Automated Density-Based Clustering of Spatial Urban Data for Interactive Data Exploration

Erica Rosalina[1], Flora D. Salim[1], and Timos Sellis[2]

1. School of Science, CSIT, RMIT University

2. School of Software and Electrical Engineering, Swinburne University

RMIT UNIVERSITY

Outline

- Introduction
- Background
- Problem Definition
- Automated Parameter Selection for Spatial Clustering
  - DBSCAN
  - HDBSCAN
- Experiment and Evaluation
- Conclusion
- Q&A

Introduction

- Interactive data exploration for Big Data analytics
  - Patterns discovery [1]
  - Hidden relationships [2]

- In many cases [3, 4, 5, 6, 7], such knowledge discovery is enabled by data visualization tools through interactive processes (such as [8]).
- Common method in summarizing and discovering interesting patterns from big data: **Clustering**

- Limited studies on the application of clustering methods on spatial-temporal data from smart cities or road networks.
- Challenging tasks for these types of data:
  - Discovering patterns by computing density of data points in space.

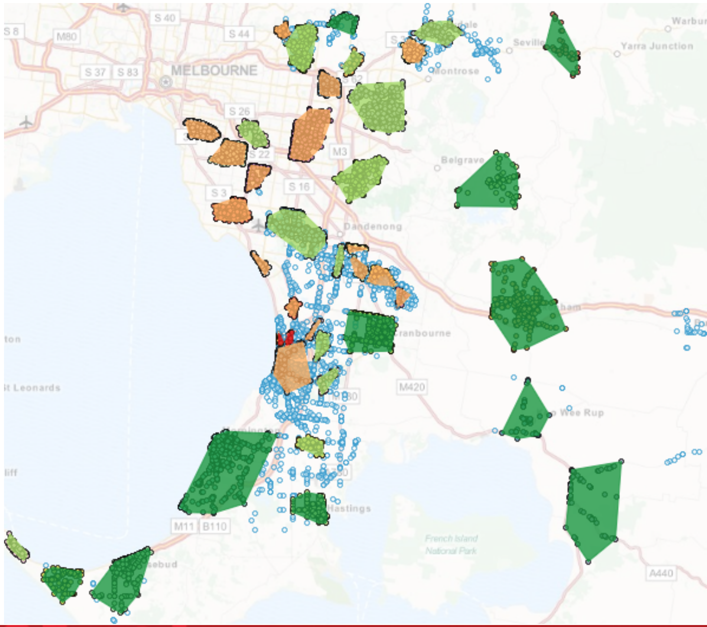
Background

- Density based clustering
  - The process of grouping of similar objects while keeping dissimilar objects in different groups based on the density in the given data space.
- Algorithms:
  - DBSCAN [9]
  - HDBSCAN [10]
  - VDBSCAN [11]
  - DMDSCAN [12]

- These algorithms require parameters: ε (Eps) and m_{pts} (minimum number of points). Moreover, the output of algorithms can be sensitive depending on the given parameters.
- Limited studies on the enhancements of density based techniques, that are parameter-less.

Problem Definition

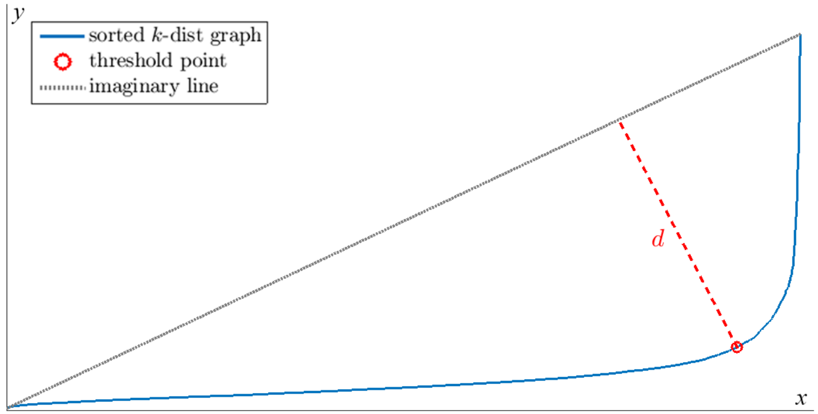
- Application domain: dynamic data exploration through map visualization.
- How to adaptively compute the density according to the level of user query (i.e. resolution changes)?
- Challenges:
  - Parameter-less approach for user based data exploration
  - Adaptive visualization for resolution changes



Automated Parameter Selection for Spatial Clustering

Density-based spatial clustering of applications with noise (DBSCAN)

- Algorithm parameters:
  - ε (Eps)
    In [9], Eps can be determined by plotting a *sorted k-dist graph* of which the first "valley" or the "knee" can be computed.



The knee corresponds to a sharp change in the density distribution amongst points. For this reason, Eps is often set to be equal to the threshold point value.

  - m_{pts} (minimum number of points)

$$m_{pts} = ln(N)$$

where N is the number of visible points on the map.

Finding Knee point of a curve (*Eps*)

- Geometric approach in finding Knee point (threshold) for *Eps*

ALGORITHM 1: Find knee point of a curve: FindKneePoint(K)

Input: K = a sorted array of curve points

$maxIndex \leftarrow 0$;

$maxDist \leftarrow -1$;

$N \leftarrow$ size of K;

$x_1 \leftarrow 0$ ▷ x_1 = first index of array K

$y_1 \leftarrow K[x_1]$ ▷ y_1 = first element of array K

$x_2 \leftarrow N - 1$ ▷ x_2 = last index of array K

$y_2 \leftarrow K[x_2]$ ▷ y_2 = last element of array K

for $i \leftarrow 0$ *to* N **do**

  $currDist \leftarrow \frac{|(y_2-y_1)x_0-(x_2-x_1)y_0+x_2y_1-y_2x_1|}{\sqrt{(y_2-y_1)^2+(x_2-x_1)^2}}$ ▷ Equation 2

  if $maxIndex = 0$ *or* $currDist > maxDist$ **then**

    $maxDist \leftarrow currDist$;

    $maxIndex \leftarrow i$;

  end

end

return $K[maxIndex]$;

Automated Parameter Selection for Spatial Clustering

Hierarchical Density-based spatial clustering of applications with noise (HDBSCAN)

- Algorithm parameters:
  - m_{pts} (minimum number of points)

$$m_{pts} = ln(N)$$

where N is the number of visible points on the map.

  - m_{clsize}
    The minimum number of samples in a group for that group to be considered a cluster; groupings smaller than this size will be left as noise.

- Typical application of HDBSCAN creates an illusion of algorithm to require only one parameter by setting m_{pts} = m_{clsize} (referred as **HDBSCAN Normal**).
- Our proposed method is referred as HDBSCAN Mode, based on most frequent number of neighbours that are within the knee of core distances.

HDBSCAN Mode

- The following algorithms are proposed to find the value of m_{clsize}.

ALGORITHM 2: m_{clSize}'s mode approach

$d_{core} \leftarrow$ core distances of every objects in dataset D;
Sort d_{core};
$d_{kneeCore} \leftarrow FindKneePoint(d_{core})$;
▷ Find knee of core distances using Algorithm 1
$C \leftarrow NeighbourCounts(D, d_{kneeCore})$ $d_{kneeCore}$ from the object. ▷ Refers to Algorithm 3
$m_{clSize} \leftarrow$ mode of set C;

ALGORITHM 3: Compute the set of neighbour counts that are within distance d

---

$C \leftarrow \emptyset$;
for *each object* $p \in D$ **do**
  Add $|N_d(p)|$ to C
end
return C

Dataset

- Victoria's road network data.
- Historical road crashes[1] data in Victoria, Australia from 1 January 2006 to 30 June 2013.
  - 72176 accident nodes
  - Our study area is limited to 63 localities in South Eastern part of Victoria, with the total number of accident nodes 7864.
  - There are 5361 affected road segments with the total length of 1425.2 km.
  - The total study area is approximately 1909.3 km^2.

1. https://www.data.vic.gov.au/data/dataset/crash-stats-data-extract

Experiment and Evaluation

- Clustering techniques for the experiment (repeated runs):
  - DBSCAN
  - HDBSCAN Normal
  - HDBSCAN Mode

- Experiment Settings:
  - High resolution
  - Low resolution

- Evaluation approaches:
  - Cluster indices
  - Visualization

Cluster Validation – Internal Indices

- Internal Criteria refers to evaluation of clustering algorithm results in terms of quantities that involve the vectors of the dataset themselves (e.g. proximity matrix), i.e. information is intrinsic to the dataset alone and no external information provided.
- Measures:
  - C index (denoted as **C**) [13]
  - Calinski-Harabasz (denoted as **CH**) [14]
  - Davies-Bouldin (denoted as **DB**) [15]
  - Dunn (denoted as **D**) [16]
  - Silhouette (denoted as **S**) [17]
  - Xie-Beni (denoted as **XB**) [18]
- Voting system is used for choosing the best clusters (based on number of best indices).

Cluster Validation – Comparisons of cluster results

- For both high and low resolutions, the validation of cluster results will be based on the following comparisons:
  1. *overall comparison*: all parameters combination from each approach are considered for selecting the best outcome of every index; only the best ones are further analyzed
  2. *indices best comparison*: only the best run(s) (according to the indices results) of each approach is/are considered
  3. *default comparison*: only the default run(i.e. with default m_{pts}) from each approach is considered, so the comparison is always done on 3 rows (1 for each approach) with identical m_{pts} value.

High Resolution Evaluation (Cluster Indices)

- Various parameter values ranging from $m_{pts} = 8$ to $m_{pts} = 15$
- 8 runs for each algorithm

TABLE I: Comparisons of all approaches in high resolution [Best (Highlighted)]

(a) *Overall* Comparison

	m_{pts}	ε	m_{clSize}	C	CH	DB	D	S	XB	
DBSCAN	-	-	-	-	-	-	-	-	-	
HDBSCAN Normal	9	-	9	0.0068	14234.6935	**0.1285**	0.0107	**0.6981**	50.9123	Fig. 3b
HDBSCAN Mode	10	-	42	**0.0062**	**22919.2289**	0.2213	0.0272	0.6374	21.5041	Fig. 4b
	13	-	55	0.0199	11660.6816	0.4520	**0.0444**	0.6404	11.3414	
	15	-	60	0.0101	17637.9780	0.2205	**0.0444**	0.6529	**8.0308**	Fig. 4c

(b) *Indices Best* Comparison

	m_{pts}	ε	m_{clSize}	C	CH	DB	D	S	XB	
DBSCAN	14	2028.24	-	0.0704	3898.0643	0.2088	0.0414	0.6640	18.3471	
HDBSCAN Normal	9	-	9	0.0068	14234.6935	**0.1285**	0.0107	**0.6981**	50.9123	Fig. 3b
	13	-	13	0.0064	17200.2566	0.1705	0.0268	0.6956	19.5653	
HDBSCAN Mode	10	-	42	**0.0062**	**22919.2289**	0.2213	0.0272	0.6374	21.5041	Fig. 4b
	15	-	60	0.0101	17637.9780	0.2205	**0.0444**	0.6529	**8.0308**	Fig. 4c

(c) *Default* Comparison

	m_{pts}	ε	m_{clSize}	C	CH	DB	D	S	XB	
DBSCAN	8	1092.41	-	0.0208	7302.8705	0.2957	0.0124	0.6396	137.5297	
HDBSCAN Normal	8	-	8	**0.0085**	12674.4550	**0.2010**	0.0187	**0.6889**	44.4553	Fig. 3a
HDBSCAN Mode	8	-	35	0.0143	**15247.0290**	0.3151	**0.0246**	0.6367	**28.6544**	Fig. 4a

Low Resolution Evaluation (Cluster Indices)

- Various parameter values ranging from $m_{pts} = 7$ to $m_{pts} = 15$
- 9 runs for each algorithm

TABLE II: Comparisons of all approaches in low resolution [Best (Highlighted)]

(a) *Overall* Comparison

	m_{pts}	ε	m_{clSize}	C	CH	DB	D	S	XB	
DBSCAN	-	-	-	-	-	-	-	-	-	
HDBSCAN Normal	13	-	13	0.0349	2329.5368	**0.1580**	0.0818	0.7074	5.5712	
	14	-	14	0.0179	3073.5154	0.4101	**0.1573**	0.7086	**2.8608**	Fig. 5b
HDBSCAN Mode	10	-	12	**0.0109**	**3303.1032**	0.1728	0.0913	0.6294	3.8294	Fig. 6b
	13	-	38	0.0180	3059.3662	0.4101	**0.1573**	**0.7086**	2.8609	Fig. 2a
	14	-	15	0.0179	3073.5154	0.4101	**0.1573**	0.7086	**2.8608**	Fig. 2b

(b) *Indices Best* Comparison

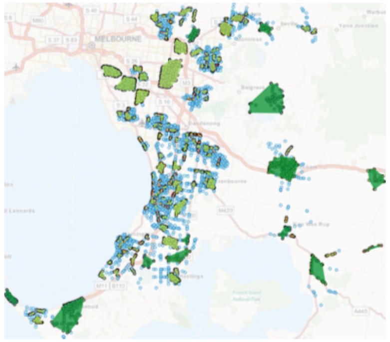
	m_{pts}	ε	m_{clSize}	C	CH	DB	D	S	XB	
DBSCAN	9	710.20	-	0.0310	1294.5558	0.4337	0.1098	0.6317	5.5642	
HDBSCAN Normal	14	-	14	0.0179	3073.5154	0.4101	**0.1573**	**0.7086**	**2.8608**	Fig. 5b
HDBSCAN Mode	10	-	12	**0.0109**	**3303.1032**	**0.1728**	0.0913	0.6294	3.8294	Fig. 6b

(c) *Default* Comparison

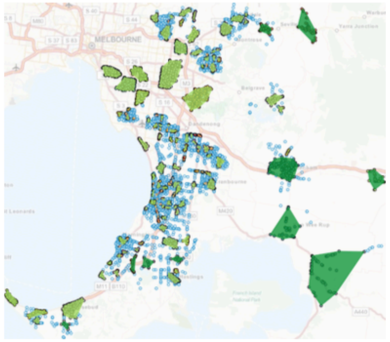
	m_{pts}	ε	m_{clSize}	C	CH	DB	D	S	XB	
DBSCAN	7	586.15	-	0.0362	627.2269	0.6294	0.0291	0.5974	78.3715	
HDBSCAN Normal	7	-	7	0.0140	2692.1316	0.2387	0.0574	0.6720	6.3584	
HDBSCAN Mode	7	-	6	**0.0124**	**2968.1770**	**0.2198**	**0.0745**	**0.6771**	**5.2215**	Fig. 6a

Visualization (High Resolution)

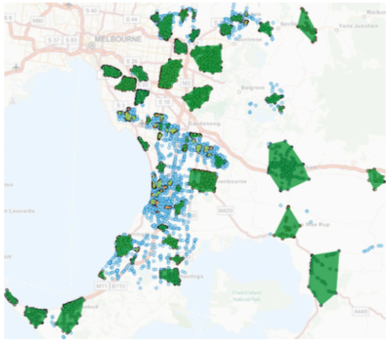
- HDBSCAN Normal (left)
- HDBSCAN Mode (right)



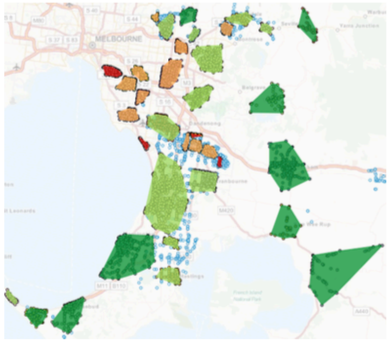
(a) $m_{pts} = 8$ (Default)



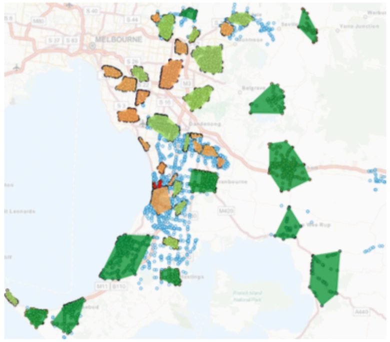
(b) $m_{pts} = 9$ (Best)



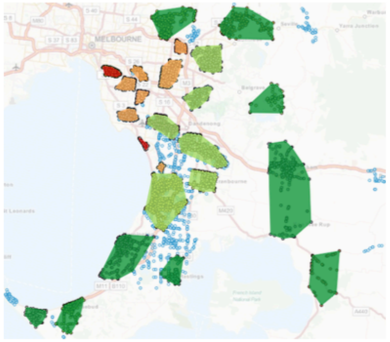
(c) $m_{pts} = 13$ (Best)



(a) $m_{pts} = 8$ (Default)



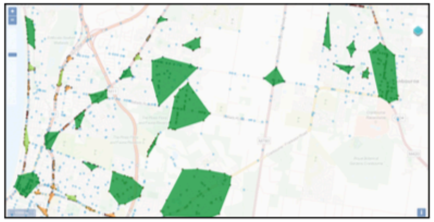
(b) $m_{pts} = 10$ (Best)



(c) $m_{pts} = 15$ (Best)

Visualization (Low Resolution)

HDBSCAN Normal



(a) $m_{pts} = 7$ (Default)

(b) $m_{pts} = 14$ (Best)

HDBSCAN Mode

(a) $m_{pts} = 7$ (Default)

(b) $m_{pts} = 10$ (Best)

Conclusion

- Adaptive clustering method is proposed to enable an interactive data exploration on the map.
- The method allows no parameter input from users for the purpose of data clustering and is able to adjust with various zoom levels for the cluster results.
- Two common density based clustering techniques are leveraged (DBSCAN and HDBSCAN).
- HDBSCAN outperforms DBSCAN for both high and low resolution experiment settings.
- The proposed HDBSCAN Mode provides better partitioning compared to HDBSCAN Normal (especially in high resolution experiment setting).
- HDBSCAN Mode is the most appropriate to run if the users need to get some reasonable partitioning on the accident data without any background knowledge required.

References

1. Van Leeuwen, M. (2014). Interactive data exploration using pattern mining. In *Interactive Knowledge Discovery and Data Mining in Biomedical Informatics* (pp. 169-182).
2. Goldstein, J., Roth, S. F., Kolojejchick, J., & Mattis, J. (1994). A framework for knowledge-based interactive data exploration. *Journal of Visual Languages & Computing*, 5(4), 339-363.
3. Liono, J., Salim, F. D., & Subastian, I. F. (2015, October). Visualization oriented spatiotemporal urban data management and retrieval. In Proceedings of the ACM First International Workshop on Understanding the City with Urban Informatics (pp. 21-26). ACM.
4. Kandogan, E. (2001, August). Visualizing multi-dimensional clusters, trends, and outliers using star coordinates. In *Proceedings of the seventh ACM SIGKDD international conference on Knowledge discovery and data mining* (pp. 107-116).
5. Zhu, J. Y., Lee, Y. J., & Efros, A. A. (2014). Averageexplorer: Interactive exploration and alignment of visual data collections. *ACM Transactions on Graphics (TOG)*, 33(4), 160.
6. Panta, S. R., Wang, R., Fries, J., Kalyanam, R., Speer, N., Banich, M., ... & Turner, J. A. (2016). A tool for interactive data visualization: Application to over 10,000 brain imaging and phantom mri data sets. *Frontiers in neuroinformatics*, 10, 9.
7. Lu, Y., Zhang, M., Li, T., Guang, Y., & Rishe, N. (2013, August). Online spatial data analysis and visualization system. In *Proceedings of the ACM SIGKDD workshop on interactive data exploration and analytics* (pp. 71-78). ACM.
8. Dimitriadou, K., Papaemmanouil, O., & Diao, Y. (2014, June). Explore-by-example: An automatic query steering framework for interactive data exploration. In *Proceedings of the 2014 ACM SIGMOD international conference on Management of data* (pp. 517-528).
9. Ester, M., Kriegel, H. P., Sander, J., & Xu, X. (1996, August). A density-based algorithm for discovering clusters in large spatial databases with noise. In *KDD* (Vol. 96, No. 34, pp. 226-231).
10. Campello, R. J., Moulavi, D., & Sander, J. (2013, April). Density-based clustering based on hierarchical density estimates. In Pacific-Asia Conference on Knowledge Discovery and Data Mining (pp. 160-172). Springer Berlin Heidelberg.
11. Liu, P., Zhou, D., & Wu, N. (2007, June). VDBSCAN: varied density based spatial clustering of applications with noise. In Service Systems and Service Management, 2007 International Conference on (pp. 1-4). IEEE.

References

12. Elbatta, M. T., & Ashour, W. M. (2013). A dynamic method for discovering density varied clusters. International Journal of Signal Processing, Image Processing and Pattern Recognition, 6(1), 123-134.
13. Hubert, L., & Schultz, J. (1976). Quadratic assignment as a general data analysis strategy. *British journal of mathematical and statistical psychology*, 29(2), 190-241.
14. Caliński, T., & Harabasz, J. (1974). A dendrite method for cluster analysis. Communications in Statistics-theory and Methods, 3(1), 1-27.
15. Davies, D. L., & Bouldin, D. W. (1979). A cluster separation measure. *IEEE transactions on pattern analysis and machine intelligence*, (2), 224-227.
16. Dunn, J. C. (1974). Well-separated clusters and optimal fuzzy partitions. *Journal of cybernetics*, 4(1), 95-104.
17. Rousseeuw, P. J. (1987). Silhouettes: a graphical aid to the interpretation and validation of cluster analysis. *Journal of computational and applied mathematics*, 20, 53-65.
18. Xie, X. L., & Beni, G. (1991). A validity measure for fuzzy clustering. *IEEE Transactions on pattern analysis and machine intelligence*, 13(8), 841-847.

Acknowledgment

• This research is partly supported by ARC Discovery Project (DP160102114) and RMIT Sustainable Urban Precinct Project ’iCommunity’.

Thank you: Q&A

- Email: flora.salim@rmit.edu.au