

# We are IntechOpen, the world's leading publisher of Open Access books Built by scientists, for scientists

**4,800**

Open access books available

**122,000**

International authors and editors

**135M**

Downloads

Our authors are among the

**154**

Countries delivered to

**TOP 1%**

most cited scientists

**12.2%**

Contributors from top 500 universities



**WEB OF SCIENCE™**

Selection of our books indexed in the Book Citation Index  
in Web of Science™ Core Collection (BKCI)

Interested in publishing with us?  
Contact [book.department@intechopen.com](mailto:book.department@intechopen.com)

Numbers displayed above are based on latest data collected.

For more information visit [www.intechopen.com](http://www.intechopen.com)



# Improvements Quality of Kohonen Maps Using Dimension Reduction Methods

Jiří Dvorský, Václav Snášel, Jana Kočíbová  
VŠB – Technical University of Ostrava  
Czech Republic

## 1. Introduction

The performance of *Self Organizing Map* (SOM) is always influenced by learn methods. The resultant quality of the SOM is also highly dependent onto the learning rate and the neighborhood function. In literature, there are plenty of studies to find a proper method to improve the quality of learning process of the SOM. They focus especially on convergence (Cottrell et al., 1998; Kohonen, 2001), the measure of the global topology preservation (Bauer et al., 1999) and, at an individual level, the sensitivity to parameters such as initialization, rate of decrease of neighborhood function, optimum learning rate etc. (de Bodt & Cottrell, 2000; Mulier & Cherkassky, 1995; Germen, 2005; Flanagan, 1996; 1994).

Although various disciplines use the SOM model in order to find solutions to broad spectrum of problems, however, there is not so much clue about the how the resultant maps are supposed to look after training.

Most articles are focused on the learning process of the SOM. The quality of the SOM is needed to measure in this process. The question is how to measure this quality. The distortion, or distortion measure, is certainly the most popular criterion for assessing the quality of the classification of the SOM (Kohonen, 2001; Rynkiewicz, 2006). Distortion measure provides an assessment of SOM properties with respect to the data and overcomes the absence of cost function in the SOM algorithm. Usually *Mean Squared Error* (MSE) is used to measure a distortion. The MSE is just a number without any dimension or scale, and may be hard to understand. What is the value of distortion is small enough? At what point should be the learning process terminated? Alternative approach for measurement of quality of learning process is the goal of our research.

In the SOM each neuron represents a set of input vectors. As the learning process continues, the set should be more and more stable, i.e. particular input vector should not move from one set to another in successive iteration in learning process. Movements, which still occur, can measure the quality of the learning process or quality of resultant SOM, if we decide to stop the learning.

Another idea is usage of a dimension reduction methods to capture most significant features of resultant SOM (Dvorský, 2007). At the beginning of the learning process, weights in the SOM are initialized with random values. In this case, there is no common, important feature in the SOM – the SOM contains only noise. How learning process continues, the map will learn significant features in the data. These features should be dominant, and if some approximation of the SOM is computed, these features must be preserved. In this moment, SVD or HOSVD, see sections 2.1 and 2.2, can be used to compute SOM approximation. In terms of

SVD, dominant features of the SOM will be represented as greater singular values. And vice versa, the small singular numbers will represent some noise in the SOM only. If small values are neglected, the approximated SOM will be very similar to the original one. In this way, quality of the SOM can be measured using data dimension reduction methods.

### 1.1 Notation of the SOM

In this chapter we will consider SOM as an array of neurons  $\mathbf{m}_i$ , usually arranged in a low dimensionality grid (1D or 2D), the map, for ease of visualization. The grid may have several forms like rectangular, hexagonal. For each input vector  $\mathbf{x}(t)$ , neuron that best matches input vector is selected. This neuron is called *Best Matching Unit (BMU)*. Then the weights of the BMU and its neighborhood will be adapted as follows:

$$\mathbf{m}_i(t+1) = \mathbf{m}_i(t) + \eta(t)h(t)(\mathbf{x}(t) - \mathbf{m}_i(t)), \quad \forall \mathbf{m}_i \quad (1)$$

where  $\eta$ ,  $0 < \eta < 1$  is the learning factor, which determines the speed of weight adaptation, and  $h(t)$  is neighborhood size determining function.

## 2. Methods of Dimension Reduction

Since our approach is based on SVD and HOSVD techniques, we first briefly review matrix SVD and then introduce tensor and the HOSVD technique. In this chapter, tensors are denoted by calligraphic upper-case letters ( $\mathcal{A}, \mathcal{B}, \dots$ ), matrices by uppercase letters ( $A, B, \dots$ ), vectors by bold lower case letters ( $\mathbf{a}, \mathbf{b}, \dots$ ), and scalars by lower case letters ( $a, b, \dots$ ).

### 2.1 Singular Value Decomposition

*Singular value decomposition (SVD)* is well known because of its application in information retrieval – *Latent semantic indexing (LSI)* (Berry & Browne, 1999; Berry et al., 1995). SVD is especially suitable in its variant for sparse matrices (Larsen, 1998).

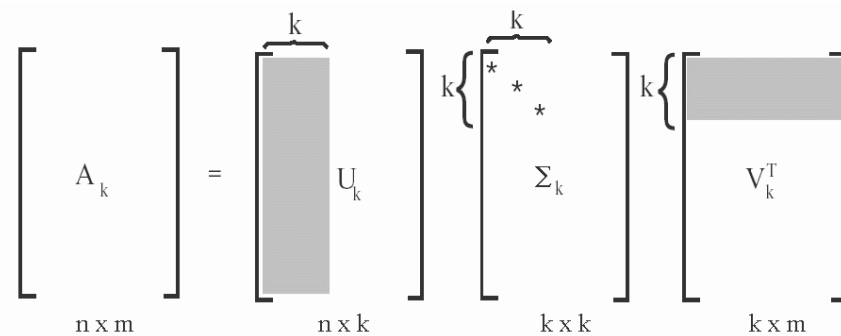
**Theorem 2.1 (Singular value decomposition)** *Let  $A$  is an  $n \times m$  rank- $r$  matrix. Be  $\sigma_1 \geq \dots \geq \sigma_r$  eigenvalues of a matrix  $\sqrt{AA^T}$ . Then there exist orthogonal matrices  $U = (\mathbf{u}_1, \dots, \mathbf{u}_r)$  and  $V = (\mathbf{v}_1, \dots, \mathbf{v}_r)$ , whose column vectors are orthonormal, and a diagonal matrix  $\Sigma = \text{diag}(\sigma_1, \dots, \sigma_r)$ . The decomposition  $A = U\Sigma V^T$  is called singular value decomposition of matrix  $A$  and numbers  $\sigma_1, \dots, \sigma_r$  are singular values of the matrix  $A$ . Columns of  $U$  (or  $V$ ) are called left (or right) singular vectors of matrix  $A$ .*

Now we have a decomposition of original matrix  $A$ . It is not needed to say, that the left and right singular vectors are not sparse. We have at most  $r$  nonzero singular numbers, where rank  $r$  is the smaller of the two matrix dimensions. Luckily, because the singular values usually fall quickly, we can take only  $k$  greatest singular values and corresponding singular vector coordinates and create a  $k$ -reduced singular decomposition of  $A$ .

**Definition 2.1** *Let us have  $k, 0 < k < r$  and singular value decomposition of  $A$*

$$A = U\Sigma V^T = (U_k U_0) \begin{pmatrix} \Sigma_k & 0 \\ 0 & \Sigma_0 \end{pmatrix} \begin{pmatrix} V_k^T \\ V_0^T \end{pmatrix}$$

We call  $A_k = U_k \Sigma_k V_k^T$  a  $k$ -reduced singular value decomposition (rank- $k$  SVD).

Fig. 1. rank- $k$  Singular Value Decomposition

For an illustration of rank- $k$  SVD see Figure 1, the gray areas determine first  $k$  coordinates from singular vectors, which are being used.

**Theorem 2.2 (Eckart-Young)** *Among all  $n \times m$  matrices  $C$  of rank at most  $k$   $A_k$  is the one, that minimizes  $\|A_k - A\|_F^2 = \sum_{i,j} (A_{i,j} - C_{w,j})^2$ .*

Because rank- $k$  SVD is the best rank- $k$  approximation of original matrix  $A$ , any other decomposition will increase the sum of squares of matrix  $A - A_k$ .

SVD rank- $k$  approximation of original matrix  $A$  can be understood in several ways. In information retrieval, matrix  $A$  represents term-document matrix, and a latent semantics is obtained using rank- $k$  approximation of original matrix  $A$ . From another point of view rank- $k$  approximation can be viewed as elimination of noise from data represents in matrix  $A$ . The SVD is computed by a batch  $O(nm^2 + n^2m + m^3)$  time algorithm (Golub & Loan, 1996), that is unfeasible for large datasets, but for our case this algorithm is fully adequate.

**Note 2.1** *From now on, we will assume rank- $k$  singular value decomposition when speaking about SVD.*

## 2.2 Tensors and HOSVD

A tensor is a higher order generalization of a vector (first order tensor) and a matrix (second order tensor). Higher order tensors are also called multidimensional matrices or multi-way arrays. The order of a tensor  $\mathcal{A} \in \mathbb{R}^{I_1 \times I_2 \times \dots \times I_N}$  is  $N$ . Elements of  $\mathcal{A}$  are denoted as  $a_{i_1 \dots i_n \dots i_N}$  where  $1 \leq i_n \leq I_n$ . In tensor terminology, matrix column vectors are referred to as mode-1 vectors and row vectors as mode-2 vectors. The mode- $n$  vectors of an  $N$ -th order tensor  $\mathcal{A}$  are the  $I_n$ -dimensional vectors obtained from  $\mathcal{A}$  by varying the index  $i_n$  and keeping the other indices fixed, that is the column vectors of  $n$ -mode matrix unfolding  $\mathcal{A}_{(n)} \in \mathbb{R}^{I_n \times (I_1 I_2 \dots I_{n-1} I_{n+1} \dots I_N)}$  of tensor  $\mathcal{A}$ . See (De Lathauwer et al., 2000) for details on matrix unfoldings of a tensor. The  $n$ -mode product of a tensor  $\mathcal{A} \in \mathbb{R}^{I_1 \times I_2 \times \dots \times I_N}$  by a matrix  $M \in \mathbb{R}^{J_n \times I_n}$  is an  $I_1 \times I_2 \times \dots \times I_{n-1} \times J_n \times I_{n+1} \times \dots \times I_N$ -tensor of which the entries are given by

$$(\mathcal{A} \times_n M)_{i_1 \dots i_{n-1} j_n i_{n+1} \dots i_N} = \sum_{i_n} a_{i_1 \dots i_{n-1} i_n i_{n+1} \dots i_N} m_{j_n i_n} \quad (2)$$

Note that the  $n$ -mode product of a tensor and a matrix is a generalization of the product of two matrices. It can be expressed in terms of matrix unfolding:

$$B_{(n)} = M A_{(n)} \quad (3)$$

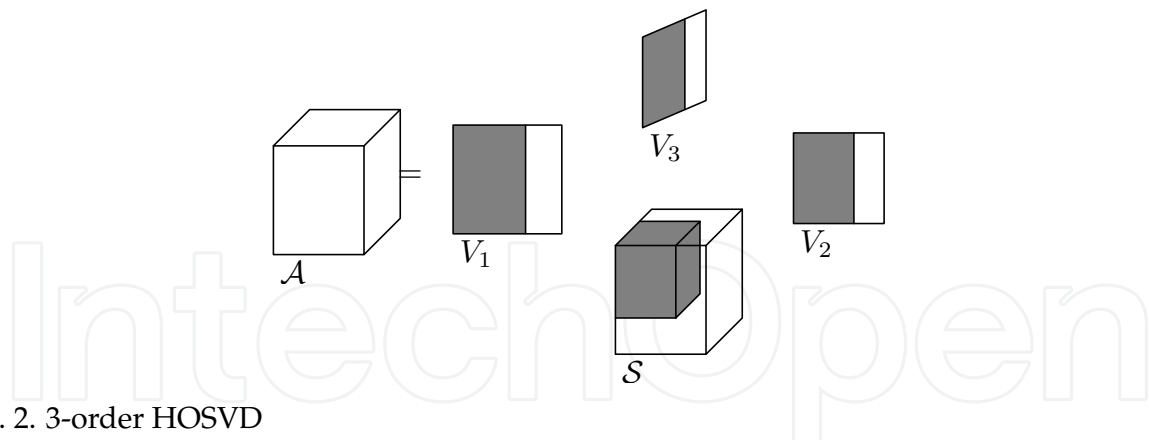


Fig. 2. 3-order HOSVD

where  $B_{(n)}$  is the  $n$ -mode unfolding of tensor  $\mathcal{B} = \mathcal{A} \times_n M$ .

In terms of  $n$ -mode products, the matrix SVD can be rewritten as  $F = S \times_1 V^{(1)} \times_2 V^{(2)}$ . By extension, *High order singular value decomposition (HOSVD)* is a generalization of matrix SVD: every  $I_1 \times I_2 \times \dots \times I_N$  tensor  $\mathcal{A}$  can be written as the  $n$ -mode product (De Lathauwer et al., 2000):

$$\mathcal{A} = S \times_1 V^{(1)} \times_2 V^{(2)} \dots \times_N V^{(N)} \quad (4)$$

HOSVD is illustrated in Figure 2 for  $N = 3$ .  $V_n$  contains the orthonormal vectors (called  $n$ -mode singular vectors) spanning the column space of the matrix  $A_{(n)}$  ( $n$ -mode matrix unfolding of tensor  $\mathcal{A}$ ).  $S$  is called core tensor. Instead of being pseudodiagonal (nonzero elements only occur when the indices satisfy  $i_1 = i_2 = \dots = i_N$ ),  $S$  has the property of all-orthogonality. That is, two subtensors  $\mathcal{S}_{i_n = \alpha}$  and  $\mathcal{S}_{i_n = \beta}$  are orthogonal for all possible values of  $n$ ,  $\alpha$  and  $\beta$  subject to  $\alpha \neq \beta$ . At the same time, the Frobenius-norms  $\sigma_i^n = \|\mathcal{S}_{i_n = i}\|$  are  $n$ -mode singular values of  $\mathcal{A}$  and are in decreasing order:  $\sigma_1^n \geq \sigma_2^n \geq \dots \geq \sigma_{I_n}^n \geq 0$ .  $S$  is in general a full tensor and governs the interactions among  $V_n$ .

### 3. SVD and HOSVD Approximation of SOM

#### 3.1 BMU Movements

BMU is found for each training vector during learning process. As the SOM learns the structure of training set, the BMU of given training vector usually changes its position within the map. Movement of the BMU at the initial phase of learning process will be probably very rapid, and as the SOM converges to stable configuration the movement of the BMU will be very tight. The learning process could be stopped, when user specific maximal number of moved BMUs is reached.

In this way, the number of moved BMUs can be taken as alternative learning process quality measurement. The number of changes of BMUs' positions between successive iterations was considered as measure in our initial work. But this approach is not very helpful. Some movements of the BMU still remain.

#### 3.2 SVD Approach

The second approach to measurement of BMU movement uses SVD decomposition of SOM. It is supposed, that movement of BMUs among original SOM and its rank- $k$  approximations will be very low, when stable configuration of the SOM is reached.

To verify this hypothesis following experiment was performed:

1. The SOM  $S$  is transformed to  $rc \times m$  matrix  $A$ , where  $r$  is the number of rows,  $c$  is the number of columns of SOM  $S$ , and  $m$  is the dimension of input.

$$A = \begin{pmatrix} \mathbf{m}_{1,1}(t) \\ \mathbf{m}_{1,2}(t) \\ \vdots \\ \mathbf{m}_{1,c}(t) \\ \mathbf{m}_{2,1}(t) \\ \vdots \\ \mathbf{m}_{r,c}(t) \end{pmatrix} \quad (5)$$

**Note 3.1** Each  $\mathbf{m}_{i,j}(t)$  is  $m$  dimensional vector.

2. Rank  $k$  approximation  $A_k$  of matrix  $A$  is computed,  $1 \leq k \leq m$ .
3. SOM  $S(k)$  is created from matrix  $A_k$ .
4. The  $BMU_S$  for each training vector is computed with original SOM  $S$ .
5. The  $BMU_S(k)$  is computed with rank  $k$  approximation  $S(k)$  for each training vector.
6. If  $BMU_S$  is different from  $BMU_S(k)$  the movement is encountered.

### 3.3 HOSVD Approach

The SOM forms usually 2D dimensional grid, each neuron is represented as  $m$  dimensional vector, where  $m$  is the dimension of input, training, vectors. From this point of view, SOM can be understood as 3-order tensor, and 3-dimensional HOSVD, see Figure 2, can be applied onto SOM. There is no need to form matrix by transformation of SOM. 3-order HOSVD can directly decompose SOM. It is expected that HOSVD preserves better relationships among neurons and structure of SOM.

Experiment similar as in SVD approach was performed:

1. The SOM  $S$  is transformed to tensor  $\mathcal{A} \in R^{r \times c \times m}$ , where  $r$  is the number of rows,  $c$  is the number of columns of SOM  $S$ , and  $m$  is the dimension of input.
2.  $\mathcal{A}_{(k_1, k_2, k_3)}$  approximation of tensor  $\mathcal{A}$  was computed. This approximation generalizes rank- $k$  approximation of matrix in SVD.
3. SOM  $S(k_1, k_2, k_3)$  is created from tensor  $\mathcal{A}_{(k_1, k_2, k_3)}$ .
4. The  $BMU_S$  for each training vector is computed with original SOM  $S$ .
5. The  $BMU_S(k_1, k_2, k_3)$  is computed with approximated SOM  $S(k_1, k_2, k_3)$  for each training vector.
6. If  $BMU_S$  is different from  $BMU_S(k_1, k_2, k_3)$  the movement is encountered.

## 4. Experimental Results

A number of experiments were carried out to prove our hypothesis. One of them is provided in this chapter. Parameters of used SOM  $S$  are given in Table 1. The input data comes from experiments done by Kudelka et al. (Kudelka et al., 2006).

Parameter	Value
# of rows	50
# of columns	50
SOM shape	toroid
input dimension	10
# of input vectors	approx 25,000
# of iterations	5000

Table 1. Experimental SOM  $S$  parameters

#### 4.1 Results of SVD Approach

The experimental results are summarized in following tables and graphs. Table 2 provides insight to learning process. Each rank- $k$ ,  $1 \leq k \leq 10$ , approximation  $S(k)$  of SOM  $S$  is computed and number of moved BMUs are stored, moreover a distance of the movement.

In initial phase of learning process, there is no significant difference among  $S(k)$  approximations. The number of moved BMUs varies from 99% to 26% in iteration 0. Average distance of movement is also near constant, from 13.6 to 15.4.

There are significant difference among each  $S(k)$  approximations after 4000 and 5000 iterations.  $S(1)$  approximation has 76% of moved input vectors, but  $S(8)$  resp.  $S(9)$  has only 9% resp. 2.6% of moved input vectors after 5000 iterations. The average distance is also very small, only 2.4 resp. 3.

Although maximal distance of the movement remains very high in all phases of learning process, the average distance of BMUs movement decreases. Histograms of these distances can provide very useful information about learning. Figure 3(a) demonstrates initial state of SOM  $S$ . The distribution of distances is random; there are plenty of short movements and also a lot of long movement of BMUs. The charts 3(d) have different shape. Histogram of distances has strong hyperbolic shape, i.e. very short movements are predominate. Cumulative histogram shows, that movements of length 1 (less than 20%) comprise slightly more than 80% of all movements for  $S(8)$  resp.  $S(9)$  approximations. These results coincide with the widely known "Pareto's 80/20 law". In other words, SOMs  $S(8)$  resp.  $S(9)$  are very close to original SOM  $S$ , consequently SOM  $S$  after 5000 iterations completed contains very small amount of noise (compare  $S(9)$  approximation after 500 iterations completed). We can conclude, that well trained SOM is resistant to SVD decomposition.

U-matrices (Ultsch, 1993) of experimental SOM  $S$  and its approximations  $S(k)$  are given in Figure 4. These figures show, that even  $S(1)$  approximation is used, the main features, main structure, of original SOM is preserved.  $S(2)$  resp.  $S(3)$  approximation add more details to basic structure of  $S(1)$  approximation.  $S(8)$  resp.  $S(9)$  approximations are not easily distinguishable from original SOM.

#### 4.2 Results of HOSVD Approach

On the other hand performance of HOSVD approximation of SOM  $S$  was disappointing. U-matrices of approximated SOM are very similar to SVD ones, see Figure 6. Also cumulative histograms of BMUs' movements show very similar characteristics. But when absolute values are compared, there are very high differences, see Table 3 e.g. only 616 BMUs were moved after 5,000 iterations in rank-9 SVD approximation, which is only 2.63 % of all input vectors. But in similar HOSVD approximation more than 9,000 BMUs were moved.

## Acknowledgement

This work is supported by Grant of Grant Agency of Czech Republic No. 205/09/1079.

## 5. Conclusion

Alternative approach of quality measurement of the SOM learning process was presented. This approach uses SVD decomposition of the SOM, and number of moved BMUs are counted among each consecutive rank- $k$  approximations of original SOM. Our experiments show some properties of SVD decomposition of SOM. That is new and good.

On the other hand HOSVD cause some problems, because number of moved BMUs is very high. It seems that SOM is not true three dimensional object – the third dimension consisting of neurons is not relevant. We should check if this behavior is feature or bug of this method or if it is a matter of coincidence. This is new and quite disappointing.

Usage of Non-negative Matrix Factorization (Golub & Loan, 1996) is our goal in future research. The second goal is application of presented method on the other type of neural network, such as Growing Neural Gas (Fritzke, 1994).

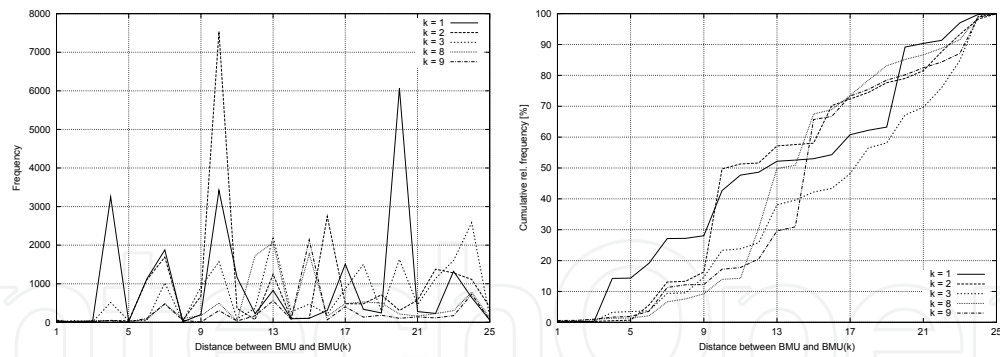
## 6. References

- Bauer, H. U., Herrmann, M. & Villmann, T. (1999). Neural maps and topographic vector quantization, *Neural Networks* **12**(4): 659–676.
- Berry, M. & Browne, M. (1999). *Understanding Search Engines, Mathematical Modeling and Text Retrieval*, Siam.
- Berry, M., Dumais, S. & Letsche, T. (1995). Computation Methods for Intelligent Information Access, *Proceedings of the 1995 ACM/IEEE Supercomputing Conference*.
- Cottrell, M., Fort, J. C. & Pages, G. (1998). Theoretical aspects of the SOM algorithm, *Neurocomputing* **21**(1): 119–138.
- de Bodt, E. & Cottrell, M. (2000). Bootstrapping self-organising maps to assess the statistical significance of local proximity, *8th European Symposium on Artificial Neural Networks. ESANN'2000. Proceedings. D-Facto, Brussels, Belgium*, pp. 245–54.
- De Lathauwer, L., Moor, B. D. & Vandewalle, J. (2000). A multilinear singular value decomposition, *SIAM Journal on Matrix Analysis and Applications* **21**(4): 1253–1278.  
**URL:** [citeseer.ist.psu.edu/delathauwer00multilinear.html](http://citeseer.ist.psu.edu/delathauwer00multilinear.html)
- Dvorský, J. (2007). Self-organizing maps and svd, *DEXA Workshops*, IEEE Computer Society, pp. 143–147.
- Flanagan, J. A. (1994). *Self-Organizing Neural Networks*, PhD thesis, École Polytechnique Fédérale de Lausanne, Lausanne.
- Flanagan, J. A. (1996). Self-organization in Kohonen's SOM, *Neural Networks* **9**: 1185–1197.
- Fritzke, B. (1994). A growing neural gas network learns topologies, in G. Tesauro, D. S. Touretzky & T. K. Leen (eds), *NIPS*, MIT Press, pp. 625–632.
- Germen, E. (2005). Improving the resultant quality of kohonen's self organizing map using stiffness factor, in L. Wang, K. Chen & Y.-S. Ong (eds), *ICNC (1)*, Vol. 3610 of *Lecture Notes in Computer Science*, Springer, pp. 353–357.
- Golub, G. H. & Loan, C. F. V. (1996). *Matrix Computations*, Johns Hopkins University Press.
- Kohonen, T. (2001). *Self-Organizing Maps*, 3rd edn, Springer-Verlag New York, LLC.
- Kudelka, M., Snasel, V., Lehecka, O. & El-Qawasmeh, E. (2006). Semantic analysis of web pages using web patterns, *2006 IEEE/WIC/ACM International Conference on Web Intelligence* **0**: 329–333.

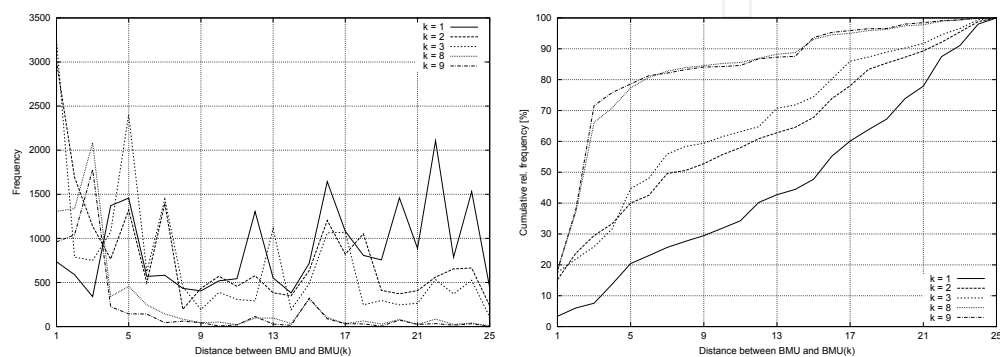


- Larsen, R. M. (1998). Lanczos bidiagonalization with partial reorthogonalization, *Technical report*, University of Aarhus.
- Mulier, F. M. & Cherkassky, V. S. (1995). Statistical analysis of self-organization, *Neural Networks* 8(5): 717–727.
- Rynkiewicz, J. (2006). Self-organizing map algorithm and distortion measure, *Neural Networks* 19(6-7): 830–837.
- Ultsch, A. (1993). Self-organizing neural networks for visualization and classification, in O. Opitz, B. Lausen & R. Klar (eds), *Information and Classification*, Springer, London, UK, pp. 307–313.

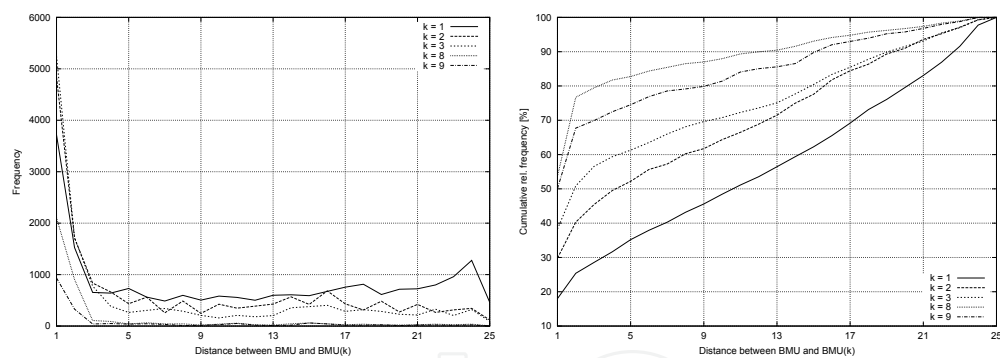
IntechOpen



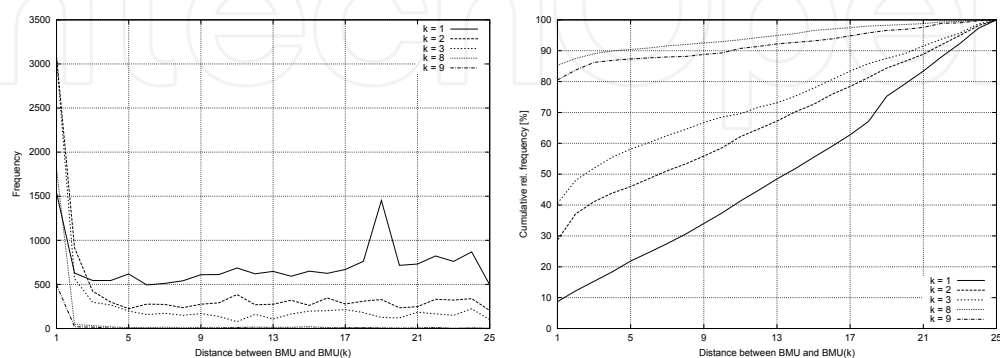
(a) Initial SOM with random weights



(b) SOM after 500 iterations



(c) SOM after 4000 iterations



(d) SOM after 5000 iterations (final state)

Fig. 3. Frequency and cumulative frequency histograms of distances between  $BMU_S$  and  $BMU_S(k)$  for SOM  $S$

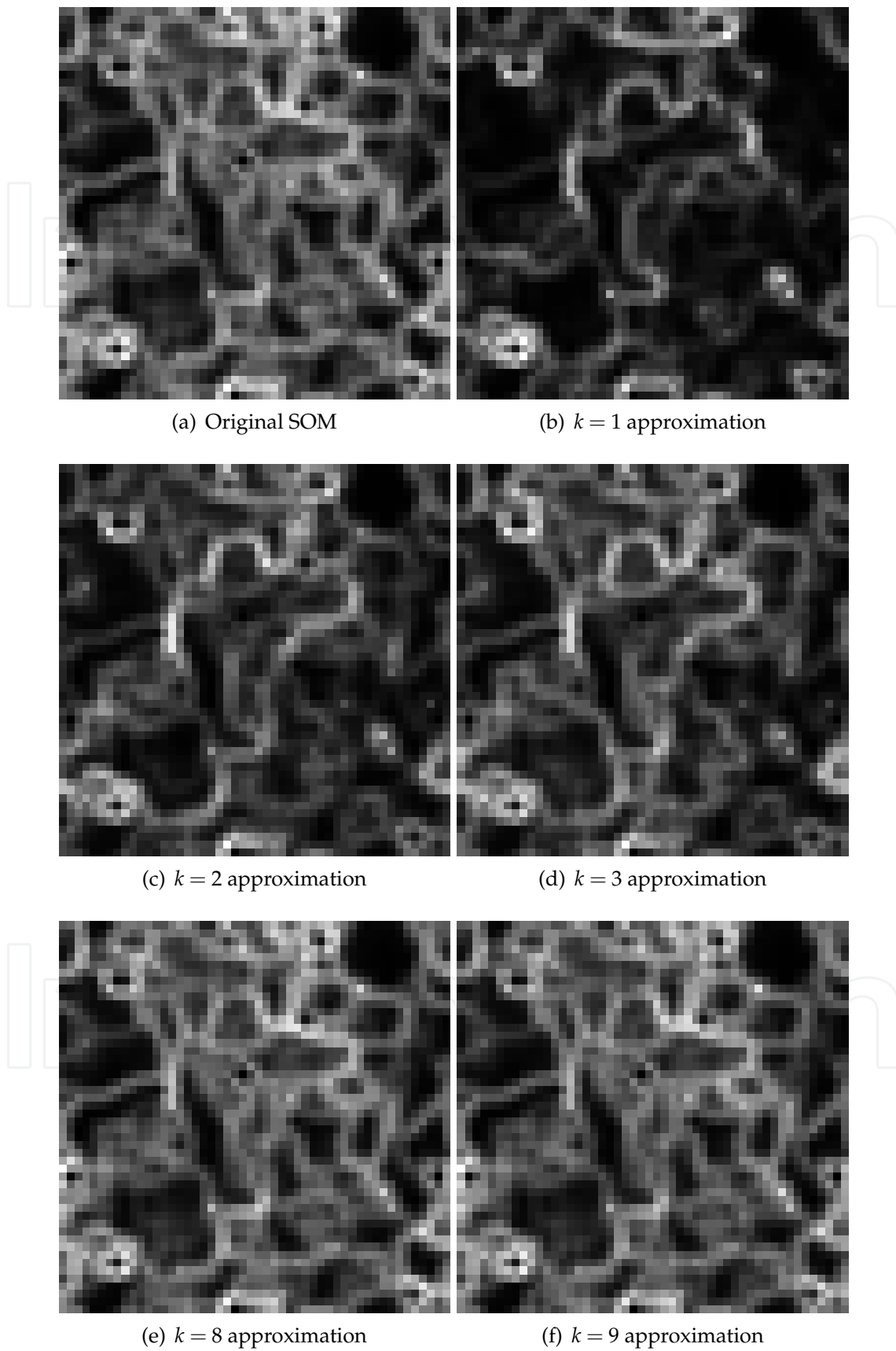
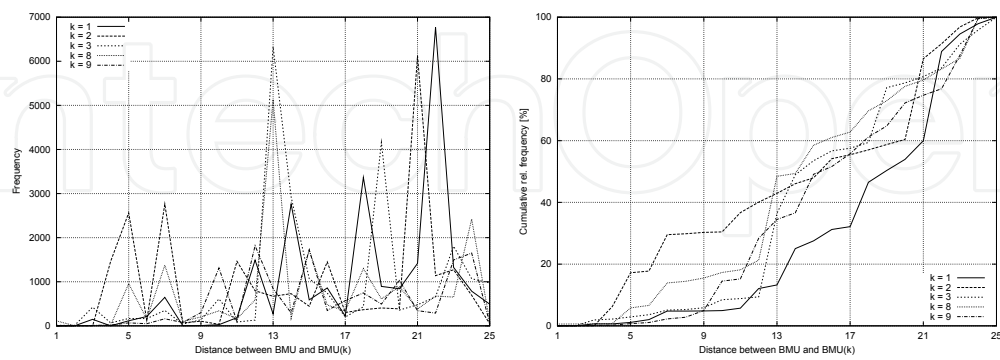
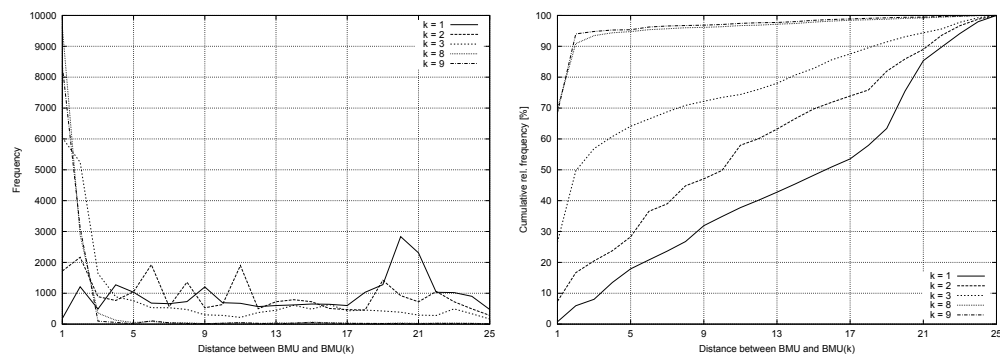


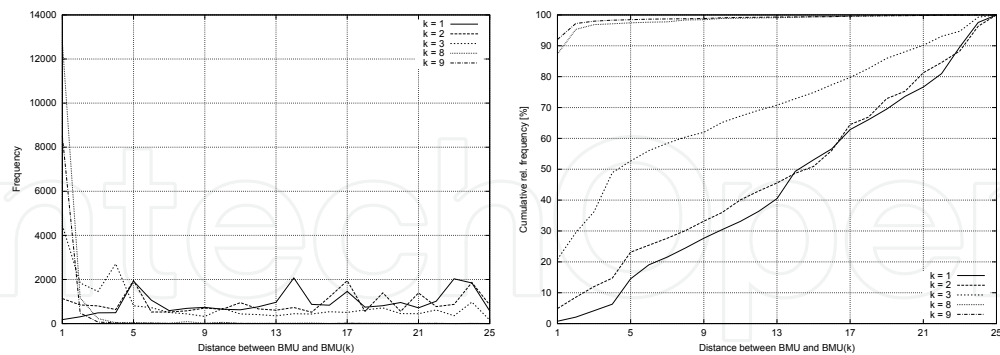
Fig. 4. SVD rank- $k$  approximations of final SOM  $S$  (after 5000 iterations)



(a) Initial SOM with random weights

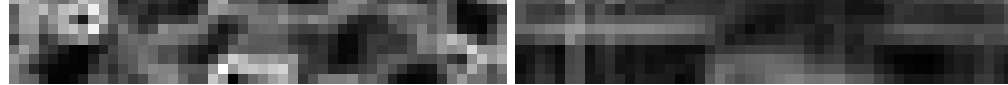


(b) SOM after 4000 iterations



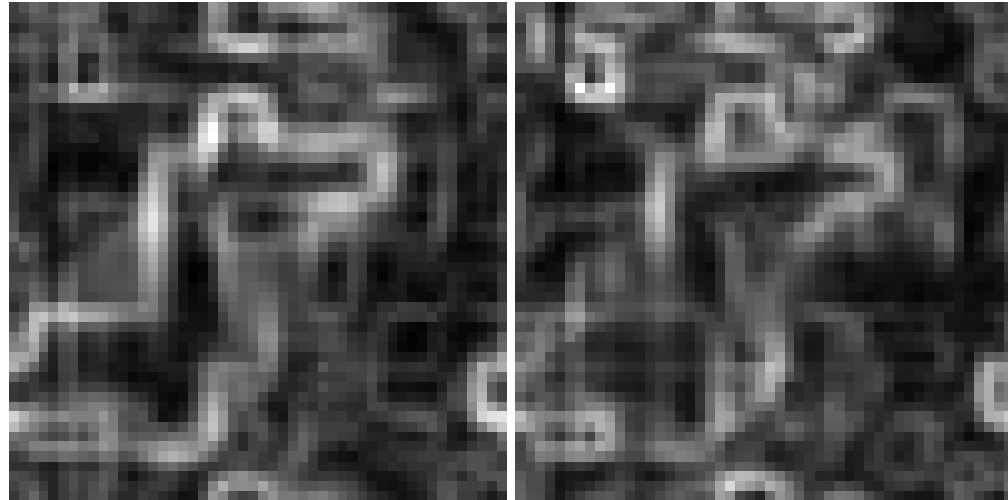
(c) SOM after 5000 iterations (final state)

Fig. 5. Frequency and cumulative frequency histograms of distances between  $BMU_S$  and  $BMU_S(k_1, k_2, k_3)$  for SOM  $S$



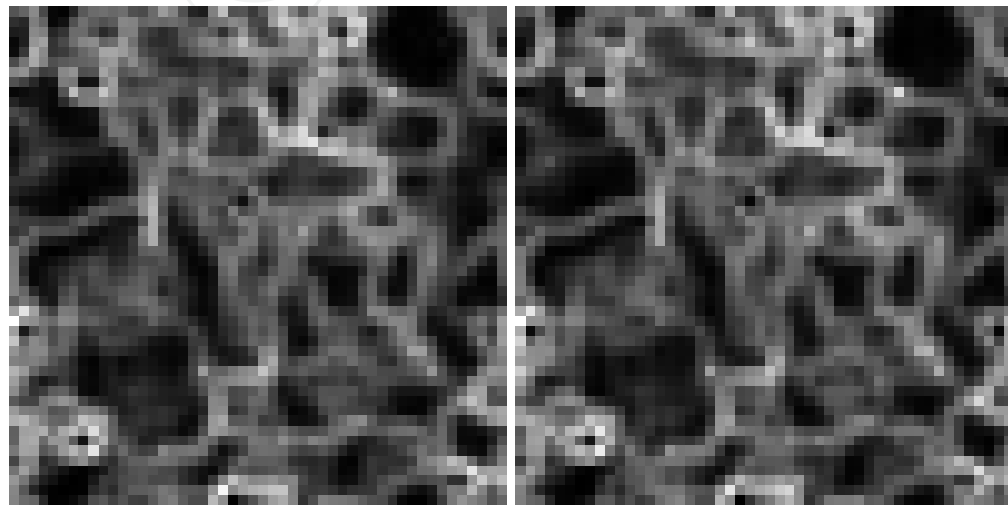
(a) Original SOM

(b)  $k_1 = 5, k_2 = 5, k_3 = 1$  approximation



(c)  $k_1 = 10, k_2 = 10, k_3 = 2$  approximation

(d)  $k_1 = 15, k_2 = 15, k_3 = 3$  approximation



(e)  $k_1 = 40, k_2 = 40, k_3 = 8$  approximation

(f)  $k_1 = 45, k_2 = 44, k_3 = 9$  approximation

Fig. 6.  $S(k_1, k_2, k_3)$  HOSVD approximations of final SOM  $S$  (after 5000 iterations)

	Number of singular values $k$							
	1	2	3	4	5	6	7	8
	0 completed iterations							
MSE	0.784	0.74	0.712	0.64	0.614	0.57	0.536	0.537
# of moved BMUs	23,381	22,570	18,053	15,897	14,963	12,326	11,031	10,654
Moved BMUs [%]	99.825	96.362	77.077	67.872	63.884	52.626	47.097	45.487
Max. distance	25	25	25	25	25	25	25	25
Average distance	13.6	13.9	16.6	17.7	17.2	16.3	17	14.9
	500 completed iterations							
MSE	0.603	0.456	0.366	0.303	0.244	0.209	0.172	0.132
# of moved BMUs	22,000	19,805	18,468	16,496	13,823	11,681	9,788	7,147
Moved BMUs [%]	93.929	84.557	78.849	70.43	59.017	49.872	41.79	30.514
Max. distance	25	25	25	25	25	25	25	25
Average distance	14.3	10	9.2	6.6	6.4	5.9	5.5	5
	4000 completed iterations							
MSE	0.595	0.452	0.356	0.284	0.236	0.184	0.143	0.092
# of moved BMUs	20,643	16,252	13,698	11,964	9,353	8,185	4,907	3,925
Moved BMUs [%]	88.135	69.388	58.483	51.08	39.933	34.946	20.95	16.758
Max. distance	25	25	25	25	25	25	25	25
Average distance	11.4	8	6.9	5.8	4.8	3.8	4.2	3.7
	5000 completed iterations							
MSE	0.595	0.452	0.355	0.284	0.229	0.18	0.119	0.083
# of moved BMUs	1,7748	10,760	7,555	6,868	5,058	3,781	2,564	2,125
Moved BMUs [%]	75.775	45.94	32.256	29.323	21.595	16.143	10.947	9.073
Max. distance	25	25	25	25	25	25	25	25
Average distance	13.2	9.1	7.5	5.5	5	4	3.7	2.4

Table 2. Parameters of SVD rank- $k$  approximations of given SOM

IntechOpen

	Approximation parameters $(k_1, k_2, k_3)$						
	(5, 5, 1)	(10, 10, 2)	(15, 15, 3)	(20, 20, 4)	(25, 25, 5)	(30, 30, 6)	(35, 35, 7)
	0 completed iterations						
MSE	1.485	1.218	1.136	0.979	0.887	0.754	0.696
# of moved BMUs	23,422	23,421	23,407	23,104	23,299	23,109	21,829
Moved BMUs [%]	100	99.996	99.936	98.642	99.475	98.664	93.199
Max. distance	25	25	25	25	25	25	25
Average distance	18.4	14.6	16.3	16.7	15	15.9	15.9
	4000 completed iterations						
MSE	0.595	0.452	0.356	0.285	0.237	0.184	0.143
# of moved BMUs	23,404	23,304	22,722	21,671	20,141	18,890	17,265
Moved BMUs [%]	99.923	99.496	97.011	92.524	85.992	80.651	73.713
Max. distance	25	25	25	25	25	25	25
Average distance	14.3	11	6.6	6.5	5	2.5	2.3
	5000 completed iterations						
MSE	0.595	0.453	0.356	0.285	0.23	0.182	0.12
# of moved BMUs	23,398	23,220	21,419	18,217	19,305	18,132	15,638
Moved BMUs [%]	99.898	99.138	91.448	77.777	82.423	77.414	66.766
Max. distance	25	25	25	25	25	25	25
Average distance	14.6	13.7	8.6	6	3.4	2.2	1.8

Table 3. Parameters of  $S(k_1, k_2, k_3)$  HOSVD approximations of given SOM  $S$



## **Self-Organizing Maps**

Edited by George K Matsopoulos

ISBN 978-953-307-074-2

Hard cover, 430 pages

**Publisher** InTech

**Published online** 01, April, 2010

**Published in print edition** April, 2010

The Self-Organizing Map (SOM) is a neural network algorithm, which uses a competitive learning technique to train itself in an unsupervised manner. SOMs are different from other artificial neural networks in the sense that they use a neighborhood function to preserve the topological properties of the input space and they have been used to create an ordered representation of multi-dimensional data which simplifies complexity and reveals meaningful relationships. Prof. T. Kohonen in the early 1980s first established the relevant theory and explored possible applications of SOMs. Since then, a number of theoretical and practical applications of SOMs have been reported including clustering, prediction, data representation, classification, visualization, etc. This book was prompted by the desire to bring together some of the more recent theoretical and practical developments on SOMs and to provide the background for future developments in promising directions. The book comprises of 25 Chapters which can be categorized into three broad areas: methodology, visualization and practical applications.

### **How to reference**

In order to correctly reference this scholarly work, feel free to copy and paste the following:

Jiri Dvorsky, Vaclav Snasel and Jana Kocibova (2010). Improvements Quality of Kohonen Maps Using Dimension Reduction Methods, *Self-Organizing Maps*, George K Matsopoulos (Ed.), ISBN: 978-953-307-074-2, InTech, Available from: <http://www.intechopen.com/books/self-organizing-maps/improvements-quality-of-kohonen-maps-using-dimension-reduction-methods>

**INTECH**  
open science | open minds

### **InTech Europe**

University Campus STeP Ri  
Slavka Krautzeka 83/A  
51000 Rijeka, Croatia  
Phone: +385 (51) 770 447  
Fax: +385 (51) 686 166  
[www.intechopen.com](http://www.intechopen.com)

### **InTech China**

Unit 405, Office Block, Hotel Equatorial Shanghai  
No.65, Yan An Road (West), Shanghai, 200040, China  
中国上海市延安西路65号上海国际贵都大饭店办公楼405单元  
Phone: +86-21-62489820  
Fax: +86-21-62489821



© 2010 The Author(s). Licensee IntechOpen. This chapter is distributed under the terms of the [Creative Commons Attribution-NonCommercial-ShareAlike-3.0 License](#), which permits use, distribution and reproduction for non-commercial purposes, provided the original is properly cited and derivative works building on this content are distributed under the same license.

IntechOpen

IntechOpen