Organization Component Analysis: The method for extracting insights from the shape of cluster

1st Kaveh Mahdavi
Computer Architecture
Universitat Politècnica de Catalunya
Barcelona Supercomputing Center
Barcelona, Spain
kaveh.mahdavi@bsc.es

2nd Jesus Labarta Mancho
Computer Architecture
Universitat Politècnica de Catalunya
Barcelona Supercomputing Center
Barcelona, Spain
jesus.labarta@bsc.es

3rd Judit Gimenez Lucas
Computer Architecture
Universitat Politècnica de Catalunya
Barcelona Supercomputing Center
Barcelona, Spain
judit.gimenez@bsc.es

***Abstract*—Clustering analysis is widely used to stratify data in the same cluster when they are similar according to specific metrics. The process of understanding and interpreting clusters is mostly intuitive. However, we observe each cluster has unique shape that comes out of metrics on data, which can represent the organization of categorized data mathematically. In this paper, we apply novel topological based method to study potentially complex high-dimensional categorized data by quantifying their shapes and extracting fine-grain insights about them to interpret the clustering result. We introduce our Organization Component Analysis method for the purpose of the automatic arbitrary cluster-shape study without assumption about the data distribution. Our method explores a topology-preserving map of a data cluster manifold to extract the main organization structure of a cluster by the leveraging of the self-organization map technique. To do this, we represent self-organization map as graph. We introduce organization components to geometrically describe the shape of cluster and their endogenous phenomena. Specifically, we propose an innovative way to measure the alignment between two sequences of momentum changes on geodesic path over the embedded graph to quantify the extent to which the feature is related to a given component. As a result, we can describe variability among stratified data, correlated features in terms of lower number of organization components. We illustrate the utilization of our method by applying it to two quite different types of data, in each case mathematically detecting the organization structure of categorized data which are much profounder and finer than those produced by standard methods.**

***Index Terms*—Self-Organization Map, Topology-preservation, Sequence similarity, Topological Data Analysis, Cluster Analysis**

I. Introduction

Cluster analysis is used as a matter of course throughout the experimental sciences to extract scientific information from data [1]. But it turns out that many well-performed segmentation results cannot be turned into profound insights easily. Because the process of making clusters is a generic mathematically oriented process but lacks the intuition and domain knowledge that is often required to interpret and drill down into the algorithmic results. However, cluster "Geometric Features" can still be disclosed from metrics on the dataset, which could represent mathematically the fine-grain structure of complex data.

Geometric features of data can reveal clusters of diverse shapes, sizes and densities as demonstrated in Fig.1. Clusters can be spherical (a), elongated (linear) (b), loop (c), tendril (d), and heterogeneous (e) [2]. We are interested in studying such features of data since we assume insight into the shape of scientifically relevant data, it has a good chance of giving insight into the science itself. Experience has shown that this assumption is a reasonable one. For example, the study of loops and their higher-dimensional analogues has recently offered insight into questions in biophysics [3] and natural-scene statistics [4]; and, the study of tendrils has recently offered insight into oncology [5]. However, aforementioned figures say we should not select a final set of model types and then we build individual model, but we should make modeling mechanism that can study all arbitrary shapes and computed easily. Therefore, the basic goal of this paper is to introduce a generalized method for studying geometric features of clustered data.

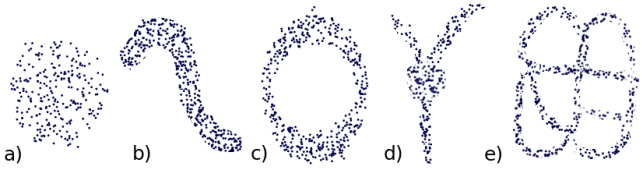


Fig. 1: Clusters of diverse shapes in $\mathbb{R}^2$

In our methodology, we propose a novel clustering result interpretation method by combining techniques from algebraic topology and statistical learning to give a quantitative basis for the study of the geometric features (shape) of a data cluster. It learns to interpret a data cluster with correct topology preservation map as modeling mechanism that doesn't make assumptions about the form of the mapping function. As a result, we can generally apply it to study the geometric feature of arbitrary cluster-shape. Moreover, this topology preservation map describes the idea of closeness in terms of relationships between sets rather than euclidean distance in the feature multidimensional space. The key objective is that once we represent high dimensional data by mapping functions on small space, we can efficiently give direct insight into the data.

Another important factor is the objective of clustering

DOI 10.1109/IJCNN52387.2021.9533650

algorithm. Since various discriminative definitions of a cluster-shape can be formulated, depending on it. For example, Density-based clustering such as DBSCAN [6], ISB-DBSCAN [7] or RNNDBSCAN [8] is famous for its capability of finding out arbitrary shape clusters from datasets. These approaches regard clusters as regions in the data space in which the data points are dense, and separated by regions of low data point density (noise). Since these methods apply a local cluster criterion to detect regions that may have an arbitrary shape and the data points inside a region may be arbitrarily distributed [9]. For such shape, it is very hectic to determine their innate structure or quantify them by applying correlation oriented techniques since the assumption of linear relationship between all variables behind these techniques do not often hold true for this arbitrary distributed data. Throughout this paper, we are interested to extract insights from arbitrary cluster-shape by not making assumptions about linearity. To do this, we propose a novel Directional Sequence Similarity method to locally measure the non-linear relationships between features.

Furthermore, in practice, not all features are important and relevant to the overall clustering task, many of them are often correlated and redundant, which may result in adverse effects such as low efficiency and poor performance, and also, dramatically increasing computational cost. Feature selection is one effective mean to identify relevant features for dimension reduction [10]. Once a reduced (active) feature subset is chosen, conventional clustering algorithm can then be applied by using the active features. Consequently, unselected (illustrative) features remain untouched that can be applicable to interpret the local prototype of cluster. In our approach, we consider active features to model the overall clustering space and also illustrative features to interpret the local establishment of each cluster with maximum comprehensiveness.

In this paper, we introduce a new method for the purpose of interpreting the clustering result through cluster-shape study which does not have assumption about data either distribution or shape. Our modeling mechanism organizes an augmented structure of a data cluster to represent its arbitrary shape by applying topology to develop tools for studying qualitative features of a cluster. This cluster-organization contains information which is equivalent to the topology-preserving map of finer-grained dense areas and their interconnection inside a cluster. We show how to efficiently and automatically interpret the topology-preserving map to extract not only the innate cluster structure, but also the causal factors associated with its formation by computing the non-linear local relationships between features. For that, we introduce a new Directional Sequence Similarity method to locally quantify it. Furthermore, we use illustrative features to obtain very detailed structure identification and a great detail on non-homogeneous local geometrical space within the identified clusters by generic clustering algorithms with using the active features.

The rest of this paper is organized as follows. The intuition motivating the cluster-shape interpretation is presented along with an overview of related work and methods in section 2. In section 3, the basic notions of cluster-shape interpreting and background techniques are defined. In section 4, our novel algorithm Organization Component Analysis (OCA) to decipher a cluster-shape with respect to its self-organization map structure is presented. The experimental result of our OCA method for the purpose of cluster-shape studying is illustrated in section 5. Section 6 concludes the paper with a summary and a brief discourse of future research.

II. Related work

There are various ways to explore the cluster-shape; through more rigorous analysis or by visualization and human interaction or external knowledge-based supervision or Topological Data Analysis.

Numerous statistical analytics techniques exist to study the multivariate data and the shape of cluster. Gaussian mixture model [11] is a probabilistic model for representing normally distributed subpopulations within an overall population. It can describe the shape of cluster as a sequence of overlapped subpopulations which have Gaussian distribution. Furthermore, in [12], [13], they discuss two multivariate analysis procedures: PCA and exploratory factor analysis, which can extract the latent structure of data. These techniques can identify a small set of synthetic variables, called eigenvectors or factors, that explain most of the total variation presented in the original variables and the shape of cluster. However, there are several requirements for a dataset such as: normality, homoscedasticity, linearity, sampling adequacy and no significant outliers. Taking advantage of the self-organization map technique, we developed an assumption-free and efficient cluster-shape analysis method.

The other approach is visual analysis that is usually a very intuitive and manual way to explore the underlying structure of the data, possibly incorporating human feedback into the process. HD-Eye method [14] explores different subspaces of the data in order to determine clusters in different feature-specific views of the data. IPCLUS [15] generates feature-specific views in which the data is well polarized. A polarized data is a 2D subset of features in which the data clearly separates out into clusters. Then a kernel-density estimator determines the views in which the data is well polarized. Finally, the shape of cluster is defined by exploring different views of the data. These methods are intuitive which make it difficult to separate the definition of cluster from the perception of an end-user and even to automatize them. Our approach mathematically combines the strengths of statistical and topological methods to eliminate the need for expert human visual analysis.

Supervision also can play an effective role, because it takes the specific goal and subjectivity of the analyst into consideration, which leads our insight of cluster-shape. In [16], they propose an interactive approach to constrain clustering in which the user can iteratively provide constraints as feedback to refine the clusters towards the desired concept. The results indicate that significant profounder insight can be made with only a few well-chosen constraints. Also, in [17], they describe an expectation maximization (EM) algorithm, penalized probabilistic clustering, which interprets pairwise

constraints as prior probabilities that two items should, or should not, be assigned to the same cluster. This formulation permits both hard and soft constraints allowing users to specify background knowledge even when it is uncertain or noisy. Normally, the supervision should be embedded inside the clustering algorithm, which lead to poor generalization and automation. Our proposed Organization Component Analysis method is not dependent on a particular clustering algorithm, therefore any clustering result can be analyzed by it efficiently.

Topological Data Analysis (TDA) is a recent and fast-growing field providing a set of new topological and geometric tools to study shape of possibly complex data [18]. In [19], they purpose the Mapper that is a mathematical tool to identify shape characteristics of datasets by applying topological method. The Mapper identifies local clusters within the data and then it studies the interaction between these small clusters by connecting them to form a graph whose shape captures aspects of the topology of the dataset. Mapper graphs associated to datasets preserve a wealth of information about the original shapes, but it is computationally expensive especially for massive datasets and its size grows rapidly with the number of data points, for that reason, Mapper takes transposed data matrix to build topological model. In our method, we apply the self-organization map (SOM) to efficiently learn and build a topology-preserving mapping that projects multi-dimensional data onto a lower 2D space by preserving the neighboring relations of the data.

The intention of our work is to purpose a novel generalized technique that can study arbitrary shape of clusters by leveraging of self-organization map and topology data analysis; thereby, that is an assumption-free and automated, and it does not dependent on any particular clustering algorithm. Especially, in our approach we apply topology-preserving mapping to optimize the computation cost and increase the efficiency.

III. BACKGROUND AND NOTATION

In this section, we discuss an overview of technique that we use leverage in our analytic approach. We firstly illustrate a brief mathematical notation, then we revisit the self-organization map algorithm.

We use $D = (d_1, d_2, ..., d_Q)$ to indicate a full dataset where $d_i \in R^M$. We suppose an active feature set M_a is selected and the rest of features M_l are illustrative ones, where $\forall M_a, M_l \subset M$, $M_a \cap M_l = \emptyset$ and $M_a \cup M_l = M$. Then, clustering algorithm (e.g. DBSCAN) is applied by using the active features to partition the Q observations into $h(\leq Q)$ clusters $\mathcal{CL} = \{cl_1, cl_2, ..., cl_h\}$ with discretionary shape. In this paper, we are interested in interpreting the shape of a particular cluster $X = (x_1, x_2, ..., x_N)$ where $X \in \mathcal{CL}$, $x_i \in R^M$ and $N \leq Q$.

A. Self-organization map

Kohonen's self-organizing map (SOM) is one of the most famous neural network models. Self-organization map fundamentally is a pattern recognition technique in multivariate data, in which intra-pattern relations among the features are grasped without the attendance of a potentially biased or subjective external influence [20].

The SOM often arranged a set of neurons in a 2D rectangular or hexagonal grid $\mathbb{T}$ in size n, to establish a discrete topological mapping of an input space, $X \in R^M$. Ω is the set of neuron indexes. The neurons are represented by set of weight vectors $V = \{v_1, v_2, ..., v_n\}$, where v_i is the weight vector associated with neuron i and is a vector of the same dimension $-M-$ of the input, n is the total number of neurons, and let r_i be the location vector of neuron i on the grid. At the start of the learning, all the weights are initialized to small random numbers. Then the algorithm repeats next two steps until the map converges in order to preserve maximum topological properties of the data on the map [21].

First at each time-step t, presents an input $x(t)$ at random, and selects the winner neuron:

$$\nu(t) = \underset{k \in \Omega}{\operatorname{argmin}} \|x(t) - v_k(t)\| \tag{1}$$

Second, update the weights of the winner and its neighbors:

$$\Delta v_k(t) = \alpha(t)\eta(\nu, k, t)[x(t) - v_\nu(t)] \tag{2}$$

where η is the neighborhood function which Gaussian form is often used in practice – more specifically:

$$\eta(\nu, k, t) = exp\left[-\frac{\|r_\nu - r_k\|^2}{2\sigma(t)^2}\right] \tag{3}$$

with σ representing the effective range of the neighborhood, and it is often decreasing with time.

The coefficients $\{\alpha(t), t \geq 0\}$, termed the 'adaptation gain', or 'learning rate', are scalar-valued that decrease monotonically, but satisfying:

$$0 < \alpha(t) < 1, \lim_{t\to\infty} \sum \alpha(t) \to \infty, \lim_{t\to\infty} \sum \alpha^2(t) < \infty \tag{4}$$

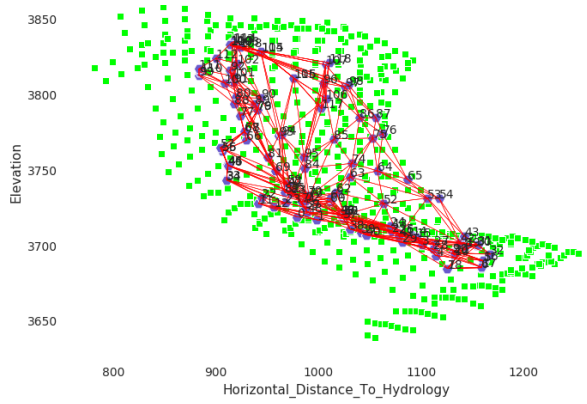
Furthermore, we apply "No Move" [20] to learn convergence mechanism in our approach. It considers stopping condition that defines no-improvement in SOM's status as no training samples changing their best match unit in a complete iteration of the training set. Using this condition, the training process is stopped as soon as it sees no-improvement.

As a result, the SOM can provide topologically preserved mapping [22] from input to output spaces, which includes grid $\mathbb{T}$ and weight vectors V. For SOM training, the weight vector associated with each neuron moves to become the centroid of a local group of input vectors. The group i is represented by its centroid vector v_i and the local groups are connected via topological space $\mathbb{T}$. We use it for vector quantization by subdividing a subset of points into micro-clusters having points close to each other locally.

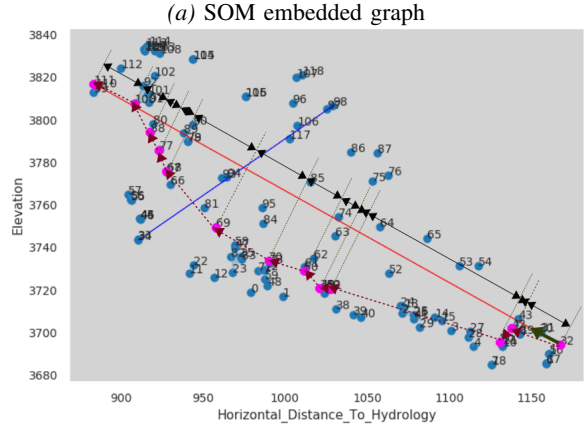
IV. ORGANIZATION COMPONENT ANALYSIS

In this section, we will introduce our Organization Component Analysis (OCA) method which analyses the shape of a particular cluster X to extract the Organization Components $OC = (oc_1, oc_2, ..., oc_C)$ where $oc_c \in R^{M_l}$ and $C \in \mathbb{Z} \cap [1, |M_a|]$.

First of all, train the SOM network to learn the topology-preserving map $(\mathbb{T}, V)$ and structure of the input data X. The key observation is that neurons that are adjacent to each other in the topology should also move close to each other in the input space, therefore it is possible to explore a high-dimensional inputs space in the two dimensions of the network topology.



(a) SOM embedded graph



(b) The blue and red lines are the first and second shape axes.

Fig. 2: Application of OCA to spatial pattern indicator from the Covertype dataset cluster C3, which is a part of our empirical study. The Horizontal_distance_to_hydrology and Elevation are selected active features. (a) shows the locations of the data points (green) and the weighted graph that is embedded in the SOM topology space. (b) The blue points are representing the SOM neurons, while the red and blue lines are major and minor axis, respectively. The rest of the items are described in the section IV in details.

In order to be able to compute pairwise relations between neurons, an undirected graph $G = (E, V, w)$ is embedded in the topological space $\mathbb{T}$. If G is represented in $\mathbb{T}$ such that the vertices (V) of G are distinct elements in $\mathbb{T}$, and an edge (E) in G is a simple arc connecting its two ends such that E preserves the grid structure of $\mathbb{T}$, also $w : E \to \mathbb{R}$ is a function mapping edges to their values which is euclidean distance between two ends of each edge in the active feature subspace $\forall M_a \subset M$. See Fig.2.a, the labeled purple points and the red lines represent the nodes and edges respectively.

Let $Va = [va_1, va_2, ..., va_n]$, $va_i \in R^{M_a}$ be active feature subset, then the weight of edge between neuron i and j is;

$$w_{ij} = \| va_i - va_j \|, \ w_{ij} \geq 0 \tag{5}$$

Then we compute the axis I_c of the cluster. The axis is the $(va_o, va_q) \in R^{M_a}$ endpoints of the longest line that can be drawn through the cluster topological space $\mathbb{T}$. Let ep be a set of the endpoints where $ep = \emptyset$ during the computing of the first (major) organization component and it will be updated gradually. The endpoints $o \neq q$ of the I_c belongs to:

$$\underset{i,j \in \{\Omega \setminus ep\}}{\operatorname{argmax}} \left[\|va_i - va_j\| + \sum_{k \in ep} (\|va_i - va_k\| + \|va_j - va_k\|) \right] \tag{6}$$

And the axis and its unit vector are (e.g., the red solid line and green solid arrow in Fig.2.b),

$$I_c = \overrightarrow{va_o va_q}, \ \hat{I}_c = \frac{I_c}{||I_c||} \tag{7}$$

Then we update endpoints set:

$$ep = ep \cup \{o, q\} \tag{8}$$

Next, we describe the organization of the cluster in the direction of $\hat{I}$ by capturing the continuing interaction between small local clusters (neurons) aligned with the I_c. To figure out this interaction we compute the geodesy path $P = (p_1, p_2, \ldots, p_{n'})$ in the graph G (where $P \subseteq \Omega$, $p_1 = o$, and $p_{n'} = q$) by Dijkstra's algorithm, which is shown as pink nodes in Fig.2.b.

Technically, the relation between the neurons in path P can describe the major axis establishment in R^M. In order to characterize the relationship between the neurons in P, we compute a sequence of forward difference vectors $\Delta = [\delta_1, \delta_2, ..., \delta_{n'-1}]$, $\delta_i \in R^M$ (Shown as red dotted arrows in Fig.2.b). Let $v_i \in R^M$ represent the weight vector of i_{th} neuron in P then

$$\delta_i \equiv |v_{i+1} - v_i| \oslash v_i, \ \ \forall i \in (P \setminus \{p_{n'}\}) \tag{9}$$

If you imagine standing at i_{th} neuron in R^M, the vector δ_i tells you the changes rate in direction of $i + 1_{th}$ neuron.

We propose a Directional Sequence Similarity (DSS) method to compute the similarity between sequence of differences in active subspace R^{M_a} and in each illustrative feature on path P. Let A be sequence of difference in active subspace R^{M_a} aligned with I_c (The length of black intervals in Fig.2.b),

$$A = ULA^T \tag{10}$$

Where L is diagonal matrix in size M with $l_{ii} = 1$ if $i \in M_a$ else $l_{ii} = 0$. To facilitate the computation, we define unit vector $U \in R^M$ with $u_i = \hat{I}_{ci}$ if $i \in M_a$ else $u_i = 0$.

Afterwards we compute sequential alignment between A and each $\delta'_i \in \Delta^T$, which are columns of Δ, to measure the relationship between directional sequence of changes in active subspace and each feature i belonged to illustrative subset M_l. Let $oc_c = [s_1, s_2, \ldots, s_{M'}]$ be the associated organization

component to the axis I_c, where $M' = |M_l|$ and s_i measures the sequence alignment between A and δ'_i:

$$s_i = \left| \frac{A.\delta'_i}{\| A \| \times \| \delta'_i \|} \right| \tag{11}$$

Where $0 \leqslant s_i \leqslant 1$, a sequence is identical with A when $s_i = 1$ or $s_i = 0$ non-identical. And if a s_i is significantly similar to A, the i_{th} illustrative feature can be interpreted as an influencing feature on the organization component oc_c.

For any cluster, we can identify $|M_a|$ organization components by repeating these steps. The first organization component corresponds to the major discrete curve that passes through the multidimensional active feature space and represents the interrelationship among a chain of local micro-clusters in direction of main axis. The next organization components correspond to the same concept that its associated endpoints have been selected by maximization sum of their euclidean distance to previously selected endpoints ep in the active feature multidimensional space. These components can describe the fine-grain interconnection inside a cluster. And the most influencing feature in each organization component can illustrate the gravitational force among a chain of local micro-clusters. Through these organization components, we can often find fine-grain patterns in categorized data that traditional methodologies fail to find. For example in, Fig.2.b, the red dotted arrows show the first organization component composition that vividly represents the main repetitive patterns among the original data points (See green points in Fig.2.a).

We summarize the complete OCA algorithm for extracting insights from the shape of cluster in Algorithm (1).

Algorithm 1 : OCA for cluster shape interpretation

Require: N data points with M features;
M_a : The active feature subset;
M_l : The illustrative feature subset;
n: Size of two-dimensional map;
$C \in \mathbb{Z} \cap [1, |M_a|]$: Number of Components;

Ensure: $ep = \{\}$ endpoints
$OC = \{\}$ Organization Compounds

1: Initiate SOM $\mathbb{T}$ in size n and train it until converges as discussed in Section $III.A$. Let $V = [v_1, v_2, ..., v_n]$, $v_i \in R^M$ contain neurons weight vectors.
2: Embedding the $G = (E, V, w)$ in the $\mathbb{T}$ and compute the w in R^{M_a} (Eq.5)
3: **for** c = 1 to C **do**
3.1: Select endpoints o, q of axis I_c (Eq.6)
3.2: Compute axis I_c and unit vector $\hat{I}_c$ (Eq.7)
3.3: $ep = ep \cup \{o, q\}$. (Eq.8)
3.4: Extract geodesy path $P = (p_1, p_2, \ldots, p_{n'})$ between o and q in G by Dijkstra's algorithm.
3.5: Compute sequence of forward difference vectors $\Delta = [\delta_1, \delta_2, ..., \delta_{n'-1}]$. (Eq.9)
3.6: Compute A sequence of changes in active subspace R^{M_a} aligned with I_c (Eq.10)
3.7: Compute $oc_c = [s_1, s_2, \ldots, s_{M'}]$ associated organization component to the axis I_c. Any s_i measures the sequence alignment between A and $\delta'_i \in \Delta^T$. (Eq.11)
3.8: $OC \leftarrow OC + oc_c$
4: **end for**
5: **return** OC

V. Application of OCA in the real data

In this section, we apply OCA to two datasets from diverse fields to show the implementation and application of our proposed OCA method for extracting insight from arbitrary cluster-shape. We analyzed datasets of (i) cartographic data of actual forest cover type; (ii) performance data of high performance computing (HPC) STREAM benchmark. We show that studying the deformation of topology preserved space is useful and efficient in detecting finer-grain pattern and relation among the stratified data. The innovation in our paper is to show that local geometrical feature of clusters is important and can mathematically lead to novel and profounder insights from the data. In continue, we discuss the OCA parameter selection and evaluation method then we go into the analyses of datasets.

A. Parameter Selection

Our OCA method has only one parameter, which is n in performing the self-organization map. The size of the map n is determined by calculating the number of neurons from the number of data points using $n \approx 5\sqrt{N}$, which is an integer close to the result of the right-hand side of the equation, and N is the number of observations [23].

B. Evaluation Quality of The Map

The important measure of the quality of the mapping is the topology preservation [24]. We calculate the topographic error, t_e, i.e. the proportion of all data vectors for which first and second Best Matching Units (BMU) are not adjacent units.

$$t_e = \frac{1}{N} \sum_{i=1}^{N} u(x_i) \tag{12}$$

where $u(x_i)$ is equal to 1 if first and second BMU are adjacent and 0 otherwise. So $t_e \in [0, 1]$, the map highly preserves topology when $t_e = 1$ or $t_e = 0$ poorly.

C. Identifying spatial patterns of wilderness sub-area.

The first application is the identification spatial patterns of wilderness sub-area. Identifying spatial patterns among potentially complex Geographical Information System (GIS) data in a consistent manner is a challenge in the field since sub-area can be small and have complex relationships. We show here that OCA can finely lead us to identify these spatial patterns by analyzing cluster-shape. We also identified interesting wilderness sub-area and their innate organization structure that may be important for geoscientists.

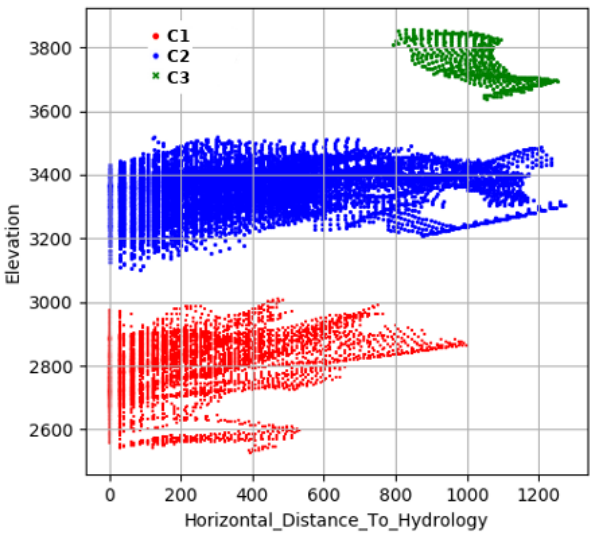


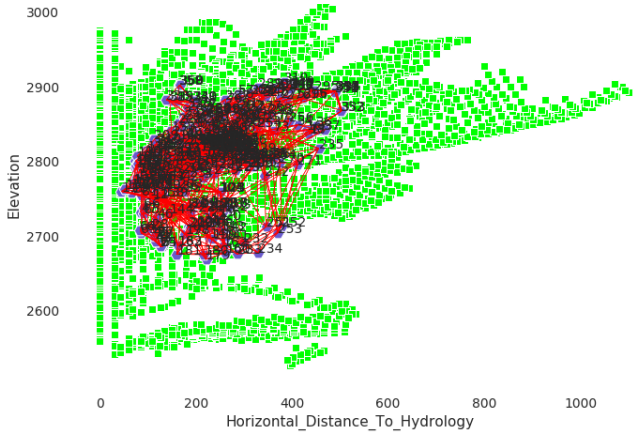
Fig. 3: Comanche Peak Wilderness Area, visualizations of the clustering result (DBSCAN $\varepsilon = 0.15$, $MinPts = 30$).

We use Covertype[1] dataset (580,112 variable pairs, 17 numerical GIS variables such as elevation, slope, aspect, distance to hydrology, and etc.), to demonstrate fine-grain spatial patterns that can be identified among GIS data by using our OCA approach. Instances in the dataset are drawn from four different wilderness areas from the Roosevelt National Forest in north Colorado: Rawah, Neota, Comanche Peak and Cache la Poudre, which are covered with seven different tree species. The organization components derived from the cover-type data of species vary from area to area, with some areas-species having particular pattern. We took the Comanche Peak areas as a benchmark (253,364 data points), which would tend to be more typical of the overall dataset, while this area would probably have Aspen as their primary major tree species, followed by Krummholz [25].

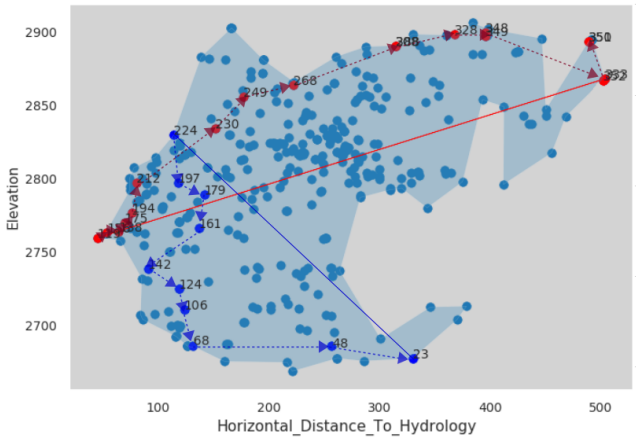
By applying Robust Independent Feature Selection [26] method, we find that "elevation" and "horizontal_distance_to_hydrology" are an excellent active feature subset for categorizing wilderness sub-area, since most of tree species in the studied wilderness areas grow within specific ranges of altitudes and available moisture in a given cell. Then we apply the DBSCAN on this area-species by tuned hyper-parameters $\varepsilon = 0.15$ and $MinPts = 30$ and using the active features. As a result, we stratify three distinct clusters in complex arbitrary shape, see Fig.3. We detect tendril cluster C1 in the lower part of that altitudinal zone. In contrast, we identify a small heterogeneous cluster C3 in highest elevation and an elongated cluster C2 with three tendrils.

We applied OCA in each cluster based on GIS data. In order to randomize the experiments, we conducted 10 OCA on each chosen cluster. Then, for each cluster, the average features influence as well as the standard deviations that have been computed over all analysis. OCA achieved to the $98\%, 95\%,$ and 84% topology preservation in average for cluster $C1$, $C2$ and $C3$ respectively by less than 300 iteration in average.

[1]https://archive.ics.uci.edu/ml/datasets/covertype



(a) SOM embedded graph



(b) The first and second Organization Component and axis

Fig. 4: Application of OCA to Covertype dataset cluster C1; (a) shows the data points and the SOM embedded graph.(b) The red and blue dashed arrows represent first and second Organization Component, respectively.

In Fig.4, as an example, we show plots of the organization components that we have derived from cluster $C1$. Plot (a) shows the embedded graph that is computed by our OCA method. The resulting graph has a structure shaped like a horizontal letter C. As shown in plot (b), our OCA method identified two spatial patterns in data. For first OC (red), the relationships are mainly aligned with increasing the active feature values where there is a long connected path, but in other OC (blue), the networks show a significantly decremental short path. Table.I presents the Directional Sequence Similarity that is computed by OCA for each cluster. In case of cluster $C1$, this very high degree of DSS (69%) is evident with "Slope" in first OC. There are geological issues that could explain such spatial pattern. The associated data points to this OC are mainly belonged to Catamount soil family that is geomorphically positioned in mountain slopes in nature. We also determined that the horizontal distance changes to roadway can describe the spatial pattern associated with the

TABLE I: Application of OCA to Comanche Peak Wilderness sub-areas, the features influencing in the first and second organization components for each cluster.

Feature	Cluster C1		Cluster C2		Cluster C3	
	OC1	OC2	OC1	OC2	OC1	OC2
Aspect	0.196	0.248	0.094	0.135	0.020	0.010
Slope	**0.688**	0.162	0.069	0.238	0.259	0.077
VDT_Hydro	0.247	0.305	0.202	0.131	0.148	0.095
HDT_Road	0.218	**0.462**	0.060	0.219	0.184	0.138
Hillshade_9am	0.169	0.206	0.049	0.216	0.192	0.120
Hillshade_Noon	0.176	0.219	0.059	0.182	0.361	**0.446**
Hillshade_3pm	0.183	0.253	0.09	0.149	0.105	0.067
HDT_Fire_Point	0.196	0.206	0.061	0.100	**0.753**	0.085
Hillshade_mean	0.175	0.222	0.060	0.182	0.191	0.144
Hillshade_9am_sq	0.171	0.215	0.053	0.203	0.353	0.246
DT_Hydro	0.164	0.201	**0.944**	**0.697**	0.293	0.358
Hillshade_Noon_sq	0.174	0.221	0.059	0.187	0.212	0.171
Hillshade_3pm_sq	0.177	0.238	0.072	0.167	0.148	0.103
cosine_slope	0.172	0.226	0.058	0.191	0.235	0.208
Interac_9amnoon	0.173	0.202	0.049	0.203	0.268	0.267
Interac_noon3pm	0.183	0.248	0.089	0.145	0.092	0.056
Interac_9am3pm	0.179	0.234	0.07	0.162	0.138	0.089

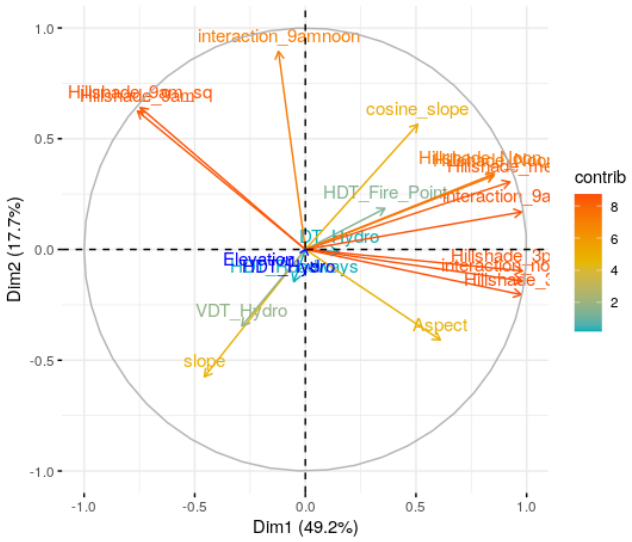
second OC. The associated data points to this OC are mainly belonged to Bullwark soil family that is geomorphically positioned in mountain faceted spurs. Note that the faceted spurs usually ends up to flat area which is appropriate place for making road. In case of cluster $C2$, we can see euclidean distance to hydrology identified as most influencing feature in first two organization components, with approximately 92% and 70% DSS. Interestingly, the rest of the features mostly do not show significant Directional Sequence Similarity with the active feature set. In cluster $C3$ case, the cluster is shaped with the natural fire lines as most influencing feature with DSS 76%.

In order to verify our result, we carried out manually map visual study via ArcGIS and UCDAVIS [2]. On the map, we just applied the GIS data which have been computed as must influencing features by our OCA method. As a result, we could easily identify these sub-areas spatial pattern on map. For example, once we recognized a spatial pattern in the hydrological map which is identical to shape of cluster $C2$, the OCA of the sub-area reveals its organization structure significantly similar to euclidean distance to hydrology. We figure out another instance that the associated sub-area to $C3$ is a flat (high hill-shade noon) near to the Comanche peak and it has not been touched with natural fires.

Moreover, we find that a PCA analysis of the same categorized data was not able to detect the indicator to detect spatial pattern. For example, in Fig.5 we presented the cluster $C1$ PCA result as contribution bi-plot. The blue vectors are presenting the coordinates of the active features that are calculated as the correlation between them and the principal components. As expected, there is a very weak linear correlation between them that means a PCA analysis of the same data was not able to identify the spatial connectivity.

In summary, we have identified any wilderness sub-area occurring consistently aligned with some spatial pattern but

[2]https://casoilresource.lawr.ucdavis.edu/see/



(a) PCA bi-plot

Fig. 5: Application of PCA to Comanche Peak Wilderness detected sub-areas C1. The larger the value of the contribution is, the more the feature contributes to the components.

non-linearly. We note that these spatial patterns are easily indicated by our methods because of the topology preservation property enjoyed by our approach. Moreover, we show that classical multivariate analysis approaches such as PCA, cannot easily detect these relevant indicators because by their nature they end up linearly separating points in the dataset that are in fact topologically close.

D. Diagnosing performance bottleneck in High Performance Computing (HPC) applications.

The next dataset we studied is a dataset that includes various Performance Hardware Counters[3] (HWC) values in the HPC STREAM [4] benchmark. Performance hardware counters values are unique metrics to understand the behavior of the application in a given hardware. Hardware counters are available in almost all modern processors, and count micro-architectural events such as L1, L2, L3: Levels of cache misses, MSP: Conditional branch instructions mispredicted, INS: Total instructions executed, and etc. Moreover, we drive a performance metric "Overlapping Index" (BOI) to indicate proportion of shared resources on-chip. The STREAM benchmark is a state-of-art HPC benchmark designed to measure sustainable memory bandwidth (in MB/s) and a corresponding computation rate for four simple vector kernels (Copy, Scale, Add and Triad). We executed STREAM application on the MareNostrum[5] where $OMP\_threads\_number = 40$ and $Loop\_size = 9M$ to collect the dataset (4264 variable pairs, 9 numerical HWC variables) by specified interval sampling mechanism [6]. We mathematically diagnose patterns in these

[3]https://icl.utk.edu/papi/
[4]http://www.cs.virginia.edu/stream/
[5]https://www.bsc.es/marenostrum/marenostrum
[6]https://tools.bsc.es/extrae

datasets that characterize the application performance losses by applying our OCA approach. Note that OCA extracted these deep insights without requiring the expertise to study the huge amount of information manually and visually.

In [27], they propose the Completed Instructions (INS) combined with Instructions Per Cycle (IPC) as an appropriate active feature subset. This combination focuses the clustering on the "performance view" of the application. Then, we applied DBSCAN algorithms to the extracted performance data by tuned hyper-parameters and using the active features in order to determine the application structure. Fig.6 shows the detected clusters (application phases). As a result, we stratified four distinct clusters in elongated shape. One can then ask the question if each cluster represents a distinct phase of application why they show heterogeneous performance behavior. To answer this question, we applied the OCA to detect the HPC systems bottleneck that can describe the variability among stratified data.

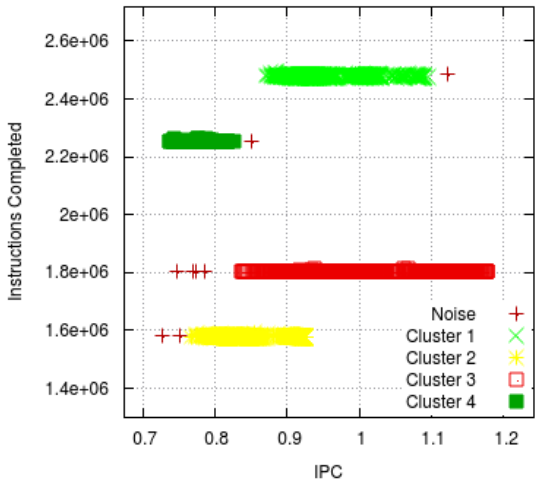
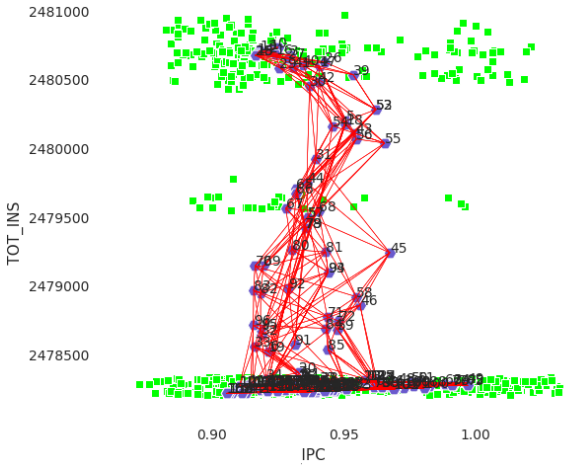


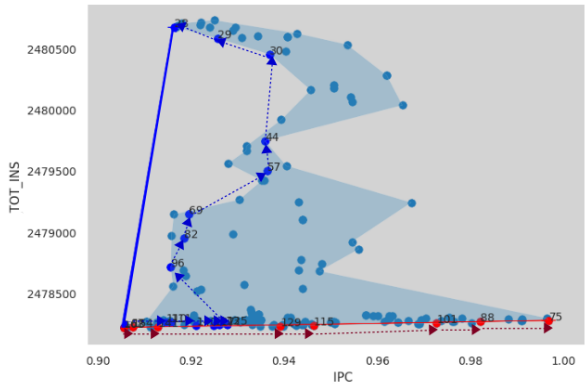
Fig. 6: Performance data extracted from STREAM benchmark execution ($OMP\_threads\_number = 40$, $Loop\_size = 9M$) , visualizations of the clustering result (DBSCAN $\varepsilon = 0.015$, $MinPts = 6$).

In this paper, we report the results of the Triad operation (e.g., Cluster1 in Fig.6), since it is the most complex scenario and is highly relevant to kernels used in HPC applications. In order to randomize the experiments, we conducted 10 OCA on Triad associated cluster and computed the average features influence as well as the standard deviations over all analysis. In all conducted experiments, OCA achieves to 94% topology preservation in average, which shows strong mapping quality.

Fig.7 shows plots of the organization components that we have identified in $cluster1$ of previously aforementioned example. Plot (a) shows the embedded graph that is computed by our OCA method. Although all data points are representatives of the same kernel, they can be categorized into three distinct sub-clusters that do not detected by DBSCAN due to the fact that we select the ε value to identify the main application trends. Note that the bottom sub-cluster includes



(a) SOM weight positions



(b) The blue and red lines are the first and second shape axes respectively.

Fig. 7: Application of OCA to STREAM dataset cluster C1; (a) shows the locations of the data points and the weight vectors.(b) The blue points are representing the SOM neurons, while the red and blue dashed arrows represent first and second Organization component, respectively.

the majority of the data points. As shown in (b), our OCA method identified two sub-structures. We identify that the first organization component indicates the main HPC application performance behavior and the second one presents the reason of the inter-cluster stratification. From Table.II, we diagnose that the magnitude of shared on-chip resources (BOI) can describe approximately 70% of the performance losses, since each thread can only use a fraction of the shared resources at specific moment. Furthermore, we identify the performance effect of L1\L2, L1 and L2 miss ratio have become the main concern when following the OC2 trajectory path, it is likely that approximately 50% of the performance problem is the L1 and L2 capacity. Also, we detect three sub-groups that represent distinctive level of L1 and L2 miss ratio.

In advanced experiment, we executed STREAM application with various combination of $OMP\_NUM\_THREAD \in \{1, 2, 4, 8, 16, 24, 32, 40, 48\}$ and $Loop\_size \in [10k, 89M]$ to collect 134 datasets (1221 to 11,600 variable pairs, 9 numerical HWC variables). We conducted the prior process on each

TABLE II: Application of OCA to STREAM data set cluster C1, the features influencing in the first and second OC.

	L1	L2	L3	L1\L2	L2\L3	MSP	BOI
OC1	0.223	0.220	0.159	0.227	0.222	0.051	**0.693**
OC2	**0.492**	**0.500**	0.086	**0.483**	0.134	0.015	0.189

obtained dataset to diagnose the HPC system bottlenecks.

Fig.8, presents the contour plots of the mean value of IPC and INS (active subspace), versus the log(loop size) and the OMP_threads_number and Fig.9 shows contour plots of illustrative feature's Directional Sequence Similarity with the major organization component. As shown in the plot 8.a, the application roughly shows highest performance by the small number of threads and it has high performance in the area under the bell-shaped component as well. We detect significantly similar pattern in the plot 9.g which identifies the performance of application mainly influenced by the magnitude of shared on-chip resources. We also identify a minor difference between two components in the right tail. It is caused by the imbalanced thread distribution between sockets that increases the IPC mean. For example, in $OMP\_NUM\_THREAD = 16$ and $log(Loop\_size) = 18$ case only two threads are assigned to the second socket; fourteen threads to the first one. From the plots 9.($a \sim e$), we can conclude that the bottleneck of the small, medium and big problem sizes is L1, L2 and L3 misses ratio respectively. The most remarkable aspect of the plot 9.a, see the yellow area in the small loop size, is that the exponential relationship between loop size and OMP_number_threads. In the same way, we recognize the similar pattern in the performance of application (IPC), you can see the lightest orange area in the small loop size in plot 8.a. Although we identify L1 miss rates as a main bottleneck of small problem size, the performance of application has still been acceptable, probably, in consideration of the relatively low latency of L1 cache. It would be worth mentioning that, on the very small loop size, the performance of application is reduced quickly by increasing the number of threads, possibly, due to the fact that parallelism overhead is too remarkable in case of very small vector size, as it can be seen in the right side of the plot 9.f. As it can be seen in plot 8.b, there is roughly a linear relationship between INS and $Loop\_size$ due to the vector size. Although the INS does not illustrate a performance issue, it helps use to segregate main application trends perfectly.

In order to verify the insight that extracted by the OCA, we performed manually the HPC performance analysis with PARAVER [7] toolkit and we identified the same performance bottleneck as well. However, it is a manual approach.

In summary, our Organization Component Analysis diagnoses performance bottleneck of HPC applications automatically rather than the visual approach that is too intuitive and laborious. In case of STREAM benchmark, we identified that higher magnitude of shared resources on-chip will ruin the application performance dramatically. Meanwhile the number of

[7]https://tools.bsc.es/paraver

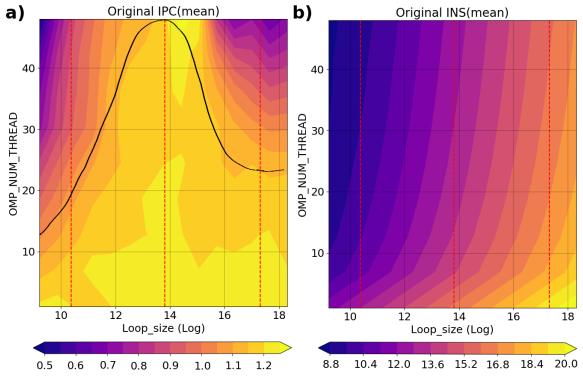


Fig. 8: STREAM benchmark, the plots are shown the contour plots of the mean value of IPC and INS, versus the log(loop size) and the OMP_threads_number.

cache misses in the higher level of cache hierarchic will be the performance bottleneck by increasing the loop size. In general, the OCA can be a canonical approach to automatically detect the HPC application performance bottleneck among complex performance data without expert human visual analysis, which can broadly be applied to any HPC application.

VI. Conclusion and Future Work

We have presented a new topology-preserving approach to study the complex and arbitrary shape of the stratified data called *Organization Component Analysis* (OCA). We propose to make the best use of the self-organization map structure of a high dimensional categorized data, which is defined on the 2D grid of neurons, both to recognize and quantify innate cluster structure and its formation, simultaneously. Whereas cluster analysis identifies regions of higher density in these data, OCA is able to extract finer-grain insights from the shape of a cluster, as it is clearly demonstrated in this article. Here OCA is a general and an efficient method that is assumption free, automated, and it can be applied on the result of any clustering algorithm. Moreover, OCA creates a graph to visualize the shape of these clusters by way of a graph. Furthermore, we propose a novel Directional Sequence Similarity method to compute the similarity between two sequences of changes, in which the rate of change is taken along a unit vector. Finally, we have shown that our novel topology-preserving approach can lead to finer and profounder insights of two real-world datasets. The usefulness of our OCA technique is not closed to these two types of applications but can generally be applied to diverse data types, such as time series, image segmentation, consumer behavior data and others.

As future work, we would like to come up with a hybrid clustering approach to obtain very detailed structure identification by giving the outer level flexibility to operate on an approximate coarse grain euclidean space with DBSCAN and great detail on non-homogeneous local space deformations with SOM.

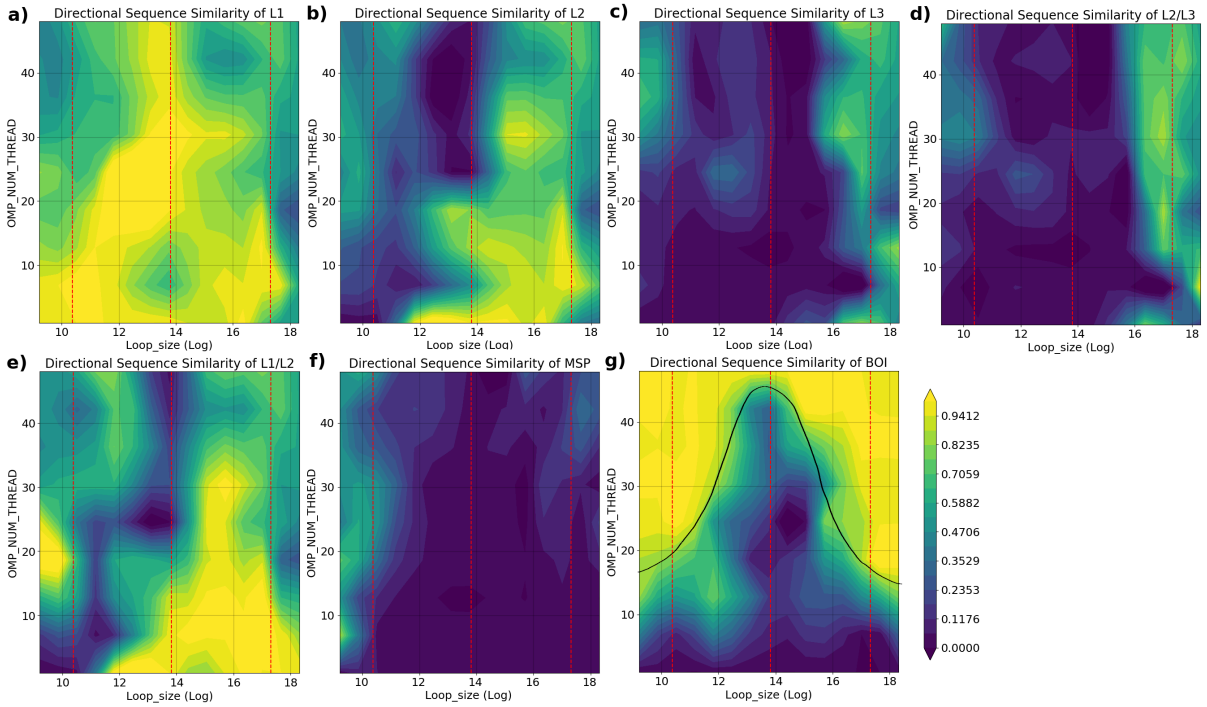


Fig. 9: STREAM benchmark, the plots present the contour plots of illustrative feature's Directional Sequence Similarity with the major organization component, versus the log(loop size) and the OMP_threads_number. The red dashed lines show the threshold of three hierarchical levels of caches (32kB, 1MB and 33MB) receptively.

REFERENCES

[1] Gan, G., Ma, C., & Wu, J. (2020). Data clustering: theory, algorithms, and applications. SIAM, pp. 3-17.

[2] Chu, S. C. (2004). Improved Clustering and Soft Computing Algorithms. Flinders University of South Australia, School of Engineering.

[3] Marcio Gameiro, Yasuaki Hiraoka, & Vidit Nanda (2012). Topological Measurement of Protein Compressibility via Persistence Diagrams, MI Preprint Series 2012-6, Faculty of Mathematics, Kyushu University.

[4] Carlsson, G., Ishkhanov, T., De Silva, V., & Zomorodian, A. (2008). On the local behavior of spaces of natural images. International journal of computer vision, 76(1), 1-12.

[5] Nicolau, M., Levine, A. J., & Carlsson, G. (2011). Topology based data analysis identifies a subgroup of breast cancers with a unique mutational profile and excellent survival. Proceedings of the National Academy of Sciences, 108(17), 7265-7270.

[6] Kriegel, & Zimek, A. (2011). Density-based clustering. Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery, 1(3), 231-240.

[7] Lv, Y., Ma, T., Tang, M., Cao, J., Tian, Y., & Al-Rodhaan, M. (2016). An efficient and scalable density-based clustering algorithm for datasets with complex structures. Neurocomputing, 171, 9-22.

[8] Bryant, A., & Cios, K. (2017). RNN-DBSCAN: A density-based clustering algorithm using reverse nearest neighbor density estimates. IEEE Transactions on Knowledge and Data Engineering, 30(6), 1109-1121.

[9] Lu, Y., Zhang, Y., Richter, F., & Seidl, T. (2020, July). k-Nearest Neighbor based Clustering with Shape Alternation Adaptivity. In 2020 International Joint Conference on Neural Networks (IJCNN) (pp. 1-8).

[10] Alelyani, S., Tang, J., & Liu, H. (2013). Feature selection for clustering: a review. Data clustering: algorithms and applications, 29(110-121), 144.

[11] Rasmussen, C. E. (2000). The infinite Gaussian mixture model. In Advances in neural information processing systems (pp. 554-560).

[12] Watkins, M. W. (2018). Exploratory factor analysis: A guide to best practice. Journal of Black Psychology, 44(3), 219-246.

[13] Jolliffe, I. T., & Cadima, J. (2016). Principal component analysis: a review and recent developments. PHILOS T R SOC A: Mathematical, Physical and Engineering Sciences, 374(2065), 20150202.

[14] Hinneburg, A., Keim, D. A., & Wawryniuk, M. (1999). HD-Eye: Visual mining of high-dimensional data. IEEE (CG&A), 19(5), 22-31

[15] Aggarwal, C. C. (2004). A human-computer interactive method for projected clustering. IEEE (TKDE), 16(4), 448-460.

[16] Cohn, D., Caruana, R., & McCallum, A. (2003). Semi-supervised clustering with user feedback. Constrained Clustering: Advances in Algorithms, Theory, and Applications, 4(1), 17-32.

[17] Lu, Z., & Leen, T. K. (2008). Pairwise constraints as priors in probabilistic clustering. Basu et al.(2008), 59-90.

[18] Lum, P. Y., Singh, G., Lehman, A., Ishkanov, T., Vejdemo-Johansson, M., Alagappan, M., ... & Carlsson, G. (2013). Extracting insights from the shape of complex data using topology. Scientific reports, 3, 1236.

[19] Singh G, Memoli F, Carlsson G (2007) Topological methods for the analysis of high dimensional data sets and 3D object recognition. Eurographics Symposium on Point- Based Graphics, pp 91–100

[20] Yin, H. (2008). The self-organizing maps: background, theories, extensions and applications. In Computational intelligence: A compendium (pp. 715-762). Springer, Berlin, Heidelberg.

[21] Mariño, L. M., & de Carvalho, F. D. A. (2020, July). A new batch SOM algorithm for relational data with weighted medoids. In 2020 International Joint Conference on Neural Networks (IJCNN) (pp. 1-8).

[22] Uriarte, E. A., & Martín, F. D. (2005). Topology preservation in SOM. International journal of applied mathematics, 1(1), 19-22.

[23] Tian, J., Azarian, M.H. and Pecht, M., (2014). Anomaly detection using self-organizing maps-based k-nearest neighbor algorithm. In Proceedings of the European Conference of the Prognostics and Health Management Society (pp. 1-9).

[24] Pena, M., Barbakh, W., & Fyfe, C. (2008). Topology-preserving mappings for data visualization. In Principal Manifolds for Data Visualization and Dimension Reduction (pp. 131-150).

[25] Moulton, R. H., & Zgraja, J. (2019). The Wilderness Area Data Set: Adapting the Covertype data set for unsupervised learning. preprint arXiv:1901.11040.

[26] Mahdavi, K., Labarta, J., & Gimenez, J. (2019). Unsupervised Feature Selection for Noisy Data. In International Conference on Advanced Data Mining and Applications (pp. 79-94).

[27] Gonzalez, J., Gimenez, J., & Labarta, J. (2009). Automatic detection of parallel applications computation phases. In IEEE (ISPDC), (pp. 1-11).