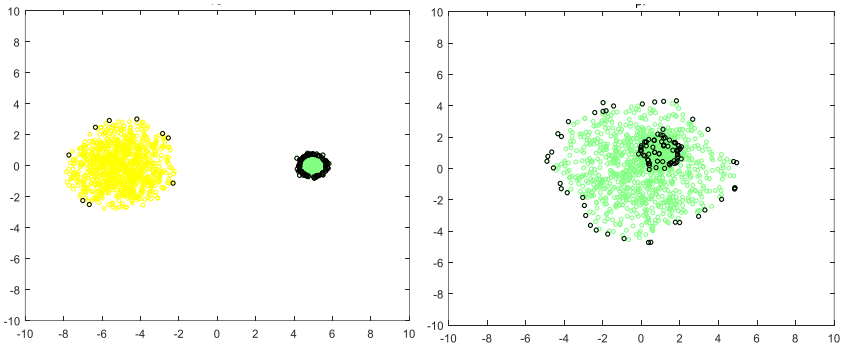


Fig. 4. The challenges of boundary extraction using SCUBI: the two clusters have different levels of density (left) and the density of a cluster is not homogeneous (right).

Due to the consideration of efficiency, the boundary extraction procedure in SCUBI is executed only once and only a small percentage of data points are selected. However, data clusters may be quite different in terms of density and the one-off extraction strategy in SCUBI may result in undesired distribution of boundary points, especially when the percentage of data points selected is relatively small. Fig. 4 (left) shows an example with two clusters of different densities. It is clear that most boundary points extracted are concentrated on the denser cluster while the boundary points on the other cluster are very sparse, insufficient as a good representation of the corresponding cluster. Furthermore, this *efficiency-robustness* dilemma cannot always be alleviated by simply increasing the number of boundary points. For example, on inhomogeneous clusters, some interior points may be incorrectly identified as boundary points due to high density level, as shown in Fig. 4 (right).

Algorithm 3 Two-stage Boundary Extraction

Input: $D, k_1, k_2, \alpha_1, \alpha_2$
Output: *Boundary*

$scores_1 \leftarrow -\infty$
for *each* $x \in D$ **do**
  $k_1NN(x) \leftarrow Find\ the\ k_1\ nearest\ neighbors\ of\ x\ in\ D$
  $u_i \leftarrow (point_i - x)\ \forall\ point \in k_1NN(x)$
  $NV_1(x) \leftarrow \sum \frac{u_i}{|u_i|}$
  $scores_1(x) \leftarrow \sum cos(NV_1(x), u_i)$
end for
$Sort\ D\ in\ descending\ order\ of\ scores_1$
$length \leftarrow round(|D| \cdot \alpha_1)$
$Boundary_{tmp} \leftarrow D[1 : length]$
$Clusters \leftarrow Cluster\ the\ Boundary_{tmp}\ using\ classical\ clustering\ algorithm$
$scores_2 \leftarrow -\infty$
for $cluster_j \in Clusters$ **do**
  for *each* $x \in cluster_j$ **do**
    $k_2NN(x) \leftarrow Find\ the\ k_2\ nearest\ neighbors\ of\ x\ in\ cluster_j$
    $v_i \leftarrow (point_i - x)\ \forall\ point \in k_2NN(x)$
    $NV_2(x) \leftarrow \sum \frac{v_i}{|v_i|}$
    $scores_2(x) \leftarrow \sum cos(NV_2(x), v_i)$
  end for
  $Sort\ cluster_j\ in\ descending\ order\ of\ scores_2$
  $length_j \leftarrow round(|cluster_j| \cdot \alpha_2)$
  $boundary_j \leftarrow cluster_j[1 : length_j]$
  $Boundary \leftarrow \{Boundary \cup boundary_j\}$
end for
return *Boundary*

This is because SCUBI calculates the scores of all data points according to Eq. 2 and sets a single threshold for distinguishing boundary points from others. If the density of a cluster is low, its data points are sparsely distributed and the neighbors of a data point are likely to spread in a wide range of directions, resulting in relatively low scores. As a result, the selection process in SCUBI tends to favor boundary points from dense clusters. Furthermore, within a cluster, if the density varies significantly, data points with large density gradients are also likely to receive high scores and be regarded as boundary points, even if they are actually interior points.

It should be mentioned that the objective of SCUBI is to conduct clustering as efficiently as possible. Our purpose is different: we want to extract a set of boundary points of high quality from the current dataset for further applications while the computational

cost is not our primary concern. In other words, we can afford extracting and clustering more boundary points and incorporating additional processing steps.

The core idea is to replace the one-off extraction strategy in SCUBI by a two-stage procedure. In the first step, SCUBI is applied on the dataset to extract a set of preliminary boundary points and clustering is performed on this point set. The number of boundary points can be set to a value larger than that in normal SCUBI to ensure robustness. In the second step, with the clustering information available, another round of boundary extraction is performed individually on boundary points belonging to the same cluster, to further refine the extraction results (Algorithm 3).

Note that the parameter k in k-nearest neighbors is a key factor in both SCUBI and our boundary extraction technique. For large k values, more data points are involved in the score calculation but noisy points and even data points from other clusters may also be included. In Algorithm 3, a smaller value (k_1) is used in the first round of extraction, which not only reduces the time cost but also enables the boundary details to be preserved. In the second round, a larger value (k_2) is used since the interference from other clusters and noisy points have largely been eliminated, resulting in more accurate normal vectors and boundary information.

4 Experiment

The major purpose of experimental studies is to investigate the effectiveness of the new boundary extraction strategy in comparison to the standard method in SCUBI. Two sets of datasets were used: 2D datasets with highly complex clusters as well as synthetic datasets created by Gaussian distributions with diagonal covariance matrices in which the number of clusters and the dimension were systematically varied from 1 to 9 and 2 to 8 (Table 1), respectively. The number of data points in each cluster was fixed to 10,000 and the variance of each dimension of each cluster was 1.

We extracted the boundary points on datasets *Letters, Circles, U-shapes* with the same parameter settings. The percentage of boundary points was set to 1% in the origin method. For the two-stage boundary extraction, the percentages of data points selected in the two stages were 10% and 10% for the three 2D datasets and 2% and 50% for *Spheres*. For the value of k (number of neighbors), it was fixed to 10 for the original method while k_1 and k_2 were 10 and 20 for our method, respectively.

Table 1. Summary of datasets

Dataset	Dim	Number of cluster	Number of points
Letters	2	19	24,160
Circles	2	30	60,000
U-shapes	2	4	84,600
Spheres	2-8	1-9	10,000-90,000

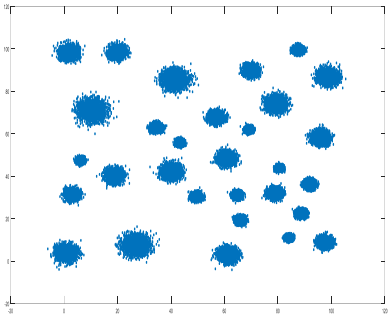
Circles is a dataset with several clusters and the densities of clusters differ significantly. In Fig. 5, SCUBI tended to extract relatively more boundary points from dense clusters while the boundary points were discontinuous on sparse clusters. Also, SCUBI

mistakenly chose some interior points as boundary points. By contrast, our extraction method produced a relatively uniform edge on almost every cluster and the misjudgment of interior points was less likely to happen. *Lines* and *Letters* are two datasets with complex clusters on which SCUBI often regarded interior point as boundary points due to the mutual interference among clusters. With the help of the cluster information, our method effectively reduced the interference of noise and other clusters, ensuring a good boundary on each cluster, as shown in Fig. 6 and Fig. 7.

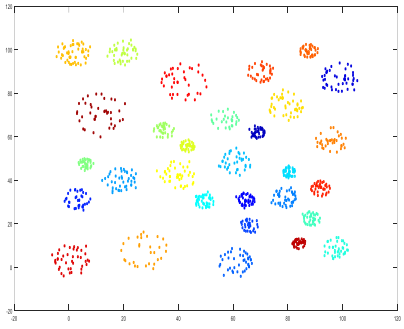
For high-dimensional cases (*Spheres*), the distance between each data point and the corresponding mean was calculated and, for each cluster, the set of outmost data points was regarded as the *ground truth* for boundary points.

The accuracy of boundary extraction is defined as below:

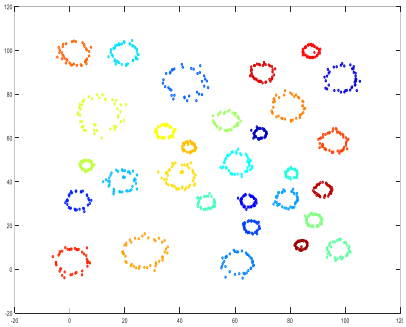
$$\text{Accuracy}\left(\alpha_{outer},\ \alpha_{boundary}\right) = \frac{|\{Outer(\alpha_{outer})\} \cap \{BoundaryExtract(\alpha_{boundary})\}|}{|\{BoundaryExtract(\alpha_{boundary})\}|} \quad (3)$$



(a) Original dataset



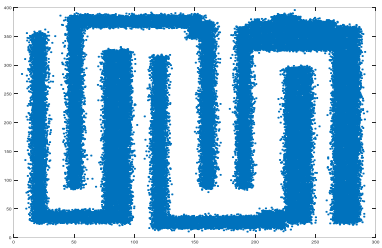
(b) SCUBI



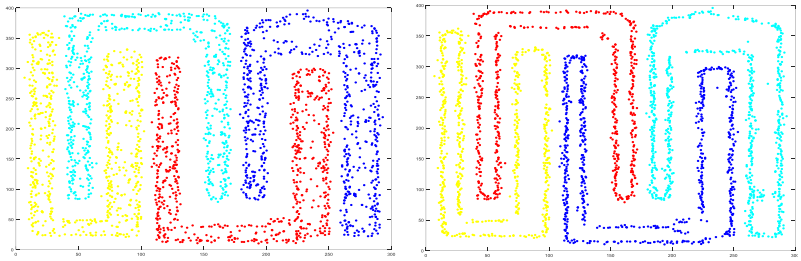
(c) Our method

Fig. 5. Boundary extraction on *Circles* dataset

In Eq. 3, α_{outer} is the percentage of data points selected as the ground truth and $\alpha_{boundary}$ is the percentage of data points extracted as boundary points. $Outer$ is the ground truth set and $BoundaryExtract$ is the set of selected boundary points. It is clear that accuracy reaches its maximum value 1 when all extracted boundary points are contained within the ground truth.

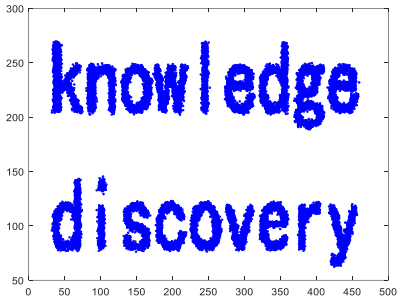


(a) Original dataset

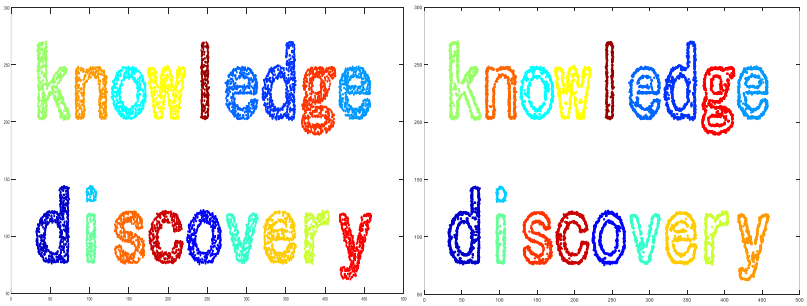


(b) SCUBI (c) Our method

Fig. 6. Boundary extraction on *U-shapes* dataset



(a) Original dataset

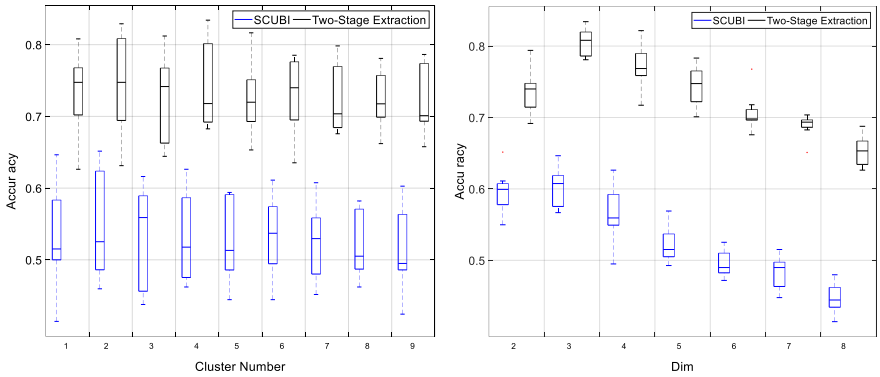


(b) SCUBI (c) Our method

Fig. 7. Boundary extraction on *Letters* dataset

We created 63 datasets with dimension from 2 to 8 and the number of clusters from 1 to 9 to demonstrate the effectiveness of boundary extraction algorithms. Fig. 8 shows the comparison of accuracy distribution between SCUBI and our method with $\alpha_{boundary} = 0.01$ and $\alpha_{outer} = 0.01$, which means that only 1% data points were selected as the boundary points and the ground truth, respectively. Fig. 8 (a) shows the relationship between accuracy and the number of clusters and each box shows the distribution of the results on 7 datasets with various dimensions. It is obvious that our method (mean accuracy around 0.75) systematically outperformed the extraction method in SCUBI (mean accuracy around 0.55). Fig. 8 (b) shows the relationship between accuracy and dimension and each box shows the distribution of the results on 9 datasets with different numbers of clusters. Again, our method was clearly superior to the extraction method in SCUBI.

Note that, as the dimension increased, the accuracy of both methods decreased gradually. The reason is largely due to the fixed number of points (10,000) in each cluster: the cluster became sparser as the dimension increased and some boundary points with very low local densities were treated as outliers and discarded. However, the ground truth set was determined by distances only, regardless of the density factor. As a result, some data points in the ground truth set were missing from the boundary points actually extracted.



(a) Accuracy vs. Cluster Numbers (b) Accuracy vs. Dimension

Fig. 8. The accuracy comparison of two boundary extraction methods on *Spheres* with varying dimensions and cluster numbers ($\alpha_{boundary} = 0.01$ and $\alpha_{outer} = 0.01$)

5 Conclusion

In the era of big data, data mining algorithms are often confronted with unprecedented volume of data. Due to the time complexity of these algorithms, most of them cannot be directly applied to the massive datasets, creating a major gap between academic research and industrial applications. Recently, SCUBI was proposed as a scalable clustering scheme, which strategically employs boundary information as a compact representation of the original dataset to accelerate the clustering procedure. As a complement

to mainstream solutions relying on high performance computing infrastructure such as GPU and distributed computing, SCUBI provides a unique perspective for handling massive datasets by exploiting the structural information of datasets while leaving existing clustering algorithms largely unchanged.

In this paper, we argue that boundary information is not only highly valuable for standard clustering tasks but also important for other key applications such as stream clustering [23], outlier/anomaly detection [24, 25] and classification. As the major contribution of our work, a novel two-stage boundary extraction technique was proposed to address the issues that we have identified with SCUBI. More specifically, we found that the current one-off extraction method in SCUBI was ineffective at handling datasets with large inter-cluster or intra-cluster variance in terms of density. Experimental results on a variety of purposefully designed datasets show that our extraction technique, which takes advantage of the clustering information, is clearly superior to the standard extraction technique in SCUBI, especially on cases with large inhomogeneity in terms of cluster density.

As to future work, it is important to conduct formal analysis of existing density-based boundary extraction techniques to provide a rigorous foundation for better understanding their effectiveness and possible limitations. Currently, these methods are mostly heuristic ones and there is still a lack of principled guidance on key issues such as parameter setting and performance evaluation. Meanwhile, with the help of a competent technique that can produce high quality boundary information, it would be very interesting to further extend its application scope to more challenging situations. For example, instead of extracting a fixed set of boundary points, a dynamic boundary set can be maintained in an online manner, which is very useful for processing data steams or when the dataset is too large to be processed as a whole in the memory.

References

1. Jain, K., Murty, N., Flynn, J.: Data clustering: A review. ACM Computing Surveys 31(3), 264-323 (1999).
2. Kaufman, L., Rousseeuw, P. J.: Finding Groups in Data: An Introduction to Cluster Analysis. Wiley (2008).
3. MacQueen, J.: Some methods for classification and analysis of multivariate observations. In: Proceedings of the 5th Berkeley Symposium on Mathematical Statistic and Probability, vol. 1, pp. 281-297 (1967).
4. Arthur, D., Manthey, B., Röglin, H.: K-means has polynomial smoothed complexity. In: Foundations of Computer Science, vol. 157, pp. 405-414 (2009).
5. Ester, M., Kriegel, H. P., Xu, X.: A density-based algorithm for discovering clusters a density-based algorithm for discovering clusters in large spatial databases with noise. In: International Conference on Knowledge Discovery and Data Mining, pp. 226-231. AAAI Press, Portland (1996).
6. Frey, B. J., Dueck, D.: Clustering by passing messages between data points. Science 315, 972-976 (2007).
7. Berkhin, P.: A survey of clustering data mining techniques. In: Kogan J., Nicholas C., Teboulle M. (eds) Grouping Multidimensional Data. Springer, 25-71 (2006).

8. Tong, Q. H., Li, X., Yuan, B.: A highly scalable clustering scheme using boundary information. Pattern Recognition Letters 89, 1-7. Elsevier (2017).
9. Edelsbrunner, H., Kirkpatrick, D., Seidel, R.: On the shape of a set of points in the plane. IEEE Transactions on Information Theory 29(4), 551–559 (1983).
10. Moreira, A. J. C., Santos, M. Y.: Concave hull: A k-nearest neighbors approach for the computation of the region occupied by a set of points. In: Proceedings of the Second International Conference on Computer Graphics Theory and Applications, vol. 3520, pp. 61-68. Springer, Barcelona (2006).
11. Li, X., Yu, W., Cervantes, J.: Border Samples Detection for Data Mining Applications Using Non Convex Hulls. In: Mexican International Conference on Artificial Intelligence, pp. 261-272. Springer (2011).
12. Hoogs, A., Collins, R.: Object boundary detection in images using a semantic ontology. In: Conference on Computer Vision and Pattern Recognition Workshop, pp. 956-963 (2006).
13. Liu, D., Nosovskiy, G. V., Sourina, O.: Effective Clustering and Boundary Detection Algorithm Based on Delaunay Triangulation. Pattern Recognition Letters 29, 1261-1273. Elsevier (2008).
14. Estivill-Castro, V., Lee, I.: AUTOCLUST: Automatic clustering via boundary extraction for mining massive point-data sets. In: International Conference on Geocomputation, vol.26, pp.23-25 (2000).
15. Yang, J., Estivill-Castro, V., Chalup, S. K.: Support vector clustering through proximity graph modelling. In: International Conference on Neural Information Processing, vol. 2, pp. 898-903. IEEE, Singapore (2002)
16. Chen, X. J., Zhang, G., Hua, X. H.: Point cloud simplification based on the information entropy of normal vector angle. Chinese Journal of Lasers 42(8), 328-336 (2015).
17. Xia, C., Hsu, W., Lee, M. L.: BORDER: Efficient computation of boundary points. IEEE Transactions on Knowledge & Data Engineering 18(3), 289-303 (2006).
18. Nosovskiy, G. V., Liu, D., Sourina, O.: Automatic Clustering and Boundary Detection Algorithm Based on Adaptive Influence Function. Elsevier (2008).
19. Zhu, F., Ye, N., Yu, W., Xu, S., Li, G.: Boundary detection and sample reduction for one-class support vector machines. Neurocomputing 123, 166-173 (2014).
20. Qiu, B. Z., Yue, F., Shen, J. Y.: BRIM: an efficient boundary points detecting algorithm. In: Advances in Knowledge Discovery and Data Mining. vol. 4426, pp. 761-768 (2007).
21. Li, Y.: Selecting training points for one-class support vector machines. Pattern Recognition Letters 32(11), 1517-1522 (2011).
22. He, Y. Z., Wang, C. H., Qiu, B. Z.: Clustering boundary points detection algorithm based on gradient binarization. Applied Mechanics & Materials 266, 2358-2363 (2013).
23. Silva, J. A., Faria, E. R., Barros, R. C.: Data stream clustering: A survey. ACM Computing Surveys 46(1), 13 (2013).
24. Pokrajac, D., Lazarevic, A., Latecki, L. J. Incremental local outlier detection for data streams. In: IEEE Symposium on Computational Intelligence and Data Mining. pp. 504-515. IEEE, Honolulu (2007).
25. Salehi, M., Leckie, C., Bezdek, J. C.: Fast memory efficient local outlier detection in data streams. IEEE Transactions on Knowledge & Data Engineering 28(12), 3246-3260 (2017).