Prediction of Aftershocks Distribution Using Artificial Neural Networks and Its Application on the May 12, 2008 Sichuan Earthquake

R. Madahizadeh[1] and M. Allamehzadeh[2*]

1. MSc Graduate, Department of Seismology, International Institute of Earthquake Engineering and Seismology (IIEES), Tehran, I.R. Iran
2. Assistant Professor, Seismology Research Center, International Institute of Earthquake Engineering and Seismology (IIEES), Tehran, I.R. Iran,

* Corresponding Author; email: mallam@iiees.ac.ir

ABSTRACT

In this paper an approach is presented to predict the concentration and the trend of aftershocks of May 12 2008 Chengdu, Sichuan, China earthquake. The method is based on inputting first aftershocks to Kohonen artificial neural network. Artificial neural networks, which are inspired from human brain, consist of several artificial neurons which are connected with some weight vectors to each other. Artificial neural networks are able to classify a large volume of input data (i.e. earthquake catalogue) simultaneously and in parallel, and can recognize seismic patterns very well. Kohonen neural networks consist of several neurons that affect mutually on each other to display important statistical characteristics of the input space (i.e. first aftershocks). Combination of associative and competitive learning rules results in formation of Kohonen's self-organizing feature map (SOFM) algorithm. SOFM algorithm has converged; the feature map computed by the SOFM algorithm indicates the concentration and the trend of aftershocks precisely. Kohonen artificial neural networks have become powerful intelligent tools in recent years, used widely in pattern recognition and data clustering.

Keywords:
Prediction; Aftershocks; Kohonen neural network; Self-organizing feature map; Clustering

1. Introduction

Earthquake aftershocks are most common immediately after the main shock. In the most popular approach, aftershocks are collected by counting all events within a predetermined space - time window following a main event [1].

The average number of occurrences of aftershocks decreases rapidly as time passes, which depends on the geological conditions and the magnitude of main shock. Earthquakes of all magnitudes have aftershocks which decay according to the Omori law [2-3],

$$v(t) \sim \frac{k}{t+c} \quad \text{for} \quad t < t_{cutoff} \qquad (1)$$

Where $v(t)$ is the rate of the occurrence of aftershocks and k, c are constant in time, but depend on the magnitude of the earthquake. Omori law persists up to a time t_{cutoff} that also depends on magnitude of the earthquake. The aftershocks distribution shows the rupture of the main shock, which is an important issue for estimating the risk of future disastrous earthquakes.

Pattern recognition of aftershocks distribution and aftershocks clustering is an important and complicated issue in seismology. It is difficult because of ununiform structures in the interested region and stochastic nature of seismic signals.

Recent developments of neural classifiers indicate that they are useful in solving many difficult problems in seismology such as discrimination analysis [4], seismic pattern classification [5], seismic phase identification [6], and earthquake prediction [7].

In the present paper by application of Kohonen's self-organizing feature map (*SOFM*) algorithm, the possible prediction of the location of aftershocks distribution of May 12, 2008 Chengdu, Sichuan, China earthquake (30.99*N*, 103.36*E*, *Mw* = 7.9, 06:28:00 *UTC*) will be described, see Figure (1). The Kohonen's self-organizing feature map (*SOFM*) algorithm has been tested on the 1997 Birjand-Ghaen, Iran and 1999 Izmit, Turkey earthquakes [8].

2. Seismotectonic Setting

The high seismicity of central and eastern Asia results from the northward collisional convergence (at about 40*mm/yr*) of the India tectonic plate against the Eurasian plate, see Figure (1). This active collision, which began about 50 million years ago, is the cause of frequent large earthquakes between India and Tibet and throughout Tibet and the surrounding areas [9-10]. The convergence has uplifted the Asian highlands and the Tibetan plateau to an average elevation of over 4876.8 meters (16,000 feet), which is the highest and largest plateau on Earth with hundreds of kilometers of displacement of crustal blocks to the east and southeast in the direction of China [11].

As India kept on moving northward and intruding into Asia by as much as 1,200*kms*, the regions north of the Himalayas moved laterally to the east and southeast along large strike slip faults such as the Altyn Tagh, pushing into central China [12], see Figure (1).

3. Seismicity of the Sichuan Province

The present seismicity of the Sichuan Province is caused by the slower-moving lateral crustal displacements which converge from the margins of the high Tibetan plateau towards the Sichuan basin and southeastern China. This deformation from the plateau results in additional extrusion of crustal materials which are pushed under the weaker sedimentary layers of the Sichuan basin and of the entire southeastern region of China.

These crustal displacements along this seismic belt are responsible for the large destructive earthquakes in the more densely populated areas of southwestern China. Thus, Sichuan is among the most seismically active regions, where frequent strong ($M \geq 6.5$) earthquakes can occur.

Tectonic stresses from the strong convergences have resulted in the formation of other major faults in Sichuan province such as the Longmen Shan fault zone, where the May 12, 2008 earthquake occurred. Several destructive earthquakes have occurred in the past and in sequence along the Longmen Shan fault zone.

4. Artificial Neural Networks

Artificial neural network models have been studied for many years in the hope of achieving human-like performance in the fields of speech and image recognition [13].

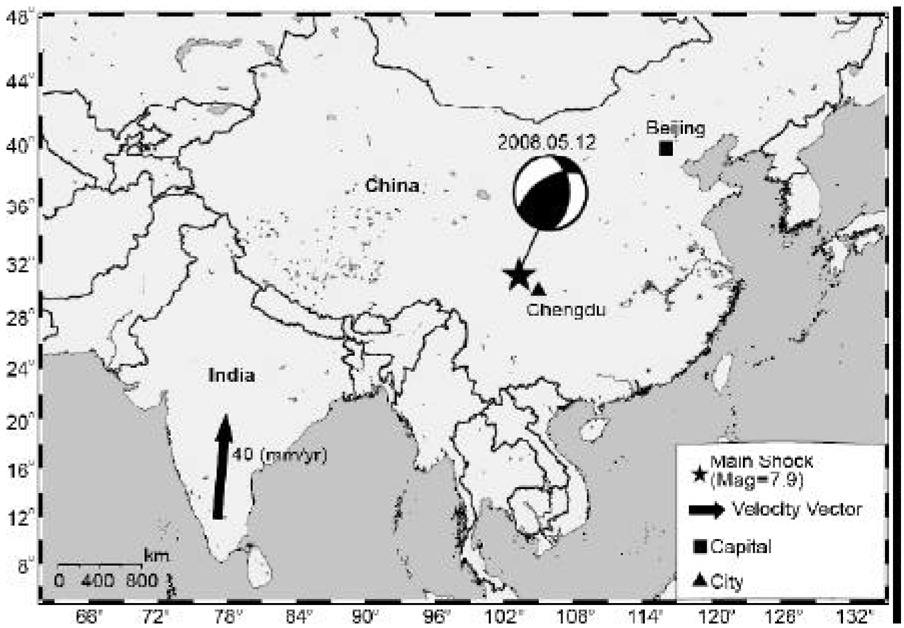

Figure 1. Map showing the epicenter (star) and focal mechanism of the May 12, 2008 Chengdu, Sichuan, China earthquake. The arrow shows the northerly motion of India, resulting easterly motion of Tibet plateau.

Artificial neural networks, which are inspired from human brain, consist of several artificial neurons which are connected with some weight vectors to each other. Artificial neural networks are able to classify a large volume of input data (i.e. earthquake catalogue) simultaneously and in parallel, and can recognize seismic patterns very well. Artificial neural networks have two useful properties (i.e. abbreviation and generalization) for learning earthquake catalogues.

A neuron is an information-processing unit that is fundamental to the operation of a neural network. The block diagram of Figure (2) shows the model of a neuron, which forms the basis for designing artificial neural networks. Here three basic elements of the neural model are identified [14]:

a) A set of synapses or connecting links, (w_{nj}), each of which is characterized by a weight or strength of its own. Specifically, a vector x_n at the input of synapse n connected to neuron j is multiplied by the synaptic weight w_{nj}. The first subscript refers to the input end of the synapse and the second subscript refers to the neuron in question to which the weight refers.
b) An adder (Σ) for summing the input vectors, weighted by the respective synapses of the neuron, the operations described here constitute a linear combiner.
c) An activation function (φ) for limiting the magnitude of the output of a neuron.

In mathematical terms, we may describe a neuron by writing the following:

Input Vector: $x = (x_1, x_2, ..., x_n)$

Weight Vector: $w = (w_{1j}, w_{2j}, ..., w_{nj})$ (2)

Net Input: $net_j = x_1 w_{1j} + x_2 w_2 + ... + x_n w_n$

Output Vector (activation): $o_j = \varphi(net_j + \theta_j)$

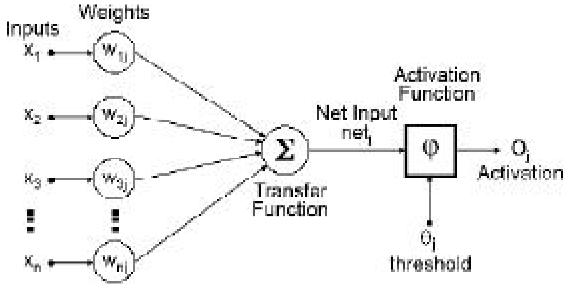

Figure 2. The block diagram of the mathematical model of an artificial neuron.

4.1. Properties of Neural Networks

The use of neural networks offers the following useful properties and capabilities.

1) **Generalization:** refers to the neural network producing reasonable outputs for inputs not encountered during training (learning).
2) **Learning:** Learning is a process by which the free parameters of a neural network are adapted through a process of stimulation by the environment in which the network is embedded. This definition of the learning process implies the following sequence of events:
   a) The neural network is stimulated by an environment.
   b) The neural network undergoes changes in its free parameters as a result of this stimulation.
   c) The neural network responds in a new way to the environment because of the changes that have occurred in its internal structure.

The procedure used to perform the learning process is called a learning algorithm, the function of which is to modify the synaptic weights of the network in an orderly fashion to attain a desired design.

4.2. Types of Learning

The type of learning is determined by the manner in which the parameter changes take place [14].

- **Learning with a Teacher**: In conceptual terms, we may think of the teacher as having the knowledge of the environment, with that knowledge being represented by a set of input -output examples. The environment is, however, unknown to the neural network of interest. Suppose now that the teacher and the neural network are both exposed to a training vector drawn from the environment. By virtue of built-in knowledge, the teacher is able to provide the neural network with a desired response for that training vector. Indeed, the desired response represents the optimum action to be performed by the neural network. The network parameters are adjusted under the combined influence of the training vector and the error signal. The error signal is defined as the difference between the desired response and the actual response of the network. This adjustment is carried out iteratively in step by step fashion with the aim of eventually making the neural network emulate the teacher.

- **Learning without a Teacher:** In the paradigm known as learning without a teacher, as the name implies, there is no teacher to oversee the learning process. That is to say, there are no labeled examples of the function to be learned by the network. In unsupervised or self-organized learning, once the network has become tuned to the statistical regularities of the input data, it develops the ability to form internal representations for encoding features of the input and thereby to create new classes automatically. To perform an unsupervised learning we may use a competitive learning rule. For example, a neural network that consists of two layers - an input layer and a competitive layer. The input layer receives the variable data. The competitive layer consists of neurons that compete with each other (in accordance with a learning rule) for the opportunity to respond to features contained in the input data. In its simplest form, the network operates in accordance with a winner-takes-all strategy. In such a strategy the neuron with the greatest total input wins the competition and turns on, all the other neurons then switch off.

5. Kohonen Artificial Neural Network

Combination of associative and competitive learning rules results in formation of self-organizing artificial neural networks. Self-organizing feature map (*SOFM*) neural network consists of several neurons that are placed at the nodes of a lattice that is usually one- or two-dimensional. These neurons effect mutually on each other to satisfy the principal goal of the self-organizing feature map which is to transform input continuous space into a one- or two-dimensional discrete map. Once *SOFM* algorithm has converged, the feature map computed by the *SOFM* algorithm displays important statistical characteristics of the input space.

The theory of self-organizing Feature Maps (*SOFM*) is fairly well understood and a number of applications of *SOFM* have also been developed [15-16]. In the neural network community, the term self-organizing (unsupervised-learning process) refers to the ability of some networks to learn without being given the correct answer for an input pattern. The *SOFM* algorithm is an unsupervised-learning process, since there is no desired output given during learning. The *SOFM* defines a mapping from the input data space onto an output layer. When the algorithm has converged, prototype vectors corresponding to nearby points on the feature map have nearby locations in input space.

Self-organizing Feature Maps are competitive neural networks in which neurons are organized in a 2-dimensional lattice (grid) representing the feature space and its algorithm creates a vector quantizer by adjusting weights from common input nodes to *m* output nodes arranged in a two dimensional grid as shown in Figure (3).

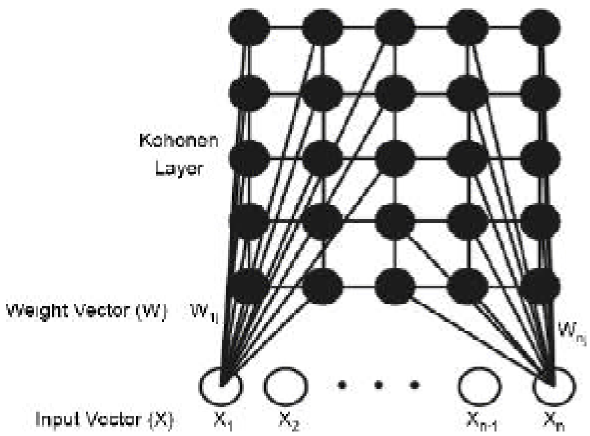

Figure 3. Self-organizing feature maps are competitive neural networks in which neurons are organized in a 2-dimensional lattice (Kohonen Layer). 'X' is input vector, 'W' is weight vector of neurons.

Output nodes are extensively interconnected with many local connections. The spatial location of an output neuron in the topologic map corresponds to a particular domain or feature of the input data. In a self-organizing map, the neurons are placed at the nodes of a lattice that is usually one-or two-dimensional. Higher-dimensional maps are also possible but not as common. The neurons become selectivity tuned to various input patterns (stimuli) or classes of input patterns in the course of a competitive learning process. The locations of the neurons so tuned (i.e., the winning neurons) become ordered with respect to each other in such a way that a meaningful coordinate system for different input features is created over the lattice.

A self-organizing map is therefore characterized by the formation of a topologic map of the input patterns in which the spatial locations (i.e., coordinates) of the neurons in the lattice are indicative of intrinsic statistical features contained in the input patterns, hence the name self-organizing map. The principal goal of the self- organizing map (*SOM*) is to transform an incoming vector pattern of arbitrary dimension into a one- or two-dimensional discrete map and to perform this transformation adaptively in a topologically ordered fashion.

5.1. Algorithm of Kohonen Artificial Neural Network

The algorithm responsible for the formation of the self-organizing map proceeds first by initializing the synaptic weights in the network [14]. This can be done by assigning them small values picked from a random number generator. In doing so, no prior order is imposed on the feature map. Once the network has been properly initialized, there are three essential processes involved in the formation of the self-organizing map, as summarized below:

a) **Competition:** For each input pattern, the neurons in the network compute their respective values of a discriminant function. The discriminant function provides the basis for competition among the neurons. The particular neuron with the largest value of discriminant function is declared winner of the competition. In simplest form of competition, the network operates in accordance with a winner- takes- all strategy. In such a strategy the neuron with the largest total input wins the competition and turns on. All the other neurons then will switch off.

b) **Cooperation:** The winning neuron determines the spatial location of a topological neighborhood of excited neurons, thereby providing the basis for cooperation among such neighboring neurons. Topological neighborhood function is symmetric about winning neuron and attains its maximum value at the winning neuron. The amplitude of the neighborhood function decreases monotonically with increasing lateral distance, see Figure (4).

c) **Synaptic Adaptation:** This last mechanism enables the exited neurons to increase their individual values of the discriminant function in relation to the input pattern through suitable adjustments applied to their synaptic weights. The adjustments made are such that the response of the winning neuron to the subsequent application of a similar input pattern is enhanced.

The algorithm is summarized as follows [14]:

1) **Initialization:** Choose random values for the initial weight vectors $w_j(0)$. The only restriction here is that the $w_j(0)$ be different for j =1, 2,…, l, where l is the number of neurons in the lattice. It may be desirable to keep the magnitude of the weights small.

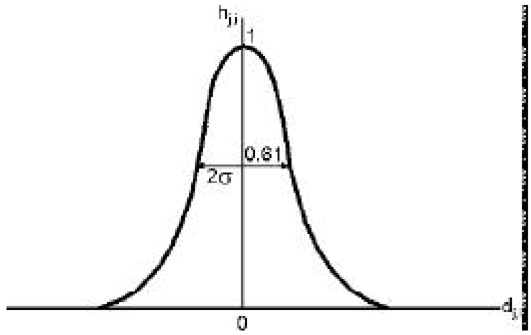

Figure 4. Gaussian topological neighborhood function ($h_{j,i}$). The winning neuron determines the center location of neighborhood function. Topological neighborhood function is symmetric about winning neuron and attains its maximum value at the winning neuron ($d_{j,i}$=0). The amplitude of the neighborhood function decreases monotonically with increasing lateral distance ($d_{j,i}$). The parameter σ is the "effective width" of the topological neighborhood and measures the degree to which excited neurons in the vicinity of the winning neuron participate in the learning process.

2) **Sampling:** Draw a sample x from the input space with a certain probability, the vector x represents the activation pattern that is applied to the lattice.

3) **Similarity Matching:** Find the best-matching (winning) neuron $i(x)$ at time step n by using the minimum- distance Euclidean criterion.

$$i(x) = arg\ \min_j \left\| x - w_j \right\|, \quad j = 1, 2, ..., l \tag{3}$$

4) **Updating:** Adjust the synaptic weight vectors of all neurons by using the update formula, see Eq. (4). Where $\eta(n)$ is the learning-rate parameter, and $h_{j,i(x)}(n)$ is the neighborhood function centered around the winning neuron $i(x)$, both $\eta(n)$ and $h_{j,i(x)}(n)$ are varied dynamically during learning for best results.

$$w_j(n+1) = w_j(n) + \eta(n)\, h_{j,i(x)}(n)\,(x - w_j(n)) \tag{4}$$

5) **Continuation:** Continue with step 2 until no noticeable changes in the feature map are observed.

Once the *SOM* algorithm has converged, the feature map computed by the algorithm displays important statistical characteristics of the input space.

5.2. *Feature Map of Kohonen Artificial Neural Network*

A nonlinear transformation called a feature map, is presented here which maps the input space onto the output space. The feature map has some important properties [14].

a) **Approximation of the Input Space:** The feature map, represented by the set of synaptic weight vectors $\{w_i\}$ in the output space, provides a good approximation to the input space. The feature map computed by the *SOM* algorithm stores a large set of input vectors by defining a smaller set of prototypes $\{w_j\}$, in order to provide a good approximation to the original input space.

b) **Topological Ordering:** The feature map computed by the *SOM* algorithm is topologically ordered in the sense that the spatial location of a neuron in the lattice corresponds to a particular domain or feature of input patterns. The topological ordering property is a direct consequence of the update formula, Eq. (4), that forces the synaptic weight vector w_i of the winning neuron to move toward the input vector x.

c) **Density Matching:** Regions in the input space from which sample vectors x are drawn with a high probability of occurrence, are mapped onto larger domains of the output space, and therefore with better resolution than regions in input space from which sample vectors x are drawn with a low probability of occurrence.

d) **Feature Selection:** Given data from an input space with a nonlinear distribution, the self-organizing map is able to select a set of best features for approximating the underlying distribution.

6. Methodology

According to Table (1), Figure (5) shows the epicenter map of aftershocks of the May 12, 2008 Sichuan earthquake. To predict aftershocks distribution of this earthquake, a two dimensional (9*9) Kohonen neural network was used (*MATLAB* 6.5 Software). The total number of weights in the neural network is 81.

The input vectors of Kohonen neural network are latitude and longitude of aftershocks which occur during two days after the main shock, see Table (1). After enough input vectors have been presented, weights will specify clusters or vector centers that sample the input space such as probability density function of the input vectors. In addition, the weights are organized such that topologically close nodes are sensitive to inputs which are physically similar.

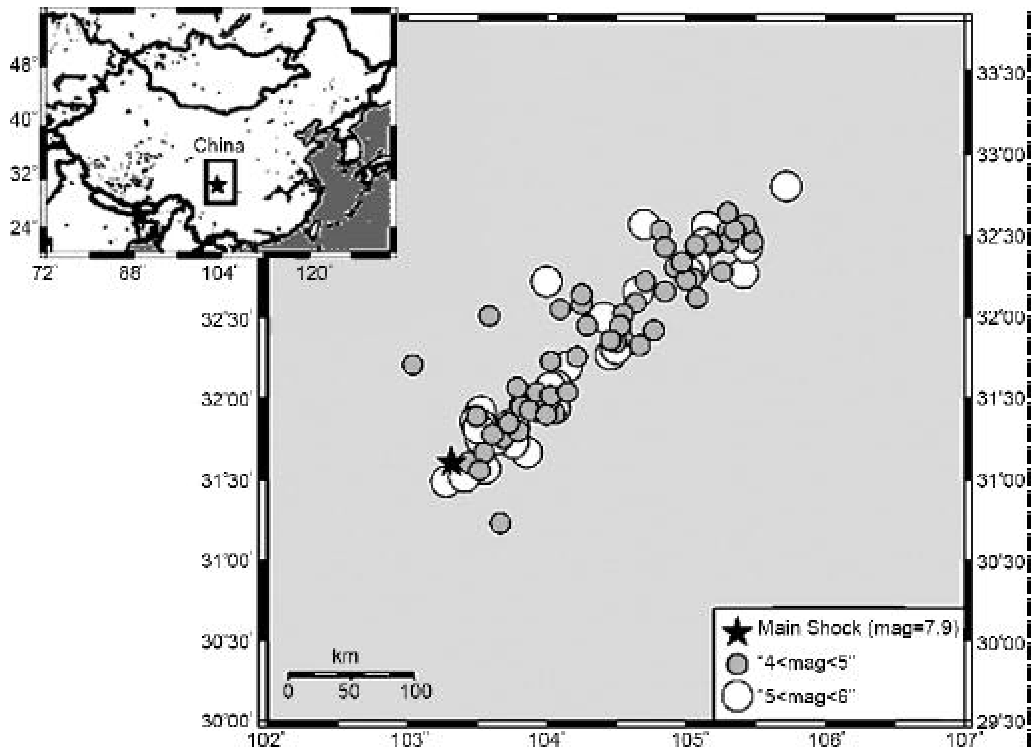

Figure 5. Epicenter map of the main shock and aftershocks of the May 12, 2008 Chengdu, Sichuan, China earthquake, see Table (1).

Table 1. Aftershocks data of May 12, 2008 Chengdu, Sichuan, China earthquake (www.emsc-csem.org).

Date	Time (UTC)	Lat. (N)	Lon. (E)	Mag. (mb)	Region
2008/05/12	06:43:13,7	31.32	103.77	5.8	Eastern Sichuan, China
2008/05/12	06:54:18,4	31.17	103.86	5.6	Eastern Sichuan, China
2008/05/12	07:34:42,2	31.27	103.64	5.5	Eastern Sichuan, China
2008/05/12	08:08:24,8	31.86	104.46	4.7	Eastern Sichuan, China
2008/05/12	08:11:03,3	31.25	103.53	5.1	Eastern Sichuan, China
2008/05/12	08:21:40,4	31.57	104.03	5.1	Eastern Sichuan, China
2008/05/12	08:26:12,2	31.40	104.00	4.9	Eastern Sichuan, China
2008/05/12	08:47:25,3	32.28	105.02	5.0	Sichuan/Gansu Border REG, China
2008/05/12	08:54:16,2	32.12	105.08	4.8	Sichuan/Gansu Border REG, China
2008/05/12	09:07:04,7	31.23	103.76	5.2	Eastern Sichuan, China
2008/05/12	09:23:34,5	32.16	104.85	4.9	Sichuan/Gansu Border REG, China
2008/05/12	09:42:25,1	31.58	104.07	5.5	Eastern Sichuan, China
2008/05/12	10:16:23,3	32.44	105.07	4.6	Sichuan/Gansu Border REG, China
2008/05/12	10:23:40,3	31.02	103.41	5.2	Eastern Sichuan, China
2008/05/12	10:43:15,3	31.52	104.03	4.5	Eastern Sichuan, China
2008/05/12	11:11:02,7	31.24	103.75	5.7	Eastern Sichuan, China
2008/05/12	11:28:40,8	31.54	103.93	4.7	Eastern Sichuan, China
2008/05/12	11:33:21,3	32.23	105.00	4.8	Sichuan/Gansu Border REG, China
2008/05/12	11:41:13,5	32.31	104.92	4.9	Sichuan/Gansu Border REG, China
2008/05/12	12:04:36,2	32.57	105.43	4.9	Sichuan/Gansu Border REG, China
2008/05/12	12:15:40,8	31.92	104.77	4.9	Eastern Sichuan, China
2008/05/12	13:40:57,2	31.06	103.52	4.9	Eastern Sichuan, China
2008/05/12	14:10:27,6	31.28	103.61	4.8	Eastern Sichuan, China
2008/05/12	14:15:25,9	32.16	104.66	5.1	Sichuan/Gansu Border REG, China
2008/05/12	14:46:09,3	32.80	105.72	5.2	Sichuan/Gansu Border REG, China
2008/05/12	15:05:31,4	31.28	103.75	5.2	Eastern Sichuan, China
2008/05/12	15:28:53,3	31.07	103.56	5.1	Eastern Sichuan, China
2008/05/12	15:54:27,3	32.45	105.18	4.5	Sichuan/Gansu Border REG, China
2008/05/12	16:28:54,9	31.35	103.73	4.8	Eastern Sichuan, China
2008/05/12	17:03:12,0	31.17	103.55	4.9	Eastern Sichuan, China
2008/05/12	17:52:23,9	31.95	104.53	4.8	Eastern Sichuan, China
2008/05/12	17:54:33,7	31.28	103.70	5.1	Eastern Sichuan, China
2008/05/12	18:26:19,7	31.73	104.03	4.6	Eastern Sichuan, China
2008/05/12	18:55:25,7	32.43	104.85	4.6	Sichuan/Gansu Border REG, China
2008/05/12	19:53:23,0	31.57	103.79	4.8	Eastern Sichuan, China
2008/05/12	20:08:48,6	31.49	103.89	5.5	Eastern Sichuan, China
2008/05/12	20:45:30,6	31.81	104.50	5.3	Eastern Sichuan, China
2008/05/12	20:51:25,5	32.34	104.97	4.9	Sichuan/Gansu Border REG, China
2008/05/12	21:08:14,5	31.39	103.50	4.6	Eastern Sichuan, China
2008/05/12	23:38:12,5	32.22	104.71	4.5	Sichuan/Gansu Border REG, China
2008/05/12	23:46:23,4	31.32	103.51	5.4	Eastern Sichuan, China
2008/05/12	23:54:50,6	31.34	103.53	5.3	Eastern Sichuan, China
2008/05/13	00:22:21,3	31.54	104.15	4.7	Eastern Sichuan, China
2008/05/13	02:15:13,9	31.83	104.67	4.9	Eastern Sichuan, China
2008/05/13	03:00:38,7	31.25	103.63	5.1	Eastern Sichuan, China
2008/05/13	05:25:46,5	32.53	105.35	4.8	Sichuan/Gansu Border REG, China
2008/05/13	05:36:27,7	32.46	105.48	4.6	Sichuan/Gansu Border REG, China
2008/05/13	07:07:09,0	30.99	103.28	5.9	Eastern Sichuan, China
2008/05/13	07:19:17,7	32.43	105.44	5.0	Sichuan/Gansu Border REG, China
2008/05/13	07:53:02,5	32.33	105.07	5.1	Sichuan/Gansu Border REG, China
2008/05/13	08:20:51,7	31.45	103.95	5.0	Eastern Sichuan, China
2008/05/13	10:16:07,3	31.77	104.46	5.0	Eastern Sichuan, China
2008/05/13	12:51:37,9	32.27	105.41	5.0	Sichuan/Gansu Border REG, China
2008/05/13	13:13:05,5	32.64	105.30	4.8	Sichuan/Gansu Border REG, China
2008/05/13	13:31:37,2	32.46	105.31	4.6	Sichuan/Gansu Border REG, China
2008/05/13	15:10:32,9	32.53	105.30	4.5	Sichuan/Gansu Border REG, China
2008/05/13	16:23:51,4	31.85	104.50	4.8	Eastern Sichuan, China
2008/05/13	19:51:56,1	31.11	103.44	4.7	Eastern Sichuan, China
2008/05/14	00:08:19,6	32.01	103.59	4.5	Western Sichuan, China
2008/05/14	02:54:36,8	31.36	103.49	5.7	Eastern Sichuan, China

Table 1. Continued...

Date	Time (UTC)	Lat. (N)	Lon. (E)	Mag. (mb)	Region
2008/05/14	05:54:58,9	32.22	104.00	5.2	Sichuan/Gansu Border REG, China
2008/05/14	09:26:44,5	31.44	104.06	5.2	Eastern Sichuan, China
2008/05/14	10:00:33,7	32.45	105.13	5.0	Sichuan/Gansu Border REG, China
2008/05/14	10:44:37,6	32.24	105.05	4.7	Sichuan/Gansu Border REG, China
2008/05/14	12:27:58,7	31.95	104.29	4.5	Eastern Sichuan, China
2008/05/14	17:17:22,3	31.50	103.99	4.5	Eastern Sichuan, China
2008/05/14	17:33:24,3	31.30	103.80	4.9	Eastern Sichuan, China
2008/05/14	21:01:07,3	31.70	104.15	5.2	Eastern Sichuan, China
2008/05/14	22:10:15,9	31.43	103.88	4.8	Eastern Sichuan, China
2008/05/15	00:09:35,1	32.08	104.25	4.7	Sichuan/Gansu Border REG, China
2008/05/15	04:27:36,1	31.49	104.02	4.6	Eastern Sichuan, China
2008/05/15	05:27:47,9	32.02	104.41	5.0	Sichuan/Gansu Border REG, China
2008/05/15	18:05:26,3	32.14	104.25	4.8	Sichuan/Gansu Border REG, China
2008/05/15	21:55:49,4	32.57	104.70	5.1	Sichuan/Gansu Border REG, China
2008/05/15	22:10:35,4	31.41	104.06	4.9	Eastern Sichuan, China
2008/05/15	22:34:35,5	31.88	104.54	4.4	Eastern Sichuan, China
2008/05/16	03:34:27,5	31.53	104.06	5.1	Eastern Sichuan, China
2008/05/16	05:25:49,0	31.42	103.53	5.5	Eastern Sichuan, China
2008/05/16	06:34:41,9	32.53	104.82	4.6	Sichuan/Gansu Border REG, China
2008/05/16	16:14:45,0	31.26	103.69	4.9	Eastern Sichuan, China
2008/05/16	20:16:52,6	31.34	103.74	5.0	Eastern Sichuan, China
2008/05/16	22:33:08,5	32.28	105.26	4.7	Sichuan/Gansu Border REG, China
2008/05/17	07:38:43,9	32.02	104.55	4.4	Sichuan/Gansu Border REG, China
2008/05/17	13:32:13,6	32.09	104.64	4.8	Sichuan/Gansu Border REG, China
2008/05/17	17:08:25,7	32.29	105.05	5.9	Sichuan/Gansu Border REG, China
2008/05/17	20:26:08,5	31.45	103.82	4.6	Eastern Sichuan, China
2008/05/18	00:45:58,1	32.05	104.10	4.4	Sichuan/Gansu Border REG, China
2008/05/18	03:51:41,8	31.73	103.67	4.6	Eastern Sichuan, China
2008/05/18	09:25:13,2	31.46	103.81	4.5	Eastern Sichuan, China
2008/05/18	12:37:07,2	31.71	103.04	4.5	Eastern Sichuan, China
2008/05/19	04:09:02,0	32.56	105.15	5.0	Sichuan/Gansu Border REG, China
2008/05/19	06:06:54,9	32.50	105.42	5.1	Sichuan/Gansu Border REG, China
2008/05/19	17:52:34,2	32.29	105.02	5.3	Sichuan/Gansu Border REG, China
2008/05/20	00:57:36,4	31.76	104.22	4.5	Eastern Sichuan, China
2008/05/20	03:42:30,9	32.45	105.20	4.4	Sichuan/Gansu Border REG, China

7. Results

Figure (6) presents the values of the synaptic weight vectors, plotted as stars in the input space. These stars (synaptic weight vectors) are the synthetic epicenter of future aftershocks of Chengdu, Sichuan, earthquake which are predicted by the two dimensional (9*9) Kohonen neural network. The epicenters of real aftershocks of this earthquake are plotted as pluses in Figure (6). As shown in Figure (6) Kohonen neural network has successfully predicted concentration of the aftershock zone and trend of future aftershocks of Sichuan earthquake.

8. Discussion

Kohonen artificial neural networks are competitive neural networks in which neurons are organized in a two-dimensional lattice (grid) representing the feature space and its algorithm creates a vector quantizer by adjusting weights from common input nodes to output nodes arranged in a two dimensional grid. The feature map computed by the *SOFM* algorithm stores a large set of input vectors by defining a smaller set of prototypes $\{w_j\}$ (i.e. stars in Figure (6)), so as to provide a good approximation to the original input space. Therefore, Kohonen neural network can be used to predict concentration of aftershock zone and the trend of future aftershocks.

Firstly, since patterns with a high probability of occurrence are mapped on to a larger area of the feature map, Kohonen's *SOFM* algorithm reflects statistical variations in the aftershocks region.

Secondly, higher density patterns have better resolution than patterns that have low probability of occurrence; therefore, the concentration of after-

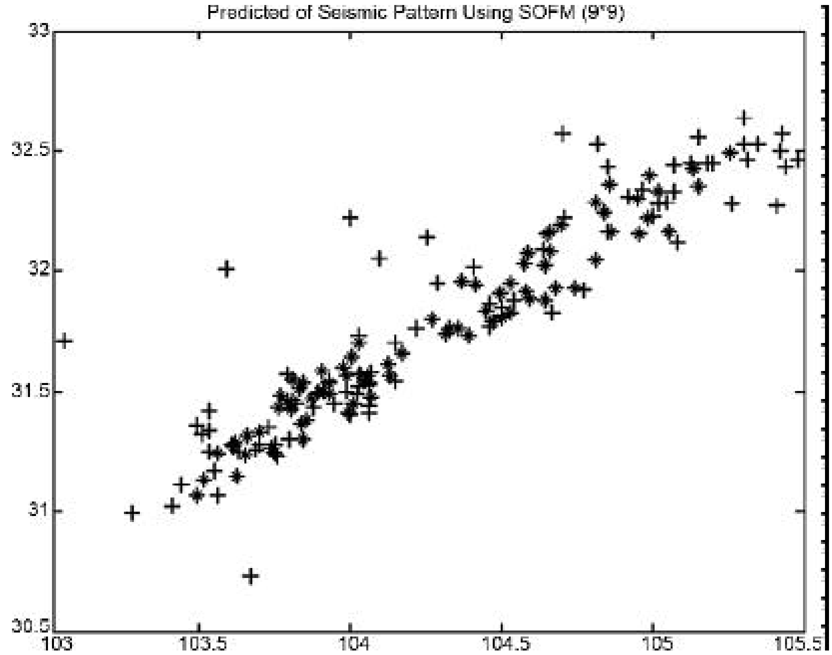

Figure 6. Stars (synaptic weight vectors) are the synthetic epicenters of future aftershocks of May 12, 2008 Chengdu, Sichuan, China earthquake which are predicted by a two dimensional (9*9) Kohonen neural network. The epicenters of real aftershocks of this earthquake are plotted as pluses.

shocks can be obviously detected.

However, Kohonen networks work best when input vectors distribution is closed, therefore, this program functions best for local aftershocks zones.

An additional application of this method is discrimination between aftershocks of two different main events, where there is no clear boundary between the aftershocks.

References

1. Baiesi, M. and Paczuski, M. (2004). "Scale-Free Networks of Earthquakes and Aftershocks", Physical Review, **69**, 1-8.

2. Omori, F. (1894). "On Aftershocks", Repl. Imp. Earth. Invest. Committe, **2**, 103-109.

3. Utsu, T., Ogata, Y., and Matsu'ura, R.S. (1995). "The Centenary of the Omori Formula for a Decay Law of Aftershock Activity", *Journal Physics Earth*, **43**, 1-33.

4. Dowla, F.U., Taylor, S.R., and Anderson, R.W. (1990). "Seismic Discrimination with Artificial Neural Networks: Preliminary Results with Regional Spectral Data", Bull. Seism. Soc. Am., **80**, 1910-1933.

5. Allamehzadeh, M. and Abbassi, M. (2008). "Recognition of Seismic Precursory Activities Using Self-Organizing Feature Map (SOFM) Neural Networks", *Journal Disaster Advances*, **1**, 29-39.

6. Dai, H.C. and MacBeth, C. (1995). "Automatic Picking of Seismic Arrivals in Local Earthquakes Data Using an Artificial Neural Network", *Geophysical Journal International*, **120**(3), 758-774.

7. Aminzadeh, F., Katz, S., and Aki, K. (1994). "Adaptive Neural Nets for Generation of Artificial Earthquake Precursors", *IEEE Trans. Geosci. Remote Sensing*, **32**, 1139-1143.

8. Allamehzadeh, M. and Mokhtari, M. (2003). "Prediction of Afershocks Distribution Using Self-Organizing Feature Maps (SOFM) and Its Application on the Birjand-Ghaen and Izmit Earthquakes", *Journal of Seismology and Earthquake Engineering (JSEE)*, **5**(3), 1-15.

9. Avouac, J.P. and Tapponnier, P. (1993). "Kinematic Model of Active Deformation in Central-Asia", Geophysical Research Letters, **20**, 895-898.

10. Gan, W.J., Zhang, P.Z., Shen, Z.K., Niu, Z.J., Wang, M., Wan, Y.G., Zhou, D.M., and Cheng, J.

(2007). “Present-Day Crustal Motion within the Tibetan Plateau Inferred from GPS Measurements”, *Journal of Geophysical Research-Solid Earth*, **112**, B08416, 14p.

11. Peltzer, G., Tapponnier, P., and Armijo, R. (1989). “Magnitude of Late Quaternary Left Lateral Displacements along the North Edge of Tibet”, *Science*, **246**, 4935, 1285-1289.

12. Zhang, P.Z., Shen, Z., Wang, M., Gan, W.J., Burgmann, R., and Molnar, P. (2004). “Continuous Deformation of the Tibetan Plateau from Global Positioning System Data”, *Geology*, **32**, 809-812.

13. Lippmann, R.P. (1987). “An Introduction to Computing with Neural Nets”, IEEE Acoustic, Speech and Signal Processing Magazine, **4**(2), 4-22.

14. Haykin, S. (1994). “Neural Networks a Comprehensive Foundation”, McMaster University, Canada.

15. Kohonen, T. (1981). “Automatic Formation of Topological Maps of Pattern in a Self-Organizing System”, *Proceedings of the 2nd Scandinavian Conference on Image Analysis*, Pattern Recognition Society of Finland.

16. Kohonen, T. (1982). “Self-Organized Formation of Topologically Correct Feature Maps”, *Biological Cybernetics*, **43**, 59-69.