**Instruction manual for Simple BL-SOM(NAIST) with Comparison Facility**

Shigehiko Kanaya, Md. Altaf-Ul-Amin, Ken Kurokawa

**Introduction**

We modified the conventional SOM and introduced batch-learning SOM (BL-SOM), to make the learning process and resulting map independent of the order of input data. Furthermore, the initial weight vectors were defined using principal component analysis (PCA) instead of random values, based on the fact that multivariate analyses, including PCA, have successfully classified gene sequences into groups corresponding to known biological categories when a relatively small number of gene sequences were analyzed. Therefore, the BL-SOM is independent of not only the data input order but also the initial condition. The following papers utilized BL-SOM as the method of analysis. So please use these papers as references when you analyze multivariate data by using this software. BL-SOM was developed in ref.1 and has been applied to various Bioinformatic researches (refs 1,2,3 and 4). Hirai et al applied this algorithm in metabolomics (refs 3 and 4).

1 S. Kanaya, M. Kinouchi, T. Abe, Y. Kudo, Y. Yamada, T. Nishi, H. Mori, T. Ikemura. Analysis of codon usage diversity for bacterial genes with a self-organizing map (SOM): characterization of horizontally transferred genes with emphasis on the *E. coli* O157 genome., *Gene*, 276, 89-99 (2001)

2 T. Abe, S. Kanaya, M. Kinouchi, Y. Ichiba, T. Kozuki, T. Ikemura, Informatics for unvailing hidden genome signature., *Genome Res.*, 13, 693-702 (2003).

3 M. Hirai, M. Yano, D. Goodenowe, S. Kanaya, T. Kimura, M.Awazuhara, M. Arita, T. Fujiwara, K. Saito, Integration of transcriptomics and metabolomics for understanding of global responses to nutritional stresses in *Arabidopsis thaliana*, *Proc. Natl. Acad. Sci., USA*, 101, 10205-10210 (2004).

4. M. Hirai, M. Klein, Y. Fujikawa, M. Yano, D.B. Goodenowe, Y. Yamazaki, S. Kanaya, Y. Nakamura, M. Kitayama, H. Suzuki, N. Sakurai, D. Shibata, J. Tokuhisa, M. Reichelt, J. Gershenzon, J. Papenbrock, K. Saito, Elucidation of gene-to-gene and metabolite-to-gene networks in arabidopsis by integration of metabolomics and transcriptomics. *J. Biol. Chem.*, 280,25590-5 (2005).

## 1. Execution of Simple BL-SOM(NAIST) with Comparison Mode

Java j2sdk-1.4.2 is required to be installed in the user's computer. First, the compressed file, SimpleSOM_forWIN.zip is to be downloaded from http://kanaya.naist.jp/SOM/. Under the 'SimpleSOM' folder, there is a folder 'Data' and an executable file 'simpleSOM.jar'.

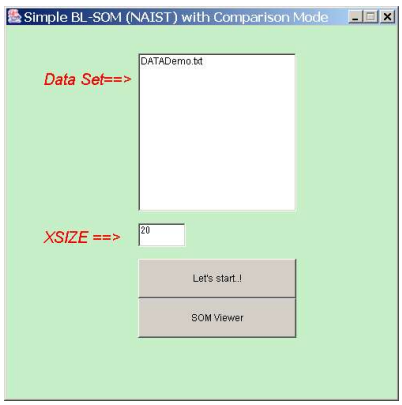

### 1.1 Format of input data

The data set must be constructed as a text file and the file name should start with 'DATA'. Each column in the data file is separated by a tab. The first column corresponds to the object (or sample) name and measurements of experiments such as microarray experiments or so on are written in from the second to the last columns. An example of the input data set is shown in the file 'DATADemo.txt' under the folder 'Data'.

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| yaaA | 0.112 | 0.018 | 0.043 | 0.040 | … | … | … |
| recF | 0.033 | -0.013 | 0.021 | 0.050 | … | ... | … |
| yaaB | 0.103 | 0.028 | 0.042 | 0.003 | … | … | … |
| yaaB | … | … | … | … | … | … | … |
| ……. | | | | | | | |

### 1.2 Execution of Simple SOM

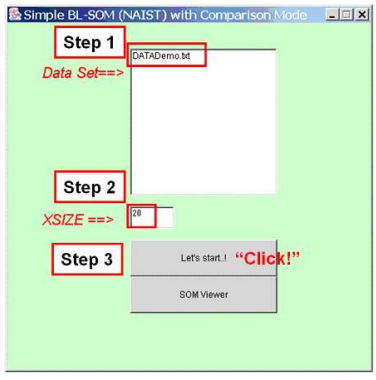
User can start by clicking the file simpleSOM.jar. The main window is shown in Panel 1. Simple BL-SOM consists of two modes of data processing, (A) construction of Self-organizing map, and (B) Visualization of Self-organizing map.



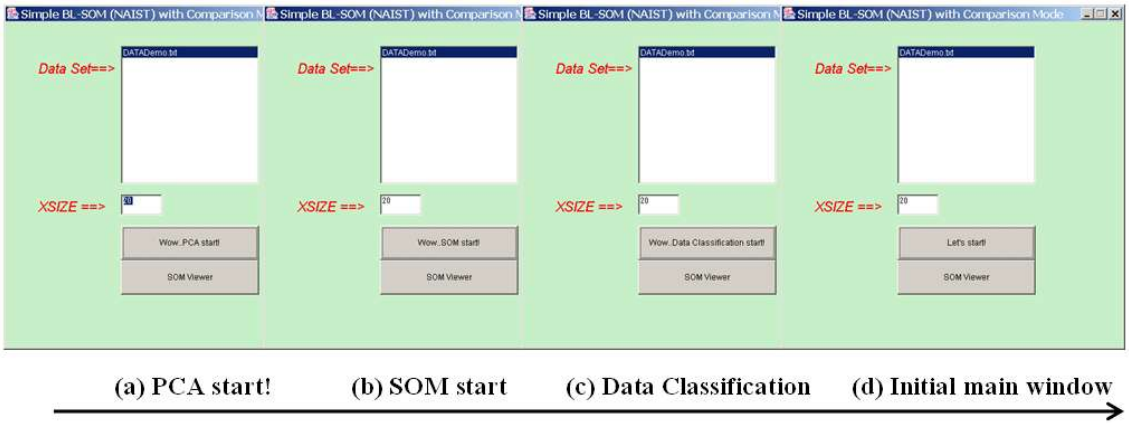

**Panel 1**

**(A) Construction of Self-organizing map**

Select the input data file of interest from the list in the 'Data Set' box (Step 1 in **Scheme 1**), set XSIZE (Step 2) and click 'Let's start.!' button, then, initial weight vectors are set by PCA (**Panel 2a**), weight vectors are updated by BL-SOM algorithm (**Panel 2b**), samples are classified based on the final weight vectors.



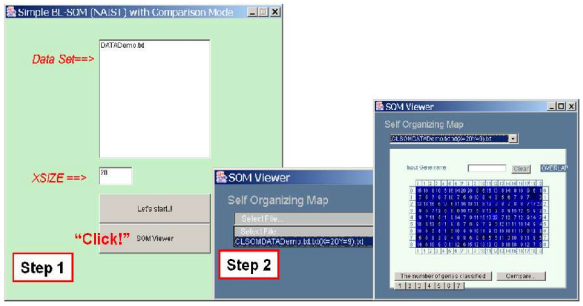

**Scheme 1**




**Panel 2**

**(B) Visualization of Self-organizing map**

After classification of objects to self-organizing map, we can visualize classification results. This process can be got done by clicking 'SOM Viewer' button (**Step 1** in **Scheme 2**) and then by selecting classification file (**Step 2**); the self-organizing map is displayed in a separate window. The number of samples included in a lattice point is shown on the corresponding lattice point in the map. The Profiles of measurements for samples included in a lattice point can be observed by just clicking the lattice point and user can search the profile of an object by entering its name in 'Input gene name' box.

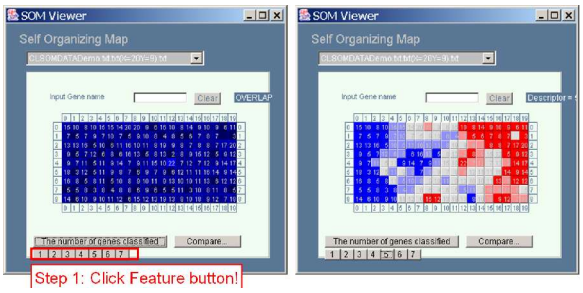



**Scheme 2**

**(B1) Feature map for individual experiment**

When a user wants to know high and low levels corresponding to individual experiments, he/she should click experiment ID (**Step 1** in **Scheme 3**). In this example the 5$^{th}$ experiment has been selected. Pink and Red lattices include only objects with measurements larger than the average for the selected experiment. Sky blue and Blue lattices include only objects with measurements smaller than the average for the

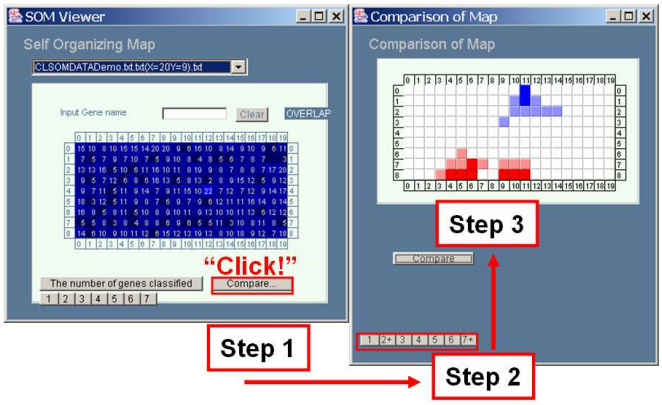
selected experiment. A red lattice indicates that at least on of the objects belonging to it is with a measurement value larger than the average plus the standard deviation and a blue lattice indicates that at least on of the objects belonging to it is with a measurement value smaller than the average minus the standard deviation.




**Scheme 3**

**(B2) Comparison between two Feature maps**

To compare two Feature maps, click the 'Compare' button (**Step 1** in **Scheme 4**) and then the 'Comparison of Map' window appears. In this window, select two experiments (**Step 2**), and click the 'Compare' button. The colors of the lattices in the comparison map reflect how the measurement values changed in the feature map for the experiment with larger ID compared to the feature map for the experiment with smaller ID. The color rules for the comparison map are presented in Table 1. Lattices are made white in all other possible cases that are not mentioned in Table 1. In Scheme 4, experiments with IDs 2 and 7 are compared.




**Scheme 4**

**Table 1** Color rule in comparison map

| Feature Map for Exp. with Small ID | Feature Map for Exp with Large ID | Comparison Map |
|---|---|---|
| red | blue | blue |
| red | sky blue | sky blue |
| pink | blue | sky blue |
| pink | sky blue | sky blue |
| sky blue | red | pink |
| sky blue | pink | pink |
| blue | pink | pink |
| blue | red | red |

## 2.Output files

Four output files ‘WTSPCA’, ‘WTSSOM’, ‘CLSOM’, and ‘Convergence.txt’ are constructed by the present software, which has been designed to analyze multidimensional data based on BL-SOM.

### 2.1 WTSPCA and WTSSOM files

WTSPCA contains initial weight vectors generated by PCA and WTSSOM contains the final weight vectors generated by the learning process of SOM. The format of these files is shown in Table 2. The first

line XSIZE=20 and the second line YSIZE=9 represent the number of lattice points in the first and second axes. In the following lines, the first and the second columns correspond to the coordinates of the lattice points, and the multidimensional weight values are written in from the third to the last columns.

**Table 2** Format of WTSPCA and WTSSOM files

```
>XSIZE=20
>YSIZE=9
0    0    -0.2017    -0.1516    -0.1457    -0.11637    -0.1292    -0.1307    -0.3151
0    1    -0.1957    -0.1588    -0.1573    -0.12823    -0.1359    -0.1366    -0.2731
……..
……..
```

**2.2 Convergence.txt file**

The summation of distances between input vectors and the corresponding nearest weights, Q(r) (see Step 2 in section 3) for each cycle of the learning process is accumulated in Convergence.txt. The first column corresponds to cycle No. and the second column corresponds to Q(r).

**Table 3** Convergence.txt

```
1        2.98766299460335
2        2.8772749098201706
3        2.9505205831352668
4        2.879867705953376
5        2.5773454311413686
6        2.711740207571155
7        2.6944859412563607
……
……
……
```

**2.3 CLSOM file**

CLSOM file accumulates information on classification of objects in the self-organizing map. The first line XSIZE=20 and second line YSIZE=9 represent the number of lattice points in the first and second axes. In

the following lines, the first column corresponds to object name, the second and third columns correspond to the coordinate of the lattice point to which the object is classified. The forth column represents p-value that the object is randomly classified to this lattice point based on numerical analysis of random vectors. The profile of the object is shown in from the fifth to the last columns.

Table 4

```
>XSIZE=20
>YSIZE=9
yaaA    17    7    0.0014    0.1128    0.0188    0.0438    0.0407    0.0610    0.05956    0.1278
recF    17    8    0.0045    0.0336    -0.0134   0.02178   0.0507    0.0487    0.04210    0.1293
……
……
```

**3. Algorithm**

**Step 1: Initialization of weight vectors by PCA**

The $s_{th}$ input vector of dimension M is represented as follows:

$$\mathbf{x}_s = (x_{s1}, x_{s2}, ..., x_{st}, ..., x_{sM})$$

where $x_{st}$ represents the measurement of the $t$th descriptor. So a data set can be represented by the following matrix.

$$\mathbf{X} = \begin{pmatrix} x_{11} & x_{12} & \cdots & x_{1t} & \cdots & x_{1M} \\ x_{21} & x_{22} & \cdots & x_{2t} & \cdots & x_{2M} \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ x_{s1} & x_{s2} & \cdots & x_{st} & \cdots & x_{sM} \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ x_{N1} & x_{N2} & \cdots & x_{Nt} & \cdots & x_{NM} \end{pmatrix}$$

Here, N means the number of input vectors.

The initial weight vectors are determined based on the first and second principal components of the M-dimensional space by PCA. Weights in the first dimension (I) are arranged into lattices corresponding to a width that is five times the standard deviation ($5\sigma_1$) of the first principal component. The second dimension (J) is defined by the nearest integer greater than ($\sigma_2/\sigma_1$) x I. The total number of weights in the first dimension I is set by a user. The weight vector on the *ij*th lattice ($\mathbf{w}_{ij}$) is represented as follows:

$$\mathbf{w}_{ij} = \mathbf{x}_{av} + \frac{5\sigma_1}{I}\left[\mathbf{b}_1\left(i - \frac{I}{2}\right) + \mathbf{b}_2\left(j - \frac{J}{2}\right)\right]$$

Here $\mathbf{x}_{av}$ is the average vector for oligonucleotide frequencies of all input vectors, and $\mathbf{b}_1$ and $\mathbf{b}_2$ are eigenvectors for the first and second principal components.

**Step 2: Adaptation of weight vectors to the input vectors.**

The minimum Euclidean distance of the input vector $\mathbf{x}_k$ with respect to all weight vectors $\mathbf{w}_{ij}$ (i = 1,2,…, I; j = 1,2,…,J) is denoted by $\mathbf{w}_{i^*j^*}$. The input vector $\mathbf{x}_k$ is classified into set $S_{ij}$ for the lattice points (i, j) satisfying $i'-\beta(r) \le i \le i'+\beta(r)$ and $j'-\beta(r) \le j \le j'+\beta(r)$ . After classification of all input vectors to the lattice pointes (i, j), weight vectors are updated by

$$\mathbf{w}_{ij}^{(new)} = \mathbf{w}_{ij} + \alpha(r)\left(\frac{\sum_{\mathbf{x}_k \in S_{ij}} \mathbf{x}_k}{N_{ij}} - \mathbf{w}_{ij}\right)$$

The two parameters $\alpha(r)$ and $\beta(r)$ are learning coefficients for the *r*th cycle, and $N_{ij}$ is the number of components of $S_{ij}$. $\alpha(r)$ and $\beta(r)$ are calculated as follows:

$\alpha(r) = \max\ \{0.01, \alpha(1)(1 - r/T)\}$

$$\beta(r) = \max\ \{0, \beta(1) - r\}$$

Here, $\alpha(1)$ and $\beta(1)$ are the initial values for the T-cycle of the learning process. The learning process is monitored by the total distance between $\mathbf{x}_k$ and the nearest weight vector $\mathbf{w}_{i'j'}$, represented as

$$Q(r) = \sum_{k=1}^{N} \left\{ \left\| \mathbf{x}_k - \mathbf{w}_{i'j'} \right\|^2 \right\}$$

Here N is the total number of sequences analyzed.

**Step 3: Classification of input vectors to weight vectors**

Each of the input vectors is classified into lattice point whose distance is the minimum from the input vector.