KOHONEN NETWORKS AS HYDROACOUSTIC SIGNALS CLASSIFIER

ANDRZEJ ŻAK

Naval University of Gdynia
Smidowicz 69, 81-103 Gdynia
andrzej-zak@wp.pl

The paper presents the method of neural network used as classifier of hydroacoustic signals made by moving ship. The main task of proposed solution is to classify the objects which made the underwater noises. From the technique of neural network the Kohonen networks which belongs to group of self organizing networks where chosen to solve the research problem. Hydroacoustic signals were acquired on the hydroacoustic range during the complex ship measurement. At the end the results of classification of underwater noises made by ship where presented.

INTRODUCTION

Classification is a procedure in which individual items are placed into groups based on quantitative information on one or more characteristics inherent in the items (referred to as traits, variables, characters, etc) and based on a training set of previously labeled items.

Formally, the problem can be stated as follows: given training data $\{(x_1, y_1), \ldots, (x_n, y_n)\}$ produce a classifier $h: X \rightarrow Y$ which maps an object $x \in X$ to its classification label $y \in Y$. Classification algorithms are very often used in pattern recognition systems.

While there are many methods for classification, they are solving one of three related mathematical problems. The first is to find a map of a feature space (which is typically a multi-dimensional vector space) to a set of labels. This is equivalent to partitioning the feature space into regions, then assigning a label to each region. Such algorithms (e.g., the nearest neighbour algorithm) typically do not yield confidence or class probabilities, unless post-processing is applied. Another set of algorithms to solve this problem first apply unsupervised clustering to the feature space, then attempt to label each of the clusters or regions.

The second problem is to consider classification as an estimation problem, where the goal is to estimate a function of the form

$$P(class \mid \vec{x}) = f(\vec{x}; \vec{\theta}) \quad (1)$$

where the feature vector input is $\vec{x}$, and the function $f(\cdot)$ is typically parameterized by some parameters $\vec{\theta}$. In the Bayesian approach to this problem, instead of choosing a single parameter vector $\vec{\theta}$, the result is integrated over all possible thetas, with the thetas weighted by how likely they are given the training data D:

$$P(class|\vec{x}) = \int f(\vec{x};\vec{\theta})P(\vec{\theta}|D)d\vec{\theta} \tag{2}$$

The third problem is related to the second, but the problem is to estimate the class-conditional probabilities $P(\vec{x}|class)$ and then use Bayes' rule to produce the class probability as in the second problem.

The most widely used classifiers are the Neural Network (Multi-layer Perceptron, Self Organizing Maps), Support Vector Machines, k-Nearest Neighbours, Gaussian Mixture Model, Gaussian, Naive Bayes, Decision Tree and RBF classifiers.

In this paper the hydroacoustics signals classification is understood as the process of automatically recognition what kind of object is generating acoustics signals on the basis of individual information included in generated sounds. Hydroacoustics signal classification is a difficult task and it is still an active research area. Automatic signal classification works based on the premise that sounds emitted by object to the environment are unique for that object. However this task has been challenged by the highly variant of input signals. The principle source of variance is the object himself. Sound signals in training and testing sessions can be greatly different due to many facts such as object sounds change with time, efficiency conditions (e.g. some elements of machinery are damaged), sound rates, etc. There are also other factors, beyond object sounds variability, that present a challenge to signal classification technology. Examples of these are acoustical noise and variations in recording environments and changes of environment itself.

In the paper the Kohonen Neural Networks were discussed as hydroacoustic signals, generated by moving ship, classifier.

1. KOHONEN NETWORKS

Kohonen network, also known as The Self-Organizing Map (SOM) is a computational method for the visualization and analysis of high-dimensional data, especially experimentally acquired information.

One of the most interesting aspects of SOMs is that they learn to classify data without supervision. With this approach an input vector is presented to the and the output is compared with the target vector. If they differ, the weights of the network are altered slightly to reduce the error in the output. This is repeated many times and with many sets of vector pairs until the network gives the desired output. Training a SOM however, requires no target vector.

For the purposes of this paper the two dimensional SOM will be discussed. The network is created from a 2D lattice of 'nodes', each of which is fully connected to the input layer. Figure 1 shows a very small Kohonen network of 4×4 nodes connected to the input layer (shown as rectangle) representing a two dimensional vector.

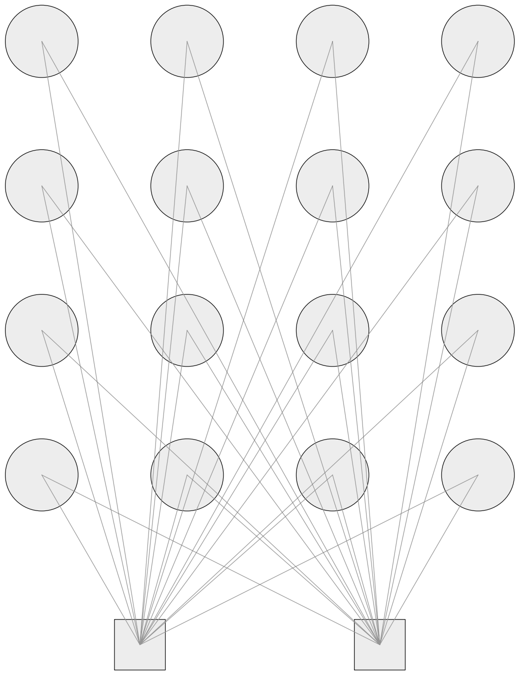

Fig.1 A simple Kohonen network

A SOM does not need a target output to be specified unlike many other types of network. Instead, where the node weights match the input vector, that area of the lattice is selectively optimized to more closely resemble the data for the class the input vector is a member of. From an initial distribution of random weights, and over many iterations, the SOM eventually settles into a map of stable zones. Each zone is effectively a feature classifier, so the graphical output can be treated as a type of feature map of the input space.

Training occurs in several steps and over many iterations:

1. Each node's weights are initialized.
2. A vector is chosen at random from the set of training data and presented to the lattice.
3. Every node is examined to calculate which one's weights are most like the input vector. The winning node is commonly known as the Best Matching Unit (BMU).
4. The radius of the neighborhood of the BMU is now calculated. This is a value that starts large, typically set to the 'radius' of the lattice, but diminishes each time-step. Any nodes found within this radius are deemed to be inside the BMU's neighborhood.
5. Each neighboring node's (the nodes found in step 4) weights are adjusted to make them more like the input vector. The closer a node is to the BMU, the more its weights get altered.
6. Repeat step 2 for N iterations.

To determine the best matching unit, one method is to iterate through all the nodes and calculate the distance between each node's weight vector and the current input vector. The node with a weight vector closest to the input vector is tagged as the BMU.

There are many methods to determine the distance for example:
the most popular Euclidean distance is given as:

$$d(x, w_i) = \|x - w_i\| = \sqrt{\sum_{j=0}^{N} (x_j - w_{ij})^2} \tag{3}$$

the scalar product is given as:

$$d(x, w_i) = 1 - xw_i = 1 - \|x\|\|w_i\| \cos(x, w_i) \tag{4}$$

the measure according to norm L1 (Manhattan) is given as:

$$d(x, w_i) = \sqrt{\sum_{j=0}^{N} |x_j - w_{ij}|} \tag{5}$$

the measure according to norm L can be written as:

$$d(x, w_i) = \max_j (|x_j - w_{ij}|) \tag{6}$$

where x is the current input vector and w is the node's weight vector.

Each iteration, after the BMU has been determined, the next step is to calculate which of the other nodes are within the BMU's neighborhood. All these nodes will have their weight vectors altered in the next step. Figure 2 shows an example of the size of a typical neighborhood close to the commencement of training.

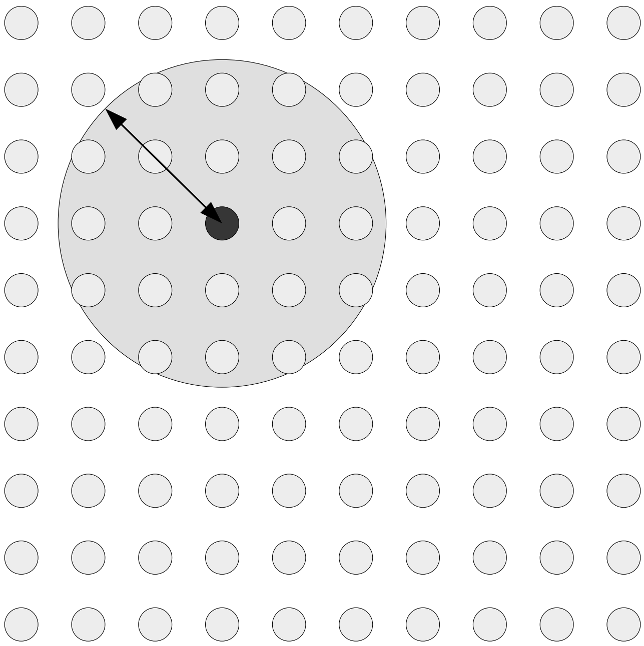

Fig.2 The BMU's neighborhood

A unique feature of the Kohonen learning algorithm is that the area of the neighborhood shrinks over time. This is accomplished by making the radius of the neighborhood shrink over time. To do this the exponential decay function can be used as follow:

$$\sigma(t) = \sigma_0 \exp\left(-\frac{t}{\lambda}\right) \quad t = 0, 1, 2, \ldots \tag{7}$$

where the σ_0 denotes the width of the lattice at time t_0 and the λ denotes a time constant, t is the current time-step (iteration of the loop).

Every node within the BMU's neighborhood (including the BMU) has its weight vector adjusted according to the following equation:

$$w_{ij}(t+1) = w_{ij}(t) + \Theta(t)\eta(t)(x_j(t) - w_{ij}(t)) \qquad (8)$$

where t represents the time-step and η is a small variable called the learning rate, which decreases with time.

The decay of the learning rate is calculated each iteration using the following equation:

$$\eta(t) = \eta_0 \exp\left(-\frac{t}{\lambda}\right) \quad t = 0,1,2,\ldots \qquad (9)$$

In Equation 8, not only does the learning rate have to decay over time, but also, the effect of learning should be proportional to the distance a node is from the BMU. Indeed, at the edges of the BMUs neighbourhood, the learning process should have barely any effect at all. Ideally, the amount of learning should fade over distance similar to the Gaussian decay according to the formula:

$$\theta(t) = \exp\left(-\frac{dist}{2\sigma^2(t)}\right) \quad t = 0,1,2,\ldots \qquad (10)$$

where $dist$ is the distance a node is from the BMU and σ, is the width of the neighbourhood function as calculated by Equation 7.

2. RESULTS OF RESEARCH

During research the six ships were measured on the hydroacoustic range of Polish Navy. Ships No. 1 and No. 4 were minesweeper project 206FM, ship No. 2 was minesweeper project 207DM, ship No. 3 was salvage ship project 570, ship No. 5 was minesweeper project 207P, and ship No. 6 was racket corvette project 1241. Every ship was measured with few, various speed of crossing. All of them were investigated at the similar hydro and meteologic conditions.

Using Kohonen networks the two dimensional neural classifier was build. Its parameter was: number of neurons: map 30x30 neurons, beginning size of area of the neighborhood was 4, beginning learning rate was 0.35, methods to determine the distance was Euclidean distance. As input data were used the amplitude spectrums, produced from time domain signals after FFT transformation for time window of length 0.5 seconds and random beginning around the loudness point of range cross. The spectrums, have the band of frequency from 5 to 500 Hz, and were normalized before presentation. After about 10000 cycles of neural network learning, was obtained the map of memberships for every presented ship as it is shown on figure 3. Because the ships No.1 and No. 4 belongs to the same kind of ships, the areas of activations which corresponds to this ships where not clearly separated. Remaining areas were clearly separated. The example results of classifier work out after learning process was presented on figure 4.

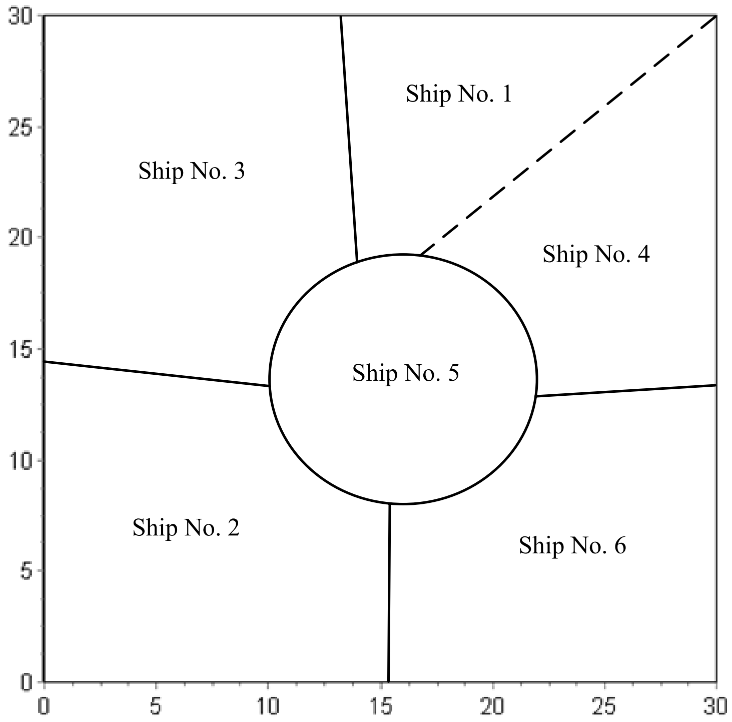

Fig.3 The map of partition for area of activation for researched ships

3. SUMMARY

As it is shown on results the used Kohonen neural network is useful for ships classification based on its hydroacoustic signature. It shows that even the ships, which belongs to the same kind, can be mapped separate quite well what suggest high resolution of classification. Presented case is quite simple because it not take into account that object sounds change with time and distance, efficiency conditions (e.g. some elements of machinery are damaged), sound rates, etc. It doesn’t consider the influence of changes of environment on acquired hydroacoustic signals. Therefore these cases should be investigated in future research. More over in future research the influence of network configuration on the quality of classification should be checked.

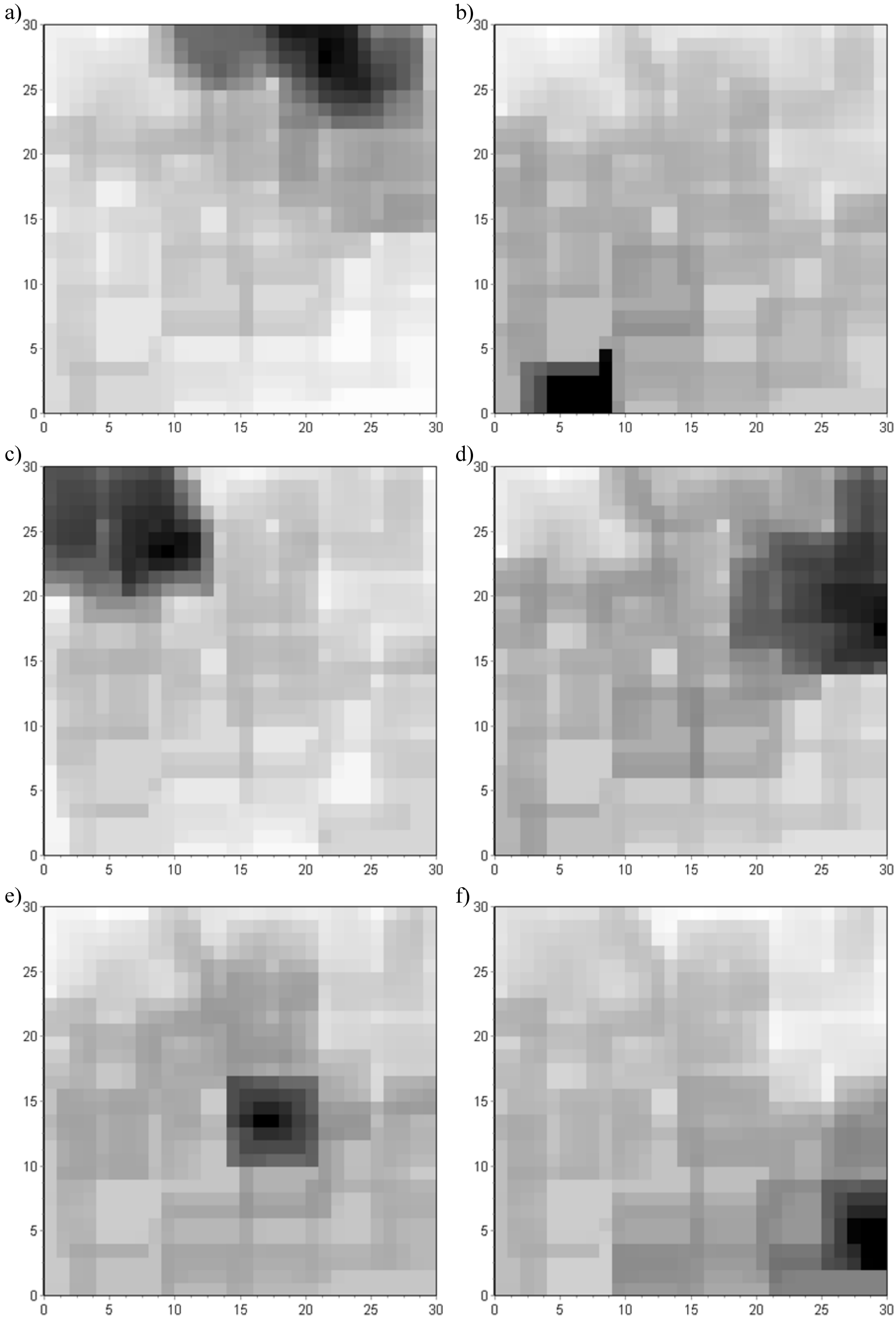

Fig.4 The results of classifier work out - maps of memberships; a) for ship no. 1; b) for ship no. 2; c) for ship no. 3; d) for ship no. 4; e) for ship no. 5; f) for ship no. 6

REFERENCES

[1] J.C. Fort, SOM's mathematics, Neural Networks, 19: 812–816, 2006.

[2] S. Haykin, Self-organizing maps, Neural networks - A comprehensive foundation, 2nd edition, Prentice-Hall, 1999.

[3] T. Kohonen, Self-organized formation of topologically correct feature maps, Biological Cybernetics, 43:59-69, 1982.

[4] T. Kohonen, Self-Organizing Maps, Third, extended edition. Springer, 2001.

[5] S. Osowski, Sieci neuronowe, Oficyna Wydawnicza Politechniki Warszawskiej, 1996.