Potential Function Method Approach to Pattern Recognition Applications

Peter Grabusts
Rezekne Academy of Technologies, Atbrivoshanas alley 115, Rezekne, Latvia

Abstract. ***Potential function method was originally offered to solve the pattern recognition tasks, then it was generalized to a wider range of tasks, which were associated with the function approximation. Potential function method algorithms are based on the hypothesis of the nature of the function that separates sets according to different classes of patterns. Geometrical interpretation of pattern recognition task includes display of patterns in the form of vector in the space of input signal that allows to perceive the learning as approximation task. The paper describes the essence of potential function method and the learning procedure is shown that is based on practical application of potential methods. Pattern recognition applications with the help of examples of potential functions and company bankruptcy data analysis with the help of potential functions are given.***

Keywords: potential functions, pattern recognition, bankruptcy prediction.

I. INTRODUCTION

It is known from physics that the distribution of electrical charge potential (depending on the distance to the charge) is determined by the formula:

$$p = a\frac{q}{r^2} \tag{1}$$

where a – coefficient;

q – charge size;

r – the distance from the given point to the charge.

Geometrical interpretation of distribution of potentials is shown in Fig. 1.

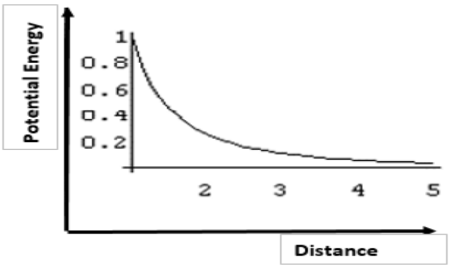

Fig.1. Distribution of potential

Thus, the potential value can serve as a starting point of the distance to the charge. When the electric field is made by a number of charges, the potential of each point of the field is equal to the amount of the potentials created by each of the charges in this point. If charges form a compact group, then the maximum value potential will take the charge internally and respectively decrease, falling away from that group.

The algorithm of the potential function method is based on the hypothesis about the function character that separates sets according to different classes of patterns [3], [5], [11]. Further description of the algorithm is given in the pattern recognition context. Assume that there are two compact groups of charges in the space. In the one group the charges are negative but in the other positive. Fig. 2 represents these groups and the set potential distribution curve.

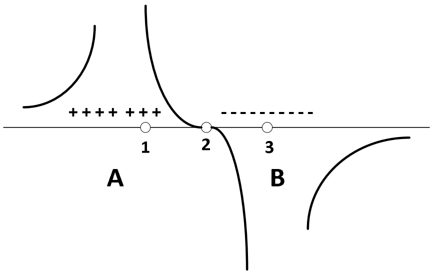

Fig.2. Potential groups

Each point of the space is closer to the set whose potential at that point is larger. Any point can be associated with specific set depending on the total potential sign at that point. Hence, the total potential curve is at the same time set separating curve. Say, point 1 has positive potential, so it can be ascribed to set A; point 3 has negative potential and it can be ascribed to set B; point 2 has zero potential on the boundary of sets.

Similar assumptions can be laid down in the pattern recognition basis by connecting vectors of learning patterns with certain potential functions. Let us consider the idea of potential function method on the basis of two-class separability.

Suppose that classes A and B do not intersect, i.e., in set X at least one separation function $\Psi(z)$ exists which assumes positive values (corresponding to class A) and negative values (corresponding to class

ISSN 1691-5402

http://dx.doi.org/10.17770/ 10.17770/etr2017vol2.2512

B). In general case there may be many such separation functions.

In the process of learning, points in space X are sequentially shown and it is declared to which class these points belong. Provided that this information is only available, it is required to determine the separation function within the finite number of steps. Then it would be possible to ascertain the conformity of test set points to class A or B depending on the separation function sign at those points.

II. LEARNING PROCEDURE, BASED ON THE POTENTIAL FUNCTION METHOD

Solving of that task deals with the following procedure: if some point x_k appears in the course of learning, potential function $K(x, x_k)$ is associated with it, which is set over the whole set X and is dependent on $x_k$as a parameter ("potential function"). In the process of learning, to point collection $x_1, x_2, \ldots$ a collection of potential functions corresponds $K(x, x_1), K(x, x_2) \ldots$ used to determine function $\Psi(x, x_1, x_2, \ldots$ by means of certain rules. These rules are determined so that $\Psi^*(x, x_1, x_2, \ldots)$ would tend to some of separation functions. The procedure when separation function is successively constructed using the functions, which are shown by space points, is called potential function method [5].

In order to ensure learning of that type, it is first necessary to restrict choice of space X and of function class $\psi(x)$. Otherwise, points may appear in testing whose membership class is misclassified. The restriction is determined so that in space X existence of such functions $\psi_i(x))i = 1,2, \ldots)$ is accepted at which it is possible for each pair of separable sets to find such a N under which separation function could be assigned as follows (description of variables see below):

$$\Psi(x) = \sum_{i=1}^{N} c_i \varphi_i(x) \qquad (2)$$

Let us introduce some important basic concepts. To each point $x \in X$ there corresponds point $z \in Z$ with co-ordinates $z_i = \varphi_i(x)$. Using formula (2), the separation function $\Psi(x)$ in space Z can be represented as linear function $\sum_{k=1}^{N} c_k z_k$. Since

$$\Psi(x) = \sum_{k=1}^{N} c_k z_k \qquad \begin{Bmatrix} > 0 \ldots x \in A \\ < 0 \ldots x \in B \end{Bmatrix} \qquad (3)$$

then points in space Z that belong to different classes are separated by hyperplane

As a potential function, we will use the function of two variable arguments:

$$K(x, y) = \sum_{i=1}^{\infty} \lambda_i^2 \varphi_i(x) \varphi_i(y) \qquad (4)$$

where $\varphi_i(x)$ $(i = 1,2 \ldots .)$ a system of linear independent functions (see condition (2) above);

λ_i – integer numbers different from 0.

In what follows we will assume that $\varphi_i(x)$ and $K(x, y)$ are restricted at $x \in A \cup B$. Variable y will be interpreted using the points that appear in the process of learning.

Suppose that in the course of learning, points $x_1, x_2, \ldots, x_k$ appear and each of them belongs either to set A or to set B. Let us assume conditionally that positive values belong to set A but negative values belong to set B. In learning, after the first point x_1 is shown, function $K_1(x)$ is constructed that is equal to the potential at point x_1 taken with the sign of the corresponding set, i.e.:

$$K_1(x) = \begin{cases} K(x, x_1) & if \;\; x_1 \in A \\ -K(x, x_1) & if \;\; x_1 \in B \end{cases} \qquad (5)$$

Further algorithm execution is based on induction. Assume that after the $r - th$ appearance, potential $K_r(x)$ is created. Suppose that at the next $(r+1) - th$ learning step, point $x_r + 1$ appears. As a result of that, four cases can occur:

$x_{r+1} \in A$,	$K_r(x_{r+1})>$	(a)
$x_{r+1} \in B$,	$K_r(x_{r+1})<$	(b)
$x_{r+1} \in A$,	$K_r(x_{r+1})<$	(c)
$x_{r+1} \in B$,	$K_r(x_{r+1})>$	(d)

In cases (a) and (b) the sign of the set to which x_{r+1} belongs and the sign of $K_r(x_{r+1})$ coincide, which means that the algorithm correctly classifies point x_{r+1}. In this case it is assumed that $K_{r+1}(x) = K_r(x)$.

In cases (c) and (d) there is an error as the sign of the set and the sign of K_{r+1} do not coincide. "Error correction" is accomplished as follows:

- in case (c) it is assumed that $K_{r+1}(x) = K_r(x) + K(x, x_{r+1})$ and
- in case (d) it is assumed that $K_{r+1}(x) = K_r(x) - K(x, x_{r+1})$.

The learning algorithm described above can be used in this way: during learning, when the $s - th$ point appears, function $K_s(x)$ is constructed. After a sufficiently long learning, the process is interrupted but function $K_s(x)$ is supposed to be a separation function. As a result of testing, after a new point x^* has appeared, the value of $K_s(x^*)$ is calculated and this point is ascribed to class A or B depending on the sign of $K_s(x^*)$.

The results of practical implementation of the method are discussed below.

III. PATTERN RECOGNITION WITH POTENTIAL FUNCTION APPLICATIONS

A. Experiment I. Artificial data

Let us consider the application of potential function method in pattern recognition tasks. Statistical data on automotive diagnostic measurements were taken as the learning data. The following measurements were made:

- mileage;

- towing capacity;
- exhaust gas;
- relevant class.

Classification is as follows:

- 1- norm;
- 2- ring wear;
- 3- cylinder group wear;
- 4- cylinder wear.

Learning set consists of 4 classes of 15 patterns in each class. After learning the testing is carried out using the given data, for which the relevance to a particular class was previously known. The recognition accuracy should be the result of testing.

The potential function takes the form of a function (see Fig. 3):

$$\varphi(R) = \frac{\lambda}{1+\alpha R^2} \qquad (6)$$

where α - learning parameter;

R – the distance between the point where the potential is calculated and the point of learning set;

λ - the value of potential that is assigned to the point in the process of learning (weight).

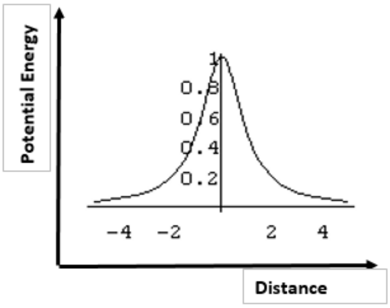

Fig. 3. Potential function within [-5,5]

The learning procedure is as follows:

1. We introduce learning set patterns of all four classes W_1, W_2, W_3, W_4.

2. We take all the patterns in turn (starting with the first X_1) and determine the corresponding class to this pattern:

a) determine the distance squares from the learning subordinated pattern to all others. Indicate classes of patterns with:

$$Class1 \;\; x_1,\, x_2, \dots, x_{15}$$
$$Class2 \;\; y_1, y_2, \dots, y_{15}$$
$$Class3 \;\; z_1, z_2, \dots, z_{15}$$
$$Class4 \;\; v_1,\, v_2, \dots, v_{15}$$

It has to be calculated:

$$R^2_{x_1x_1}; R^2_{x_1x_2}, \dots\dots\dots\dots\dots\dots\dots R^2_{x_1x_{15}}$$
$$R^2_{x_1y_1}; R^2_{x_1y_2}, \dots\dots\dots\dots\dots\dots\dots R^2_{x_1y_{15}}$$
$$R^2_{x_1z_1}; R^2_{x_1z_2}, \dots\dots\dots\dots\dots\dots\dots R^2_{x_1z_{15}}$$
$$R^2_{x_1v_1}; R^2_{x_1v_2}, \dots\dots\dots\dots\dots\dots\dots R^2_{x_1v_{15}}$$

b) We calculate out the potential functions:

$$\varphi_{x_1x_1}, \varphi_{x_1x_2}, \dots\dots\dots\dots\dots\dots\dots \varphi_{x_1x_{15}}$$
$$\varphi_{x_1y_1}, \varphi_{x_1y_2}, \dots\dots\dots\dots\dots\dots\dots \varphi_{x_1y_{15}}$$
$$\varphi_{x_1z_1}, \varphi_{x_1z_2}, \dots\dots\dots\dots\dots\dots\dots \varphi_{x_1z_{15}}$$
$$\varphi_{x_1v_1}, \varphi_{x_1v_2}, \dots\dots\dots\dots\dots\dots\dots \varphi_{x_1v_{15}} \;,$$

using the formula

$$\varphi_{x_1x_n} = \frac{\lambda_n}{1+\alpha R^2_{x_1x_n}}$$
$$\varphi_{x_1y_n} = \frac{\lambda_n}{1+\alpha R^2_{x_1y_n}}$$
$$\varphi_{x_1z_n} = \frac{\lambda_n}{1+\alpha R^2_{x_1z_n}}$$
$$\varphi_{x_1v_n} = \frac{\lambda_n}{1+\alpha R^2_{x_1v_n}}$$

c) we calculate the average value of the potential created by the patterns of each class with respect to the learning pattern:

$$\psi_{w_1x_1} = \frac{1}{N_{x_n}} \sum_{i=1}^{N_{x_n}} \varphi_{x_1x_n}$$
$$\psi_{w_2x_1} = \frac{1}{N_{y_n}} \sum_{i=1}^{N_{y_n}} \varphi_{x_1y_n}$$
$$\psi_{w_3x_1} = \frac{1}{N_{z_n}} \sum_{i=1}^{N_{z_n}} \varphi_{x_1z_n}$$
$$\psi_{w_4x_1} = \frac{1}{N_{v_n}} \sum_{i=1}^{N_{v_n}} \varphi_{x_1v_n} \;,$$

where $N_{x_n}, N_{y_n}, N_{z_n}, N_{v_n}$ - the number of patterns in each class (in this case – 15 patterns in each class)

d) We compare the values of $\psi_{w_1x_1}, \psi_{w_2x_1}, \psi_{w_3x_1}, \psi_{w_4x_1}$ and the learner pattern is attributed to the class which creates the greatest potential for this pattern.

3) We set the pattern recognition accuracy i.e. compare the resulting class value with the previously known.

4) If the pattern is recognized correctly (classes match) - move to the next pattern.

5) If the image is not recognized correctly, we increase the "weight" value by 1 and move to the next pattern.

6) After the last pattern recognition the first learning cycle ends. We start a new cycle - again

repeat learning points 2,3,4,5 (starting from the first and ending with the last pattern). Learning shall be done as long as all the patterns are recognized correctly at the current cycle - resulting class values coincide with the previously given.

7) We save the resulting values for each pattern (which will be required for the testing phase).

8) At the given value the learning is completed.

For each pattern to be tested we do the activities listed in the second point, and the pattern is applied to one of the four classes. Comparing the computed value of the class with previously known - we find out whether the pattern is recognized correctly or incorrectly.

We take the next test pattern and so on. At the end of the testing we process the results obtained, for example, the number of incorrectly recognized patterns, etc.

As a learning sample the following data were taken (60 patterns - 4 class of 15). Learning coefficient α changed within [0.005; 1.000].

Analyzing the learning and testing results, it can be concluded that the algorithm converges at all α values, but the number of learning cycles decrease with increasing of α value. At $\alpha > 0.03$ the learning is performed in the first step. Table I shows the number of learning steps and the number of unrecognized patterns at different values of α coefficient.

Table I.
LEarning steps and Unrecognized patterns

α	Learning steps	Unrecognized patterns
0.01	3	12
0.02	2	7
0.04	1	8
0.05	1	8
0.06	1	8
0.07	1	8
0.08	1	8
0.09	1	8
0.1	1	8
0.2	1	8
0.3	1	9
0.4	1	9
0.5	1	9
0.6	1	9
1.0	1	9

The Fig. 3 shows number of learning steps dependence on α, Fig. 4 – number of unrecognized patterns dependence on α.

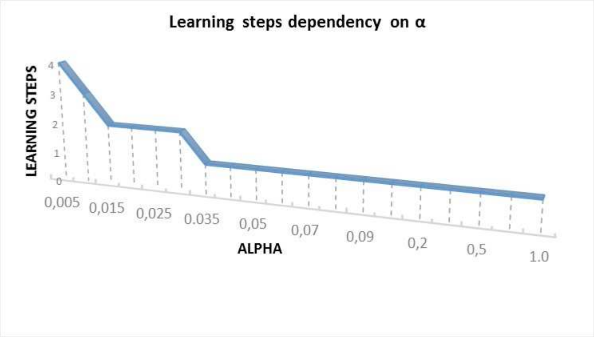

Fig. 3. Learning dependency on α

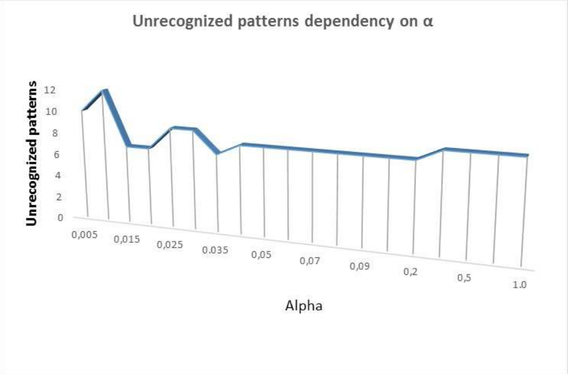

Fig. 4. Unrecognized patterns number dependency on α

The result of the experiment - correct pattern recognition is performed for 92.5% of all tested patterns (9 patterns remained unrecognized: 3 from class W_2, 5 from class W_3, and 1 from class W_4).

Thus, 100% recognition in the result of data experiment is obviously not possible for the reason that classes overlap, and hence it is not possible to construct a "good" separating function.

B. Experiment 1I. Bankruptcy data

Bankruptcy diagnostics is a directed financial analysis system whose area is crisis situation control at the enterprises. In the analysis of the general financial situation of the company a separate group of financial ratios is made, using which it is possible to reason about the threat of bankruptcy [6], [7], [10], [12], [14].

The task of this experiment is to examine bankruptcy data by potential function method ability to correctly perform classification - bankrupt or non-bankrupt (only two-class separability).

The data on firm bankruptcy were taken from [13], [15]. For the purpose of experiments, balance sheet data of 63 companies were used (46 - bankruptcy and 17 - not bankruptcy). It was decided to calculate the following financial ratios on the basis of the data available and further use them in all the experiments:

- R2: Cash Flow / Current Liabilities;
- R3: Cash Flow / Total Assets;
- R7: Current Assets / Current Liabilities;
- R9: Current Assets / Total Assets;
- R31: Working capital / Total assets.

In the first part of the experiment, the effect of learning parameter on the duration of learning (epochs) was examined. As a learning set, the bankruptcy data were used. A sample of the same firm bankruptcy data served as a test data. Learning parameter α varied within the limits [0.1, 2.0]. Learning results are given in Fig. 5.

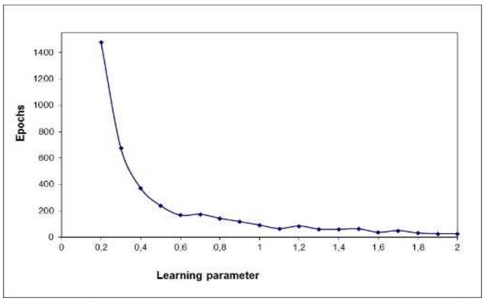

Fig.5. Graph of parameter's dependence on the number of epochs

By analyzing the results of learning, one can conclude that the algorithm converges at all the α values assigned but the number of learning cycles goes down as α value increases.

In the second part of the experiment, testing and analysis of learning algorithm execution depending on the value of potential, were performed. It was found that after the learning the points of the given test set were identified correctly, i.e., α values calculated in the course of learning enabled one to determine correctly the input data class: bankrupt or not bankrupt. After application of potential functions in the algorithm, the initial α values are equal to 0. If a point is not classified correctly during learning, the value of that parameter, is increased by 1. As a result of the experiment it was found that in bankruptcy data set there were 7 points for which correction of parameter α was performed most frequently during learning. Table II represents these points and the number of corrections performed at different α values. Note that at the rest α values the situation does not change essentially.

Table II.
Experimental results (misclassified cases and its λ parameter)

	α=0.3	α=0.4	α=0.5
Cases/λ	4/249	4/148	4/92
	8/274	8/153	8/94
	14/357	14/206	14/168
	26/393	26/212	26/112
	37/424	37/228	37/140
	50/482	50/265	50/165
	59/104	59/60	59/90

The correspondence is shown in the graphic form in Fig. 6.

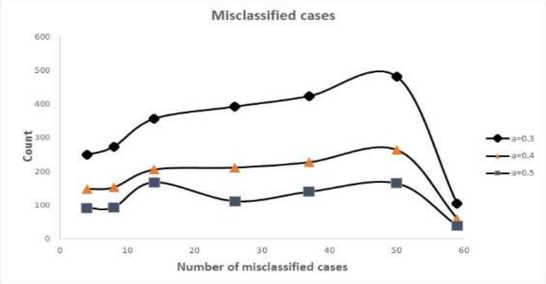

Fig.6. Misclassified cases dependence on the count of parameter λ

So, at these bankruptcy data misclassified cases for the case of potential functions are 4, 8, 14, 26, 37, 50, 59. Hence, a conclusion can be made that potential function method can be used for bankruptcy data analysis, such as neural networks [1], [2], [4], [8], [9], [13]. However, one has to be careful in interpreting the results.

IV. CONCLUSIONS

We have presented potential functions implementation possibility in bankruptcy prediction (the experiments have been performed in the Matlab environment). The experiments have shown that these methods can be viewed as alternatives to traditional bankruptcy prediction methods. Popular neural network models need significant parameter debugging resources to achieve valid results, whose correctness could be checked with traditional methods.

It can be concluded that different methods yield different results and they have to be analyzed carefully.

REFERENCES

[1] C.M. Bishop, *Neural networks for pattern recognition*. Oxford University Press, 1995.

[2] J. Hertz, A. Krogh and R.G. Palmer, *Introduction to the theory of neural computation*. Addison Wesley, 1991.

[3] M. Niranjan, A.J. Robinson and F. Fallside, *Pattern recognition with potential functions in the context of neural networks. Proceedings Sixth Scandinavian Conference on Image Analysis*, Oulu, Finland, Volume 1, pp. 96-103, 1989.

[4] M. Odom and R. Sharda, *A neural network model for bankruptcy prediction. Proc. Int. Joint Conf. Neural Networks*, San Diego, CA, 1990.

[5] M.A. Aizerman, E.M. Braverman and L.I. Rozonoer, "Theoretical foundations of the potential function method in pattern recognition learning," Automation and Remote Control, 25, 821-837, 1964.

[6] E. Altman, "Financial ratios, discriminant analysis and the prediction of corporate bankruptcy," Journal of Finance, vol. 13, pp. 589-609, 1968.

[7] E. Altman, R. Haldeman and P. Narayanan, "ZETA analysis. A new model to identify bankruptcy risk of corporations," Journal of Banking and Finance 1, 29-54, 1977.

[8] A. Atiya, "Bankruptcy prediction for credit risk using neural networks: A survey and new results," IEEE Transactions on Neural Networks, Vol. 12, No. 4, P. 929-935, 2001.

[9] B. Back, T. Laitinen and K. Sere, "Neural networks and bankruptcy prediction: funds flow, accrual ratios and accounting data, " Advances in Accounting 14, P. 23-37, 1996.

[10] R. Beaver, "Financial ratios as predictors of failure, Empirical Research in Accounting: Selected Studies, " J.Accounting Research, vol. 4, pp.71-111, 1966.

[11] T. Hofmann, B. Scholkopf and A.J. Smola, "Kernel methods in machine learning," The Annals of Statistics, vol. 36, No. 3, 1171-1220, 2008.

[12] J. Ohlson, "Financial ratios and the probabilistic prediction of bankruptcy, " J.Accounting Res., vol. 18, pp. 109-131, 1980.

[13] G. Rudorfer, "Early bankruptcy detecting using neural networks," APL Quote Quad, ACM New York, vol. 25, N. 4, P. 171-178, 1995.

[14] K. Tam and M. Kiang, "Managerial applications of the neural networks: The case of bank failure predictions," Management Science, vol. 38, P. 416-430, 1992.

[15] G. Rudorfer, "Home page – Gottfried Rudorfer," [Online]. Available: http://godefroy.sdf-eu.org/apl95/ratios95.zip [Accessed: Jan. 7, 2017].