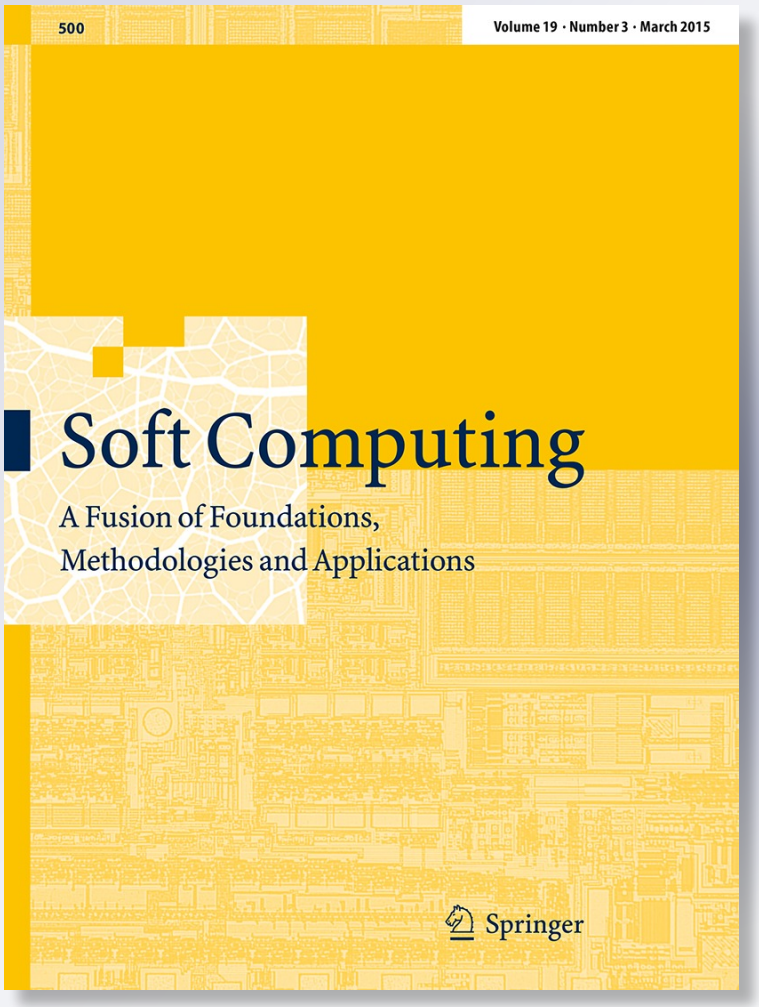
A probabilistic artificial neural network-based procedure for variance change point estimation

Amirhossein Amiri, S. T. A. Niaki & Alireza Taheri Moghadam

Soft Computing
A Fusion of Foundations, Methodologies and Applications

ISSN 1432-7643
Volume 19
Number 3

Soft Comput (2015) 19:691-700
DOI 10.1007/s00500-014-1293-x



Springer

Soft Comput (2015) 19:691–700
DOI 10.1007/s00500-014-1293-x

METHODOLOGIES AND APPLICATION

A probabilistic artificial neural network-based procedure for variance change point estimation

Amirhossein Amiri · S. T. A. Niaki · Alireza Taheri Moghadam

Published online: 8 May 2014

Abstract Control charts are useful tools of monitoring quality characteristics. One of the problems of employing a control chart is that the time it alarms is not synchronic with the time when assignable cause manifests itself in the process. This makes difficult to search and find assignable causes. Knowing the real time of manifestation of assignable cause (change point) helps to find assignable cause(s) sooner and eases corrective actions to be taken. In this paper, a probabilistic neural network (PNN)-based procedure was developed to estimate the variance change point of a normally distributed quality characteristic. The PNN was selected based on trial and error among different types of artificial neural networks and on the basis of its advantages such as fast training process, converging to optimal classifier and adding or removing samples without extensive retraining. In the proposed procedure, the signal is first received by an S^2 control chart and then based on the designed tests of hypothesis, which distinguish the size of shift in the variance, a suitable PNN is activated. The performance of the proposed procedure is evaluated through extensive simulation studies. In addition, the results of a comparison study with themaximum likelihood estimation (MLE) method show that the proposed procedure outperforms MLE in estimating the real time of the step change in variance of a normal quality characteristic. Finally, an illustrative example is presented to clarify the procedure step by step.

Keywords Change point · Variance change point · Probabilistic artificial neural network · Statistical process control

1 Introduction

Statistical process control (SPC) methods are used to monitor quality characteristics of a process over time. When a control chart alarms a signal, indicating the presence of an assignable cause, a search to identify the cause is started and then corrective actions are performed to remove it. The problem is that often the time of manifestation of assignable causes is not synchronic with the time, an out-of-control signal is observed. Estimating the real time of the process change (called a change point) certainly helps to find the source(s) of variation easier, because knowing the time of manifestation of assignable cause limits the search range over the operator, material, instrument, and the like. Therefore, one can remove the root cause of the problem easier and the production process will return to its in-control state sooner.

The change-point estimation problem has been investigated in depth by many researchers, where several methodologies have been proposed in the literature. These research works can be classified based on both the type of process change and the approach taken for estimation.

The type of process change considered in the literature includes single step, multiple step, linear trend, monotonic change, and sporadic change (Amiri and Allahyari 2012).

Communicated by V. Loia.

A. Amiri (✉) · A. Taheri Moghadam
Department of Industrial Engineering, Shahed University, Tehran, Iran
e-mail: amirhossein.amiri@gmail.com; amiri@shahed.ac.ir

A. Taheri Moghadam
e-mail: a.taherimoghadam@shahed.ac.ir

S. T. A. Niaki
Department of Industrial Engineering, Sharif University of Technology, Tehran, Iran
e-mail: niaki@sharif.edu

The single-step change occurs in an unknown time and its effect remains until the corrective action is taken. Since many real-world processes are subject to this type of change, most of the change point estimation research works such as Pignatiello and Samuel (2001), Park and Park (2004), Perry and Pignatiello (2008), and Niaki and Khedmati (2012) have been performed in this class.

John (2000), Li et al. (2006), and Perry (2010) proposed methodologies to estimate multiple step change points, in which several step changes occur in different times of the manufacturing process under consideration.

In the linear trend change, where the process starts to change with a linear trend in an unknown time, Perry et al. (2006), Zandi et al. (2011) and Amiri and Khosravi (2012) developed different estimation methods.

The monotonic change includes multiple step and trend changes. This type of change has been considered by Perry et al. (2006, 2007), Noorossana and Shadman (2009) and Amiri and Khosravi (2013). Finally, sporadic change is a type of change in which the process changes in indefinite order.

The change-point estimation approach can be classified into four classes of maximum likelihood estimator (MLE), cumulative sum (CUSUM), exponentially weighted moving average (EWMA), and artificial neural network (ANN).

MLE is the most popular approach among the ones mentioned above. CUSUM and EWMA are control charts that can be used to identify the change points as well. Although these two control charts are primarily used for detection, they do have "built-in" change point estimators that are effective in estimating the process change point following control chart signals (Amiri and Allahyari 2012). ANN also can be used to estimate process change points and some researchers such as Ahmadzadeh (2011), Atashgar and Noorossana (2011) employed it to estimate change point of the mean vector in a bivariate normal process.

Several researchers in the literature attacked the variance change-point estimation problem. Park and Park (2004) investigated the performances of an MLE approach to estimate the real time of a single-step shift in variance and mean of a process monitored by the $(\bar{X}, S)$ control chart. Samuel et al. (1998) proposed an MLE approach to estimate the time of step shift in variance of a process monitored by the S control chart. Zhao et al. (2010) applied a ratio test to find the time of a single-step shift in the process variance. Hawkins et al. (2003) used MLE, CUSUM, and EWMA approaches to estimate the change point of a single-step shift in the process variance and mean, simultaneously. Lee and Park (2007) developed an MLE approach to estimate the single-step variance shift of processes monitored by variable sample rate (VSR) Shewhart, CUSUM, and EWMA control charts. Oh et al. (2005) proposed a non-parametric and data adaptive approach for variance change point (VCP) detection in a time series model. They used back-propagation neural network (BPN) as a pattern classification tool for change point detection when the observations are autocorrelated. The inputs of the network are differences between successive observations, moving average and moving variance statistics under period 5, 20, and 60. An output neuron is a decimal number in the range of [1, 3]. The output neuron classifies data into three categories including "1" low volatility period (in-control condition), "2" transition period (VCP) and "3" high volatility period (out-of-control condition). They considered VCP detection in a time series model as a pattern classification problem.

In this paper, a new approach is proposed to estimate the real time of a step change in the variance parameter of a univariate normal process. It is assumed that the shift is single step and that the observations are independent. Also, an S^2 control chart is used to monitor the variance of a univariate normal process. In addition, we specifically concentrate on Phase-II monitoring and assume that the parameters of the normal process, (μ, σ^2), are either known or have been estimated in Phase-I.

We utilize a procedure based on probabilistic neural networks (PNN) for VCP estimation after getting a signal from the S^2 control chart. Probabilistic neural networks are a special type of radial basis network suitable for classification problems. As it is explained in next section, the application of the proposed neural network-based procedure is easy. Because there is no need for extra calculation, except hypothesis tests, for selecting an appropriate network. Inputs of the selected network are sample variances, which are the statistics plotted on the S^2 control chart. Besides, PNNs can be used for VCP estimation in any normal processes, because the observations are first standardized. In other words, our trained neural network can be easily used for the VCP estimation of any normal process without any retraining process. Meanwhile, computational results show that neural network provides more accurate estimates than the ones obtained by the popular MLE approach under medium and large shifts.

The structure of the paper is as follows: In Sect. 2, the proposed ANN-based procedure as well as its training algorithm is explained. In Sect. 3, the precision of the proposed procedure is evaluated through computational experiments. A comparison of the proposed procedure with the MLE method of Samuel et al. (1998) is also performed in this section. In Sect. 4, an illustrative example is used to clarify the proposed approach. Our concluding and suggestions for future research are given in the final section.

2 Artificial neural network

Artificial neural network is an approach in information processing that does not require specific algorithm or rule development. The three essential features of a neural com-

puting network include the computing units, the connections between the computing units, and the training algorithm used to find values of the network parameters.

Different types of ANNs such as feed forward (FF), BPN network, and PNN with different architectures (number of hidden layers, neurons of each layer, transfer function, learning function, training process, inputs and outputs) have been tested in this research. We were not be able to obtain acceptable results using either FF or BPN networks under different architectures. However, we were able to achieve accurate and precise results using PNN. Hence, based on the initial advantages (as follows) and the advantages of the PNN in comparison with FF and BPN, we selected PNN among the mentioned networks and used it in the proposed procedure to estimate the real time of the step change point in variance parameter of a normally distributed quality characteristic. The training process of PNNs is much faster than the ones for other networks, especially BPNs. Moreover, PNNs possess inherent parallel structure. In addition, as the size of the representative training set increases, PNNs guarantee to converge to an optimal classifier. Finally, the training samples can be added or removed without extensive retraining. Despite many advantages of the PNN, one of the drawbacks of this neural network is that it requires exactly a representative sample from the dataset as training set.

The first layer of the PNN computes the distance from the input vector to the training input vectors. The results are stored in a vector where its elements indicate how close the input is to the training input. The second layer produces its net output as a vector of probabilities. Finally, a transfer function on the output of the second layer picks the maximum of these probabilities and produces a 1 which indicates VCP and 0 for other samples.

In this paper, several neural networks such as FF, BPN, and PNN with various architectures are examined for the VCP estimation problem and the results pointed out that a two-layer PNN can perform better than the other structures. The PNN of this research is a two-layer network, where the first layer has a radial-basis transfer function and calculates its weighted inputs with Euclidean distance weight function. The second layer has a competitive transfer function and calculates its weighted input using a dot product weight function. Its net input includes the sum net input function. Only the first layer involves biases.

2.1 Training PNN

Since the proposed model is non-parametric, we have to standardize datasets first to ensure that the networks perform well for all normally distributed datasets. We have tested different statistics as the input and output of the network. The results show that S^2 statistic is the best input of the networks. Also, the S^2 control chart that is the most popular control chart for monitoring the variance of a univariate normal process is used for detecting the out-of-control state.

Initial data set is first generated to train the PNN. If the networks are trained by small number of initial data, they will not perform properly. Also, if a large number of initial data are chosen, the networks will need much time and memory to estimate VCP. We suggest 5,000 initial data for training phase of the networks. This amount of data will train the networks accurate and fast. We generate 5,000 initial data using both in-control and out-of-control normal processes under different shift sizes. Then, S^2 control chart with different sample sizes (between 5 and 20) is employed for monitoring purposes. After getting a signal by the S^2 control chart, the PNN is used to estimate the time of change in process variance (VCP). The statistic and the corresponding control limits of S^2 control chart in Phase II are given in Eqs. (1)–(3), respectively.

$$S^2 = \frac{\sum_{i=1}^{n}(x_i - \mu)^2}{n}, \tag{1}$$

$$\mathrm{UCL} = \frac{\chi^2_{\alpha/2,n}}{n}\sigma^2, \tag{2}$$

$$\mathrm{LCL} = \frac{\chi^2_{1-\alpha/2,n}}{n}\sigma^2, \tag{3}$$

where x_i represents the ith observation, n denotes the sample size, and $\chi^2_{\alpha/2,n}$ is the upper $\alpha/2$ percentile point of a χ^2 distribution with n degrees of freedom. In addition, the Type-I error probability is assumed to be $\alpha = 0.05$. Hence, the in-control average run length of the chart (ARL_0) is 20.

As mentioned previously, PNNs require a representative training set. Therefore, a very well-representative training set is needed for training phase. Since the range of change in variance can be so extensive, we should separate the range to smaller ranges. This approach provides more representative training sets. Through some experiments, it was found that, few parallel networks will perform much better than a single network. Hence, four PNNs (A, B, C, and D) are proposed in parallel to estimate VCPs based on shift sizes of $[1.0\sigma–1.2\sigma]$, $[1.2\sigma–1.3\sigma]$, $[1.3\sigma–1.4\sigma]$, and $[1.4\sigma–\infty]$, respectively. For the shift size of $[1.0\sigma–1.2\sigma]$, the number of input neurons is set at 80, because the probability of exceeding run length over 80 under the smallest shift 1.0σ is very low (0.016). The numbers of input neurons for the other three PNNs are set at 40, 15, and 10, respectively. Moreover, all PNNs share a unique number of 80 output neurons. Note that in the simulation studies, we intend to achieve the result for which in almost 100.00 percent of the time the run length is less than the threshold considered. These thresholds are obtained using 30,000 replications.

The inputs of each PNN are the sample variance (S^2), the output being a vector containing one and zero elements, where the elements corresponding to the change point are one and the other elements are zeros. For instance, if 50th

neuron of the output layer is one, it denotes that the last $80-50+1=31$ samples are out-of-control and the control chart has alarmed after 31 samples. The inputs, the outputs, and the change point are illustrated in Fig. 1.

As mentioned former, in all PNNs designed, all x values are first standardized using Eq. (4) and then are used to calculate the sample variance (S^2), (x_i may come from either in or out-of-control processes).

$$z_i = \frac{x_i - \mu_0}{\sigma_0}, \tag{4}$$

where z_i is the normalized value of x_i, and μ_0 and σ_0 are the assumed known mean and standard deviation of the process.

Because of the randomness of the process, the time of an out-of control alarm is unknown. In addition, the number of input and output neurons of the networks is assumed to be fixed. Hence, specific number of datasets should be inserted as input neurons, which contain both in-control and out-of-control data. We first generate out-of-control dataset with random shifts until the S^2 control chart signals to handle the randomness of the process. Then, we enter the dataset (S^2 statistic) to the last neurons as inputs. Thereafter, we generate in-control data and compute the corresponding S^2 statistics, which are the inputs of the initial neurons.

As indicated previously, the output neuron vector contains 80 elements of all zero values, except the one that indicates the starting time of disturbance taking the value of one. For example, if the S^2 control chart alarms a signal after 10 samples from the change point, the 10th neuron from the last should get the value of one and the other ones should be zero. In other words, if we have 70 in-control data, the 71th output neuron should be one and the others should be

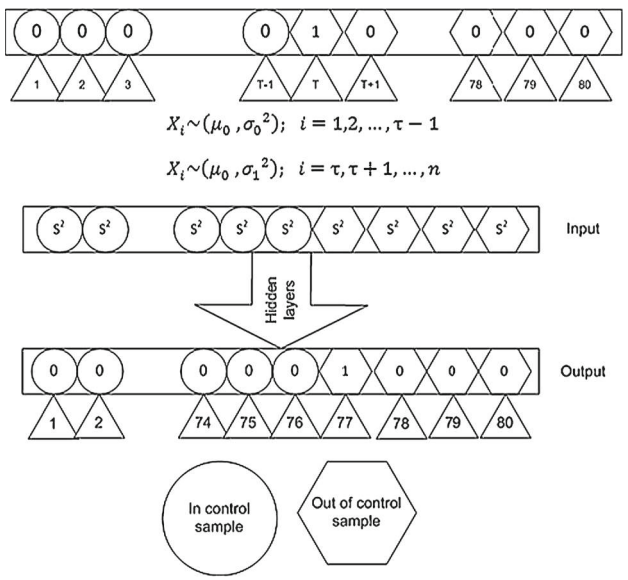


Fig. 1 PNN input and output

zero. As it is mentioned before, the probability of exceeding run length over 80 under the smallest shift 1.0σ is very low (0.016). Hence, 80 input neurons for network A are convenient. Similarly, we define the number of input neurons for other networks. Note that for networks B, C and D, in almost 100 percent of the time the run length is less than the number of input neurons. Table 1 shows the number of input and output neurons of the networks. Figure 2 shows how PNN can be used to estimate VCP. For VCP estimation, a network should be activated from four candidate networks. Since there is no idea about magnitude of the variance step change, the hypothesis test is suggested for choosing a network. Obviously just the last sample is fallen out of the control limits because the control chart has signaled first at the last sample. Hence, we just test the last sample for choosing a network. P value of the test related to network A is always less than the other P values, because this test consists of the other tests. Similarly P value of the test related to network B is always less than the P value of the tests related to networks C and D. Hence, if we choose the lowest P value, network A will always be chosen. Therefore, the largest P value which is less than $\alpha = 0.05$ is chosen and the corresponding neural network is activated. All PNNs are trained using 5,000 input and output data sets.

3 Computational experiments

To evaluate the performances of the proposed PNN approach in estimating the variance-change point of a normal process, we generate data sets similar to the training phase, in which certain shifts in process variance are considered. The real change point corresponds to 10th sample and the number of out-of-control samples depends on the power of the control chart in detecting shifts.

The mean absolute errors of the proposed PNNs, the error being the difference between the real and the estimated change points, based on different variance shift magnitudes using the averages of 10,000 independent replications are given in Table 2. In this table, for example, if the process variance shifts to 1.5σ, then the mean of the absolute error is 0.3631. In other words, for a shift of 1.5σ in the process variance, in average, PNN shows 0.3631 samples greater or smaller than the real change point. Furthermore, the preci-

Table 1 Specifications of the neural networks

Neural network	Magnitude of shift	Number of input neurons	Number of output neurons
A	σ–1.2σ	80	80
B	1.2σ–1.3σ	40	80
C	1.3σ–1.4σ	15	80
D	1.4σ–3σ	10	80

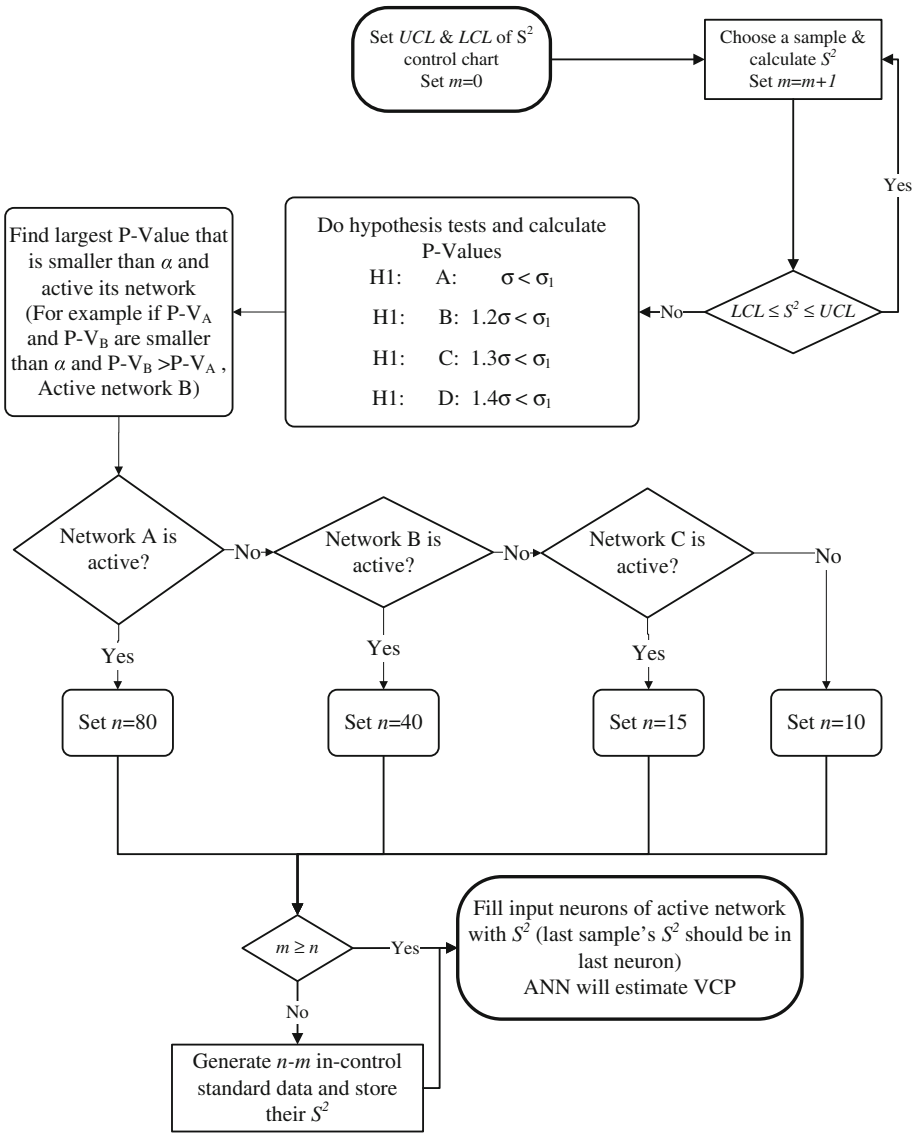


Fig. 2 A flowchart to use the designed PNNs in practice

Table 2 Mean absolute error of PNNs under different shifts

Shift	1.25σ	1.5σ	2σ	2.5σ
Mean of $E = \lvert\hat{\tau} - \tau\rvert$	1.1839	0.3631	0.0733	0.0324

sions of the proposed estimator in terms of the cumulative probability of the error being in different intervals based on 10,000 independent replications are given in Table 3 and Fig. 3. As shown in these tables and the figure, the proposed procedure performs well especially when the shifts are medium or large.

Note that in the real applications, negative shifts in the variance are in less attention because they imply process improvement. Hence, we avoid evaluating the proposed method under negative shifts of variance.

3.1 Comparison with the MLE method

In this section, the performance of the proposed approach is compared with the one of the MLE methods proposed by Samuel et al. (1998) using exactly the same values of the parameters n, μ and σ given in Samuel et al. (1998). The results are reported in Table 4. As shown in this table, the proposed method provides more accurate estimates than the one using the MLE method in all positive shifts of the process variance.

Table 3 Precision of the estimator

Shift	1.25σ	1.5σ	2σ	2.5σ
Probability of error	$P(E \le 0) = 0.5678$	$P(E \le 0) = 0.7875$	$P(E \le 0) = 0.9450$	$P(E \le 0) = 0.9769$
	$P(E \le 1) = 0.7915$	$P(E \le 1) = 0.9258$	$P(E \le 1) = 0.9890$	$P(E \le 1) = 0.9951$
	$P(E \le 2) = 0.8779$	$P(E \le 2) = 0.9656$	$P(E \le 2) = 0.9956$	$P(E \le 2) = 0.9979$
	$P(E \le 3) = 0.9136$	$P(E \le 3) = 0.9815$	$P(E \le 3) = 0.9982$	$P(E \le 3) = 0.9988$
	$P(E \le 4) = 0.9355$	$P(E \le 4) = 0.9897$		
	$P(E \le 5) = 0.9515$	$P(E \le 5) = 0.9943$		
	$P(E \le 6) = 0.9628$			$E = \left\|\hat{\tau} - \tau\right\|$
	$P(E \le 10) = 0.9861$			

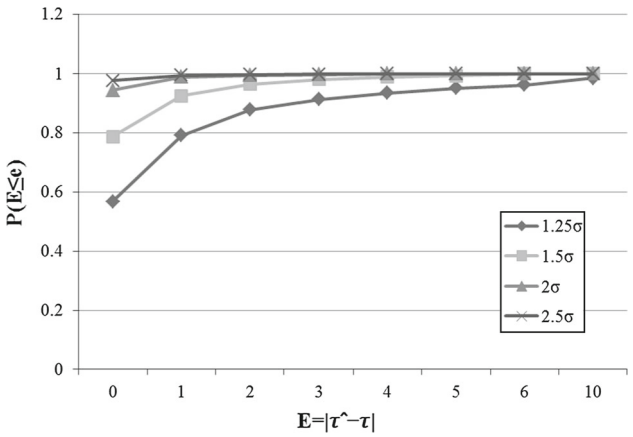


Fig. 3 Performance of the proposed methodology

4 Illustrative example

To demonstrate the application of the proposed methodology, an illustrative example is presented in this section using Monte Carlo simulation. In this example, the sample size is 10, $\mu = 0.2$, $\sigma = 4.0$, and $\alpha = 0.05$. The x values are first standardized using Eq. (4) (μ_0 and σ_0 are known), and then the corresponding S^2 statistic is computed. The UCL and LCL of the standardized S^2 control chart are 2.04832 and 0.322470, respectively (using Eqs. 2 and 3). Note that the UCL and LCL of standardized datasets just depend on α and n, because $\mu_1 = 0$ and $\sigma_1^2 = 1$ (under in-control situation).

The x values and their corresponding standardized z values are shown in Tables 5 and 6, respectively. The control chart signals an out-of-control condition and the last 40 samples until the alarm are shown in Table 5. The variance of the process is shifted 1.2σ and the control chart detected it after 8 samples (see Fig. 4) In other words, the last 8 samples are coming from the out-of-control process.

Based on the algorithm given in Fig. 2, test of hypotheses on the samples of the last subgroup is first conducted to select and activate the proper PNN. The corresponding P values of the test are shown in Table 7.

Table 4 Precision of the PNN estimator against MLE

Shift	1.1σ		1.2σ		1.3σ		1.4σ		1.5σ		2.0σ	
	MLE	PNN	MLE	PNN	MLE	PNN	MLE	PNN	MLE	PNN	MLE	PNN
$\hat{P}\left(\left\|\hat{\tau} - \tau\right\| \le 0\right)$	0.07	0.31	0.19	0.49	0.33	0.65	0.45	0.67	0.55	0.78	0.82	0.95
$\hat{P}\left(\left\|\hat{\tau} - \tau\right\| \le 1\right)$	0.14	0.52	0.37	0.71	0.56	0.83	0.71	0.86	0.79	0.92	0.95	0.99
$\hat{P}\left(\left\|\hat{\tau} - \tau\right\| \le 2\right)$	0.20	0.63	0.48	0.81	0.69	0.90	0.82	0.92	0.89	0.96	0.97	0.99
$\hat{P}\left(\left\|\hat{\tau} - \tau\right\| \le 3\right)$	0.26	0.70	0.57	0.85	0.77	0.93	0.89	0.94	0.93	0.98	0.98	1.00
$\hat{P}\left(\left\|\hat{\tau} - \tau\right\| \le 4\right)$	0.30	0.75	0.64	0.88	0.83	0.95	0.93	0.96	0.95	0.99	0.99	
$\hat{P}\left(\left\|\hat{\tau} - \tau\right\| \le 5\right)$	0.34	0.78	0.69	0.91	0.87	0.96	0.95	0.97	0.96	0.99		
$\hat{P}\left(\left\|\hat{\tau} - \tau\right\| \le 6\right)$	0.38	0.80	0.74	0.92	0.90	0.97	0.96	0.98	0.97	1.00		
$\hat{P}\left(\left\|\hat{\tau} - \tau\right\| \le 7\right)$	0.41	0.82	0.77	0.93	0.92	0.98	0.97	0.99	0.98			
$\hat{P}\left(\left\|\hat{\tau} - \tau\right\| \le 8\right)$	0.45	0.84	0.80	0.95	0.93	0.98	0.97	1.00	0.98			
$\hat{P}\left(\left\|\hat{\tau} - \tau\right\| \le 9\right)$	0.47	0.85	0.83	0.95	0.95	0.99	0.98		0.98			
$\hat{P}\left(\left\|\hat{\tau} - \tau\right\| \le 10\right)$	0.50	0.87	0.85	0.96	0.96	0.99	0.98		0.98			

Table 5 Simulated dataset

Subgroup	Samples									
	x_{i1}	x_{i2}	x_{i3}	x_{i4}	x_{i5}	x_{i6}	x_{i7}	x_{i8}	x_{i9}	x_{i10}
1	6.58	−3.55	4.67	−2.85	10.14	0.85	0.54	−0.20	−1.67	3.50
2	0.61	−1.07	6.11	4.00	−2.68	1.44	−0.11	−5.72	1.88	1.69
3	−3.47	6.05	5.38	−2.72	0.88	0.48	−0.22	2.24	−0.94	3.01
4	−0.87	3.88	6.00	4.59	−1.92	4.01	5.39	6.68	−3.36	3.75
5	−0.10	1.23	6.98	3.97	−4.27	1.44	0.03	6.46	8.52	1.32
6	0.71	7.41	−1.26	1.51	5.96	−0.35	1.55	4.51	5.48	3.06
7	1.15	1.60	−0.04	9.18	0.15	0.89	−0.38	5.97	1.18	0.52
8	3.02	5.80	−3.54	−4.43	−4.81	8.56	1.29	−1.72	6.59	−0.08
9	10.41	1.99	−5.86	4.14	−2.10	2.26	−1.15	−3.03	2.33	−0.04
10	−1.35	9.16	−1.74	9.12	5.94	6.25	4.06	3.84	3.89	2.32
11	−0.67	5.32	−3.25	3.24	4.59	0.48	3.06	1.38	3.72	−2.04
12	9.01	4.61	2.21	4.83	2.45	−0.81	5.39	−2.74	3.21	−0.79
13	11.87	−1.42	0.50	1.26	−4.34	7.57	3.01	4.82	2.05	1.52
14	−0.93	0.24	3.74	5.53	−0.08	4.45	3.31	7.52	1.96	3.43
15	−5.66	3.54	−1.73	−5.19	−1.85	−0.41	2.76	3.04	−0.89	1.18
16	3.98	3.16	3.36	−2.53	7.35	7.50	8.19	3.12	6.71	1.99
17	−0.82	−3.71	2.78	−2.36	−0.65	1.05	1.49	2.50	0.81	−0.52
18	1.69	3.11	−4.56	−2.58	5.40	1.73	0.56	−0.83	1.83	−4.09
19	1.60	3.41	−5.84	11.57	8.78	1.80	3.12	2.08	−2.29	−0.42
20	3.29	−5.07	−7.33	8.92	7.35	1.38	0.40	5.42	2.13	−0.37
21	−3.23	−2.60	−0.62	−0.48	4.49	3.13	−3.30	−2.57	8.00	7.72
22	2.95	7.85	2.33	3.77	−3.16	−3.72	−0.38	5.81	−2.45	6.10
23	0.33	−0.18	4.05	−4.61	11.30	4.20	−0.23	7.02	4.88	4.04
24	−3.96	−2.53	−1.98	−0.82	−3.29	−7.83	3.88	1.65	−3.86	6.89
25	1.43	0.56	0.30	0.83	−0.74	1.93	4.13	1.54	4.14	2.33
26	−2.18	6.99	−7.08	3.22	1.93	−2.50	1.17	7.48	8.15	−0.28
27	13.46	3.72	9.13	6.82	4.99	3.83	2.93	3.90	2.50	5.43
28	2.20	−7.00	−1.15	4.47	3.09	−2.04	−1.56	3.78	−2.62	−0.71
29	1.61	−0.20	3.50	1.92	−1.21	5.47	1.61	1.40	8.88	−1.12
30	1.08	−4.36	−4.41	6.97	−1.98	−0.81	0.78	−1.26	−2.20	4.17
31	1.53	−0.44	0.03	1.26	−4.01	−1.77	4.77	0.19	−0.69	−0.33
32	6.87	−5.60	5.84	4.62	0.48	3.09	5.46	3.67	3.75	−2.17
33	2.59	11.29	2.22	0.31	8.68	7.55	10.20	6.63	4.68	8.67
34	1.20	0.60	−1.87	10.32	13.27	6.88	−0.83	−3.78	−2.33	2.28
35	−4.17	−5.44	−5.24	4.16	−7.79	4.36	4.03	−3.07	−2.54	−0.78
36	−1.06	6.07	1.31	−2.09	4.42	6.70	−6.28	7.78	−2.94	−7.34
37	2.36	7.43	7.87	0.11	5.31	1.40	1.08	7.52	3.15	6.62
38	4.35	−1.10	4.64	1.26	−3.88	2.98	1.96	−0.52	2.57	−2.07
39	11.13	−5.66	−5.39	7.05	−1.68	−1.07	−1.06	6.31	2.57	4.39
40	−6.93	−6.55	3.05	−7.03	11.12	3.54	8.27	−6.51	4.55	4.35

As shown in Table 7, since the P values of the hypothesis A and B are less than $\alpha = 0.05$, and that the P value of test B is larger than the one for test A, network B should be activated.

As network B requires 40 inputs, the last 40 subgroups are collected and standardized to calculate the corresponding S^2 statistic as an input vector of 40×1 in network B. The output of network B shows the change point (note that the output

Table 6 Standardized simulated datasets and their corresponding S^2 statistic

Subgroup	Samples										S_i^2
	z_{i1}	z_{i2}	z_{i3}	z_{i4}	z_{i5}	z_{i6}	z_{i7}	z_{i8}	z_{i9}	z_{i10}	
1	1.15	−1.39	0.67	−1.21	2.03	−0.29	−0.36	−0.55	−0.92	0.38	1.08
2	−0.35	−0.77	1.03	0.50	−1.17	−0.14	−0.53	−1.93	−0.03	−0.08	0.74
3	−1.37	1.01	0.85	−1.18	−0.28	−0.38	−0.55	0.06	−0.74	0.25	0.61
4	−0.72	0.47	1.00	0.65	−0.98	0.50	0.85	1.17	−1.34	0.44	0.74
5	−0.53	−0.19	1.24	0.49	−1.57	−0.14	−0.49	1.11	1.63	−0.17	0.88
6	−0.32	1.35	−0.81	−0.12	0.99	−0.59	−0.11	0.63	0.87	0.27	0.52
7	−0.21	−0.10	−0.51	1.79	−0.46	−0.28	−0.60	0.99	−0.21	−0.37	0.53
8	0.25	0.95	−1.39	−1.61	−1.70	1.64	−0.18	−0.93	1.15	−0.52	1.35
9	2.10	0.00	−1.97	0.54	−1.02	0.07	−0.79	−1.26	0.08	−0.51	1.21
10	−0.84	1.79	−0.94	1.78	0.99	1.06	0.52	0.46	0.47	0.08	1.08
11	−0.67	0.83	−1.31	0.31	0.65	−0.38	0.26	−0.15	0.43	−1.01	0.48
12	1.75	0.65	0.05	0.71	0.11	−0.70	0.85	−1.19	0.30	−0.70	0.72
13	2.47	−0.86	−0.37	−0.19	−1.59	1.39	0.25	0.71	0.01	−0.12	1.20
14	−0.73	−0.44	0.44	0.88	−0.52	0.61	0.33	1.38	−0.01	0.36	0.45
15	−1.91	0.38	−0.93	−1.80	−0.96	−0.60	0.19	0.26	−0.72	−0.20	0.99
16	0.50	0.29	0.34	−1.13	1.34	1.38	1.55	0.28	1.18	0.00	0.93
17	−0.70	−1.43	0.20	−1.09	−0.66	−0.24	−0.13	0.12	−0.30	−0.63	0.48
18	−0.08	0.28	−1.64	−1.15	0.85	−0.07	−0.36	−0.71	−0.04	−1.52	0.78
19	−0.10	0.35	−1.96	2.39	1.70	−0.05	0.28	0.02	−1.07	−0.60	1.42
20	0.32	−1.77	−2.33	1.73	1.34	−0.16	−0.40	0.86	0.03	−0.59	1.47
21	−1.31	−1.15	−0.66	−0.62	0.62	0.28	−1.32	−1.14	1.50	1.43	1.17
22	0.24	1.46	0.08	0.44	−1.29	−1.43	−0.60	0.95	−1.11	1.02	0.97
23	−0.42	−0.54	0.51	−1.65	2.32	0.55	−0.56	1.26	0.72	0.51	1.18
24	−1.49	−1.13	−0.99	−0.71	−1.32	−2.46	0.47	−0.09	−1.46	1.22	1.66
25	−0.14	−0.36	−0.42	−0.29	−0.69	−0.02	0.53	−0.11	0.54	0.08	0.15
26	−1.05	1.25	−2.27	0.31	−0.02	−1.12	−0.21	1.37	1.54	−0.57	1.38
27	2.86	0.43	1.78	1.20	0.75	0.46	0.23	0.47	0.12	0.86	1.48
28	0.05	−2.25	−0.79	0.62	0.27	−1.01	−0.89	0.44	−1.16	−0.68	0.99
29	−0.10	−0.55	0.37	−0.02	−0.80	0.87	−0.10	−0.15	1.72	−0.78	0.54
30	−0.23	−1.59	−1.60	1.24	−0.99	−0.70	−0.31	−0.82	−1.05	0.54	1.03
31	−0.12	−0.61	−0.49	−0.18	−1.50	−0.94	0.69	−0.45	−0.67	−0.58	0.53
32	1.22	−1.90	0.96	0.65	−0.38	0.27	0.86	0.42	0.44	−1.04	0.89
33	0.15	2.32	0.06	−0.42	1.67	1.39	2.05	1.16	0.67	1.67	1.91
34	−0.20	−0.35	−0.97	2.08	2.82	1.22	−0.71	−1.44	−1.08	0.07	1.86
35	−1.54	−1.86	−1.81	0.54	−2.45	0.59	0.51	−1.27	−1.14	−0.69	1.94
36	−0.77	1.02	−0.17	−1.02	0.61	1.18	−2.07	1.44	−1.24	−2.33	1.78
37	0.09	1.36	1.47	−0.47	0.83	−0.15	−0.23	1.38	0.29	1.15	0.83
38	0.59	−0.77	0.66	−0.19	−1.47	0.25	−0.01	−0.63	0.14	−1.02	0.51
39	2.28	−1.91	−1.85	1.26	−0.92	−0.77	−0.77	1.08	0.14	0.60	1.74
40	−2.23	−2.14	0.26	−2.26	2.28	0.38	1.57	−2.13	0.64	0.59	2.78

vectors of networks all have 80 neurons). Table 8 shows the input neurons and the last 40 output neurons of network B. The results in Table 8 show that network B detects the change point at 33rd subgroup; i.e., 33rd subgroup is "1" in both output and target columns. The estimated change point based on the result of Table 8 is also illustrated in Fig. 4 using a

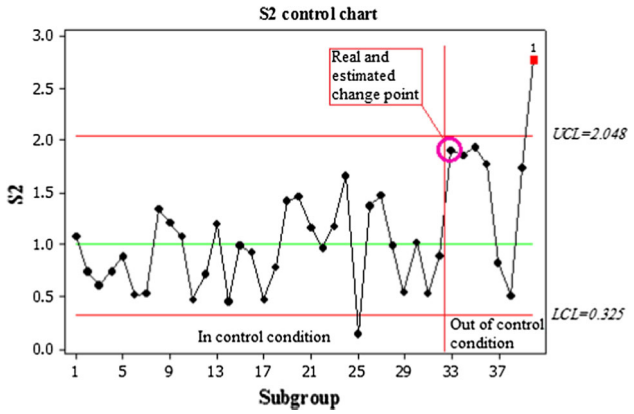


Fig. 4 Control chart of the standardized simulated datasets

Table 7 P values of tests of hypotheses

H_1	P value
A: $\sigma_0 < \sigma_1$	0.0019
B: $1.2\sigma_0 < \sigma_1$	0.0366
C: $1.3\sigma_0 < \sigma_1$	0.0875
D: $1.4\sigma_0 < \sigma_1$	0.1648

circle. As mentioned in Table 4, we note that roughly in 50 percent of the times the proposed PNN approach performs similar to this example $\hat{P}(|\hat{\tau} - \tau| \leq 0) = 0.49$ for the shift of 1.2σ and sample size of 10).

5 Conclusions and future researches

In this paper, a procedure based on probabilistic ANN was proposed to estimate VCPs of univariate normal processes. In the proposed procedure, the S^2 control chart was used to detect shifts in the variance parameter of the process. The results obtained using extensive simulation experiments showed that the proposed PNN-based method performs better than the MLE method in all shifts. An illustrative simulation example was given at the end to demonstrate the application of the proposed methodology.

As a future research, one can consider a similar methodology to find the real time of a change in covariance matrix of multivariate normal quality characteristics.

Table 8 Inputs and outputs of network B

Subgroup	Input	Number of output neuron	Output	Target
1	1.08	41	0	0
2	0.74	42	0	0
3	0.61	43	0	0
4	0.74	44	0	0
5	0.88	45	0	0
6	0.52	46	0	0
7	0.53	47	0	0
8	1.35	48	0	0
9	1.21	49	0	0
10	1.08	50	0	0
11	0.48	51	0	0
12	0.72	52	0	0
13	1.20	53	0	0
14	0.45	54	0	0
15	0.99	55	0	0
16	0.93	56	0	0
17	0.48	57	0	0
18	0.78	58	0	0
19	1.42	59	0	0
20	1.47	60	0	0
21	1.17	61	0	0
22	0.97	62	0	0
23	1.18	63	0	0
24	1.66	64	0	0
25	0.15	65	0	0
26	1.38	66	0	0
27	1.48	67	0	0
28	0.99	68	0	0
29	0.54	69	0	0
30	1.03	70	0	0
31	0.53	71	0	0
32	0.89	72	0	0
33	1.91	73	1	1
34	1.86	74	0	0
35	1.94	75	0	0
36	1.78	76	0	0
37	0.83	77	0	0
38	0.51	78	0	0
39	1.74	79	0	0
40	2.78	80	0	0

Acknowledgments The authors are thankful to anonymous reviewers for their precious comments which led to significant improvement in the paper.

References

Ahmadzadeh F (2011) Change point detection with multivariate control charts by artificial neural network. J Adv Manuf Technol (published online). doi:10.1007/s00170-009-2193-6

Amiri A, Allahyari S (2012) Change point estimation methods for control chart post signal diagnostics: a literature review. Qual Reliab Eng Int 28:637–685

Amiri A, Khosravi R (2013) Identifying time of a monotonic change in the fraction nonconforming of a high quality process. J Adv Manuf Technol 68:547–555

Amiri A, Khosravi R (2012) Estimating the change point of cumulative count of conforming control chart under a drift. Scientia Iranica 19:856–861

Atashgar K, Noorossana R (2011) An integrating approach to root cause analysis of a bivariate mean vector with a linear trend disturbance. Int J Adv Manuf Technol 52:407–420

Hawkins DM, Qiu P, Kang CW (2003) The change point model for statistical process control. J Qual Technol 35:355–366

John A (2000) Multiple change point detection with genetic algorithm. Soft Comput 4:68–75

Lee J, Park C (2007) Estimation of the change point in monitoring the process mean and variance. Commun Stat Simul C 36:1333–1345

Li F, Runger GC, Tuv E (2006) Supervised learning for change point detection. Int J Prod Res 44:2853–2868

Niaki STA, Khedmati M (2012) Detecting and estimating the time of a step-change in multivariate poisson processes. Scientia Iranica 19:862–871

Noorossana R, Shadman A (2009) Estimating the change point of a normal process mean with a monotonic change. Qual Reliab Eng Int 25:79–90

Oh KJ, Moon MS, Kim TY (2005) Variance change point detection via artificial neural networks for data separation. Neurocomputing 68:239–250

Park J, Park S (2004) Estimation of the change point in the x and s control charts. Commun Stat Simul C 33:1115–1132

Perry MB, Pignatiello JJ, Simpson JR (2006) Estimating the change point of a poisson rate parameter with a linear trend disturbance. Qual Reliab Eng Int 22:371–384

Perry MB, Pignatiello JJ, Simpson JR (2007) Change point estimation for monotonically changing poisson rates in SPC. Int J Prod Res 45:1791–1813

Perry MB, Pignatiello JJ (2008) A change point model for the location parameter of exponential family densities. IIE Trans 40:947–956

Perry MB (2010) Identifying the time of polynomial drifts in the mean of auto correlated processes. Qual Reliab Eng Int 26:399–415

Pignatiello JJ, Samuel TR (2001) Estimation of the change point of a normal process mean in SPC applications. J Qual Technol 33:82–95

Samuel TR, Pignatiello JJ, Calvin JA (1998) Identifying the time of a step change with x control charts. Qual Eng 10:521–527

Zandi F, Niaki STA, Nayeri MA, Fathi M (2011) Change-point estimation of the process fraction non-conforming with a linear trend in statistical process control. Int J Comp Integr M 24:939–947

Zhao w, Tian Z, Xia Z (2010) Ratio test for variance change point in linear process with long memory. Stat Paper 51:397–407