On Tensor Distances for Self Organizing Maps: Clustering Cognitive Tasks

Georgios Drakopoulos[1]([envelope]) [ORCID], Ioanna Giannoukou[2] [ORCID], Phivos Mylonas[1] [ORCID], and Spyros Sioutas[3]

[1] Department of Informatics, Ionian University,
Tsirigoti Sq. 7, 49100 Kerkyra, Greece
`{c16drak,fmylonas}@ionio.gr`
[2] Department of Management Science and Technology, University of Patras,
26504 Patras, Greece
`igian@upatras.gr`
[3] CEID, University of Patras, 26504 Patras, Greece
`sioutas@ceid.upatras.gr`

Abstract. Self organizing maps (SOMs) are neural networks designed to be in an unsupervised way to create connections, learned through a modified Hebbian rule, between a high- (the input vector space) and a low-dimensional space (the cognitive map) based solely on distances in the input vector space. Moreover, the cognitive map is segmentwise continuous and preserves many of the major topological features of the latter. Therefore, neurons, trained using a Hebbian learning rule, can approximate the shape of any arbitrary manifold provided there are enough neurons to accomplish this. Moreover, the cognitive map can be readily used for clustering and visualization. Because of the above properties, SOMs are often used in big data pipelines. This conference paper focuses on a multilinear distance metric for the input vector space which adds flexibility in two ways. First, clustering can be extended to higher order data such as images, graphs, matrices, and time series. Second, the resulting clusters are unions of arbitrary shapes instead of fixed ones such as rectangles in case of ℓ_1 norm or circles in case of ℓ_2 norm. As a concrete example, the proposed distance metric is applied to an anonymized and open under the Creative Commons license cognitive multimodal dataset of fMRI images taken during three distinct cognitive tasks. Keeping the latter as ground truth, a subset of these images is clustered with SOMs of various configurations. The results are evaluated using the corresponding confusion matrices, topological error rates, activation set change rates, and intra-cluster distance variations.

Keywords: SOM · Cognitive maps · Dimensionality reduction · Hebbian learning · Multilinear distance · Tensor algebra · Higher order data · Multi-aspect clustering · Cognitive tasks · fMRI imaging

1   Introduction

Self organizing maps (SOMs) are grids of computational neurons which can efficiently learn through a modified Hebbian process to approximate any higher dimensional manifold, called the *input vector space* and represented by a set of data points or data vectors, with a low dimensional space termed the *cognitive map*, which is one- or two-dimensional. This map can be then employed for clustering and visualization purposes. For this reason in a typical data science pipeline SOMs can be placed either at the beginning in order to discover outliers or at the end in order to cluster or visualize high dimensional results.

The approximation and the associated properties are made feasible by rearranging the namely connections between the input vector space and the cognitive map. Distances in the former are reflected to the latter with some loss of minor information, provided the cognitive map contains sufficient neurons to capture the desired input vector features in the geometry of the cognitive map. This ability as well as the segmentwise continuity of the latter can be attributed to the training process and to the distance metric in the input vector space, which can be theoretically arbitrary to a great extent allowing thus the progressive construction of variable shaped clusters. This is a distinct advantage over clustering algorithms which are based on fixed shaped regions. Nonetheless, most common distance metrics do not take advantage of this.

The primary research objective of this conference paper is the development of a multilinear distance metric for the data points of the input space. This lays the groundwork for more flexible cluster shapes in the cognitive map, as they can be the union of arbitrary shapes instead of fixed ones, clustering higher order data points such as graphs and matrices, and for multi-aspect clustering where each object in the input space can be represented by multiple but distinct data points. In order to evaluate the above, they have been tested with an open dataset, under the Creative Commons license from *openneuro.org*, containing fMRI images taken during three cognitive tasks conducted by elderly people. Since ground truth data is available, the confusion matrices have been derived for SOMs of various configurations. Moreover, the corresponding topological error rates, the activation set change rates, and the intra-cluster distance variations have been computed for each epoch during the cognitive map formulation.

The remaining of this work is structured as follows. In Sect. 2 the relevant scientific literature regarding the applications of blockchains, including process mining and IoT, is briefly summarized. Section 3 overviews the various SOM aspects. Section 4 overviews the clustering evaluation metrics, which can be either SOM-specific or generic. The proposed family of tensor-based distance metrics is proposed in Sect. 5, whereas the activation set variation is introduced in Sect. 6 where also the experimental results are given and commented on. Future research directions are described in Sect. 7. Tensors are represented by capital calligraphic letters, matrices with capital boldface, vectors with small boldface, and scalars with small letters. Vectors are assumed to be columns unless stated otherwise. When a function or a set depends on a number of param-

eters, then the latter are shown in the respective definition after a colon. Finally, Table 1 summarizes the notation of this work.

Table 1. Notation of this conference paper.

Symbol	Meaning
$\triangleq$	Definition or equality by definition
$\{s_1, \ldots, s_n\}$	Set with elements $s_1, \ldots, s_n$
$(t_1, \ldots, t_n)$	Tuple with elements $t_1, \ldots, t_n$
$|S|$	Set or tuple cardinality
$\mathrm{loc}\,(\mathbf{s})$	Grid location for data point $\mathbf{s}$
$\mathrm{invloc}\,(u)$	Set of data points assigned to neuron u
$\mathrm{weight}\,(u)$	Synaptic weights of neuron u
$\mathrm{neighb}\,(u_1, u_2)$	Indicator function of neighboring neurons u_1 and u_2
$\mathrm{vec}\,(\mathcal{T})$	Vector operator for tensor $\mathcal{T}$

2   Previous Work

SOMs have been proposed in [11] for dimensionality reduction, clustering, and visualization and since then their popularity remains unabated. The latter can be mainly attributed to the preservation of the topological properties of the original high dimensional space in the final low dimensional one [8,10]. Among the SOM applications can be found climatology models [7], knowledge discovery in gene expressions [25], and the novel clustering of ECG complexes [14]. A clustering method for strings based on SOMs is presented in [13]. Finding patterns with SOMs in collections of Web documents is explored in [9]. SOMs have been also used as a visualization tool in finance clustering long feature vectors [2]. Additionally, due to SOM popularity, a number of implementations exists. In [24] a Simulink library aiming at FPGAs is proposed, while [23] discusses a parallel SOM library based on the MapReduce paradigm.

Tensors have found multiple applications in fields such as signal processing [1], computer vision [22], and deep learning [29]. Unsupervised methods for face recognition based on the separation of higher order subspaces are described in [28] and extended in [20]. The term-document matrix of information retrieval is extended to a term-keyword-document tensor in [4] in order to exploit the added semantic value of keywords. Face recognition in the context of emotion discovery is discussed in [27]. In [6] a genetic algorithm is proposed for discovering communities based on linguistic and geographical criteria. Multilinear discriminant analysis is a higher order clustering approach based on tensor algebra [16]. A distinction between its primary approaches with an application to gait recognition is given in [19]. Finally, an LMS-based update scheme for tensor entries has been used in adaptive system identification as shown in [5].

3   Cognitive Maps

3.1   Input and Coordinate Spaces

In an SOM each neuron is placed on a one- or two-dimensional grid, with the latter being more common. In the following analysis it will be assumed that a two dimensional $p \times q$ grid is used. Without loss of generality, the following concepts can be applied also to one dimensional girds [12,15]. Once the training process is complete, the SOM is a cognitive map, namely a low dimensional representation of the data point space with the former preserving topological similarities and neighborhoods of the latter. This technique is believed to be performed by the hippocampus of the human brain in order to construct mental representations of the physical or perceived world [3,21,26].

Let n denote the number of available data vectors or data points and d their fixed dimensionality. The latter holds true even when the data points are of variable length such as strings. In this case, d is taken to be the maximum string length.

Definition 1 (Input vector space). *The input space $\mathcal{V}$ is the d-dimensional space spanned by the set of the n data points $S \triangleq \{\mathbf{s_k}\}_{k=1}^{n}$.*

In this work each data vector is assumed to consist of d numerical features extracted from the dataset of Sect. 6:

$$\mathbf{s_k} \triangleq \begin{bmatrix} s_{k,1} & s_{k,2} & \dots & s_{k,d} \end{bmatrix}^T \tag{1}$$

Definition 2 (Coordinate space). *The coordinate space $\mathcal{C}$ is a $p \times q$ two dimensional discrete space composed of the $m = pq$ locations of the neurons.*

Each neuron u_k represents a location vector in $\mathcal{C}$ as follows:

$$u_k \leftrightarrow \begin{bmatrix} x_k & y_k \end{bmatrix}^T \in \mathcal{C}, \quad 0 \le k \le m - 1 \tag{2}$$

Since multiple data vectors can be mapped to the same neuron, $\mathrm{loc}\,(\cdot)$ is not a function as its inverse returns the set of data vectors $S_k \subseteq S$ defined as:

$$\mathrm{invloc}\,(u_k) \triangleq \left\{ \mathbf{s_{k_1}}, \dots, \mathbf{s_{k_{f_k}}} \right\} \triangleq S_k \tag{3}$$

Definition 3 (Neighboring neurons). *Any neuron adjacent in $\mathcal{C}$ to a given neuron u_i is termed to be a neighbor of u_i.*

The designation of neighborhood depends on the grid number of dimensions, grid shape, and neighborhood strategy. Thus, in a square grid the neighboring neurons can form a cross, a square, or a hexagon around u_i. In this work the first choice is used. The following indicator function codifies the above:

$$\mathrm{neighb}\,(u_i, u_j) \triangleq \begin{cases} 1, u_i \text{ and } u_j \text{ are neighboring} \\ 0, \text{otherwise} \end{cases} \tag{4}$$

Therefore, for each neuron u_i the neighborhood set $\Gamma(u_i)$ is defined as follows:

$$\Gamma(u_i) \triangleq \{u_j \in \mathcal{C} \mid \text{neighb}(u_i, u_j) = 1\} \tag{5}$$

Each neuron u_i in the grid has its own vector of synaptic weights $\mathbf{w_i}$:

$$\text{weight}(u_i) \triangleq \mathbf{w_i} \in \mathcal{V} \tag{6}$$

3.2 Distance Metrics and Weight Function

SOMs require three functions in order to construct the final cognitive map, namely the distance metrics of $\mathcal{V}$ and $\mathcal{C}$ and the weight function. As its name suggests, the distance metrics in $d(\cdot, \cdot)$ used between data points and synaptic weights. Common choices, depending on the nature of data points, are: ℓ_2, ℓ_1, or ℓ_0 norms for numerical vectors, Hamming distance for binary vectors, Tanimoto coefficient for sets, and Levenshtein distance for strings.

Common distance metrics $g(\cdot, \cdot)$ in $\mathcal{C}$ include the triangular, square, circular, and Gaussian distances. Notice that it is not necessary to use the same distance metric in both $\mathcal{V}$ and $\mathcal{C}$. In fact, because these spaces are of different nature, a given metric may not be even applicable in one of these spaces.

Finally, the weight function $h(\cdot, \cdot)$, which is usually normalized such that its maximum value equals one, depends on the location of two neurons in the grid. In fact, in most cases the weight depends on some form of the grid distance, not to be confused with the function $g(\cdot, \cdot)$ defined earlier, between two neurons and not on their absolute locations. In certain scenaria $h(\cdot, \cdot)$ and $g(\cdot, \cdot)$ are related, by they do not need to be so in the general case. In this work these functions are independent of each other. Common options include:

– **Constant:** Each neuron in the proximity set receives the same weight.
– **Gaussian:** The Gaussian kernel has a smooth decay rate which drops fast enough to ensure the clusters remain relatively compact. It is defined as:

$$h(u_i, u_j; \sigma_0) \triangleq \frac{1}{\sigma_0\sqrt{2\pi}} \exp\left(-\frac{\|u_i - u_j\|_1^2}{2\sigma_0^2}\right) \tag{7}$$

– **Triangular:** This function has a linear decay rate since:

$$h(u_i, u_j; \gamma_0) \triangleq 1 - \frac{\|u_i - u_j\|_1}{\gamma_0} \tag{8}$$

– **Circular:** The weight function forms a semicircle around neuron u_j:

$$h(u_i, u_j; \rho_0) \triangleq \frac{\sqrt{\rho_0^2 - \|u_i - u_j\|_2^2}}{\rho_0} \tag{9}$$

The weight function also plays a central role in the formation for each neuron of its proximity set, which is defined as follows:

Definition 4 (Proximity set). *The proximity set $\Delta(u_i; \xi_0)$ for a neuron u_i given a threshold ξ_0 is the set of neurons for which $h(\cdot, \cdot)$ remains above ξ_0:*

$$\Delta(u_i; \xi_0) \triangleq \{u_j \in \mathcal{C} \mid h(u_i, u_j) \geq \xi_0\} \tag{10}$$

The proximity set $\Delta(u_i; \xi_0)$ of a given neuron u_i differs from its the neighborhood $\Gamma(u_i)$ introduced in Definition 3. The former corresponds to the effective area of a cluster around u_i whereas the latter to the central area of that cluster. Therefore, $\Delta(u_i; \xi_0)$ always includes $\Gamma(u_i)$ but extends beyond that. The limiting case where the two sets coincide is very rarely used in practice, if at all.

3.3 Learning Rate

The learning rate η of an SOM controls as in other neural network architectures the convergence rate of the training process.

- **Constant:** Under this policy the learning rate is a fixed constant η_0.
- **Exponential decay:** In order to ensure that early epochs receive a larger weight in comparison to the later ones, the following scheme can be used:

$$\eta[r; \alpha_0, \gamma_0] = \alpha_0 \exp(-\gamma_0 r) \tag{11}$$

- **Cosine decay:** This option is based on the following function:

$$\eta[r; \alpha_0, r_0] = \alpha_0 \cos\left(\frac{\pi r}{2r_0}\right), \quad 0 \leq r \leq r_0 - 1 \tag{12}$$

- **Inverse polynomial:** This rate attempts to achieve a smoother decay as:

$$\eta[r; \alpha_0, \beta_0, \gamma_0, \delta_0] = \frac{\delta_0}{(\alpha_0 r + \beta_0)^{\gamma_0}} \tag{13}$$

- **Inverse logarithm:** Finally, this choice yields the formula:

$$\eta[r; \alpha_0, \beta_0, \delta_0] \triangleq \frac{\delta_0}{a_0 \log(r) + \beta_0} \tag{14}$$

The learning rate can be allowed to vary within the same epoch or remain fixed and change only with each epoch. In the former case, the learning rate can very well be a product of two factors, one taking into account the epoch and the other depending on the projection within the epoch. As a general note, when the learning parameter is allowed to change within the same epoch, then it should be combined with a policy which does not preserve the order which the input vectors are projected to the SOM. Here η remains constant during each epoch.

3.4 Grid Dimensions

In the general case there is no straightforward criterion to determine the grid size. This work follows the empirical rule proposed in [10] which takes into consideration only the total number of data points:

$$p = q = \lceil 5\sqrt{n} \rceil \tag{15}$$

3.5   Training Process

The synaptic weights of the winning neuron u^* are updated in each projection of the epoch r based on a modified Hebbian update rule as follows:

$$\Delta \text{weight}(u^*)[r] = \eta[r] \cdot d(\mathbf{s_i}, \text{weight}(u^*)) \qquad (16)$$

The winning neuron u^* for the specific input vector $\mathbf{s_i}$ is the neuron whose weight vector is closest in $\mathcal{V}$ to $\mathbf{s_i}$:

$$u^* \triangleq \text{argmin}\{d(\mathbf{s_i}, \text{weight}(u))\} \qquad (17)$$

Notice that the distance metric in (17) is computed in $\mathcal{V}$ and not in $\mathcal{C}$ since data points belong in the former. Additionally, u^* is added to the set $D[r]$ of activated neurons for that given epoch is formed, assuming that $D[r]$ is empty at the beginning of the epoch:

$$D[r] \leftarrow D[r] \cup u^* \qquad (18)$$

Additionally, besides the winning neuron u^* the synaptic weights of each neighboring neuron $u_i \in \Delta(u^*; \xi_0)$ are also updated as follows:

$$\Delta \text{weight}(u_j)[r] = \eta[r] \cdot h(u^*, u_j) \cdot d(\mathbf{s_k}, \text{weight}(u_j)) \qquad (19)$$

Both updates (16) and (19) drive the weights of the corresponding neurons closer to the data point $\mathbf{s_k}$, gradually creating a cluster. Neurons in $\Delta(u^*; \xi_0)$ receive a reduced update because of the added factor of the weight function.

4   Performance Metrics

Perhaps the most common and intuitive SOM error metric is the topological error. As local coherence preservation is of primary importance in the final cognitive map, it is only natural to measure the average probability that a data point is assigned to the periphery of a cluster.

Definition 5 (Topological error rate). *The topological error $T[r]$ for epoch r is the ratio of the number of data points which are mapped to $\Delta(u^*) \setminus \Gamma(u^*)$.*

$$T[r] \triangleq \frac{1}{n} \sum_{\mathbf{s_k} \in S} |\{\text{loc}(\mathbf{s_k}) \notin \{u \cup \Gamma(u) \mid u \in D[r]\}\}| \qquad (20)$$

Another SOM performance metric is the activation set evolution, essentially the number of clusters formed at each epoch and the centroids thereof. The latter is not static but rather dynamic and it is controlled by the data points.

Definition 6 (Activation set change rate). *The change of activation set is the percentage of the activation set cardinalty during the current epoch.*

$$K[r] \triangleq \frac{|D[r]| - |D[r-1]|}{|D[r]|}, \quad r \geq 1 \qquad (21)$$

Clustering performance is usually difficult to evaluate. One common metric for assessing clustering is the average intra-clustering distance, namely the arithmetic mean of the average distsance of each data point from the corresponding centroid. In the case of the SOMs this metric takes the following form.

Definition 7 (Intra-cluster distance). *The intra-cluster distance is the average of the average distance of each data point from its respective centroid.*

$$Q\,[r] \;\triangleq\; \frac{1}{|D\,[r]|} \sum_{u_i \in D[r]} \frac{1}{|C_i|} \sum_{u_j \in C_i} g\,(u_j, u_i) \qquad (22)$$

5   Tensor Distance

Here the multilinear distance metric, based on work previously done [17], is introduced. First, an auxiliary definition is in order.

Definition 8 (Tensor vector operator). *The tensor vector operator maps a P-th order tensor $\mathcal{T} \in \mathbb{R}^{I_1 \times I_2 \times \dots \times I_P}$ to a vector $\mathbf{v} \in \mathbb{R}^N$ by stacking from top to bottom the columns of dimension I_k from left to right. Also $N = I_1 \cdot \dots \cdot I_P$.*

The proposed distance is defined as follows:

Definition 9 (Multilinear distance). *The multilinear distance metric is defined as the squared root weighted quadratic form of the vectorized difference of the data points where the latter are represented as tensors.*

$$J\,(\mathcal{X}, \mathcal{Y}; \mathbf{G}) \;\triangleq\; \left(\mathrm{vec}\,(\mathcal{X} - \mathcal{Y})^T \; \mathbf{G} \; \underbrace{\mathrm{vec}\,(\mathcal{X} - \mathcal{Y})}_{\mathbf{s}} \right)^{\frac{1}{2}} = \sqrt{\mathbf{s}^T \mathbf{G} \mathbf{s}} \qquad (23)$$

Metric J includes bilinear and quadratic functions as special cases. This allows more flexibility as different tensor components can receive different weights.

The weight matrix $\mathbf{G}$ is defined in [17] elementwise as the Gaussian kernel whose values depend only on the locations of the elements of $\mathbf{s}$ involved. Although this resulted in a matrix with non-negative entries, a better way is to define $\mathbf{G}$ as positive definite, meaning that the value of the quadratic function under the square root in (23) is guaranteed to be positive.

Moreover, it makes sense to select a $\mathbf{G}$ tailored to the underlying nature of the data points. Since in the experiments the data points are stacked images, the following two properties of the discrete cosine transform (DCT) will be exploited: First, since images consist of real values, the transformation spectrum will also contain only real values. Second this spectrum will be very sparse, with non-zero values concentrated along the low spatial frequencies. Therefore, subtracting and weighing through a linear filter only these values instead of every pixel saves tremendous computational time.

The original data points are three stacked fMRI images for each subject:

$$\mathbf{X}_k \;\triangleq\; \left[\;\mathbf{X}_{k,1} \;\middle|\; \mathbf{X}_{k,2} \;\middle|\; \mathbf{X}_{k,3}\; \right] \qquad (24)$$

Therefore, $\mathbf{G}$ is built as a matrix product involving the discrete DCT matrix $\mathbf{D}$, the linear filtering matrix $\mathbf{F}$, the identity matrix $\mathbf{I}$, and the 3×1 auxiliary vector $\mathbf{1}_{3,1}$ whose elements equal 1 as follows:

$$\mathbf{G} \triangleq (\mathbf{1}_{3,1} \otimes \mathbf{D})^T \cdot (\mathbf{I} \otimes \mathbf{F}) \cdot (\mathbf{1}_{3,1} \otimes \mathbf{D}) \tag{25}$$

In Eq. (25) $\otimes$ denotes the Kronecker tensor product which is used to create the following structured matrices:

$$\mathbf{1}_{3,1} \otimes \mathbf{D} = \begin{bmatrix} \mathbf{D} \\ \mathbf{D} \\ \mathbf{D} \end{bmatrix} \qquad \mathbf{I} \otimes \mathbf{F} = \begin{bmatrix} \mathbf{F} & & \\ & \mathbf{F} & \\ & & \mathbf{F} \end{bmatrix} \tag{26}$$

The linear filter matrix $\mathbf{F}$ has been selected to be the first order lowpass or smoothing Butterworth filter.

The advantages of using a higher order distance metric such as that of Eq. (23) are the following:

– It enables multi-aspect clustering, meaning that the various objects of the input vector space can be clustered in more than one ways. Thus, different aspects of the same object may contribute to clustering. Alternatively, this can be interpreted as that each object is represented by multiple attribute vectors, which is the basis for many advanced clustering algorithms.
– There is more flexibility in defining weights for the various attribute vectors or aspects, which can help understand the underlying clustering dynamics.
– Multilinear metrics are suitable for evaluating the distance between higher order data such as graphs, tensors, time series, and images in a natural way.

6   Results

The anonymized dataset for the experiments was first published in [18] as part of a project for studying the ageing human brain and uploaded to *openneuro.org* under the Creative Commons license[1]. Following the warnings of the respository, the files pertaining to one subject were ignored. According to the dataset authors, the latter consists of 34 participants (33 in our case), 14 male and 20 (19 in our case) female, all right handed with no health problems. Each participant was presented for three seconds an image of neutral or negative affective polarity selected from the International Affective Picture System and then was issued an audio instruction to *suppress*, *maintain*, or *enhance* their emotions. These are the three cognitive tasks, respectively denoted as **T1**, **T2**, and **T3**.

Again according to [18], the fMRI images used here were collected with the following parameters. There were 30 sagittal images with each such image having a 3 mm thickness with repetition time (TR) and echo time (TE) being respectively equal to 2000 msec and 30 msec, a slice gap of 33%, a field of view (FoV)

---

of 192×192 mm^2, and a spatial resolution of 3 mm. From these images were kept three, one each at 10%, 50%, and 90% of the block length. Finally, after normalization, registration, and correcting for magnetic field inhomogenetity and head motion, spatial preprocessing took place with a Gaussian kernel of 6 mm FWHM. Table 2 has the total and marginal frequency distribution for the three cognitive tasks and the image polarity. It follows that not only the three classes but also their subclasses are well balanced.

Table 2. Distribution of tasks.

	T1	T2	T3	Subtotal
Neutral	6	5	5	16
Negative	6	6	5	17
Subtotal	12	11	10	33

The configuration of an SOM is represented by a tuple c_k:

$$c_k \triangleq (p, q, d\left(\cdot, \cdot\right), g\left(\cdot, \cdot\right), h\left(\cdot, \cdot\right), \eta) \tag{27}$$

The possible options for each field of c_k are given in Table 3. Out of the 27 possible combinations of metrics for $\mathcal{V}$, $\mathcal{C}$, and learning parameter rate, in total 9 were chosen with emphasis given to the learning parameter rate.

Table 3. SOM configuration options.

$d\left(\cdot, \cdot\right)$	$g\left(\cdot, \cdot\right)$	$h\left(\cdot, \cdot\right)$	η
T: tensor	G: Gaussian	G: Gaussian	S: cosine
L2: ℓ_2 norm	C: circular	C: circular	E: exponential
L1: ℓ_1 norm	R: triangle	R: triangle	P: inverse polynomial

The values used in each function as needed by each SOM configuration are shown in Table 4. Recall that p_0 and q_0 are the SOM tableau dimensions, as

Table 4. Values for each configuration parameter.

Param.	Value	Param.	Value
r_0	25	S	α_0: 1.0
G	σ_0: 4	E	(α_0, γ_0): $(1.0, 0.5)$
C	ρ_0: 5	P	$(\alpha_0, \beta_0, \gamma_0, \delta_0)$: $(1.0, 1.0, 2, 1.0)$
R	γ_0: 0.2	p_0, q_0	30, 30

determined by (15), and r_0 is the fixed number of epochs used. Notice that p_0 and q_0 were selected to be identical resulting in square tableau, but this is by no means mandatory.

The actual configurations used in the experiments along with their numbering are shown in Table 5. As it can be seen, the exact same number of combinations was used with the ℓ_1 norm, the ℓ_2 norm, and the proposed multilinear metric. Configurations achieving the best confusion matrix in terms of maximizing its trace, one for each distance metric in $\mathcal{V}$, are marked in boldface.

Table 5. SOM configurations.

#	Configuration	#	Configuration	#	Configuration
1	$(p_0, q_0, L1, C, C, E)$	10	$(p_0, q_0, L2, C, C, E)$	19	(p_0, q_0, T, C, C, E)
2	$(p_0, q_0, L1, C, C, P)$	11	$(p_0, q_0, L2, C, C, P)$	20	(p_0, q_0, T, C, C, P)
3	$(p_0, q_0, L1, C, C, S)$	12	$(p_0, q_0, L2, C, C, S)$	21	(p_0, q_0, T, C, C, S)
4	$(p_0, q_0, L1, R, R, E)$	13	$(p_0, q_0, L2, R, R, E)$	22	(p_0, q_0, T, R, R, E)
5	$(p_0, q_0, L1, R, R, P)$	14	$(p_0, q_0, L2, R, R, P)$	23	(p_0, q_0, T, R, R, P)
6	$(p_0, q_0, L1, R, R, S)$	15	$(p_0, q_0, L2, R, R, S)$	24	(p_0, q_0, T, R, R, S)
7	$(p_0, q_0, L1, G, G, E)$	16	$(p_0, q_0, L2, G, G, E)$	25	(p_0, q_0, T, G, G, E)
8	$(\mathbf{p_0, q_0, L1, G, G, P})$	17	$(p_0, q_0, L2, G, G, P)$	26	(p_0, q_0, T, G, G, P)
9	$(p_0, q_0, L1, G, G, S)$	18	$(\mathbf{p_0, q_0, L2, G, G, S})$	27	$(\mathbf{p_0, q_0, T, G, G, S})$

Table 6. Best (left, configuration 8) and average confusion matrix for ℓ_1.

Task(%)	T1	T2	T3	T1	T2	T3
T1	78.66	8.33	13.00	77.11	11.89	11.00
T2	10.66	75.00	14.33	11.00	74.50	14.50
T3	4.00	7.00	79.00	5.20	8.20	76.60

Tables 6, 7, and 8 are the confusion matrices when the SOM distance metric is the ℓ_1 norm, the ℓ_2 norm, and that of Eq. (23) respectively for the SOM configuration which achieved the least misclassification rate, in other words it maximized the trace of the corresponding confusion matrix. In all three cases the best case entries are distinct from but not very far from their average case countrparts. This points to the best case scenario being actually achievable and not being merely a fortuitous but rare result.

Moreover, from the entries of Tables 6, 7, and 8 follows that the proposed multilinear distance metric consistently results in lower misclassification rates. This can be attributed to the additional flexibility offered by it, in terms of partitioning the data input space to arbitrary regions and of discovering latent

Table 7. Best (left, configuration 18) and average confusion matrix for ℓ_2.

Task(%)	T1	T2	T3	T1	T2	T3
T1	79.66	8.00	12.33	78.00	7.50	14.50
T2	9.00	81.00	10.00	7.30	79.40	13.30
T3	7.66	8.33	83.00	5.00	13.50	81.50

Table 8. Best (left, config. 27) and average confusion matrix for multilinear metric.

Task(%)	T1	T2	T3	T1	T2	T3
T1	91.01	6.99	2.00	88.50	11.16	1.33
T2	2.33	85.33	12.33	2.00	84.00	14.00
T3	5.00	8.66	87.33	3.00	11.50	86.50

Table 9. Performance scores vs configuration.

#	Scores	#	Scores	#	Scores
1	$7.58, 0.66, 15.11, 1.15$	10	$6.22, 0.74, 12.54, 1.13$	19	$6.02, 0.82, 10.34, 7.22$
2	$7.61, 0.64, 15.43, 1.13$	11	$6.71, 0.73, 13.53, 1.11$	20	$6.17, 0.81, 10.84, 7.21$
3	$7.65, 0.61, 15.04, 1.19$	12	$6.43, 0.69, 13.42, 1.14$	21	$6.39, 0.79, 10.72, 7.28$
4	$8.56, 0.68, 14.48, 1.17$	13	$6.25, 0.76, 12.52, 1.12$	22	$5.82, 0.83, 9.11, 7.25$
5	$8.71, 0.67, 14.94, 1.15$	14	$6.36, 0.73, 12.55, 1.12$	23	$5.97, 0.81, 9.34, 7.23$
6	$8.62, 0.64, 14.65, 1.13$	15	$6.35, 0.74, 12.67, 1, 11$	24	$6.08, 0.81, 9.56, 7.22$
7	$6.93, 0.71, 13.43, 1.13$	16	$6.23, 0.77, 12.13, 1.13$	25	$5.35, 0.84, 8.56, 7.24$
8	$7.03, 0.67, 13.46, 1.11$	17	$6.55, 0.74, 12.62, 1.11$	26	$5.55, 0.82, 8.99, 7.25$
9	$7.11, 0.69, 13.77, 1.12$	18	$6.42, 0.75, 12.37, 1.13$	27	$5.47, 0.81, 8.84, 7.22$

features in the image triplets. Also, since there were divisions by 33, the total number of subjects, fractions like 0.33 and 0.66 were very frequent.

In Table 6 there is a trend for **T2** to be confused with **T3** and vice-versa. Specifically, the entries denoting a confusion between **T2** and **T3** and vice versa were significantly higher than those recording confusion between **T1** and **T3** and vice versa as well as **T1** and **T3** and vice versa.

From Table 7 is clear that ℓ_2 norm outperforms ℓ_1. This is an indication that spheres may be a better way to partition the data input space, perhaps due to their isotropic curvature. The strong confusion between **T2** and **T3** continues.

Table 8 contains the best values both in the optimal and in the mean case. The confusion trend between **T2** and **T3** is also present.

Table 9 shows the average topological error rate, the average absolute activation set change rate, the average inter-cluster distance variation, and the mean wallcklock time (in seconds) in this order for each configuration. Each SOM configuration was executed eleven times, with the first time being considered a

test and thus not contributing at all to the time and performance measurements presented here. The numbering is exactly the same as that of Table 5.

Since the SOM training is computationally intensive, it is a reasonable indicator of the true computational time. Although the proposed metric is more expensive, its cost is not prohibitive. There is minimum variation among the time measurements for each distance metric with the ℓ_1 and ℓ_2 norms achieving very similar scores.

Figure 1 depicts the mean topological error rate scores of Table 9 in order to obtain a intuitive evaluation of the indicative performance of each configuration. The remaining three scores were omitted to avoid cluttering.
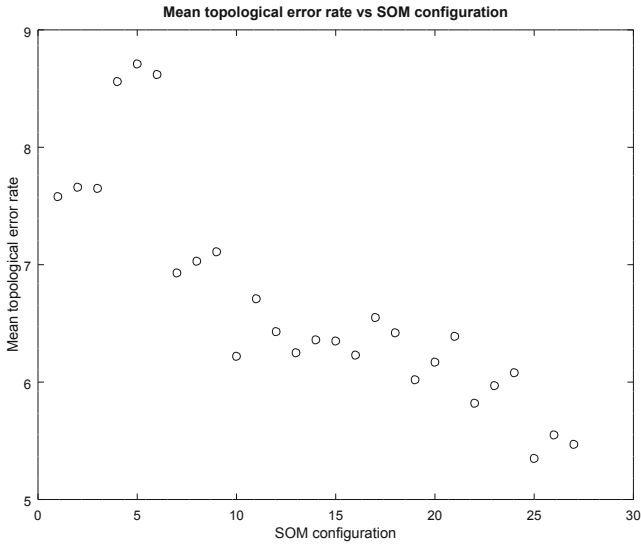
Fig. 1. Mean topological error vs SOM configuration.

Certain conclusions can be drawn regarding the various SOM configurations. Partitioning the cognitive map to Gaussian regions leads to better performance scores, followed by spherical and rectangular regions. The same can be said about the cosine decay rate. Also, the tensor distance metric yield systematically better scores at all three performance metrics.

The conclusion which can be drawn about the three cognitive tasks is that there is a systematic trend for confusion between **T2** and **T3** as in all cases the confusion between these two was consistently higher regardless of metric distance. This may indicate a tendency for negative emotions to persist in older adults, a view also shared by [18].

7   Conclusions and Future Work

This conference paper describes the application of SOMs to the clustering of fMRI image triplets in order to distinguish between three cognitive tasks. The SOMs configurations had different combinations of distance metrics for input and neuron space, learning rate decay rate, and neighbourhood shape. The three distance metrics for the data input space were the ℓ_1 norm, the ℓ_2 norm, and a multilinear metric. The distinction between the cognitive tasks is in compliance with established results. Moreover, the proposed metric achieves lower topological error, intra-cluster distance, and activation set change rate, whereas it resulted in a confusion matrix with the highest trace.

This work can be extended in a number of ways. First and foremost, the selection of other features which are appropriate for high dimensional cultural datasets can be explored. Moreover, evaluating both the SOM performance and the activation set variation with larger benchmark datasets can shed more light in their behavior. Concerning the SOM itself, the link between the data point selection strategies and the total number of epochs should be explored. Also, an approriate estimation for the grid dimensions, based perhaps on information theoretic measures such as the AIC, BIC, or MDL, is a possible research direction. The development of training termination criteria connected with the overall quality of the cognitive map should be sought. The role of bias in ensuring that no inactive neurons exist, especially for large grids with randomly initalized synaptic weights is another possible topic. How SOMs can be applied to online clustering, especially in big data pipelines, is worth researching.

References

1. Cichocki, A., et al.: Tensor decompositions for signal processing applications: from two-way to multiway component analysis. IEEE Signal Process. Mag. **32**(2), 145–163 (2015)
2. Deboeck, G., Kohonen, T.: Visual Explorations in Finance with Self-Organizing Maps. Springer, Heidelberg (2013). https://doi.org/10.1007/978-1-4471-3913-3
3. Downs, R.M., Stea, D.: Cognitive maps and spatial behavior: process and products. Adaline (1973)
4. Drakopoulos, G., Kanavos, A., Karydis, I., Sioutas, S., Vrahatis, A.G.: Tensor-based semantically-aware topic clustering of biomedical documents. Computation **5**(3), 34 (2017). https://doi.org/10.3390/computation5030034
5. Drakopoulos, G., Mylonas, P., Sioutas, S.: A case of adaptive nonlinear system identification with third order tensors in TensorFlow. In: INISTA, pp. 1–6 (2019). https://doi.org/10.1109/INISTA.2019.8778406

6. Drakopoulos, G., et al.: A genetic algorithm for spatiosocial tensor clustering. EVOS **1**(11) (2019). https://doi.org/10.1007/s12530-019-09274-9
7. Hewitson, B., Crane, R.G.: Self-organizing maps: applications to synoptic climatology. Clim. Res. **22**(1), 13–26 (2002)
8. Kangas, J.A., Kohonen, T.K., Laaksonen, J.T.: Variants of self-organizing maps. IEEE Trans. Neural Netw. **1**(1), 93–99 (1990)
9. Kaski, S., Honkela, T., Lagus, K., Kohonen, T.: WEBSOM-self-organizing maps of document collections. Neurocomputing **21**(1–3), 101–117 (1998)
10. Kiviluoto, K.: Topology preservation in self-organizing maps. In: ICNN, vol. 1, pp. 294–299. IEEE (1996)
11. Kohonen, T.: The self-organizing map. Proc. IEEE **78**(9), 1464–1480 (1990)
12. Kohonen, T.: Exploration of very large databases by self-organizing maps. In: ICNN. vol. 1, pp. PL1-PL6. IEEE (1997)
13. Kohonen, T., Somervuo, P.: Self-organizing maps of symbol strings. Neurocomputing **21**(1–3), 19–30 (1998)
14. Lagerholm, M., Peterson, C., Braccini, G., Edenbrandt, L., Sornmo, L.: Clustering ECG complexes using Hermite functions and self-organizing maps. IEEE Trans. Biomed. Eng. **47**(7), 838–848 (2000)
15. Lampinen, J., Oja, E.: Clustering properties of hierarchical self-organizing maps. J. Math. Imaging Vis. **2**(2–3), 261–272 (1992)
16. Li, Q., Schonfeld, D.: Multilinear discriminant analysis for higher-order tensor data classification. TPAMI **36**(12), 2524–2537 (2014)
17. Liu, Y., Liu, Y., Zhong, S., Chan, K.C.: Tensor distance based multilinear globality preserving embedding: a unified tensor based dimensionality reduction framework for image and video classification. Expert Syst. Appl. **39**(12), 10500–10511 (2012)
18. Lloyd, W., Morriss, J., Macdonald, B., Joanknecht, K., Sigurd, J., van Reekum, C.: Longitudinal change in executive function is associated with impaired top-down frontolimbic regulation during reappraisal in older adults. bioRxiv (2019)
19. Lu, H., Plataniotis, K., Venetsanopoulos, A.: Uncorrelated multilinear discriminant analysis with regularization for gait recognition. In: Biometrics Symposium, pp. 1–6. IEEE (2007)
20. Lu, H., Plataniotis, K.N., Venetsanopoulos, A.N.: Uncorrelated multilinear discriminant analysis with regularization and aggregation for tensor object recognition. IEEE Trans. Neural Netw. **20**(1), 103–123 (2008)
21. O'Keefe, J., Nadel, L.: The Hippocampus as a Cognitive Map. Clarendon Press, Oxford (1978)
22. Shashua, A., Hazan, T.: Non-negative tensor factorization with applications to statistics and computer vision. In: ICML, pp. 792–799. ACM (2005)
23. Sul, S.J., Tovchigrechko, A.: Parallelizing BLAST and SOM algorithms with MapReduce-MPI library. In: International Symposium on Parallel and Distributed Processing, pp. 481–489. IEEE (2011)
24. Tisan, A., Cirstea, M.: SOM neural network design- a new Simulink library based approach targeting FPGA implementation. Math. Comput. Simul. **91**, 134–149 (2013)
25. Törönen, P., Kolehmainen, M., Wong, G., Castrén, E.: Analysis of gene expression data using self-organizing maps. FEBS Lett. **451**(2), 142–146 (1999)
26. Tversky, B.: Cognitive maps, cognitive collages, and spatial mental models. In: Frank, A.U., Campari, I. (eds.) COSIT 1993. LNCS, vol. 716, pp. 14–24. Springer, Heidelberg (1993). https://doi.org/10.1007/3-540-57207-4_2

27. Vasilescu, M.A.O., Terzopoulos, D.: Multilinear analysis of image ensembles: TensorFaces. In: Heyden, A., Sparr, G., Nielsen, M., Johansen, P. (eds.) ECCV 2002. LNCS, vol. 2350, pp. 447–460. Springer, Heidelberg (2002). https://doi.org/10.1007/3-540-47969-4_30

28. Yan, S., Xu, D., Yang, Q., Zhang, L., Tang, X., Zhang, H.J.: Multilinear discriminant analysis for face recognition. IEEE Trans. Image Process. **16**(1), 212–220 (2006)

29. Yu, D., Deng, L., Seide, F.: The deep tensor neural network with applications to large vocabulary speech recognition. IEEE Trans. Audio Speech Lang. Process. **21**(2), 388–396 (2012)