SUPPORTING INFORMATION

Source Apportionment of Particulate Matter by Application of Machine Learning Clustering Algorithms

Vikas Kumar[1], Manoranjan Sahu[2,1,3*], Pratim Biswas[4]

[1]Interdisciplinary Program in Climate Studies, Indian Institute of Technology Bombay, Mumbai 400076, India

[2]Aerosol and Nanoparticle Technology Laboratory, Environmental Science and Engineering Department, Indian Institute of Technology Bombay, Mumbai 400076, India

[3]Centre for Machine Intelligence and Data Science, Indian Institute of Technology Bombay, Mumbai 400076, India

[4]Aerosol and Air Quality Research Laboratory, University of Miami, College of Engineering, Coral Gables, FL, 33146, USA

Article Submitted to: Aerosol and Air Quality Research

Original supporting information prepared: October 17, 2021

Pages: 20

Contents: Four Figures (S1-S4); Two Tables (S1-S2)

2. Methodology

2.2.2. Clustering

The major steps involved in the clustering process are briefly discussed as follows:

- *Feature selection or extraction*: Clustering may exhibit different results depending upon the features (variables in the dataset, i.e., the chemical species mentioned in Table S1 in Supporting Information for this study), criteria, and approach used. Therefore, pre-processing the data before availing for analysis is essential for clustering to extract meaningful results and enhance cluster quality (Halkidi et al., 2001; Olivas et al., 2009; Aggarwal and Reddy, 2014). The objective of this step is to prepare the data for analysis with the help of pre-processing methods. Some pre-processing methods are cleaning and formatting the data, handling missing values, feature selection, and/or extraction. Pre-processing is done to facilitate the understanding of the underlying structure of the data. Feature selection refers to selecting the most prominent and salient features from the original data relevant to the analysis. The other features are pruned because their presence might degrade the quality of clusters and increase the computational time, memory, and complexity. Feature selection assures that the selected features actual physical meaning is retained (Jain et al., 1999; Halkidi et al., 2001; Xu and Wunsch, 2008; Olivas et al., 2009; Aggarwal and Reddy, 2014; Li et al., 2017).

  On the contrary, feature extraction comprises transformation processes for dimensionality reduction, normalization, and scaling on the data's original input features. Feature extraction produces novel features that may or may not have any physical significance (Jain et al., 1999; Xu and Wunsch, 2008; Olivas et al., 2009; Aggarwal and Reddy, 2014; Li et al., 2017). Due to the data being used earlier, missing values were already dealt with (refer to Sahu et al. (2011) for details). No feature selection was conducted so that the input data remains similar to Sahu et al. (2011), as this is a comparative study. Dimensionality reduction was performed using Principal Component Analysis (PCA) to visualize the clustering results on a 2D plot. PCA forms the covariance matrix from the set of eigenvectors computed from the data. PCA reduces the actual data into small dimension data by grouping eigenvectors with similar magnitude represented by eigenvalue (Flach, 2012).

- *Optimal number of clusters*: A fundamental and notable dilemma faced by researchers during clustering analysis is the estimation of the optimal number of clusters denoted by *k*, which is a trial-and-error process and needs to be pre-defined for most of the clustering algorithms (Milligan and Cooper, 1985; Gordon, 1998; Sugar and James, 2003; Pham et al., 2005). Various approaches to solve this problem have been developed whose details can be found in literature Milligan and Cooper (1985) and Pham et al. (2005). A non-parametric method proposed by Sugar and James (2003) known as the jump method was implemented in this study. In this method, distortion (d_k) is the average distance per dimension between each observation and its closest cluster center and measures the *within-cluster dispersion*. Mathematically, minimum distortion is the average Mahalanobis distance per dimension between X (a p dimensional random variable having a mixture distribution of G components, each with covariance Γ) and c_x (the cluster center closest to X) where $c_x \epsilon (c_1, c_2, \dots, c_k)$ (Mahalanobis, 1936).

$$d_k = \frac{1}{p} \min_{c_1,\dots,c_k} E \ [(X - \ c_x)^T \, \Gamma^{-1}(X - \ c_x)]$$

The approach is based on plotting a consistently decreasing monotonous curve between d_k and k known as distortion curve, which when transformed to an appropriate negative power (assumed $p/2$), exhibits a sharp jump at the optimal number of clusters. The distortion curves were plotted for the 2C and 8C datasets and shown in Figure S2 (a) and (b) respectively, in the Supporting Information. The distortion plot for the 2C dataset indicates a jump at k = 5 and a slightly smaller jump at k = 6, while the 8C dataset behaves similarly at k = 6 and 7, which shows an agreement with the findings of Sahu et al. (2011), where 6 and 7 sources were identified from 2C and 8C dataset respectively. To make the comparison relevant with the previous study, 6 and 7 were chosen as the optimal number of clusters for 2C and 8C datasets.

- *Clustering algorithm selection*: Various clustering algorithms have been developed and used extensively in various domains. However, there is no universal clustering algorithm to solve all problems (Xu and Wunsch, 2008). Different clustering algorithms produce different results. Therefore, it is essential to properly analyze the problem's domain and attributes before choosing the clustering strategy. This is the most crucial step of the entire clustering process. Two distinct clustering algorithms, viz. kMC and SC were applied on each dataset and the results were analyzed and compared with each other and Sahu et al. (2011) later in the paper. kMC was selected for this study as it is one of the most popular clustering algorithms. kMC finds its application in variety of domains because of its simple implementation, speed, and memory efficiency (Gan et al., 2007; Xu and Wunsch, 2008; Aggarwal and Reddy, 2014). SC was of interest in this study due to its similarity with PMF which is the mostly used receptor model technique for SA. Another feature of kMC and SC algorithms are its applicability in cases with less data which is not present in some advanced algorithms such as DBSCAN, OPTICS, BIRCH etc. which are generally applied when large databases are present. kMC and SC allows the modeler to choose number of clusters which is absent in algorithms such as DBSCAN and OPTICS (Zhang et al., 1996; Ankerst et al., 1999; Xu and Wunsch, 2008; Schubert et al., 2017). Option to choose the number of sources is an important aspect of this study as well as SA since we want to keep the number of sources constant with Sahu et al., (2011) for comparison with PMF results for validation. However, in future with bigger datasets more algorithms should be tested and compared for their suitability for SA.

- *Cluster validation*: Since clustering primarily deals with unlabelled data, assessing and validating clustering results is an indispensable step of the process. Clustering validation helps decide the algorithm or parameters that produce the result best suited for the data (Jain et al., 1999; Halkidi et al., 2001; Xu and Wunsch, 2008; Xiong and Li, 2013). Clustering validation measures/indexes evaluate the quality of clustering results and are broadly classified into external and internal categories. External indices analyze and compare the results with an external pre-specified structure, not available to the input data. Internal indices try to examine the results based on quantities involved in the data itself and are the only option when no external information is available which is the case for real-world data (Jain et al., 1999; Halkidi et al., 2001; Xu and Wunsch, 2008; Xiong and Li, 2013). The internal indices are primarily based on two criteria: intra-cluster homogeneity (a measure of how densely packed or compact the clusters are) and inter-cluster separability (a measure of how distant the clusters are from each other) (Milligan and Cooper, 1985; Halkidi et al., 2001;

Rokach and Maimon, 2005; Xiong and Li, 2013; Aggarwal and Reddy, 2014). Two internal indices are used for validation of the results in this study which are explained below.

(a) Calinski-Harabasz Index (CH): For a dataset with *n* number of data points and *k* clusters, CH (Caliński and Harabasz, 1974) index is the ratio of the sum of between and within-cluster dispersion (sum of distances squared) and can be calculated as:

$$CH\ (k) = \frac{(n-k)\ Tr(B(k))}{(k-1)\ Tr(W(k))}$$

where *Tr(B(k))* and *Tr(W(k))* are the traces of the between and within-cluster scatter matrices *B(k)* and *W(k)* respectively and can be computed as (Milligan and Cooper, 1985; Rokach and Maimon, 2005; Gan et al. 2007):

$$Tr\ \big(B(k)\big) = \sum_{i=1}^{k} |C_i|(z_i\ -\ z)^T(z_i - z)$$

$$Tr\ \big(W(k)\big) = \sum_{i=1}^{k} \sum_{x\ \epsilon\ c_i} (x\ -\ z_i)^T(x\ -\ z_i)$$

where z and z_i are the means of the entire dataset and cluster c_i respectively. A Higher CH score represents dense and well-separated clustering (Caliński and Harabasz, 1974).

(b) *Davis Bouldin Index (DB)*: The Davies-Bouldin (DB) index (Davies and Bouldin, 1979) does not depend on the number of clusters as well as the clustering algorithm but attempts to maximize the between-cluster distance while minimizing the distance between the centroid and other points of the cluster. The DB index can be calculated as:

$$DB\ (k) = \frac{1}{k}\sum_{i=1}^{k} R_i$$

$$R_i = \max_{j \neq i} \left(\frac{e_i + e_j}{D_{ij}} \right)$$

where R_i is the individual cluster index, D_{ij} is the distance between the centroids of clusters i and j, and e_i and e_j are the average errors for clusters i and j respectively. The DB index varies from 0 to 1, where values closer to 0 represent better clustering (Milligan and Cooper, 1985; Rokach and Maimon, 2005; Gan et al., 2007).

- *Result interpretation*: The main objective of clustering is to unravel hidden structures in the data that provide significant and relevant information about the data and the application domain for which experts in the field are required to thoroughly investigate the clustering results and validate them with the experimental or practical evidence (Halkidi et al., 2001; Xu and Wunsch, 2008; Olivas et al., 2009). Since this is a source apportionment study, the clusters obtained represent various sources of $PM_{2.5}$. The sum of aerosol species was calculated for each cluster. Afterward, each cluster was assigned to a source based on the source profiles available in the literature, which will be discussed in detail in the results.

2.2.3 k- Means Clustering

Consider a dataset $D = \{x_1, x_2, x_3, \ldots, x_n\}$ having k number of clusters where $C = \{c_1, c_2, c_3, \ldots, c_k\}$ is the set of cluster centroids. The SSE can be calculated using:

$$SSE = \sum_{j=1}^{k}\sum_{i=1}^{n} |x_i - c_k|^2$$

where x_i is a point and c_k is the centroid of cluster C_k.

Morissette and Chartier (2013) compared three kMC algorithms namely Forgy/Lloyd (Forgy, 1965; Lloyd, 1982), MacQueen (MacQueen, 1967), and Hartigan and Wong (Hartigan and Wong, 1979). The comparison concluded that the Forgy/Lloyd algorithm could create empty clusters that might not suit this study. Hartigan and Wong only optimize the within-cluster sum of squares. The MacQueen algorithm was used in this study as its optimization function is the total sum of squares:

k- Means Clustering Algorithm (MacQueen, 1967)

```
Input: Dataset D, Number of Clusters k

Initialization: Select k initial cluster centers
repeat
    Assign each object in the dataset to the nearest cluster using Euclidean distance
    Recompute the centroid of each cluster based on the current partition
until the convergence criterion is achieved i.e., no change in cluster centroid

Output: Clusters {$C_1$, $C_2$, $C_3$, …, $C_k$} for the dataset D
```

2.2.4 Spectral Clustering

Consider an undirected graph $G = (V, E)$ where V and E are the set of vertices and edges respectively of the graph G. The Laplacian matrix also known as unnormalized graph Laplacian $L(G)$ for the graph G is a n x n matrix defined by,

$$L(G) = D(G) – W(G)$$

where $D(G)$ is the diagonal matrix constructed with the degrees of node or vertex. A vertex's degree can be defined as the number of edges connected to a vertex for an undirected graph. $W(G)$ is a symmetric matrix (since G is undirected) known as adjacency or similarity matrix for *G. w (i, j), t*he elements of $W(G)$ represent the weight of the edge connected by the nodes i and j where

$$w\,(i,j) = \begin{Bmatrix} 1, if\ there\ is\ an\ edge\ joining\ vertices\ i\ and\ j \\ 0, otherwise \end{Bmatrix}$$

and each node or vertex represents a data point (Pothen et al., 1990; Merris, 1994; Meila and Shi, 2001; Newman, 2006; Nadler and Galun, 2006; Filippone et al., 2008; Chi et al., 2009; Clarke et al., 2009; Hastie et al., 2009; Flach, 2012; Albalate and Minker, 2013; Celebi and Aydin, 2016). The unnormalized Laplacian matrix L(G) is a symmetric matrix with n non-negative, real eigenvalues where $0 = \lambda_1 \leq \lambda_2 \leq \ldots \leq \lambda_n$. The smallest eigenvalue of L(G) is 0 and the corresponding eigenvector is constant and equivalent to 1 (Von, 2007; Liu and Han, 2013; Aggarwal and Reddy, 2014). Similar to unnormalized graph Laplacian, there are two variants of normalized graph Laplacian defined by

$$L_{sym}(G) = D(G)^{\frac{-1}{2}} L(G) D(G)^{\frac{-1}{2}} = I - D(G)^{\frac{-1}{2}} W(G) D(G)^{\frac{-1}{2}}$$

$$L_{rm}(\mathrm{G}) = D(G)^{-1} L(G) = I - D(G)^{-1} W(G)$$

where $L_{sym}(G)$ and $L_{rm}(G)$ refer to a symmetric matrix and the random walk perspective Laplacian matrix respectively and *D(G)* and *L(G)* have usual meanings.

SC reduces the partitioning problem to the classic linear algebra problem of solving the eigenvalues and vectors for a matrix ($Lx = \lambda Dx$) (Shi and Malik, 2000; Meila and Shi, 2001; Ng et al., 2002; Filippone et al., 2008; Celebi and Aydin, 2016). The SC family which has multiple variants, can be summarized as a three-step process: (a) construction of similarity graph for all the data points; (b) formation of low dimensional representation of the data known as spectral embedding in a space with the application of eigenvectors spectrum of the Laplacian matrix of the similarity graph; (c) application of classical clustering algorithm such as *k*-means for the spectral embedding partition (Kannan et al., 2004; Aggarwal and Reddy, 2014).

The first step for SC is to construct a similarity matrix for the nodes of graph G. The objective of the similarity matrix is to model the local geometric structure of the data points. The similarity matrix can be constructed in three ways: k-nearest neighbor (kNN), ε-neighborhood, and fully connected graphs. For this study, kNN graphs had been used which aims at connecting the vertex v_i with v_j if v_j is among the kNN of v_i and vice-versa and ignores the direction of the edges making the graph G undirected. Thereafter, the weight of the edges is calculated by the similarity of their endpoints (Von, 2007; Liu and Han, 2013; Aggarwal and Reddy, 2014). Let us assume that the dataset consists *n* arbitrary data points $x_1, x_2, x_3, \ldots x_n$ whose pairwise similarities $s_{ij} = s\,(x_i, x_j)$ has been calculated and stored as a matrix $S = (s_{ij})_{i,\,j\,=\,1\ldots n}$. The random walk version of the normalized SC used in the paper is shown below:

Normalized spectral clustering (random walk version) (Meila and Shi, 2001; Liu and Han, 2013)

Input: Similarity matrix $S \in R^{nxn}$, number k of clusters to construct.

```
Construct the similarity graph G. Let W(G) be the adjacency matrix and D(G) be the degree matrix.
Compute the unnormalized graph Laplacian L(G) where L(G) = D(G) – W(G).
Compute the top k eigenvectors $f_1, f_2, \ldots f_k$ of $Lf = \lambda Df$.
Construct the matrix $F \in R^{nxk}$ from $f_1, f_2, \ldots f_k$.
For i = 1, …, n, let $y_i \in R^k$ be the vector corresponding to the $i^{th}$ row of F.
Cluster the points $(y_i)_{i\,=\,1,\,\ldots,\,n}$ in $R^k$ with the k-means algorithm into clusters $C_1, \ldots, C_k$.
```

Output: Clusters $A_1, \ldots, A_k$ with $A_i = \{j \mid y_j \in C_i\}$

Table S-1. Statistical summary of aerosol species used in this study

Species	Concentration			
	Min	Max	AM	SD
$PM_{2.5}$	4488.97	44761.12	18232.56	10144.93
Al	2.73	663.81	43.56	74.16
Si	10.25	1308.15	119.12	161.25
S	203.21	5575.22	1647.39	1249.18
K	15.26	225.02	63.65	34.51
Ca	9.71	927.27	93.41	103.94
Ti	0	48.13	5.93	5.94
Mn	0.46	33.67	3.78	3.73
Fe	19.54	922.57	124.30	112.48
Cu	0.31	23.92	3.48	2.96
Zn	2.38	487.75	25.68	45.06
Se	0	13.30	2.92	2.38
Pb	0.18	70.66	5.10	6.73
OC[a]	372.35	9537.68	3401.67	1735.65
EC[a]	124.49	3208.81	693.91	553.14
O1TC[b]	0	1145.73	232.41	231.46
O2TC[b]	4.21	1890.96	632.52	401.81
O3TC[b]	32.02	1803.35	539.51	312.47
O4TC[b]	153.26	1828.98	542.21	268.29
OPTRC[b]	3.53	2683.82	740.58	414.7
E1TC[b]	128.53	4152.61	1031.61	791.02
E2TC[b]	0	575.52	27.21	61.55
E3TC[b]	0	8.13	0.12	0.85

Note: a: Species in 2C dataset only, b: Species in 8C dataset only, $PM_{2.5}$ is in µg/m^3 while other elements are in ng/m^3

Table S-2. Validity indices for clustering performance

Index	**Dataset**			
	2C		**8C**	
	kMC	SC	kMC	SC
CH	129.35	436.95	140.92	468.14
DB	0.69	0.26	0.57	0.19

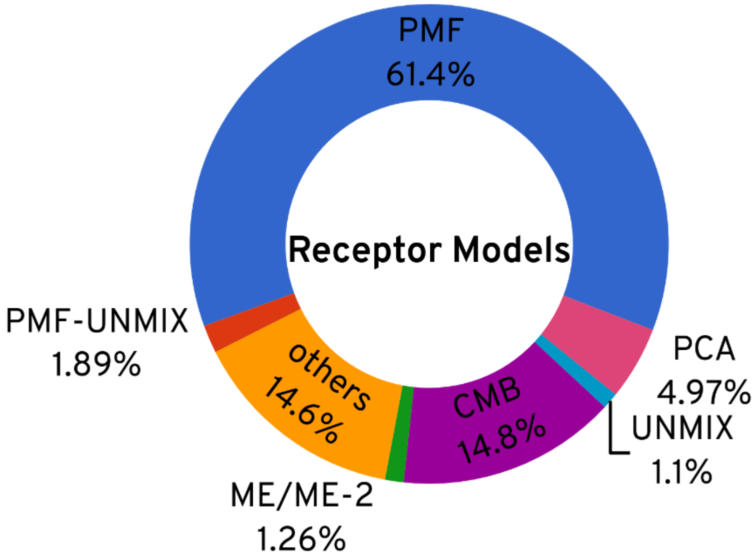

Figure S-1. Receptor models used in SA studies during 1990-2019 (Karagulian et al., 2015; Hopke et al., 2020)

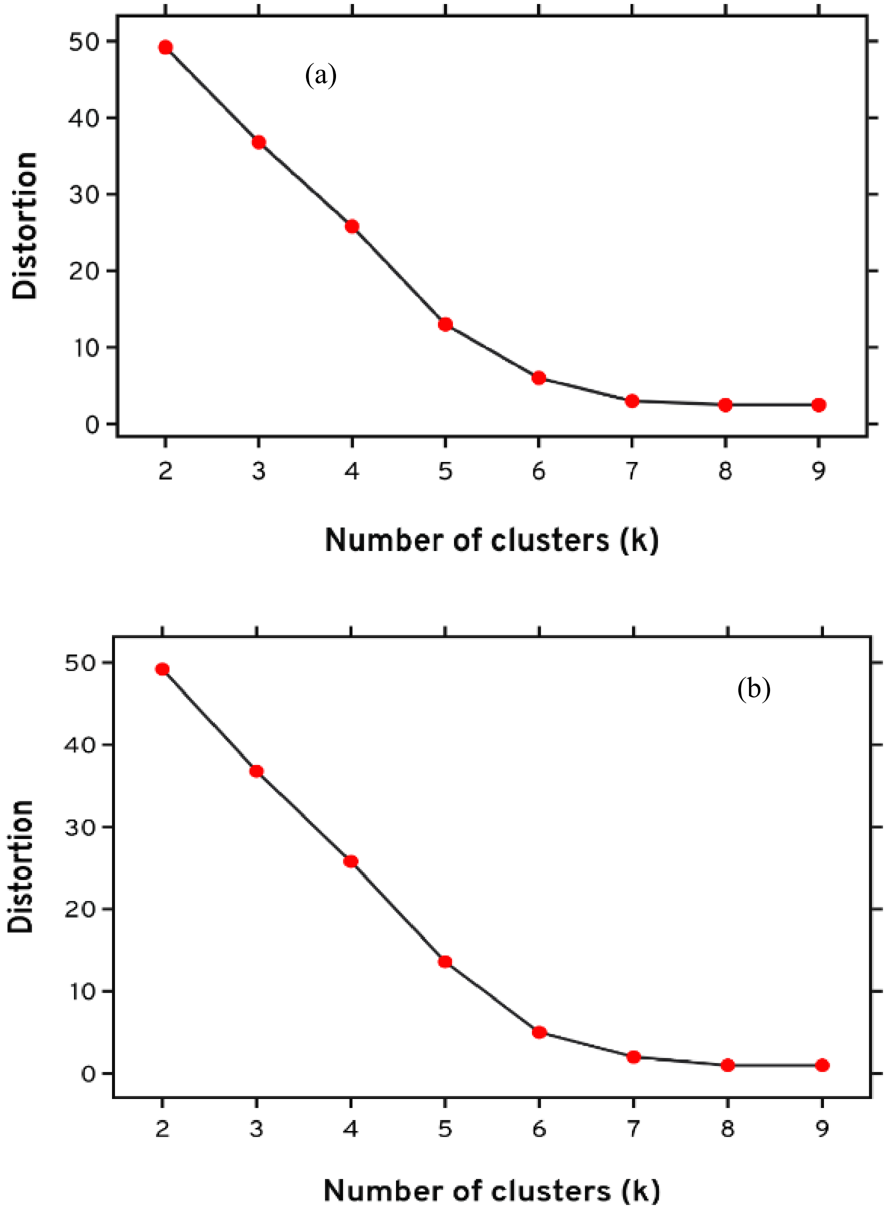

Figure S-2. Distortion curve for (a) 2C and (b) 8C data set

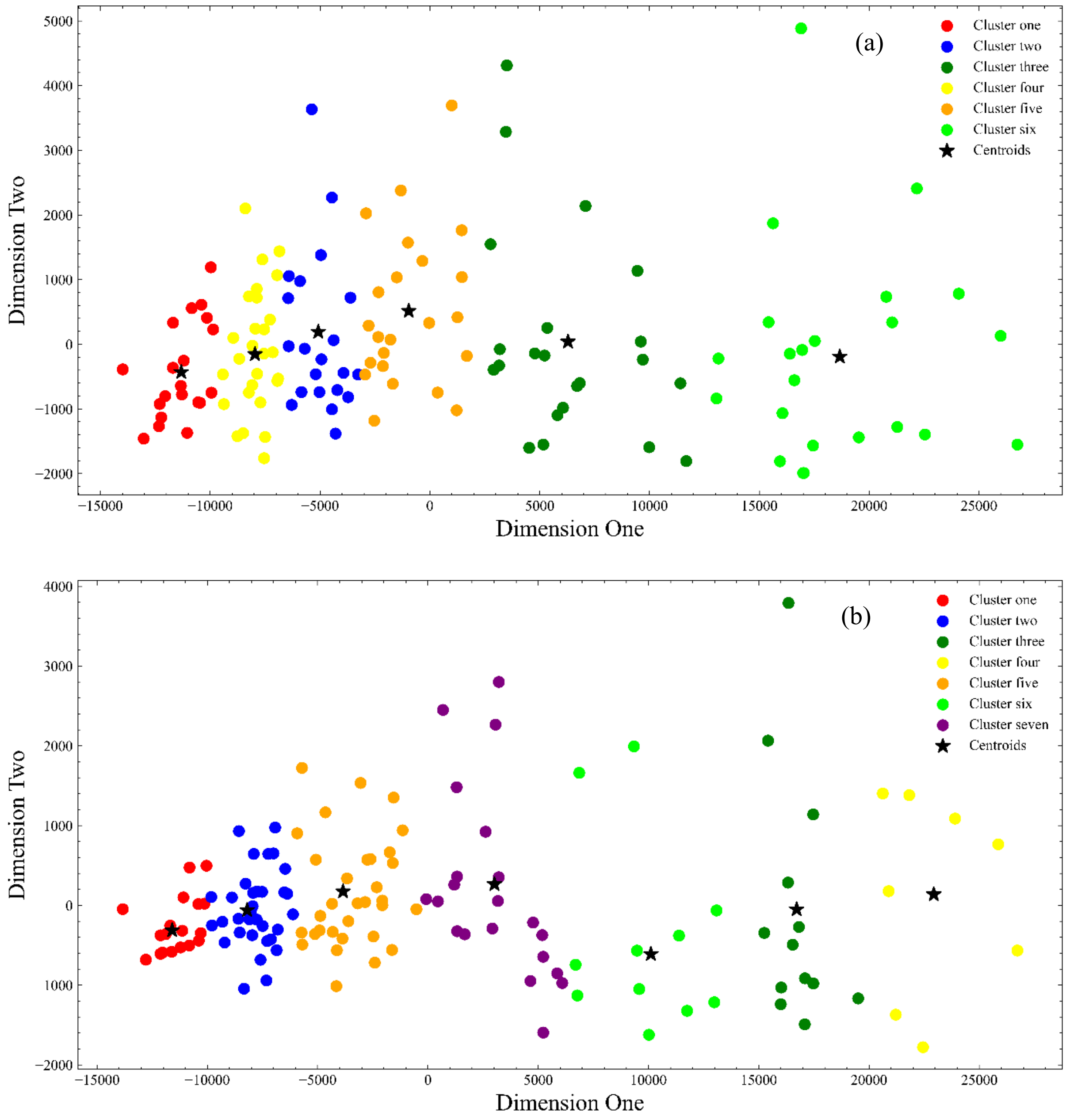

Figure S-3. kMC algorithm result for (a) 2C and (b) 8C data set

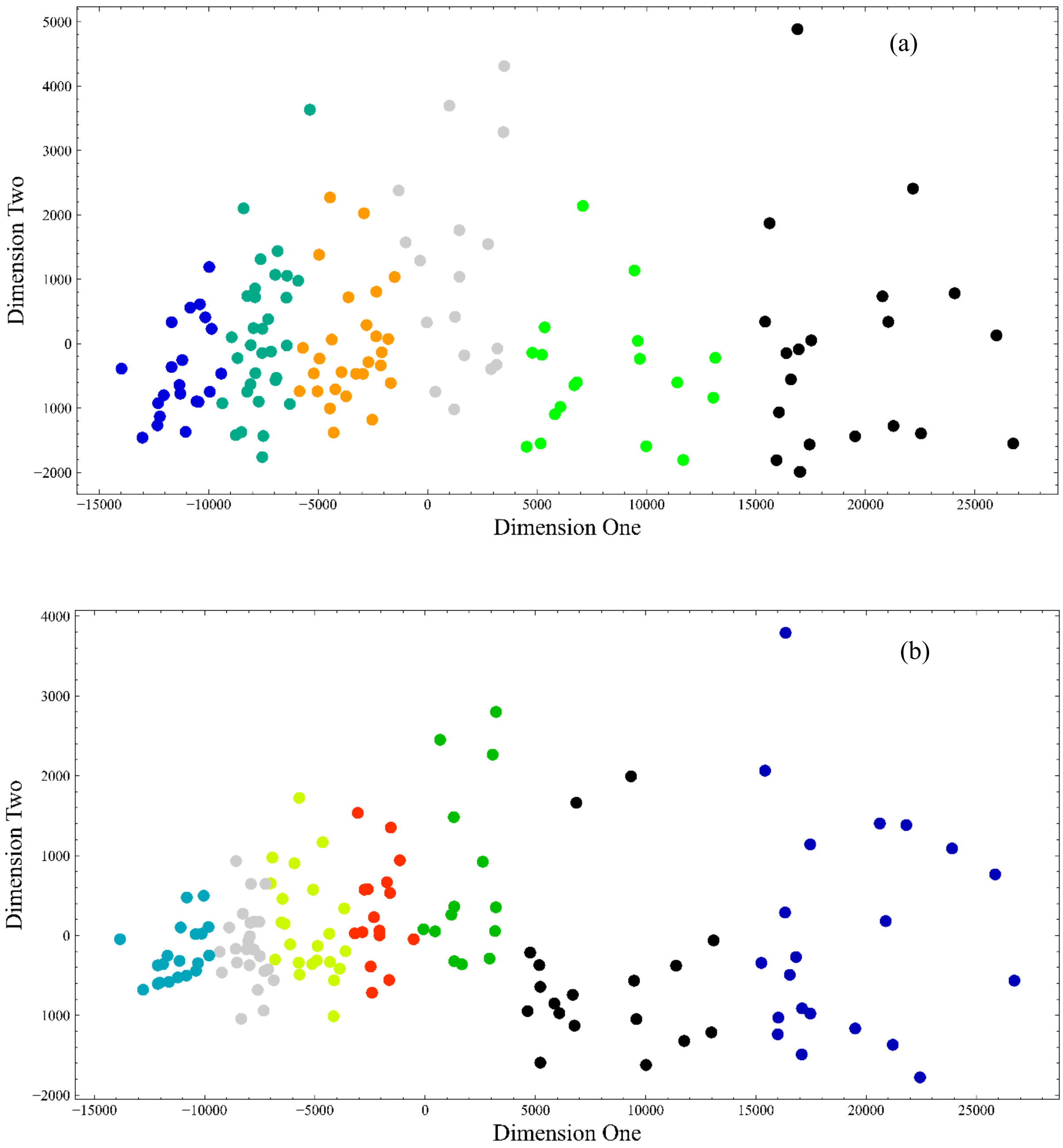

Figure S-4. SC algorithm results for (a) 2C and (b) 8C data set

References

Aggarwal, C.C., Reddy, C.K. (2014). Data clustering: algorithms and applications. Chapman And Hall/CRC.

Albalate, A., Minker, W. (2013). Semi-supervised and unsupervised machine learning novel strategies. Hoboken, NJ, USA John Wiley & Sons, Inc.

Ankerst, M., Breunig, M.M., Kriegel, H.-P., Sander, J. (1999). OPTICS: ordering points to identify the clustering structure. ACM SIGMOD Record 28, 49–60. https://doi.org/10.1145/304181.304187

Calinski, T., Harabasz, J. (1974). A dendrite method for cluster analysis. Commun. Stat. Theory Methods 3, 1–27. https://doi.org/10.1080/03610927408827101

Celebi, M.E., Aydin, K. (2016). Unsupervised learning algorithms. Springer, Cham.

Chi, Y., Song, X., Zhou, D., Hino, K., Tseng, B.L. (2009). On evolutionary spectral clustering. ACM Trans Knowl Discov Data 3, 1–30. https://doi.org/10.1145/1631162.1631165

Clarke, B., Fokoue, E., Zhang, H.H. (2009). Principles and theory for data mining and machine learning. Springer, New York.

Davies, D.L., Bouldin, D.W. (1979). A cluster separation measure. IEEE Trans. Pattern Anal. Mach. Intell., 224–227. https://doi.org/10.1109/TPAMI.1979.4766909

Filippone, M., Camastra, F., Masulli, F., Rovetta, S. (2008). A survey of kernel and spectral methods for clustering. Pattern Recognit. 41, 176–190. https://doi.org/10.1016/j.patcog.2007.05.018

Flach, P. (2012). Machine learning: the art and science of algorithms that make sense of data. Cambridge University Press.

Forgy, E. (1965). Cluster analysis of multivariate data: efficiency versus interpretability of classification. Biometrics 21, 768–769.

Gan, G., Ma, C., Wu, J. (2007). Data clustering: theory, algorithms, and applications. Soc. Ind. Appl. Math.

Gordon, A.D. (1998). Cluster validation, data Science, classification, and related methods. Springer Japan, Tokyo. https://doi.org/10.1007/978-4-431-65950-1

Halkidi, M., Batistakis, Y., Vazirgiannis, M. (2001). On clustering validation techniques. J. Intell. Inf. Syst. 17, 107–145. https://doi.org/10.1023/a:1012801612483

Hartigan, J.A., Wong, M.A. (1979). Algorithm AS 136: a k-means clustering algorithm. Appl. Stat. 28, 100. https://doi.org/10.2307/2346830

Hastie, T., Tibshirani, R., Friedman, J. (2009). The elements of statistical learning: data mining, inference, and prediction. Springer, New York.

Hopke, P.K., Dai, Q., Li, L., Feng, Y. (2020). Global review of recent source apportionments for airborne particulate matter. Sci. Total Environ. 740, 140091. https://doi.org/10.1016/j.scitotenv.2020.140091

Jain, A.K., Murty, M.N., Flynn, P.J. (1999). Data clustering: a review. ACM Comput. Surv. 31, 264–323. https://doi.org/10.1145/331499.331504

Kannan, R., Vempala, S., Vetta, A. (2004). On clusterings: Good, bad and spectral. J. ACM 51, 497–515. https://doi.org/10.1145/990308.990313

Karagulian, F., Belis, C.A., Dora, C.F.C., Prüss-Ustün, A.M., Bonjour, S., Adair-Rohani, H., Amann, M. (2015). Contributions to cities' ambient particulate matter (PM): A systematic review of local source contributions at global level. Atmos. Environ. 120, 475–483. https://doi.org/10.1016/j.atmosenv.2015.08.087

Li, J., Cheng, K., Wang, S., Morstatter, F., Trevino, R.P., Tang, J., Liu, H. (2017). Feature selection: a data perspective. ACM Comput. Surv. 50, 1–45. https://doi.org/10.1145/3136625

Liu, J., Han, J. (2013). Spectral clustering. In Data Clustering: Algorithms and Applications.

Lloyd, S. (1982). Least squares quantization in PCM. IEEE Trans. Inf. Theory 28, 129–137. https://doi.org/10.1109/tit.1982.1056489

MacQueen, J. (1967). Some methods for classification and analysis of multivariate observations, in: In proceedings of the fifth Berkeley symposium on mathematical statistics and probability, pp. 281–297.

Mahalanobis, P.C. (1936). On the generalized distance in statistics, in: Proceedings of the National Institute of Science, India, pp. 49–55.

Meila, M., Shi, J. (2001). A random walks view of spectral segmentation, in: Proceedings of the Eighth Int. Conf. Artif. Intell.

Merris, R. (1994). Laplacian matrices of graphs: a survey. Linear Algebra Its Appl. 197-198, 143–176. https://doi.org/10.1016/0024-3795(94)90486-3

Milligan, G.W., Cooper, M.C. (1985). An examination of procedures for determining the number of clusters in a data set. Psychometrika 50, 159–179. https://doi.org/10.1007/bf02294245

Morissette, L., Chartier, S. (2013). The k-means clustering technique: general considerations and implementation in Mathematica. Tutor Quant Methods Psychol 9, 15–24. https://doi.org/10.20982/tqmp.09.1.p015

Nadler, B., Galun, M. (2006). Fundamental limitations of spectral clustering, in: Adv. Neural Inf. Process. Syst., pp. 1017–1024.

Newman, M.E.J. (2006). Finding community structure in networks using the eigenvectors of matrices. Phys. Rev. E 74. https://doi.org/10.1103/physreve.74.036104

Ng, A., Jordan, M., Weiss, Y. (2002). On spectral clustering: analysis and an algorithm. Adv. Neural Inf. Process. Syst. 14, 849–856.

Olivas, E.S., Guerrero, J.D.M., Martinez-Sober, M., Magdalena-Benedito, J.R., Serrano, L. (Eds.). (2009). Handbook of research on machine learning applications and trends: algorithms, methods, and techniques. IGI Global.

Pham, D.T., Dimov, S.S., Nguyen, C.D. (2005). Selection of K in K-means clustering. Proc. Inst. Mech. Eng., Part C: J. Mech. Eng. Sci. 219, 103–119. https://doi.org/10.1243/095440605x8298

Pothen, A., Simon, H.D., Liou, K.-P. (1990). Partitioning sparse matrices with eigenvectors of graphs. SIAM J. Matrix Anal. Appl. 11, 430–452. https://doi.org/10.1137/0611030

Rokach L., Maimon O. (2005) Clustering methods. In: Maimon O., Rokach L. (eds) Data Mining and Knowledge Discovery Handbook. Springer, Boston, MA. https://doi.org/10.1007/0-387-25465-X_15

Schubert, E., Sander, J., Ester, M., Kriegel, H.P., Xu, X. (2017). DBSCAN revisited, revisited: why and how you should (still) use DBSCAN. ACM Transactions on Database Systems 42, 1–21. https://doi.org/10.1145/3068335

Shi, J., Malik, J. (2000). Normalized cuts and image segmentation. IEEE Trans. Pattern Anal. Mach. Intell. 22, 888–905. https://doi.org/10.1109/34.868688

Sugar, C.A., James, G.M. (2003). Finding the number of clusters in a dataset. J Am Stat Assoc 98, 750–763. https://doi.org/10.1198/016214503000000666

Von Luxburg, U. (2007). A tutorial on spectral clustering. Stat. Comput. 17, 395–416. https://doi.org/10.1007/s11222-007-9033-z

Xiong, H., Li, Z. (2013). Clustering validation measures. In Data clustering: algorithms and applications.

Xu, R., Wunsch, D.C. (2008). Clustering. John Wiley and Sons, Piscataway, NJ.

Zhang, T., Ramakrishnan, R., Livny, M. (1996). BIRCH: an efficient data clustering method for very large databases. ACM SIGMOD Record 25, 103–114. https://doi.org/10.1145/235968.233324