



SOM IN HILBERT SPACE

*J. Snor, J. Kukul, Q. Van Tran**

Abstract: The self organization can be performed in an Euclidean space as usually defined or in any metric space which is generalization of previous one. Both approaches have advantages and disadvantages. A novel method of batch SOM learning is designed to yield from the properties of the Hilbert space. This method is able to operate with finite or infinite dimensional patterns from vector space using only their scalar product. The paper is focused on the formulation of objective function and algorithm for its local minimization in a discrete space of partitions. General methodology is demonstrated on pattern sets from a space of functions.

Key words: *Hilbert space, cluster analysis, optimal partition, heuristic optimization, SOM*

Received: November 13, 2016

DOI: 10.14311/NNW.2019.29.002

Revised and accepted: January 17, 2019

1. Introduction

The Hilbert space [17] is frequently used for data analysis, recognition and classification. Especially, kernel methods operating in the Hilbert space are modern and efficient tools in many applications. We have been inspired by kernel methods of data processing [3] related to Kernel Principal Component Analysis, Kernel Clustering techniques and Kernel SOM. The paper is oriented to the self-organization in the Hilbert space via generalization of Kohonen SOM batch learning [7]. The method is motivated by infinite-dimensional and high-dimensional cases of pattern description including the self-organization of functions on given domain using adequate weight function in scalar product. The second useful case is a direct application to large signals and images which are results of regular sampling of functions on rectangular domains. The self-organization is also useful for investigation of words and texts using p -spectrum [5] for all $p \in \mathbb{N}$. The novel learning algorithm is based on the data partition over SOM graph nodes and their local rearrangement. All the procedures of pattern processing are based on the properties of the Hilbert space. Necessary theoretical background is formulated in three theorems about the relationships between patterns and their cluster. Based on principles mentioned above, novel algorithm of SOM learning has been developed in the Matlab environment. Our SOM learning approach is verified on two examples using whitened Kernel PCA [12] as referential method.

The rest of the paper is structured as follows. The second section is devoted to

*Jakub Snor; Jaromir Kukul – Corresponding author; Quang Van Tran; Czech Technical University in Prague, Faculty of Nuclear Sciences and Physical Engineering, Department of Software Engineering, Trojanova 13, Prague 2, Czech Republic, E-mail: jaromir.kukul@jfi.cvut.cz

the Hilbert space introduction. Clusters in the Hilbert space and their properties are discussed in the third section which is followed by the description of clustering in the Hilbert space in the fourth section. After these basic facts the fifth section introduces the self-organization in this space as the main result of this paper. Last two sections are focused on the case study of the SOM in the space of functions and concluding remarks.

2. Hilbert space preliminaries

Data processing is frequently performed in a vector space. In particular cases one can analyze data in metric space or in another special structure. Having *field* \mathcal{F} [1] with addition, multiplication, zero element, unit element, and axioms, we can define *vector space* [2] \mathcal{V} with addition \oplus , scalar multiplication \odot , and adequate axioms. *Metric space* [10] \mathcal{M} with metric d enables distance measurement $d(x, y)$ for any $x, y, \in \mathcal{M}$. Structure \mathcal{M} is more general than structure \mathcal{V} , but \mathcal{M} is not too useful for data processing except particular cases. The *Hilbert space* [17] \mathcal{H} as a vector space \mathcal{V} with inner product $(x | y)$ is the preferred structure in this paper. We will use $\|x\| = \sqrt{(x | x)}$ as norm and $d(x, y) = \|x - y\|$ as distance for $x, y \in \mathcal{H}$ respecting well known axioms [17]. Any \mathcal{H} is also \mathcal{M} with $d(x, y) = \sqrt{(x - y | x - y)}$ and Euclidean space \mathcal{E} as $\mathcal{V} = \mathbb{C}^n$ is a special case of \mathcal{H} with

$$(x | y) = \sum_{k=1}^n x_k^* y_k. \quad (1)$$

Therefore, a data analysis in \mathcal{H} will extend the abilities of the Euclidean space \mathcal{E} but it is more strict than in the metric space \mathcal{M} . The main advantage of pattern analysis in \mathcal{H} is in simplification of the batch learning algorithms. The Hilbert space is frequently used in many kernel methods [3, 12, 16] in combination with kernel trick in the hidden layer, but we design the novel algorithms for the patterns from the Hilbert space, i.e., presented in the input layer.

3. Cluster representation in the Hilbert space

In accordance with the description of clustering in the Euclidean, metric, and kernel space, we have to introduce a cluster and its properties, first. Having $m \in \mathbb{N}$ patterns in given *cluster* $\mathcal{C} = \{x_k \in \mathcal{H} : k = 1, \dots, m\}$ we define *cluster quality* as the minimal value of

$$f^+(y) = \sum_{k=1}^m \|x_k - y\|^2 \quad (2)$$

for $y \in \mathcal{H}$. Any value $y_{\text{opt}}^+ \in \text{argmin} f^+(y)$ is called *centroid* [16].

But individual patterns should have various positive weights related to \mathcal{C} . Let $w_k \in \mathbb{R}^+$ be weights for $k = 1, \dots, m$ satisfying

$$\sum_{k=1}^m w_k = 1. \quad (3)$$

This approach can be also useful for the SOM description in the main section. The definitions of centroid and cluster quality in \mathcal{H} can be directly generalized as

follows. *Weighted centroid* is any $t \in \mathcal{H}$ satisfying $t \in \operatorname{argmin} f(y)$ where *cluster quality* is defined as

$$f(y) = \sum_{k=1}^m w_k \|x_k - y\|^2 \quad (4)$$

for non-empty \mathcal{C} and in the case of empty \mathcal{C} we set $f(y) = 0$. As will be shown, the unique weighted centroid exists.

Theorem 1. When \mathcal{C} is not empty, function f reaches its unique minimum for $t = \sum_{k=1}^m w_k x_k$.

Proof. We express the objective function as

$$\begin{aligned} f(y) &= \sum_{k=1}^m w_k (x_k - y \mid x_k - y) = \sum_{k=1}^m w_k (\|x_k\|^2 - (x_k \mid y) - (y \mid x_k) + \|y\|^2) = \\ &= \|y\|^2 + \sum_{k=1}^m w_k \|x_k\|^2 - \sum_{k=1}^m ((x_k \mid y) + (y \mid x_k)) w_k. \end{aligned}$$

Applying function f to t , we obtain

$$f(t) = \|t\|^2 + \sum_{k=1}^m w_k \|x_k\|^2 - \sum_{k=1}^m ((x_k \mid t) + (t \mid x_k)) w_k.$$

Now we express

$$\begin{aligned} f(y) - f(t) &= \|y\|^2 - \|t\|^2 - \sum_{k=1}^m ((x_k \mid y) + (y \mid x_k) - (x_k \mid t) - (t \mid x_k)) w_k = \\ &= \|y\|^2 - \left(\sum_{k=1}^m w_k x_k \mid \sum_{j=1}^m w_j x_j \right) - \sum_{k=1}^m w_k \left(\left(x_k \mid \sum_{j=1}^m w_j x_j \right) + \right. \\ &\quad \left. + \left(\sum_{j=1}^m w_j x_j \mid x_k \right) \right) = \\ &= \|y\|^2 - \sum_{k=1}^m \sum_{j=1}^m w_k w_j (x_k \mid x_j) - \sum_{k=1}^m w_k ((x_k \mid y) + (y \mid x_k)) + \\ &\quad + \sum_{k=1}^m \sum_{j=1}^m w_k w_j ((x_k \mid x_j) + (x_j \mid x_k)) = \\ &= (y \mid y) - \sum_{k=1}^m ((x_k \mid y) + (y \mid x_k)) w_k + \sum_{k=1}^m \sum_{j=1}^m w_k w_j (x_j \mid x_k) = \\ &= (y \mid y) - \sum_{j=1}^m w_j (x_j \mid y) - \sum_{k=1}^m w_k (y \mid x_k) + \sum_{k=1}^m \sum_{j=1}^m w_k w_j (x_j \mid x_k) = \\ &= \left(y - \sum_{j=1}^m w_j x_j \mid y - \sum_{k=1}^m w_k x_k \right) = \|y - t\|^2 \geq 0. \end{aligned}$$

Therefore, $f(y)$ reaches its minimum just for unique t . □

Theorem 1 is useful for the explicit approach to weight centroid construction but direct calculation of centroid t in \mathcal{H} is impractical, time expensive or unrealisable. Fortunately, both cluster analysis and batch SOM can be based only on the mutual pattern distances and cluster penalization.

Theorem 2. Cluster quality of non-empty \mathcal{C} is equal to

$$f(t) = \sum_{j=2}^m \sum_{i=1}^{j-1} w_i w_j \|x_i - x_j\|^2. \quad (5)$$

Proof. The optimum value is

$$\begin{aligned} f(t) &= \sum_{k=1}^m w_k \|x_k - t\|^2 = \|t\|^2 + \sum_{k=1}^m w_k \|x_k\|^2 - \sum_{k=1}^m w_k ((x_k | t) + (t | x_k)) = \\ &= \left(\sum_{k=1}^m w_k x_k \mid \sum_{j=1}^m w_j x_j \right) + \sum_{k=1}^m w_k \|x_k\|^2 - \sum_{k=1}^m w_k \left(\left(x_k \mid \sum_{j=1}^m w_j x_j \right) + \right. \\ &\quad \left. + \left(\sum_{j=1}^m w_j x_j \mid x_k \right) \right) = \\ &= \sum_{k=1}^m \sum_{j=1}^m w_k w_j (x_k | x_j) + \sum_{k=1}^m w_k \|x_k\|^2 - \sum_{k=1}^m \sum_{j=1}^m w_k w_j ((x_k | x_j) + (x_j | x_k)) = \\ &= \sum_{k=1}^m w_k \|x_k\|^2 - \sum_{k=1}^m \sum_{j=1}^m w_k w_j (x_j | x_k). \end{aligned}$$

The right hand side of (5) can be expressed as

$$\begin{aligned} \sum_{j=2}^m \sum_{i=1}^{j-1} w_i w_j \|x_i - x_j\|^2 &= \frac{1}{2} \sum_{k=1}^m \sum_{j=1}^m w_k w_j (x_k - x_j | x_k - x_j) = \\ &= \frac{1}{2} \left(\sum_{k=1}^m \sum_{j=1}^m w_k w_j \|x_k\|^2 - \sum_{k=1}^m \sum_{j=1}^m w_k w_j ((x_k | x_j) + \right. \\ &\quad \left. + (x_j | x_k)) + \sum_{k=1}^m \sum_{j=1}^m w_k w_j \|x_j\|^2 \right) = \\ &= \frac{1}{2} \left(\sum_{k=1}^m w_k \|x_k\|^2 - \sum_{k=1}^m \sum_{j=1}^m w_k w_j (x_k | x_j) - \right. \\ &\quad \left. - \sum_{k=1}^m \sum_{j=1}^m w_k w_j (x_j | x_k) + \sum_{j=1}^m w_k \|x_j\|^2 \right) = \\ &= \sum_{k=1}^m w_k \|x_k\|^2 - \sum_{k=1}^m \sum_{j=1}^m w_k w_j (x_j | x_k) \end{aligned}$$

which is identical. □

The problem of cluster penalization in \mathcal{H} is then reduced to a pair-wise evaluation of

$$\|x_i - x_j\|^2 = (x_i | x_i) - (x_i | x_j) - (x_j | x_i) + (x_j | x_j). \quad (6)$$

Therefore, when we are able to calculate scalar product in \mathcal{H} , we are also able to evaluate cluster quality without necessity to identify the weighted centroid t . The evaluation of $d(x, t)$ for $x \in \mathcal{H}$ is also useful and possible.

Theorem 3. The distance between $x \in \mathcal{H}$ and centroid t is

$$d(x, t) = \sqrt{\sum_{k=1}^m w_k d^2(x_k, x) - \sum_{j=2}^m \sum_{k=1}^{j-1} w_k w_j d^2(x_k, x_j)}. \quad (7)$$

Proof. According to Theorem 1 the left hand side squared can be expressed as

$$\begin{aligned} d^2(x, t) &= d^2\left(x, \sum_{k=1}^m w_k x_k\right) = \left\|x - \sum_{k=1}^m w_k x_k\right\|^2 = \\ &= \|x\|^2 + \left\|\sum_{k=1}^m w_k x_k\right\|^2 - \left(x \mid \sum_{k=1}^m w_k x_k\right) - \left(\sum_{k=1}^m w_k x_k \mid x\right) = \\ &= \|x\|^2 + \left(\sum_{k=1}^m w_k x_k \mid \sum_{j=1}^m w_j x_j\right) - \sum_{k=1}^m w_k (x \mid x_k) - \sum_{k=1}^m w_k (x_k \mid x) = \\ &= \|x\|^2 + \sum_{k=1}^m \sum_{j=1}^m w_k w_j (x_k \mid x_j) - \sum_{k=1}^m w_k ((x \mid x_k) + (x_k \mid x)). \end{aligned}$$

The right hand side squared can be expressed as

$$\sum_{k=1}^m w_k d^2(x_k, x) - \sum_{j=2}^m \sum_{k=1}^{j-1} w_k w_j d^2(x_k, x_j)$$

and according to Theorem 2 one obtains

$$\begin{aligned} \sum_{k=1}^m w_k d^2(x_k, x) &- \sum_{j=2}^m \sum_{k=1}^{j-1} w_k w_j d^2(x_k, x_j) = \sum_{k=1}^m w_k d^2(x_k, x) - \\ &- \sum_{j=2}^m \sum_{k=1}^{j-1} w_k w_j \|x_k - x_j\|^2 = \sum_{k=1}^m w_k \|x_k - x\|^2 - \\ &- \left(\sum_{k=1}^m w_k \|x_k\|^2 - \sum_{k=1}^m \sum_{j=1}^m w_k w_j (x_k \mid x_j) \right) = \|x\|^2 - \end{aligned}$$

$$\begin{aligned}
 & - \sum_{k=1}^m w_k ((x | x_k) + (x_k | x)) + \sum_{k=1}^m w_k \|x_k\|^2 + \\
 & + \sum_{k=1}^m \sum_{j=1}^m w_k w_j (x_k | x_j) - \sum_{k=1}^m w_k \|x_k\|^2 = \|x\|^2 + \\
 & + \sum_{k=1}^m \sum_{j=1}^m w_k w_j (x_k | x_j) - \sum_{k=1}^m w_k ((x | x_k) + (x_k | x))
 \end{aligned}$$

which is identical. \square

Previous three theorems form a theoretical background for both batch clustering and batch SOM learning in the Hilbert space.

4. Clustering in the Hilbert space

Let $M, H \in \mathbb{N}$ be number of patterns from \mathcal{H} and given number of clusters. Pattern set is then $\mathcal{S} = \{x_k \in \mathcal{H} : k = 1, \dots, M\}$. Traditional *batch clustering* [7] is driven by *partition vector* [8] $\mathbf{p} = (p_1, \dots, p_M) \in \{1, \dots, H\}^M$. It is a kind of lookup table of length M where p_k is the class membership of k^{th} pattern. Cluster $\mathcal{C}_j \subset \mathcal{S}$ is defined as $\mathcal{C}_j = \{x_k \in \mathcal{S} : p_k = j\}$. The clusters $\mathcal{C}_1, \dots, \mathcal{C}_H$ represent *partition* of \mathcal{S} as trivial to verify. The corresponding weights inside the cluster \mathcal{C}_j are defined here as $w_k = 1/\text{card}(\mathcal{C}_j)$ or undefined for empty cluster. *Partition quality* is defined [16] as

$$Q(\mathbf{p}) = \sum_{j=1}^H \sum_{k \in \mathcal{C}_j} \|x_k - t_j\|^2 \quad (8)$$

where t_j is centroid of \mathcal{C}_j . After small rearrangement we directly obtained

$$Q(\mathbf{p}) = \sum_{j=1}^H \text{card}(\mathcal{C}_j) \sum_{k \in \mathcal{C}_j} w_{j,k} \|x_k - t_j\|^2 = \sum_{j=1}^H q_j \quad (9)$$

where q_j is quality of cluster \mathcal{C}_j calculated according (5) as

$$q_j = \text{card}(\mathcal{C}_j) \cdot \sum_{k=2}^{\text{card}(\mathcal{C}_j)} \sum_{l=1}^{k-1} w_{j,k} w_{j,l} \|x_k - x_l\|^2. \quad (10)$$

The general aim of batch cluster analysis is to obtain *optimal clustering (partition)* driven by partition vector $\mathbf{p}_{\text{opt}} \in \text{argmin} Q(\mathbf{p})$. The batch clustering is therefore a kind of nonlinear integer optimization task. According to [4], optimal clustering is a kind of NP-hard task in \mathcal{E} which is a special case of finite \mathcal{H} . Therefore, optimum clustering in \mathcal{H} is also NP-hard task.

In this case, any optimization heuristics can be employed:

- K-means [16] as procedure beginning with random partition can be modified to perform partition revisions according to (7) until local minimum is found,
- Simulated Annealing (SA) [6] as traditional optimization heuristics based on simulation of physical annealing,

- Fast Simulated Annealing (FSA) [14] as an improvement of SA based on non-physical concept of importance sampling,
- Integer Cuckoo Search (ICS) [9] as population based heuristics with integer Lévy flights [15],
- Steepest Descent (SD) [13] as referential heuristics for local minima finding,
- Random Descent (RD) [11] as another referential local heuristics.

5. SOM in the Hilbert space

$$\mathcal{R} \subset \binom{\mathcal{N}}{2} \quad (11)$$

be a set of edges (relations). Then a pair $\mathcal{G} = \langle \mathcal{N}, \mathcal{R} \rangle$ is an undirected connected graph which typically represents the topology of SOM [7]. The number of output nodes $H = \text{card}(\mathcal{N})$. Let $d^* : \mathcal{N} \times \mathcal{N} \rightarrow \mathbb{N}_0$ be a vertex distance in the graph of SOM. Then $\mathcal{M} = \langle \mathcal{N}, d^* \rangle$ is a metric space over the SOM. The maximum value of the vertex distance over $\mathcal{N} \times \mathcal{N}$ is called *diameter* D of the SOM. The SOM in Hilbert space is a function $\text{SOM} : \mathcal{S} \rightarrow \mathcal{N}$, while the *SOM learning* is an algorithm of the SOM function design according to the set of patterns \mathcal{S} [8].

Batch learning of SOM in \mathcal{H} is also driven by vector $\mathbf{p} = (p_1, \dots, p_M) \in \{1, \dots, H\}^M$ which places patterns from $\mathcal{S} \subset \mathcal{H}$ into nodes from \mathcal{N} of given SOM graph. Novel penalization strategy is based on weighted centroids around individual SOM nodes and corresponding centroid quality. When the pattern is placed into the investigated node, its weight is maximum possible but when it is placed in the node neighborhood, the weight is a function of d^* according to

$$w_{i,k} = \frac{\chi(d^*(i, p_{x_k}))}{\sum_{j=1}^M \chi(d^*(i, p_{x_j}))} \quad (12)$$

where characteristics $\chi : \{0, \dots, D\} \rightarrow \mathbb{R}^+$ is a non-increasing function satisfying $\chi(0) = 1, \chi(D) \in (0, 1)$. In the case of batch SOM learning [7], only direct neighbors are involved in centroid calculations with full weight. Therefore, $\chi(d^*) = 1$ for $d^* \leq 1$ and $\chi(d^*) = 0$ otherwise.

The other characteristics are also useful as:

- *rectangular* with characteristics $\chi(d^*) = I(d^* \leq R)$,
- *Gaussian* with characteristics $\chi(d^*) = \exp\left(-\frac{1}{2} \left(\frac{d^*}{R}\right)^2\right)$,
- *Exponential* with characteristics $\chi(d^*) = \exp\left(-\frac{d^*}{R}\right)$,
- *Cauchian* with characteristics $\chi(d^*) = \left(1 + \left(\frac{d^*}{R}\right)^2\right)^{-1}$

where $R > 0$ is learning radius. Gaussian characteristics is frequently used in the traditional Kohonen SOM learning and should be preferred also in this method. As seen from (12) the weights $w_{i,k}$ are well defined and positive satisfying

$$\sum_{k=1}^M w_{i,k} = 1, \text{ for all } i = 1, \dots, H \quad (13)$$

which represents node views to given data set \mathcal{S} . SOM quality is then defined as a sum of node view penalizations according to

$$S(\mathbf{p}) = \sum_{i=1}^H s_i \quad (14)$$

where s_i is “cluster” quality with weights $w_{i,k}$ for given node i according to (10). The main difference between cluster analysis and SOM in \mathcal{H} is in the interaction among patterns from various nodes. SOM partition is formally the same as clustering partition but the influence of any pattern overcome its node limitations. Therefore, SOM learning is just an integer minimization of $S(\mathbf{p})$ for given pattern set \mathcal{S} .

This task is similar to optimum clustering. General integer minimization heuristics (SA, FSA, ICS, SD, RD) can be employed again. Alternative way is to modify K-means heuristic without necessity of node centroid calculations. The novel heuristics of SOM learning in \mathcal{H} begins with random partition \mathbf{p} . The main loop performs SOM partition revisions till penalization $S(\mathbf{p})$ decreases as follows: For every pattern $x_k \in \mathcal{S}$ and every node i we calculate $d(x_k, t_i)$ according to (7) where t_i is hidden centroid of \mathcal{C}_i around node i . New pattern position in SOM is then given by formula $p_k \in \underset{i=1, \dots, H}{\operatorname{argmin}} d(x_k, t_i)$.

6. Case study: SOM in the space of functions

The abilities of novel algorithm will be demonstrated in infinite dimensional Hilbert space of functions. Let $a, b \in \mathbb{R}, a < b$ be boundaries. Let $f, g : [a, b] \rightarrow \mathbb{R}$ be functions from \mathcal{H} with scalar product

$$(f|g) = \int_a^b f(x)g(x)dx. \quad (15)$$

Two pattern sets of functions were studied. In the first case we studied power functions on interval $[0, 1]$ for various parameter $\alpha \geq 0$. Four types of functions are included as

$$f_1(x) = x^\alpha, \quad (16)$$

$$f_2(x) = (1 - x)^\alpha, \quad (17)$$

$$f_3(x) = 1 - x^\alpha, \quad (18)$$

$$f_4(x) = 1 - (1 - x)^\alpha. \quad (19)$$

These functions represent increasing, decreasing, convex, and concave continuous functions on give interval. The list of functions is included in Tab. I.

In the second case, we studied four types of trigonometric functions as

$$f_1(x) = \sin \alpha x, \tag{20}$$

$$f_2(x) = \cos \alpha x, \tag{21}$$

$$f_3(x) = \sin^\alpha x, \tag{22}$$

$$f_4(x) = \cos^\alpha x. \tag{23}$$

These functions were studied on interval $[-\pi, \pi]$ for various parameter $\alpha \in \mathbb{N}_0$. The list of functions is included in Tab. I.

label	power function	trigonometric function
A	0	0
B	1	1
C	$x^{\frac{1}{4}}$	$\sin x$
D	$x^{\frac{1}{2}}$	$\sin 2x$
E	$x^{\frac{3}{4}}$	$\sin 3x$
F	x	$\sin 4x$
G	$x^{\frac{3}{2}}$	$\sin 5x$
H	x^2	$\sin 6x$
I	$(1-x)^{\frac{1}{4}}$	$\cos x$
J	$(1-x)^{\frac{1}{2}}$	$\cos 2x$
K	$(1-x)^{\frac{3}{4}}$	$\cos 3x$
L	$1-x$	$\cos 4x$
M	$(1-x)^{\frac{3}{2}}$	$\cos 5x$
N	$(1-x)^2$	$\cos 6x$
O	$1-x^{\frac{1}{4}}$	$\sin^2 x$
P	$1-x^{\frac{1}{2}}$	$\sin^3 x$
Q	$1-x^{\frac{3}{4}}$	$\sin^4 x$
R	$1-x^{\frac{3}{2}}$	$\sin^5 x$
S	$1-x^2$	$\sin^6 x$
T	$1-(1-x)^{\frac{1}{4}}$	$\cos^2 x$
U	$1-(1-x)^{\frac{1}{2}}$	$\cos^3 x$
V	$1-(1-x)^{\frac{3}{4}}$	$\cos^4 x$
W	$1-(1-x)^{\frac{3}{2}}$	$\cos^5 x$
X	$1-(1-x)^2$	$\cos^6 x$

Tab. I Two pattern sets.

The choice of SOM topology is driven by data processing traditional and therefore only regular planar SOM graphs will be used to demonstrate the mapping from the Hilbert space into discrete 2D space with integer coordinates. Two SOM topologies were used for the self organization. In the first case, we applied rectangular topology 3×3 consisting of 9 nodes. In the second case, hexagonal topology of 7 nodes was used. Gaussian characteristic was used for different radii R . Whitened

Kernel PCA [12] with kernel $K(x, y) = (x|y)$ was used as referential method and its results were mapped into 2D and 3D spaces. SOM batch learning was organized in 1000 independent runs of random shooting heuristics with embedded K-means local search according to 7 and the best solutions of minimum penalization was collected.

Pattern set of power functions was self organized first. Results of 2D and 3D Kernel PCA are depicted on Fig. 1. As seen, 2D KPCA provides good insight into pattern set and 3D case is not necessary for the self-organization. Therefore, the mapping into 2D SOM has to be trivial. The influence of learning radius in the case of rectangular SOM topology is demonstrated on Fig. 2 and in the case of hexagonal SOM topology is demonstrated on Fig. 3. As seen, the large value of learning radius R forms separated clusters inside SOM meanwhile lower value of R caused data spreading and therefore detailed pattern description. The best subjective result for the set of power functions was obtained for seven nodes of hexagonal topology with Gaussian characteristics and radius $R = 1$. In this case, there is a very good correspondence between Kernel PCA's as referential method and SOM partition.

Pattern set of trigonometric functions was organized as the second case. Results of 2D and 3D Kernel PCA are depicted on Fig. 4. Due to similarity of several trigonometric functions, 2D Kernel PCA is less informative in comparison with 3D Kernel PCA which distinguish among the trigonometric functions. Therefore, the

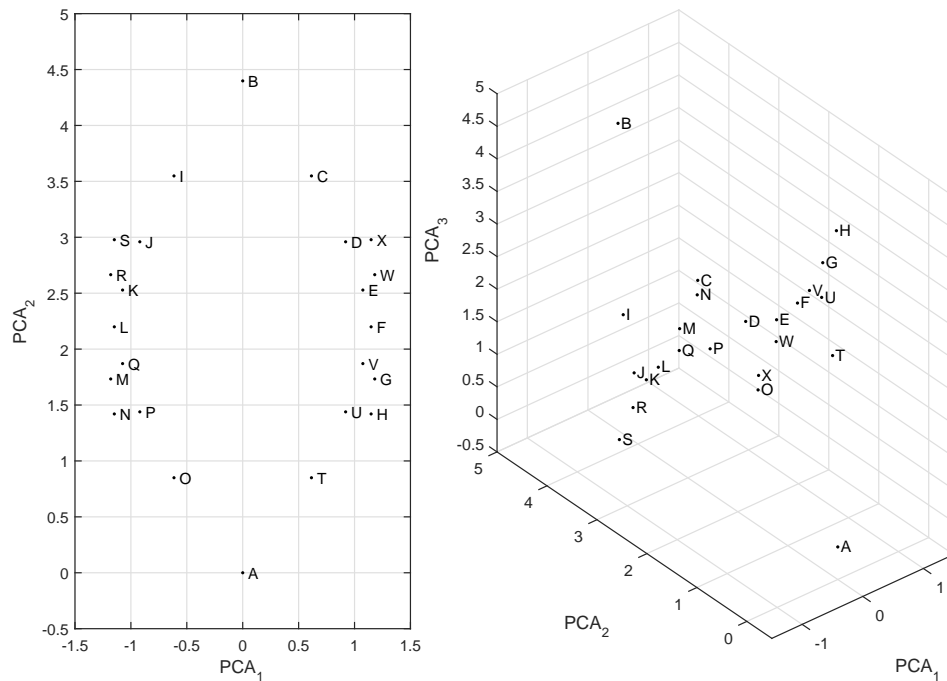


Fig. 1 Kernel PCA for set of power functions: $D = 2$ (left), $D = 3$ (right).

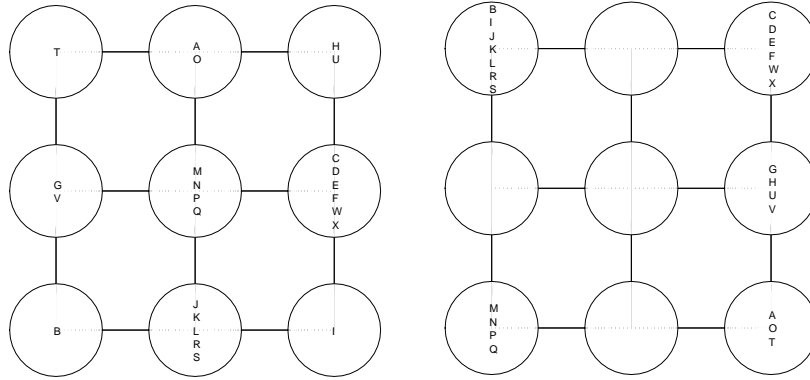


Fig. 2 Rectangular SOM for $H = 9$ and power functions: $R = 0.3$ (left), $R = 0.5$ (right).

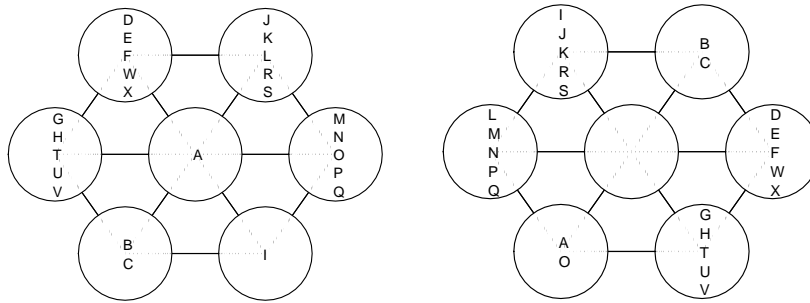


Fig. 3 Hexagonal SOM for $H = 7$ and power functions: $R = 0.2$ (left), $R = 1$ (right).

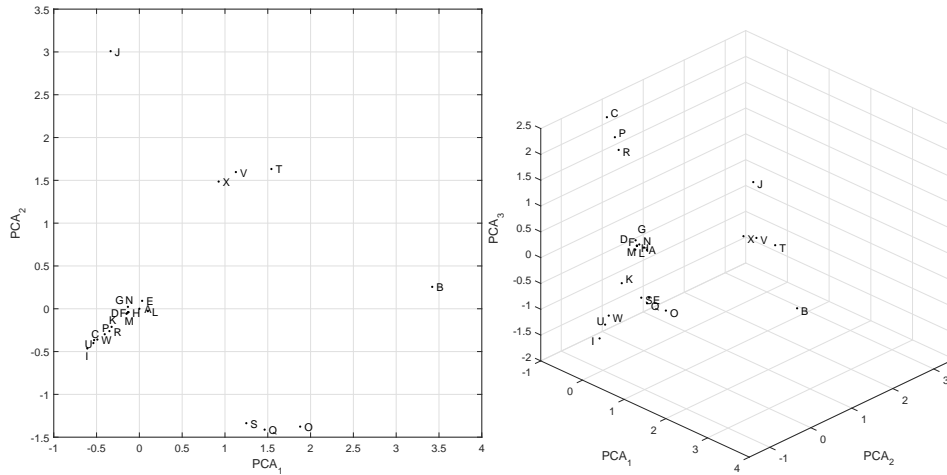


Fig. 4 Kernel PCA for set of trigonometric functions: $D = 2$ (left), $D = 3$ (right).

mapping of these functions into the 2D planar SOM will have similar properties as 2D Kernel PCA. The influence of learning radius in the case of rectangular SOM topology is demonstrated on Fig. 5 and in the case of hexagonal SOM topology is demonstrated on Fig. 6. As in the case of pattern set of power functions, the large value of R forms separated clusters inside SOM meanwhile lower value of R caused data spreading and therefore detailed pattern description. The best subjective result for the set of trigonometric functions was obtained for seven nodes of hexagonal topology with Gaussian characteristics and radius $R = 0.5$ for which the correspondence between Kernel PCA's as referential method and SOM partition was the best.

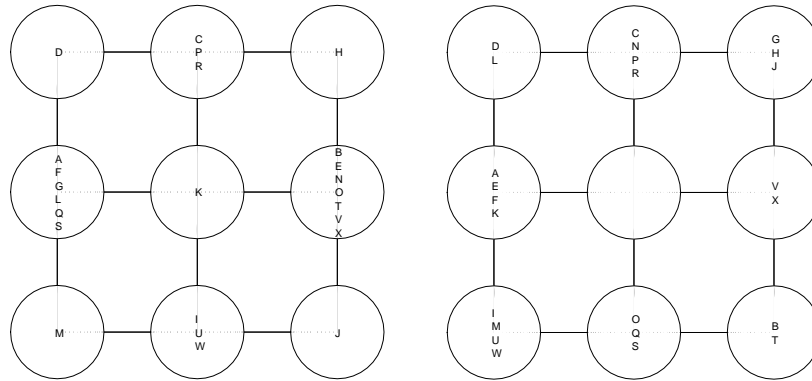


Fig. 5 Rectangular SOM for $H = 9$ and trigonometric functions: $R = 0.2$ (left), $R = 1$ (right).

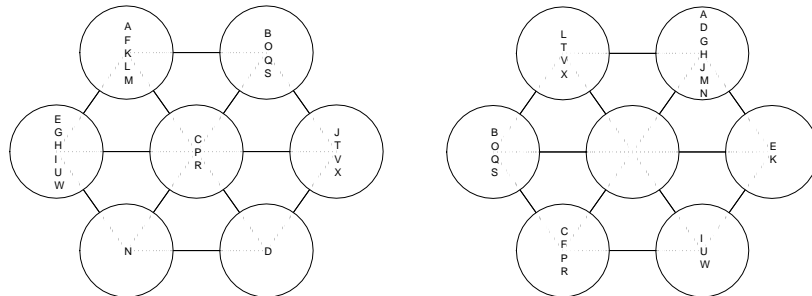


Fig. 6 Hexagonal SOM for $H = 7$ and trigonometric functions: $R = 0.2$ (left), $R = 0.5$ (right).

7. Conclusions

Using basic properties of general Hilbert space, both weighted clustering and batch learning of SOM is possible without coordinate-wise processing. Three basic theorems were proved and then used for SOM learning. General algorithm was able

to self-organize the functions from the Hilbert space as demonstrated on two sets of functions. The effects of SOM topology and repulsing diameter R were studied experimentally using Gaussian characteristics and random shooting optimization heuristics with local hybridization. Too high repulsion radius R caused pattern occurrence only in the outer nodes of SOM which is inefficient. Therefore, the lower values of R are recommended due to better pattern spreading over SOM nodes which means better data description. The general method of self-organization can be applied to functions, long series, large 2D and 3D images, and words as patterns from the Hilbert space whenever they can be efficiently mapped into 2D space. We can also perform the cluster analysis or use special non-planar SOM topology like 2^D or 3^D cube. The applicability of clustering and SOM in the Hilbert space outperform the abilities of the pattern analysis in the Euclidean space.

Acknowledgement

The paper has been created under the support of the grant SGS17/196/OHK4/3T/14. The authors also acknowledged Miloslav Havlicek from FNSPE CTU in Prague for the support related to the Hilbert space theory.

References

- [1] ARTIN M. *Algebra*. Englewood Cliffs, N.J.: Prentice Hall, 1991.
- [2] BROWN W.A. *Matrices and vector spaces*. New York: M. Dekker, 1991.
- [3] FILIPPONE M., CAMASTRA F., MASULLI F. ROVETTA S. A Survey of Kernel and Spectral Methods for Clustering. *Pattern Recognition*. 2008, 41(1), pp. 176–190, Retrieved from <http://eprints.gla.ac.uk/56336/1/56336.pdf>.
- [4] GAREY M.R., JOHNSON D.S. *Computers and intractability: a guide to the theory of NP-completeness*. New York: W.H. Freeman and Company, 1997.
- [5] GUSFIELD D. *Algorithms on String, Trees, and Sequences*. Cambridge: Cambridge University Press, 1997.
- [6] KIRKPATRICK S., GELLAT C.D., VECCHI M.P., Optimization by Simulated Annealing. *Science*. 1983, 220(4598), pp. 671–680, doi: [10.1126/science.220.4598.671](https://doi.org/10.1126/science.220.4598.671).
- [7] KOHONEN T. *Self-Organizing Maps*. New York: Springer Verlag, 1995.
- [8] KUKAL J. SOM in Metric Space. *Neural Network World*. 2004, 14(6), pp. 469–488.
- [9] KUKAL J., MOJZES M., TRAN Q.V., BOSTIK J. Integer Cuckoo Search. In: *Proceedings of 18th International Conference on Soft Computing MENDEL*, 2012, Brno: VUT Press, pp. 298–303.
- [10] SEARCOID M. *Metric spaces*. London: Springer, 2006.
- [11] NESTEROV Y. Efficiency of Coordinate Descent Methods on Huge-Scale Optimization Problems. *SIAM Journal on Optimization*. 2012, 22(2), pp. 341–362, doi: [10.1137/100802001](https://doi.org/10.1137/100802001).
- [12] SCHOLKOPF B., SMOLA A., MULLER K.R. Nonlinear Component Analysis as a Kernel Eigenvalue Problem. *Neural Computation*. 1998, 10(5), pp. 1299–1319, doi: [10.1162/089976698300017467](https://doi.org/10.1162/089976698300017467).
- [13] SNYMAN J.A. *Practical mathematical optimization: an introduction to basic optimization theory and classical and new gradient-based algorithms*. New York: Springer, 2005.
- [14] SZU H., HARTLEY R. Fast simulated annealing. *Physics Letters A*. 1987, 122(3–4), pp. 157–162.
- [15] VISWANATHAN G.M. Lévy flights and Superdiffusion in the Context of Biological Encounters and Random Searchers. *Physics of Life Reviews*. 2008, 5(3), pp. 133–150, doi: [10.1017/cbo9780511902680.014](https://doi.org/10.1017/cbo9780511902680.014).
- [16] XU R., WUNSCH D.C. *Clustering*. Piscataway, NJ: IEEE Press, 2009.
- [17] YOUNG N. *An introduction to Hilbert space*. New York: Cambridge University Press, 1988.