RESEARCH PAPERS
FACULTY OF MATERIALS SCIENCE AND TECHNOLOGY IN TRNAVA
SLOVAK UNIVERSITY OF TECHNOLOGY IN BRATISLAVA

2019 Volume 27, Number 45

FRÉCHET METRIC IN NEURAL NETWORK THEORY

Renáta MASÁROVÁ[1]

[1]SLOVAK UNIVERSITY OF TECHNOLOGY IN BRATISLAVA,
FACULTY OF MATERIALS SCIENCE AND TECHNOLOGY IN TRNAVA,
INSTITUTE OF APPLIED INFORMATICS, AUTOMATION AND MECHATRONICS
ULICA JÁNA BOTTU 2781/25 TRNAVA, SLOVAK REPUBLIC
e-mail: renata.masarova@stuba.sk
Received 26 August 2019, accepted 23 October 2019, published 29 November 2019

Abstract

This paper deals with application of a modified Fréchet metric to self-organizing neural networks, called Kohenen maps. The methodology used allows us to put more emphasis on the selected parameters in the input data. It can simplify finding the minimal distance dF_j, since $dF_j \in \langle 0,1 \rangle$.

Key words

Neural networks, self-organizing maps, Kohenen algorithm, Fréchet metric

INTRODUCTION

This paper is focused on one of the basic models of neural networks – self-organizing maps (SOM). Two of the earliest people to focus on the problem of topological mapping using neural networks were Willshaw and von der Malbury in [7]. A computationally simplified model was published by Kohonen in 1982, and therefore SOM are also referred to as Kohonen maps. The new modification of his algorithm will allow the input vector coordinates to be of different importance.

KOHONEN MAPS

Kohonen maps belong to the category of self-organizing neural networks. They are learning networks without a teacher, which does not require an ideal template to be configured. All that is required is a group of input signals, and, during learning, the network will find the common characteristics, on which it will base its decisions.

Their goal is to find a special representation of the complex data structures – i.e. that similar vectors are represented by neurons that are close to a given topology. They can recognize similarity or differences between various elements by themselves. We can use them to process unknown data or signals. These maps are used in altering sound, image processing, removing unknown interference, automatized sorting etc.

In this paper, we will describe the algorithm only schematically. More details can be found in papers [2], [5], [6].

Kohonen maps represent a two-layer neural network. It has n neurons in the input layer and m neurons in the output layer. Kohonen algorithm is characterized by the following steps:

1. Defining the weight values w_{ij} (by small random numbers), setting the initial average of the area around neuron $N_j(0)$ to maximum, defining the initial learning parameter $\eta(t)$, $0 < \eta(t) < 1$.
2. Submitting the initial vector $\left(x_0(t), x_1(t), \dots, x_{n-1}(t)\right)$.
3. Computing the distance $d_j = \sum_{i=0}^{n-1} \left(x_i(t) - w_{ij}(t)\right)^2$ for every $j = 1, 2, \dots, m$.
4. Finding the winning neuron j^* for which the following holds true $d_{j^*} = \min_{j=1,\dots,m} d_j$.
5. Adjusting the weights of neuron j^* and its neighbours. The new weights are given by the formula
   $w_{ij}(t+1) = w_{ij}(t) + \eta(t)\left(x_i(t) - w_{ij}(t)\right)$ for $j \in N_{j^*}(t), 0 \le i \le n-1$.
   Updating the learning parameter and the average of its surroundings. Values $\eta(t)$ and $N_{j^*}(t)$ are reduced in time t to stabilize weights and localize of areas of maximum activity.
6. Testing the end condition. If it is not satisfied, go back to step 2.

The Kohonen algorithm hides many questions and mathematical problems. For example:

Does the Kohonen algorithm always converge?

Does an optimal value exist for choosing the learning parameter $\eta(t)$ and the average of the surroundings of neurons $N_j(t)$?

How to define the distance function d_j?

We will dedicate this paper to the last of these questions. For calculating the distance, the most commonly used is the Euclidian metric (i.e. distance defined by the relation $d_j = \sum_{i=0}^{n-1} \left(x_i(t) - w_{ij}(t)\right)^2$) , or alternatively its maximization (i.e. distance defined by the relation $d_j = \max_{0 \le i \le n-1} \left(x_i(t) - w_{ij}(t)\right)$). Improvements of these methods can be found in many papers (for example [3] and [4]). All of them are based on different modification of the Euler metric. In the following section, we will show a new approach. We will modify the Fréchet metric which is used in the metric space theory for calculating distances between infinite numeric sequences. In comparison with previous metrics, for which all coordinates are “equal”, the coordinates of the initial vector in Fréchet metric can be each given a different priority.

FRÉCHET METRIC FOR FINITE DIMENSIONAL VECTORS

In this section, we will modify the definition of the Fréchet metric [8] for ordered n-tuples, where n is any natural number.

Let $n \in N$ and X be the set of all ordered n-tuples or real numbers. Let $x = (x_1, x_2, \dots, x_n)$ and $y = (y_1, y_2, \dots, y_n)$ be two elements of this set. Their distance can be defined as follows:

$$d_F(x, y) = \sum_{k=1}^{n} \frac{1}{2^k} \cdot \frac{|x_k - y_k|}{1 + |x_k - y_k|}.$$

The distance defined as such satisfies three properties of a metric:
1. $\forall x, y \epsilon X \;\; d_F(x,y) \geq 0$, at most $d_F(x,y) = 0 \Leftrightarrow x = y$
2. $\forall x, y \epsilon X \;\; d_F(x,y) = d_F(y,x)$
3. $\forall x, y, z \epsilon X \;\; d_F(x,y) \leq d_F(x,z) + d_F(z,y)$.

Proof:
1. Non-negativity is obvious, because $d_F(x,y)$ is a sum of finite positive elements. If $x = y$ then $d_F(x,y) = \sum_{k=1}^{n} \frac{1}{2^k} \cdot \frac{0}{1} = 0$.
   If $d_F(x,y) = 0$ then $\frac{|x_k - y_k|}{1+|x_k - y_k|} = 0$ for each $k = 1,2,\dots,n$
   $\Rightarrow |x_k - y_k| = 0 \Rightarrow x_k = y_k$ for each $k = 1,2,\dots,n \Rightarrow x = y$.
2. $d_F(x,y) = \sum_{k=1}^{n} \frac{1}{2^k} \cdot \frac{|x_k - y_k|}{1+|x_k - y_k|} = \sum_{k=1}^{n} \frac{1}{2^k} \cdot \frac{|y_k - x_k|}{1+|y_k - x_k|} = d_F(y,x)$.
3. To prove triangle inequality, we first need to prove the auxiliary implication:
$$\forall a, b, c \in R^+ : a \leq b + c \Rightarrow \frac{a}{1+a} \leq \frac{b}{1+b} + \frac{c}{1+c}.$$
   Let a, b, c be three positive real numbers that satisfy the inequality $a \leq b + c$.
$$\frac{a}{1+a} \leq \frac{b}{1+b} + \frac{c}{1+c}$$
$$a(1+b)(1+c) \leq b(1+a)(1+c) + c(1+a)(1+b)$$
$$a + ab + ac + abc \leq b + ab + bc + abc + c + ac + bc + abc$$
$$a \leq b + bc + c + bc + abc$$
$$a \leq b + c + (2bc + abc)$$
$$2bc + abc > 0 \text{ because } a, b, c > 0$$
$$a \leq b + c.$$

For each x_k, y_k, z_k:
$$|x_k - y_k| = |x_k - z_k + z_k - y_k| \leq |x_k - z_k| + |z_k - y_k|.$$

Using the auxiliary implication, we get
$$\forall k = 1,2,\dots,n: \frac{|x_k - y_k|}{1+|x_k - y_k|} \leq \frac{|x_k - z_k|}{1+|x_k - z_k|} + \frac{|z_k - y_k|}{1+|z_k - y_k|}$$
$$\sum_{k=1}^{n} \frac{1}{2^k} \frac{|x_k - y_k|}{1+|x_k - y_k|} \leq \sum_{k=1}^{n} \frac{1}{2^k} \frac{|x_k - z_k|}{1+|x_k - z_k|} + \sum_{k=1}^{n} \frac{1}{2^k} \frac{|z_k - y_k|}{1+|z_k - y_k|}$$
$$d_F(x,y) \leq d_F(x,z) + d_F(z,y).$$

Note: The Fréchet metric takes on values from the interval $\langle 0,1 \rangle$, because
$$0 \leq d_F(x,y) = \sum_{k=1}^{n} \frac{1}{2^k} \frac{|x_k - y_k|}{1+|x_k - y_k|} \leq \sum_{k=1}^{n} \frac{1}{2^k} \leq \sum_{k=1}^{\infty} \frac{1}{2^k} = 1.$$

USE OF THE FRÉCHET METRIC IN KOHONEN MAPS

Now we will return to the Kohonen algorithm. Its steps will be as follows:
1. Defining the weight values w_{ij} (by small random numbers), setting the initial average for the surrounding of neuron $N_j(0)$ to maximum, defining the initial learning parameter $\eta(t)$, $0 < \eta(t) < 1$.
2. Submitting the initial vector $\left(x_0(t), x_1(t), \dots, x_{n-1}(t)\right)$.

3. Calculating the distance $d_j = \sum_{i=0}^{n-1} \frac{1}{2^{i+1}} \cdot \frac{|x_i(t) - w_{ij}(t)|}{1 + |x_i(t) - w_{ij}(t)|}$ for every $j = 1, 2, \dots, m$.
4. Finding the winning neuron j^* for which the following holds true $d_{j^*} = \min_{j=1,\dots,m} d_j$.
5. Adjusting the weights for neuron j^* and its neighbours. The new weights are given by the relation
   $w_{ij}(t+1) = w_{ij}(t) + \eta(t)\left(x_i(t) - w_{ij}(t)\right)$ pre $j \in N_{j^*}(t), 0 \le i \le n-1$,
   while updating the learning parameter and the average of the surroundings. Values $\eta(t)$ and $N_{j^*}(t)$ are reduced in time t to stabilize weights and localize the areas of maximum activity.
6. Testing the end condition. If it is not satisfied, go back to step 2.

The difference is that, in the third step, we replace the most common distance calculation

$$d_j = \sum_{i=0}^{n-1} \left(x_i(t) - w_{ij}(t)\right)^2$$

by

$$dF_j = \sum_{i=0}^{n-1} \frac{1}{2^{i+1}} \cdot \frac{|x_i(t) - w_{ij}(t)|}{1 + |x_i(t) - w_{ij}(t)|}.$$

By breaking down this sum we get

$$dF_j = \frac{1}{2} \cdot \frac{|x_0(t) - w_{0j}(t)|}{1 + |x_0(t) - w_{0j}(t)|} + \frac{1}{4} \cdot \frac{|x_1(t) - w_{1j}(t)|}{1 + |x_1(t) - w_{1j}(t)|} + \frac{1}{8} \cdot \frac{|x_2(t) - w_{2j}(t)|}{1 + |x_2(t) - w_{2j}(t)|} + \cdots$$
$$+ \frac{1}{2^n} \cdot \frac{|x_{n-1}(t) - w_{n-1j}(t)|}{1 + |x_{n-1}(t) - w_{n-1j}(t)|}.$$

Here we can see the most significant difference between both metrics. In comparison with the Euclidian metric, the difference between the first coordinates is the most significant. The effect difference between each following pair is halved, compared to the one before it. We can use this if we consider some elements of the input signal to be more important than others. In addition, the value of this distance is at most equal to 1.

We will show the modified algorithm on an example from [5].

Example. To illustrate, we will show a simple example of Kohonen map learning with a one-dimensional arrangement of two neurons (more precisely the first cycle).
We have the following vectors $(1,1,0,0), (0,0,0,1), (1,0,0,0), (0,0,1,1)$. We choose initial learning parameter $\eta(t) = 0.6$, and learning function $\eta(t+1) = 0.5 \cdot \eta(t)$.
The radius around the neuron will be 0 (only one neuron updates its weight values at each step)

We define the weigh matrix

$$\begin{pmatrix} 0.2 & 0.8 \\ 0.6 & 0.4 \\ 0.5 & 0.7 \\ 0.9 & 0.3 \end{pmatrix}.$$

For the first input vector

$$dF_1 = \frac{1}{2} \cdot \frac{0.8}{1.8} + \frac{1}{4} \cdot \frac{0.4}{1.4} + \frac{1}{8} \cdot \frac{0.5}{1.5} + \frac{1}{16} \cdot \frac{0.9}{1.9} = 0.365$$

$$dF_2 = \frac{1}{2} \cdot \frac{0.2}{1.2} + \frac{1}{4} \cdot \frac{0.6}{1.6} + \frac{1}{8} \cdot \frac{0.7}{1.7} + \frac{1}{16} \cdot \frac{0.3}{1.3} = 0.302$$

The input vector is closer to node 2. We are updating the weight matrix

$$\begin{pmatrix} 0.2 & 0.92 \\ 0.6 & 0.64 \\ 0.5 & 0.28 \\ 0.9 & 0.12 \end{pmatrix}.$$

We continue with the step 3 (calculation dF_j) for the second input vector.

CONCLUSION

The methodology mentioned in the article allows us to put more emphasis on the selected parameters in the input data (for example, to prioritize a factor in image recognition). At the same time, it can simplify finding the minimal distance dF_j, because $dF_j \in \langle 0,1 \rangle$.

Acknowledgement

This paper was supported by the VEGA Slovak Scientific Grant Agency within the Project 1/0232/18 Using the methods of multi-objective optimization in a production processes control.

This publication has been written thanks to support of the Operational Program Research and Innovation for the project Research, modeling and simulation of industrial production processes using progressive technologies, ITMS code: NFP313010T589 co-financed by the European Regional Development Fund.

References

[1] KOHONEN, T. 1995. *Self-Organizing Maps*. Springer.
[2] KVASNIČKA, V., BEŇUŠOVÁ, L., POSPÍCHAL, J., FARKAŠ, I., TIŇO, P., KRÁL, A. 1997. *Úvod do teórie neurónových sietí (Introduction into the theory of neural networks)*. Bratislava: Iris, 285 p. ISBN 9788088778301.
[3] RITTER, H., SCHULTEA, K. 1998. Kohonen self-organizing maps. In: *Exploring their computational capabilities*. Proc. IEEE ICNN, San Diego, 109-116.
[4] RUŽIČKA, P. 1993. On convergence of learning results for topological maps. *NeuralNetwork World,* **4**, 413-424.
[5] VOLNÁ, E. 2008. *Neuronové síte 1 (Neural networks 1).* Ostrava: University in Ostrava. 87 p.
[6] VOLNÁ, E. 2012. *Evoluční algoritmy a neuronové síte (Evolution algorithms and neural networks).* Ostrava: University in Ostrava. 152 p.
[7] WILLSHAW, D.J., MALSBURG VON DER, D. 1976. How patterned neural connection can be set up by self-organization. In: *Proc.of the Royal Society of London B*, 194, 431-445.
[8] ŠALÁT, T. 1981. *Metrické priestory (Metric spaces).* Bratislava: Alfa. 224 p.