Artificial Intelligence Based Pattern Recognition

Dr. N.V.N. Indra Kiran[1*] , B. Pradeep Kumar [2] & S. Phani Kumar[3]

[1]Professor, Department of Mechanical Engineering, ANITS, Visakhapatnam, Andhra Pradesh, India.
*Corresponding Author E-mail: *indrakiran.me@anits.edu.in*

[2,3]Assistant Professor, Department of Mechanical Engineering, ANITS, Visakhapatnam, Andhra Pradesh, India
[2]E-mail: *pradeep.me@anits.edu.in*

Abstract: Control charts pattern recognition is one of the most important tools in statistical process control to identify process problems. Unnatural patterns exhibited by such charts can be associated with certain assignable causes affecting the process. Most of the previous work in intelligent process control used raw data as input vector representation. The objective of this study was to evaluate the relative performance of a feature-based optimized Recognizer compared with the raw data-based optimized recognizer. The study focused on recognition of seven commonly researched patterns plotted on the X-bar chart.

Keywords: Statistical Process Control, Optimized Recognizer, X-Bar Chart

I. INTRODUCTION

There are seven basic CCPs, e.g. normal (NOR), systematic (SYS), cyclic (CYC), increasing trend (IT), decreasing trend (DT), upward shift (US) and downward shift (DS) [1-6]. All other patterns are either special forms of basic CCPs or mixed forms of two or more basic CCPs. Only the NOR pattern is indicative of a process continuing to operate under controlled condition [1-6]. All other CCPs are unnatural and associated with impending problems requiring pre-emptive actions. Advances in manufacturing and measurement technology have enabled real-time, rapid and integrated gauging and measurement of process and product quality. ANN learns to recognize patterns directly through a typical sample patterns during a training phase. Neural nets may provide required abilities to replace the human operator. Neural network also can identify an arbitrary Pattern not previously encountered. Back propagation network (BPN) has been widely used to recognize different abnormal patterns of a control chart [1, 2, 7-10]. BPN is a supervised-learning network and its output value is continuous, usually between [0, 1]. It is usually used for detecting, forecasting and classification tasks, and is one of the most commonly used networks [3].

II. PATTERN RECOGNIZER DESIGN

a) Sample patterns

Sample patterns should be collected from a real manufacturing process. Since, many patterns are required for developing and validating a CCP recognizer, and as those are not economically available, simulated data are often used. Since a large window size can decrease the recognition efficiency by increasing the time required to detect the patterns, an observation window with 32 data points is considered here. The values of different parameters for the unnatural patterns are randomly varied in a uniform manner. A set of 3500 (500x7) sample patterns are generated from 500 series of standard normal variates. It may be noted that each set contains equal number of samples for each pattern class. This is so done because if a pattern class is trained more number of times, the neural network will become biased towards that pattern. The equations used for simulating the seven CCPs are given in appendix.

b) Sample patterns for Statistical features

The choice of statistical features to be extracted from the raw data to be presented as the input vector into the recognizer is very important. The presence of too many input features can burden the training process and lead to inefficient recognizers. Features low in information content or redundant should be eliminated whenever possible. Redundant here refers to features with marginal contribution given that other features are present [5, 8, 9, 11].

Based on the pair wise correlation coefficients median, mode, range, skewness, kurtosis and standard deviation are selected and mean, moment and mean absolute deviation are omitted.

c) Training algorithms

It is very difficult to know which training algorithm will be the fastest for a given problem. It depends on many factors, including the complexity of the problem, the number of data points in the training set, the number of weights and biases in the network, the error goal, and this section compares the various training algorithms.

Backpropagation algorithm uses the gradient of the performance function to determine how to adjust the weights to minimize performance. In backpropagation, the gradient is determined by performing computations backwards through the network [3]. There are many variations of backpropagation, some of them provide faster convergence while others give smaller memory requirement. In this study five training algorithms are evaluated they are gradient descent algorithm (traindx) and resilient backpropagation (trainrp), Conjugate Gradient Algorithm (trainscg), Quasi-Newton Algorithm (trainbfg) and Levenberg-Marquardt (trainlm) [4]. The variable learning rate algorithm traindx is usually much slower than the other methods, and has about the same storage requirements as trainrp, but it can still be useful for some problems [5].

d) Neural Network Configuration

The recognizer was developed based on multilayer perceptions (MLPs) architecture; its structure comprises an input layer, one or more hidden layer(s) and an output layer. Figure 1 shows an MLP neural network structure comprising these layers and their respective weight connections. Before this recognizer can be put into application, it needs to be trained and tested. In the supervised training approach, sets of training data comprising input and target vectors are presented to the MLP. The learning process takes place through adjustment of weight connections between the input and hidden layers and between the hidden and output layers. These weight connections are adjusted according to the specified performance and learning functions. The input node size was equal to the size of the observation window, i.e. 32 when using raw data based recognizers and six in case of statistical features based recognizer. The number of output nodes in this study was set corresponding to the number of pattern classes, i.e. seven. The labels, shown in table 1, are the targeted values for the recognizers' output nodes. The maximum value in each row (0.9) identifies the corresponding node expected to secure the highest output for a pattern to be considered correctly classified.

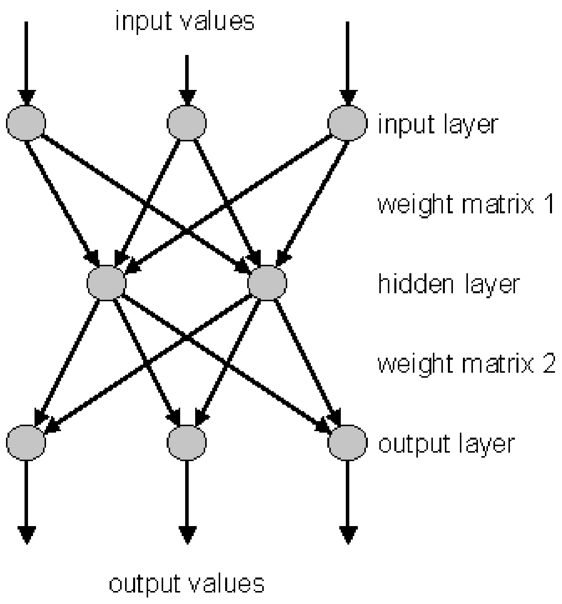


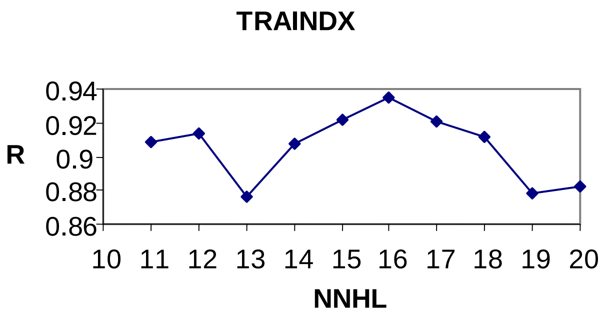
Figure 1. MLP Neural Network Architect

The general rule is that the network size should be as small as possible to allow efficient computation. The number of nodes in the hidden layer is selected based on the results of many experiments conducted by varying the number of nodes from 11 to 20. All those experiments are coded in MATLAB® using its ANN toolbox [4]. The transfer functions used are hyperbolic tangent (tansig) for the hidden layer and sigmoid (logsig) for the output layer. The hyperbolic tangent function transforms the layer inputs to output range from −1 to +1 and the sigmoid function transforms the layer inputs to output range from 0 to 1 [12]. Coefficient of correlation performance of the neural network for the traindx is the maximum when the number of nodes in the hidden layer is 16 and 12 for raw data and statistical features as input respectively and it is shown in figure 2 & 3. The selected ANN architecture is given below. The general rule is that the network size should be as small as possible to allow efficient computation. The number of nodes in the hidden layer is selected based on the results of many experiments conducted by varying the number of nodes from 11 to 20. All those experiments are coded in MATLAB® using its ANN toolbox [4]. The transfer functions used are hyperbolic tangent (tansig) for the hidden layer and sigmoid (logsig) for the output layer. The hyperbolic tangent function transforms the layer inputs to output range from −1 to +1 and the sigmoid function transforms the layer inputs to output range from 0 to 1 [12]. Coefficient of correlation performance of the neural network for the traindx is the maximum when the number of nodes in the hidden layer is 16 and 12 for raw data and statistical features as input respectively and it is shown in figure 2 & 3. The selected ANN architecture is given below.

1. ANN Configuration for Raw Data

Network details: traindx

Architecture: 32-16-7 network, with tansig TF and logsig TF in hidden and output layer respectively. Training: traindx algorithm



NNHL: number of neurons in hidden layer

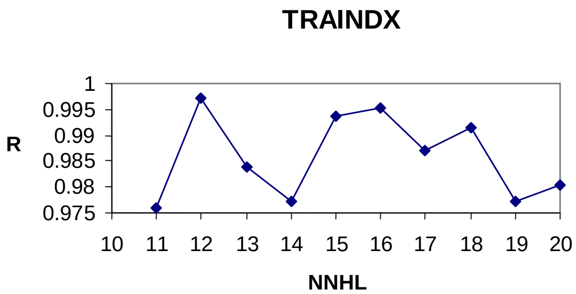
Figure 2. NNHL VS R

2. ANN Configuration for Statistical features

Network details: traindx

Architecture: 6-12-7 network, with tansig TF and logsig TF in hidden and output layer respectively. Training: traindx algorithm

Examples, but it has not learned to generalize to new situations for improving generalization early stopping is used and it is implemented in Neural Network Toolbox software [4].



NNHL: number of neurons in hidden layer

Figure 3. NNHL VS R

ANN recognizers were developed using raw data and statistical features as the input vector. This section discusses the procedures for the training and recall (recognition) phases of the recognizers. The recognition task was limited to the seven previously mentioned common SPC chart patterns. All the procedures were coded in MATLAB using its ANN toolbox [4].

It was noted during training that feature-based recognizers were more easily trained. The recognition accuracy, between actual targets and predicted targets are higher for traindx algorithm in case of raw feature based recognizers.

III. CONCLUSIONS

The objective of this study was to evaluate the relative performance of training algorithms with the optimum structure for raw data and statistical features based optimized CCP recognizer. Feature based recognizers achieved a statistically significant improvement in recognition performance. Further, the use of the statistical feature set required less training effort and resulted in better recall performance. The MLP neural network was used as a generic recognizer to classify seven different types of SPC chart patterns. In this study five training algorithms are studied and traindx is identified to be the best algorithm for this problem.

Conflict of interest: The authors declare that they have no conflict of interest.

Ethical statement: The authors declare that they have followed ethical responsibilities.

REFERENCES

[1]. Amin, A., 2000, Recognition of printed Arabic text based on global features and decision tree learning techniques. Pattern Recognition, 33, 1309–1323.

[2]. Anagun, A. S., 1998, A neural network applied to pattern recognition in statistical process control. Computers Industrial Engineering, 35, 185–188.

[3]. B. Yegananarayana, "Artificial Neural Network". Prentice-Hall India

[4]. Demuth,H. and Beale, M., 1998, Neural Network Toolbox User's Guide (Natick, MA: Math Works)

[5]. Hassan, A., Nabi Baksh, M. S., Shaharoun, A. M., & Jamaluddin, H. (2003). Improved SPC chart pattern recognition using statistical features. International Journal of Production Research, 41(7), 1587–1603.

[6]. Montgomery, D. C., 2001, Introduction to Statistical Quality Control, 4th Ed. (New York: Wiley).

[7]. N.V.N.Indra Kiran, M.Pramila Devi, G.Vijaya Lakshmi (2010). Effective control chart pattern recognition using artificial neural networks. IJCSNS,10(3),194-199.

[8]. Pham, D. T., & Oztemel, E. (1992). Control chart pattern recognition using neural networks. Journal of System Engineering, 2,256–262.

[9]. Pham, D. T., & Wani, M. A. (1997). Feature-based control chart pattern recognition. International Journal of Production Research, 35(7),1875–1890.

[10]. Pham, D. T., & Sagiroglu, S. (2001). Training multilayered perceptrons for pattern recognition: a comparative study of four training algorithms. International Journal of Machine Tools and Manufacture, 41, 419–430.

[11]. Battiti, R., 1994, Using mutual information for selecting features in supervised neural net learning. IEEE Transactions on Neural Networks, 5, 537–550.

[12]. Smith M (1993) Neural networks for statistical modeling. Van Norstrand Reinhold, New York.

APPENDIX

The following equations are used to generate different patterns for the training and testing data sets:

Normal pattern $yi = \mu + ri\, \sigma$

Systematic patterns $yi = \mu + ri\, \sigma + d \times (-1)i$

Increasing or decreasing trend $yi = \mu + ri\, \sigma \pm ig$

Upward or downward shift $yi = \mu + ri\, \sigma \pm ks$

Cyclic patterns $yi = \mu + ri\, \sigma + a \sin(2\pi i/T)$

Where, i is the discrete time point at which the pattern is sampled ($i = 1, \ldots, 32$), k is 1 if $i \geq P$ (point of shift); otherwise $k = 0$, ri is the random value of a standard normal variate at i th time point and yi is the sample value at i th time point.

