Supervised Potentiality Actualization Learning for Improving Generalization Performance

Ryotaro Kamimura
IT Education Cente and Graduate School of Science and Technology, Tokai Univerisity
1117 Kitakaname, Hiratsuka, Kanagawa 259-1292, Japan
ryo@keyaki.cc.u-tokai.ac.jp

Abstract— *The present paper proposes an application of potentiality learning to supervised learning. The potentiality has been developed as a measure of the importance of components in the self-organizing maps (SOM) to extract important input neurons. The main characteristics lies in its simplicity and thus it can be easily implemented. If it is possible to use it for conventional supervised learning, better performance can be expected with much simpler computational method. The potentiality is defined by the variance of input neurons and it is incorporated into supervised learning. Using the potentiality inside, two data sets were used to evaluate the performance. The results show that the potentiality method outperformed ones without it and other conventional methods in terms of generalization performance.*

Keywords: Potentiality, selective potentiality, determination, actualization, generalization

1. Introduction

1.1 Potentiality and Its Actualization

Neural networks have been applied to many problems with better performance than that by the conventional statistical methods. Though the performance of neural networks has been improved, it can be said that the potentiality of components of neural networks cannot be fully explored [1] . The potentiality is considered as the implicit capability of neural networks. The potentiality can be actualized or realized in terms of a number of different forms. For example, the potentiality is realized as the properties of components which can be used to interpret network behaviors or to improve generalization performance. One of the main problems is that little attempts have been made to determine the main potentiality of components of neural networks.

In the present paper, the simple potential method is proposed with two main characteristics, namely, variance and separation. First, the potentiality is supposed to be represented in the form of variance of connection weights. The potentiality is considered to be higher when the neurons respond to input patterns as differently as possible. Second, the potentiality determination and use phase are separated. There have been many attempts to interpret network behaviors and to improve generalization [2], [3], [4], [5], [6] [7], [8], [9]. One of the main difficulties inherent to those approaches is that the errors between targets and actual outputs are minimized and simultaneously generalization performance is improved or interpretation is improved. Error minimization and performance improvement are sometimes contradictory to each other. For example, to have more interpretable networks, internal representations should be simplified as much as possible, which may degrade the performance of neural networks. To overcome those problems, a new method is proposed, where potentiality determination and actualization phase are completely separated. For example, the potentiality is determined roughly and then this potentiality is incorporated into the process of error minimization. Then, contradiction between error minimization and potentiality determination is minimized.

1.2 Relations to the Input Neuron Selection

To demonstrate the potentiality method, the method is applied to the detection of important input neurons (variables) [10], [11], [12], [13], [14] . The variable selection has played important roles in improving the performance of neural networks. In particular, in application, the interpretation of input variables is necessary. However, in this interpretation, neural networks are said to be weaker than the conventional methods such as the regression analyses. The regression analysis has been used in many practical problems, because the coefficients obtained by the regression analysis can be interpreted, though the actual generalization performance is much weaker.

To have more interpretable input variables or input neurons, the potentiality is introduced. The potentiality is defined as the capability of neurons responding to input patterns as differently as possible. Thus, the potentiality is defined as a variance of connection weights. Because the potentiality is an abstract concept, it can be actualized. In the potentiality actualization phase, the potentiality is actualized so as to represent the importance of input neurons.

1.3 Outline

Section 2 introduces the potentiality in the supervised learning. The method is composed of two phase. First, the potentiality determination phases is applied to detect the important input neurons with higher potentiality. Then, the potentiality is normalized and the corresponding connection weights are modified. Then, the final fine tuning phase is performed. In Section 3, the method was applied to the two data sets. In both sets, generalization performance was improved by the potentiality.

2. Theory and Computational Methods

2.1 Potentiality Actualization Learning

The potential actualization learning aims to determine the potentiality of neurons and actualize its potentiality as much as possible. As mentioned, in the potentiality method, the determination of the potentiality and its actual use is separated to facilitate learning. The computational procedure is composed of two phases, namely, potentiality determination and actualization phase in Figure 1. In the potentiality determination phase, the potentiality of neurons is determined by using the variance of connection weights. Then, connection weights are given into the potentiality actualization phase as initial connection ones. In addition, connection weights are weighted by the relative potentiality to take into account the importance or potential importance of input neurons as shown in Figure 1(b). Thus, in the potential actualization phase, connection weights are actually updated to take into account the potentiality and realize or actualize potentiality.

2.2 Individual Potentiality

For this, it is needed to define the potentiality of individual input neurons. The potentiality of an input neuron is defined by

$$v_k = \exp\left(R\sum_{j=1}^{M}(w_{jk} - w_k)^2\right), \tag{1}$$

where w_{jk} denote connection weights from the kth input neuron to the jth hidden neuron and

$$w_k = \frac{1}{M}\sum_{j=1}^{M} w_{jk}. \tag{2}$$

The coefficient R determines the intensity of the variation of connection weights and should be experimentally determined. The potentiality is based on the variance of input neurons toward output neurons. It is natural to suppose that when input neurons respond to output neurons with large variation, the input neurons surely play important roles. This means that the neurons with large variation have high potentiality to represent input patterns. In addition, by the exponential function, when the variation of neurons becomes larger, the expected potentiality increases exponentially or excessively. This property is needed to intensify a few number of important neurons.

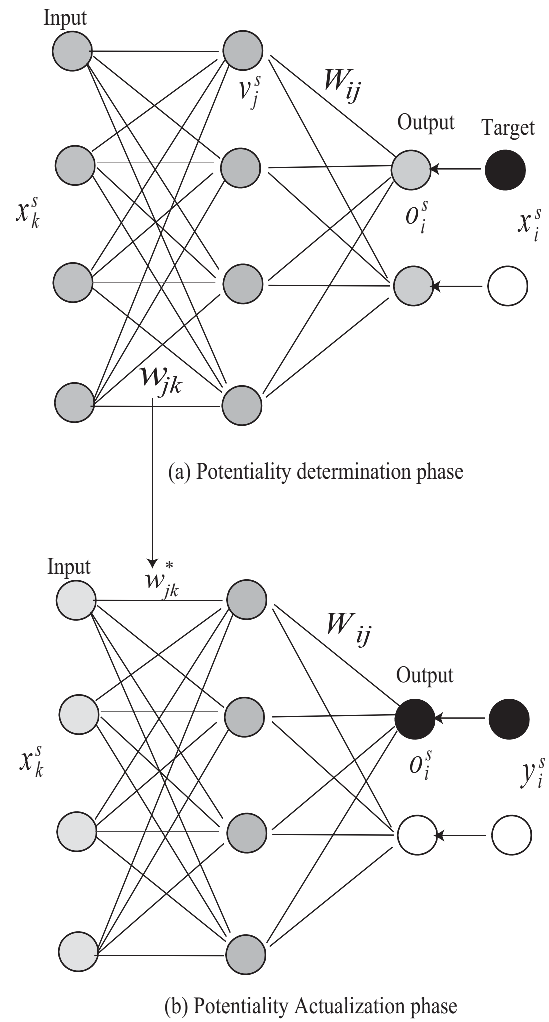

Fig. 1: Network architecture with the potentiality determination (a) and actualization (b) phase.

2.3 Selective Potentiality

The selective potentiality is defined by using the concept of information in the information-theoretic methods [15], [16], [17], [18], [19]. When the information increases in competitive learning, only one neuron finally fires, while all the other neurons cease to do so. This concept of information-theoretic competitive learning is directly translated into the potentiality. When the selective potentiality increases, finally only one neuron tend to have the maximum potentiality.

For using the information theoretic concepts, it is needed to normalize the individual potentiality

$$p(k) = \frac{v_k}{\sum_{l=1}^{L} v_l}. \tag{3}$$

The selective potentiality is defined by the decrease from maximum uncertainty to observed uncertainty

$$I = 1 + \frac{\sum_{k=1}^{L} p(k) \log p(k)}{\log L}. \quad (4)$$

When this potentiality increases, a smaller number of input neurons tend to have larger individual potentiality.

2.4 Potentiality Determination and Actualization Phase

The method is composed of the potentiality determination and actualization phase. In the determination phase, after finishing the learning, the potentiality is computed with the parameter

$$R = \frac{r}{L-1}, \quad (5)$$

where L and r denote the number of input neurons and the learning parameter.

Then, the relative potentiality is computed and with this potentiality, the potentiality actualization is initialized

$$^{new}w_{jk} = {}^{old}w_{jk} p(k). \quad (6)$$

With these connection weights, the errors between targets and outputs are minimized

$$E = \frac{1}{2S} \sum_{s=1}^{S} \sum_{i=1}^{N} (y_i^s - o_i^s)^2, \quad (7)$$

where S and N denote the number of input patterns and output neurons, and y_i^s are the targets for the outputs o_i^s[1]. This means that the potentiality is incorporated into the leaning processes as initial weights. The experiments results show that the graduate decent learning is much affected by the initial conditions and this method is simple and effective to take into account the potentiality.

3. Results and Discussion

3.1 German Credit Approval Data Set

The first data set is the German credit data set from the machine learning database. The number of input patterns was 1000 with 24 input variables [20].

3.1.1 Selective Potentiality Increase

Figure 2 shows the selective potentiality as a function of the parameter r. As shown in the figure, the selective potentiality increased gradually when the parameter r increased.

Figure 3 show the relative potentiality when the parameter r increased from 1.0(a) to 5.0(h). When the parameter r is 1.0 in Figure 3(a), the relative potentiality distributed almost uniformly. Then, when the parameter r increased from 1.2 (b) to 1.6 (d), the potentiality became gradually differentiated. Then, the parameter increased further from 2.5 (e) to 5.0 (h), several input neurons tended to have much higher relative potentialities.

[1]The hidden and output activation function were the the hyperbolic tangent sigmoid and linear one and the early stopping method was used.

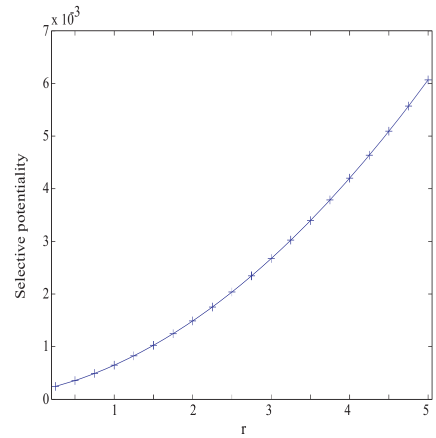

Fig. 2: Selective potentiality as a function of the parameter r for the German credit data set.

Table 1: Summary of experimental results of generalization for the German data set with ten different runs.

Methods	R	Average	Std dev	Min	Max
Potential	4.4	**0.2187**	0.0340	**0.1600**	**0.2600**
Early stopping		0.2367	**0.0273**	0.1867	0.2667
SVM		0.2613	0.0344	0.2067	0.3133
Logistic		0.2313	0.0308	0.1733	0.2733

3.1.2 Generalization Performance

Figure 4(a) shows generalization errors as a function of the parameter r. When the parameter r increased or the selective potentiality increased, the generalization errors tended to decrease and seem to reach the stable states. Figure 4(b) shows the standard deviation of the generalization errors. One of the main characteristics is that the standard deviation increased when the parameter increased. This means that the generalization errors fluctuated when the parameter R increased.

3.1.3 Summary of Results

Table 1 shows the summary of experimental results related to the generalization performance. In the table, the values in bold faces show the minimum values. As can be seen in the table, except the standard deviation, the potential method shows the best performance with the minimum values in the average, minimum and maximum values. On the other hand, the standard deviation was the largest by the potential method. As pointed out in the previous section, the standard deviation tended to be larger by the potentiality method.

Experimental results confirm that generalization performance was improved by increasing the potentiality but the errors tended to fluctuate for the larger parameter values.

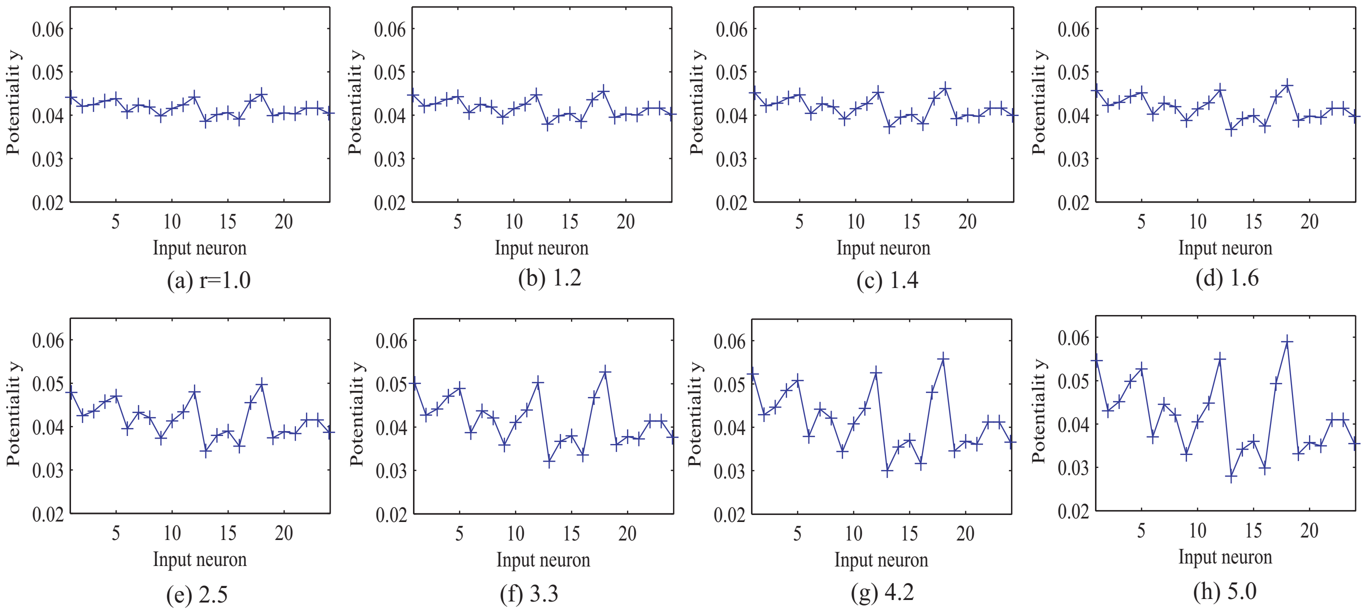

Fig. 3: Potentiality $p(k)$ of input neurons for four input neurons for the German credit data set.

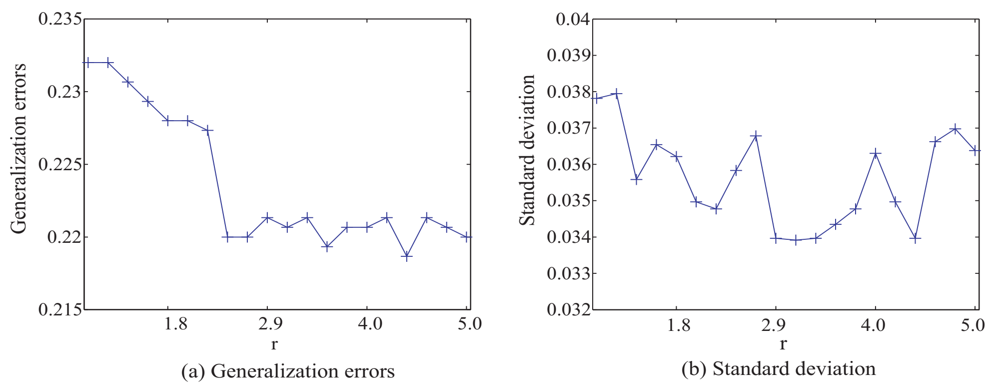

Fig. 4: Generalization errors (a) and the standard deviation of the errors (b) by the potentiality method for the German credit data set.

3.2 Biodegradation Data Set

The second data set is also from the machine learning data set where 41 attributes and 1055 patterns, which must be classified into 2 classes (ready and not ready biodegradable) [20].

3.2.1 Selective Potentiality

Figure 5 shows the selective potentiality as a function of the parameter r. The selective potentiality increased gradually when the parameter r increased.

Figure 6 shows the relative potentiality when the parameter r increased from 1.0(a) to 5.0(h). When the parameter r was 1.0 in Figure 6(a), the potentiality was almost uniform. Then, when the parameter r increased gradually, several potentialities became larger. Finally, when the parameter r was 5.0 in Figure 6(h), some potentialities were clearly differentiated.

3.2.2 Generalization Performance

Figure 7(a) shows generalization errors as a function of the parameter r. The generalization errors decreased for the smaller values of the parameter and then fluctuated. Figure 7(b) shows the standard deviation of the generalization errors. As can be seen in the figure, the standard deviation decreased gradually for the smaller parameter values. Then, the standard deviation became larger when the values became larger.

These results seem to suggest that the potentiality is not related to the improved generalization performance as shown in Figure 7(a) and 5. This can be explained by seeing the standard deviation of generalization errors. When the parameter R increased, the generalization errors fluctuated

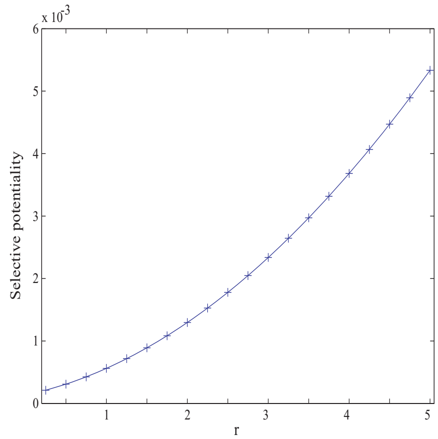

Fig. 5: Selective potentiality as a function of the parameter r for the bio-degeneration data set.

when the parameter R was larger in Figure 5(a). However, the standard deviation in Figure 7(b) greatly fluctuated when the parameter r became larger. This large standard deviation surely affected the overall generalization performance.

3.2.3 Summary of Results

Table 2 shows the summary of experimental results. The potentiality method showed the best performance in terms of the average and maximum errors. On the other hand, for the minimum errors, the logistic regression method showed the best performance and the potentiality method showed the second best performance. The potentiality method had the second largest standard deviation.

The experimental results also show that the present method of potentiality is good at improving generalization performance. The good performance is explained by two points, namely, the effectiveness of potentiality and separation of two phases. First, the potentiality as the variance of input neurons is effective in improving the generalization performance. When neurons respond to input patterns as differently as possible, the neurons play very important roles in learning. For example, naturally, neurons, responding only uniformly to input patterns, are considered to be unimportant. Second, in the method, the potentiality determination and use phase were separated. Only when the potentiality is determined, it is used in learning. This separation contributes to the improved performance.

4. Conclusion

The present paper proposes a new type of learning called "potentiality actualization learning". The potentiality implies the potentiality of input neurons, which is supposed to be realized in many different forms. In this paper, the potentiality is represented in terms of the variance of input neurons. The learning is conducted to realize this potentiality

Table 2: Summary of experimental results of generalization for the bio-degeneration data set with ten different runs. The logistic function was used to normalize the data.

Methods	R	Average	Std dev	Min	Max
Potential	3.7	**0.1184**	0.0221	0.0696	**0.1456**
Early stopping		0.1215	0.0206	0.0823	**0.1456**
SVM		0.1234	**0.0192**	0.0886	0.1519
Logistic		0.1316	0.0303	**0.0633**	0.1646

of input neurons. The potentiality actualization learning is composed of two phases. In the first phase of potentiality determination, the potentiality is determined. In the second phase of the potentiality actualization, the learning is conducted, incorporating the information on the potentiality.

The method was applied to two data sets, namely, German credit approval data set and bio-degradation data set. In both cases, the potentiality could be increased by changing the parameter. In addition, generalization performance was improved. Comparing with those by the other conventional methods like the SVM, performance was better. However, the standard deviation of the generalization errors tended to be larger than that by the other methods. If it is possible to reduce this large standard deviation by some methods, the generalization by the present method can be more improved. Thus, it is needed to develop a method to stabilize learning processes for the potentiality actualization learning.

References

[1] R. Kamimura and R. Kitajima, "Selective potentiality maximization for input neuron selection in self-organizing maps," in *Proceedings of IJCNN 2015 (to appear)*, IEEE, 2015.
[2] L. I. Nord and S. P. Jacobsson, "A novel method for examination of the variable contribution to computational neural network models," *Chemometrics and Intelligent Laboratory Systems*, vol. 44, pp. 153–160, 1998.
[3] A. Micheli, A. Sperduti, and A. Starita, "Analysis of the internal representations developed by neural networks for structures applied to quantitative structure-activity relationship studies of benzodiazepines," *Journal of Chemical Information and Computer Sciences*, vol. 41, pp. 202–218, 2001.
[4] G. G. Towell and J. W. Shavlik, "Extracting refined rules from knowledge-based neural networks," *Machine learning*, vol. 13, pp. 71–101, 1993.
[5] M. Ishikawa, "Structural learning with forgetting," *Neural Networks*, vol. 9, no. 3, pp. 509–521, 1996.
[6] M. Ishikawa, "Rule extraction by successive regularization," *Neural Networks*, vol. 13, pp. 1171–1183, 2000.
[7] P. Howes and N. Crook, "Using input parameter influences to support the decisions of feedforward neural networks," *Neurocomputing*, vol. 24, p. 1999, 1999.
[8] R. Feraud and F. Clerot, "A methodology to explain neural network classification," *Neural Networks*, vol. 15, pp. 237–246, 2002.
[9] R. Setiono, W. K. Leow, and J. M. Zurada, "Extracting of rules from artificial neural networks for nonlinear regression," *IEEE Transactions on Neural Networks*, vol. 13, no. 3, pp. 564–577, 2002.
[10] I. Guyon and A. Elisseeff, "An introduction to variable and feature selection," *Journal of Machine Learning Research*, vol. 3, pp. 1157–1182, 2003.

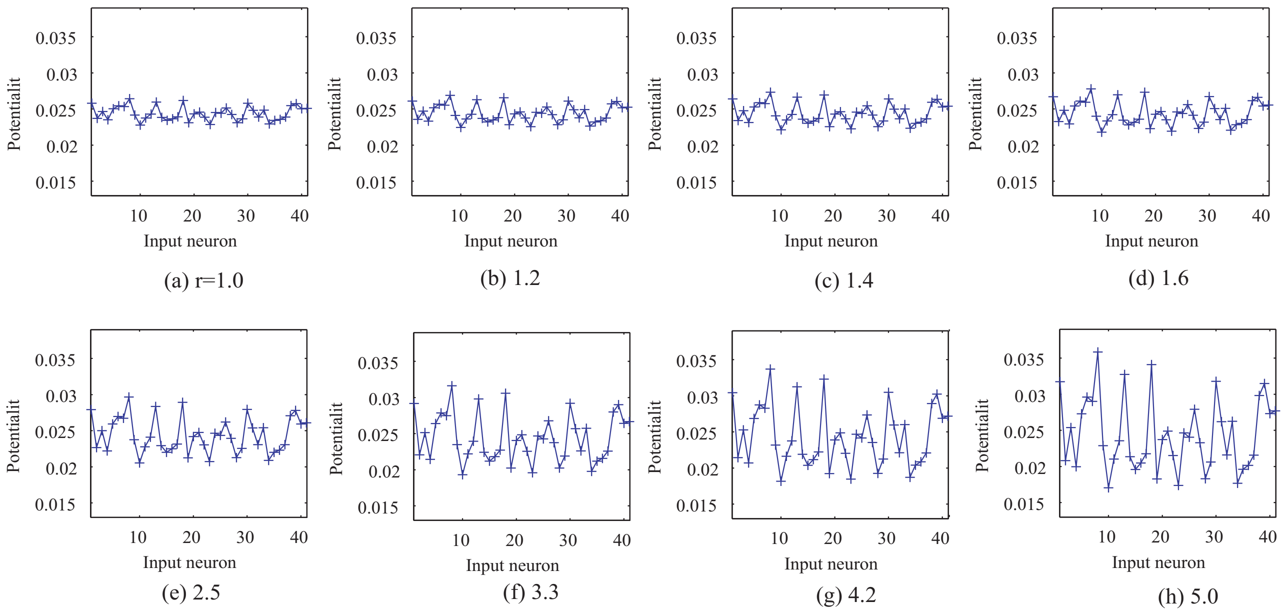

Fig. 6: Potentiality $p(k)$ of input neurons for four input neurons for the bio-degeneration data set.

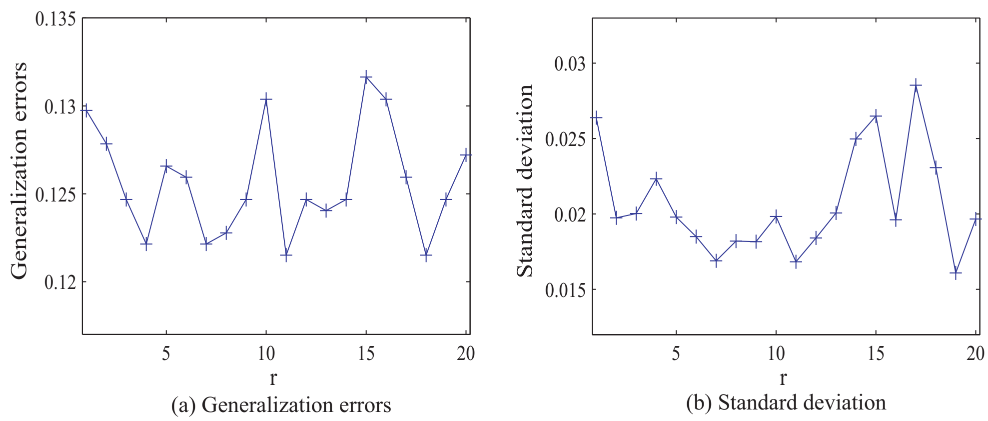

Fig. 7: Generalization errors (a) and the standard deviation of the errors (b), by the potentiality method for the bio-degeneration data set.

[11] A. Rakotomamonjy, "Variable selection using SVM-based criteria," *Journal of Machine Learning Research*, vol. 3, pp. 1357–1370, 2003.

[12] S. Perkins, K. Lacker, and J. Theiler, "Grafting: Fast, incremental feature selection by gradient descent in function space," *Journal of Machine Learning Research*, vol. 3, pp. 1333–1356, 2003.

[13] J. Reunanen, "Overfitting in making comparison between variable selection methods," *Journal of Machine Learning Research*, vol. 3, pp. 1371–1382, 2003.

[14] G. Castellano and A. M. Fanelli, "Variable selection using neural-network models," *Neurocomputing*, vol. 31, pp. 1–13, 1999.

[15] R. Kamimura and T. Kamimura, "Structural information and linguistic rule extraction," in *Proceedings of ICONIP-2000*, pp. 720–726, 2000.

[16] R. Kamimura, "Information-theoretic competitive learning with inverse Euclidean distance output units," *Neural Processing Letters*, vol. 18, pp. 163–184, 2003.

[17] R. Kamimura, "Teacher-directed learning: information-theoretic competitive learning in supervised multi-layered networks," *Connection Science*, vol. 15, pp. 117–140, 2003.

[18] R. Kamimura, "Progressive feature extraction by greedy network-growing algorithm," *Complex Systems*, vol. 14, no. 2, pp. 127–153, 2003.

[19] R. Kamimura, "Information theoretic competitive learning in self-adaptive multi-layered networks," *Connection Science*, vol. 13, no. 4, pp. 323–347, 2003.

[20] K. Bache and M. Lichman, "UCI machine learning repository," 2013.