RBF Neural Network for Probability Density Function Estimation and Detecting Changes in Multivariate Processes

Ewa Skubalska-Rafajłowicz

Institute of Computer Engineering, Control and Robotics, Wrocław University of Technology, Wybrzeże Wyspiańskiego 27, 50 370 Wrocław, Poland
ewa.rafajlowicz@pwr.wroc.pl

Abstract. We propose a new radial basis function (RBF) neural network for probability density function estimation. This network is used for detecting changes in multivariate processes. The performance of the proposed model is tested in terms of the average run lengths (ARL), i.e., the average time delays of the change detection. The network allows the processing of large streams of data, memorizing only a small part of them. The advantage of the proposed approach is in the short and reliable net training phase.

1 Introduction

Statistical control charts are designed in order to detect abnormalities (out-of control states) in the process under consideration. The most common abnormalities are mean shifts, variance changes and trends.

Suppose $X_1, X_2, \ldots$ are independent random vectors observed sequentially and X_1 to X_{q-1} have a distribution function with probability density f_0 while $X_q, X_{q+1}, \ldots$ have a distribution function with probability density $f_1 \neq f_0$.

q is unknown and some action should be taken after undesirable change in the process. One has to decide, on the basis of given observations

$$X_t = (x_{t1}, \ldots, x_{td}) \ ,$$

whether X_t is r.v. with pdf f_0 , i.e., process is "in-control" or if X_t is another r.v. - process is "out-of-control", i.e., changes in the process occurred. We assume, that probability densities f_0 and f_1 exist but are unknown.

There is extensive literature on statistical methods for statistical process control (SPC) and control charts, see [14] and the bibliography cited therein.

Classical control charts require prior assumptions about the probability density distribution of the observed process variables. Typically it is assumed that monitored data follow univariate or multivariate Gaussian (or sometimes other known) distribution. For multivariate statistical process control with individual observations, the Hotelling T^2 control chart or charts (based on Mahalanobis distance) are usually recommended.

L. Rutkowski et al. (Eds.): ICAISC 2006, LNAI 4029, pp. 133–141, 2006.

A neural network based approach to statistical process control and out-of control state detection allows in-control data density distribution to be non-Gaussian. Most of the neural network models designed for detecting changes in (mostly univariate) statistical process work in pattern recognition settings, i.e. on the assumption that also abnormal observation (out-of control states) are available and their class-membership (in-control and out-of control labels) are known [8], [3], [7], [6], [10], [9], [4], [5], [2], [12].

A neural network-based approach used when only in-control data is available has been considered in only a few papers (see [18] and [22]).

In the former paper the author proposes a vector quantization neural network with Kohonen‘s type learning algorithm to define the acceptance region. The multi-variate data is transformed onto unit interval using quasi-inverse of a space-filling curve [19], [20]. The method uses only one current vector observation to decide about the state of the process and for normal (Gaussian) data it is comparable to the Hotelling T^2 control chart [18], [13]. Zorriassatine et al.[22] uses a novelty detection method [1] for bivariate time series.

When constructing a control chart it is desirable to have a long average run length (ARL) in the in-control state, since this means a low level rate of false alarms. On the other hand, a short out-of control ARL is desired, which guarantees that any unacceptable changes will be identified as soon as possible.

Here we propose a new, easy to learn, radial basis function (RBF) neural network model for detecting changes in a multivariate process. The detection is based on one vector observation as in the classical T^2 control chart. Furthermore, we assume that the a‘priori probabilities of in-control and out of control process states are not given.

In this paper the possibilities of detecting changes in the process mean vector (mean shifts) are investigated in terms of in-control and out-of-control ARL‘s.

2 RBF Neural Network Model for Detection of Changes

The radial basis function networks have been extensively applied to pattern recognition, function approximation or regression function estimation.

A basic radial-basis function (RBF) network consists of three layers having entirely different roles: an input layer, a hidden layer, which applies a nonlinear transformation from the input space to the hidden space and a linear output layer. Hence,

$$f_N(x) = \sum_{i=1}^{N} w_i G(||x - c_i||) \ , \tag{1}$$

where $x \in R^d$, $c_i \in R^d$, are tunable vectors, w_i are tunable weights, and N is a number of neurons.

Usually $||x||$ is the Euclidean norm, however also generalized weighted norm $||x||_{Q_i}$, defined by the quadratic form $||x||^2_{Q_i} = x^T Q_i^T Q_i x$ can be used, where Q_i are (usually tunable) $d \times d$ matrices.

The most popular are Gausssian RBF nets:

$$G(r) = \exp\left(-\frac{r^2}{2\sigma^2}\right) \text{ for some } \sigma > 0 \text{ and } r \in \mathcal{R} .$$

There are three groups of parameters in the RBF networks which may be learnable or arbitrarily chosen: the weights w_i, the centers c_i and some parameters of radial basis functions, for example σ or Q_i matrices.

RBF networks can be related to Parzen window estimators of a probability density [16] or to Nadaraya-Watson regression estimators [21], [1], [15]. Similarities between the RBF network structure and kernel regression estimators lead to RBF networks with the centers chosen to be a subset of the training input vectors and associated weights which directly correspond to Y_i's [21]. Other approaches related to Nadaraya-Watson regression estimators were proposed in [17] and [11]

Usually, parameters of the network (1) are obtained from an n-sample observation data set (learning sequence) $L_n = ((X_1, Y_1), \ldots, (X_n, Y_n)$.

As regards our problem, labels $Y_i,\ i = 1, \ldots, n$ are set as "in-control" and do not carry any information. Thus, we need a net which will self-organize and generalize information about distribution of the in-control states.

In this context, we choose RBF neural networks related to Parzen kernel estimators. Bishop [1] discusses a number of heuristics for learning RBF parameters in such a way, that the basis functions approximate the distribution of the input data.

The Parzen window estimator [16], [1] with Gaussian kernel functions takes the form:

$$\frac{1}{n(2\pi\sigma^2)^{d/2}} \sum_{i=1}^{n} \exp\left(-\frac{||X - X_i||^2}{2\sigma^2}\right) , \tag{2}$$

where d is the dimensionality of the input data.

Let N be a number of centers. Assuming that the centers should be distributed according the same probability distribution as the learning data, the centers are simply a subset of the training input vectors. One can take, for example, N first elements from the leaning sequence $(X_1, \ldots, X_n)$.

Note that if X_i is close to a center C, then

$$G(||X - X_i||) - G(||X - C||) \approx 0$$

So, we can replace each X_j in the sum (2) by its nearest neighbor among a set of centers $\{C_1, C_2, \ldots, C_N\}$ breaking ties at random. Note, that the same C_i can be the nearest neighbor for several X_j's and that each C_i has at least one point from the learning sequence (namely itself) as a neighbor, since every center is taken from the learning set.

Let n_j stands for the number of points closest to the center C_j, i.e.,

$$n_j = \text{card}[\{X_i : ||X_i - C_j|| < ||X_i - C_k||\}] .$$

Thus, we obtain the approximate version of (2):

$$y(X) = \frac{1}{n(2\pi\sigma^2)^{d/2}} \sum_{j=1}^{N} n_j \exp\left(-\frac{||X - C_j||^2}{2\sigma^2}\right) . \tag{3}$$

Observe, that this kind of the probability density approximation appears in [17], where it is used as a common denominator in the RBF neural network mimicking the Nadaraya-Watson regression estimators.

The decision about in-control or out of control state of the current vector observation X is made according to the estimate of the probability density $y(X)$.

If $y(X) < \lambda$, where λ is chosen acceptance level, then classify X as abnormal (out of control) state of the process. Otherwise, accept X as an in-control state. The set $y(X) \geq \lambda$ forms the confidence region.

2.1 Algorithm for Tuning RBF Net

Step 1. Choose centers C_j, $j = \overline{1,N}$ at random from the learning sequence $\{X_1, X_2, \ldots, X_n\}$.

Step 2. Set $n_j = 0$, $j = \overline{1,N}$.

Step 3. For $i = \overline{1,n}$ perform the following steps.

1. Find $j^* = arg \min_{1 \leq j \leq N} ||X_i - C_j||$.
2. Update the corresponding weight:$n_{j^*} = n_{j^*} + 1$.

Step 4. Form the net

$$y(X) = \frac{1}{n(2\pi h\sigma^2)^{d/2}} \sum_{j=1}^{N} n_j \exp\left(-\frac{||X - C_j||^2}{2\sigma^2}|\right) \ .$$

Step 5. Choose the acceptance level (threshold) λ. If $y(X)$ is greater than λ accept vector observation as in-control, otherwise alarm, since an out-of-control state is detected.

This algorithm should be accompanied by a method of selecting the bandwidth $\sigma > 0$ and threshold parameter λ. One can choose any known method, e.g., the cross-validation for selecting $h\sigma$. Having selected centers and using a formula (3) one can considerably reduce the computational burden needed for selecting σ in a data-driven way.

Furthermore, reducing the number of kernels (to the number of centers) leads to the formula less sensitive to the σ choice. The threshold level λ governs the false alarm probability α. The average run length to the false alarm (the in-control ARL) equals to $1/\alpha$[14], but the distribution function of in-control states is usually not (fully) known. Thus, as in classical control charts, the value of the threshold, which guarantees desired in-control ARL should be chosen experimentally.

3 Experimental Results

In the following sections we present the results of applying the RBF control chart to a series of simulated data sets. We have tested the proposed method using three different data sets:

A. A 2-D normal distribution with $(0, 0)$ mean and covariance matrix I

B. A mixture of two equiprobable 2-D normal distributions with vector means: $(0, 0)$ and $(2, 0)$ and the same I.

C. A mixture of three equiprobable 2-D normal distributions with vector means: $(0, 0)$, $(2, 0)$ and $(0, 0-2)$ and the same I (see Figure 1).

We compared the performance of the proposed RBF control chart with the results given by the classical T^2 chart based on the Hotelling statistic (also known as the squared Mahalanobis distance) [14], [13]:

$$T^2(x) = (x - m)^T \Sigma^{-1}(x - m) \ ,$$

where x is a given observation vector, m is a mean vector and Σ the covariance matrix. Both of them, m and Σ, are estimated from the learning data. If $T^2(x) > t$, where t is an experimentally chosen value, it is assumed that the observation x is rejected as out-of-control data. Thus, the region of acceptance of T^2 control chart form an ellipsoid with the center m and the other parameters defined by the covariance matrix Σ.

In two cases (A and B) the neural network model was tuned using 10^5 in-control learning samples. We examined the changes in the process caused by the following mean shifts $||\Delta m|| = 0.5,\ 1, 2,$ and 3. The number of centers was equal to 100.

The in-control ARL‘s were obtained from 10^6 examples and out-of-control ARL‘s were estimated from 10^5 repetitions. The results were averages over four different shift directions. The comparisons for examples A and B are given in Table 1.

Table 1. Comparison of RBF neural network chart with T^2 chart, $d = 2$

$\|\|\Delta m\|\|$	Example A		Example B	
	T^2	RBF net	T^2	RBF net
	$t = 10.6$	$\lambda = 0.00049$	$t = 9.8$	$\lambda = 0.0003$
0.0	200.0	199.0	200.4	198.2
0.5	116.0	117.6	124.0	118.2
1.0	42.0	44.5	53.3	48.1
2.0	6.9	7.7	10.1	8.7
3.0	2.2	2.3	3.2	2.9

The second column of Table 1 contains analytically obtained ARL‘s for T^2 Hotelling chart applied to multivariate normal data (see for example [13]). The RBF net based control chart attains almost the same ARL times, however the knowledge about probability density distribution is not used in the process of designing the RBF net chart. The value of parameter λ was chosen experimentally on test data in such a way as to obtain ARL0=200.

The third column consists of empirically determined ARL‘s for T^2 Hotelling chart applied to multivariate non-normal data (example B). This time, the RBF

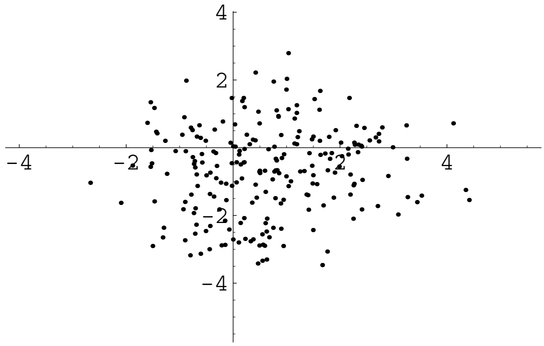

Fig. 1. Learning data - example C

net chart leads to shorter average detection times (smaller out-of-control ARL's), than the T^2 chart.

Experiments on the third example (the mixture of three normals – C) were performed on 300,000 learning samples (see Figure 1). In this case the changes in the process were introduced by the mean shifts $||\Delta m||/\ (det\Sigma_C)^{1/4} = 0.5,\ 1, 2, 3,$ where Σ_C is the empirical covariance matrix obtained for the distribution C (estimated on the basis of all 300000 learning samples). The results for every shift's length were averages over eight different shift directions. The same empirical covariance matrix Σ_C was used in T^2 calculations. The number of centers of the RBF network was equal to 200. The value of parameter λ was chosen experimentally on test data in such a way as to obtain ARL0=200. The comparisons for the example C are given in Table 2.

Table 2. Comparison of RBF neural network chart with T^2 chart for mixture of three normals $d = 2$

$\|\|\Delta m\|\|/det\Sigma_C^{1/4}$	Example C	
	T^2	RBF net
	$t = 41.5$	$\lambda = 0.00029$
0.0	201.6	200.0
0.5	137.1	97.6
1.0	85.8	29.3
2.0	21.8	4.64
3.0	3.76	1.82

The RBF control chart proposed here gave this time evidently better results than that obtained with T^2 control chart. The ARL times estimated for T^2 chart are even worse than relative ARL's computed for examples A and B, since the

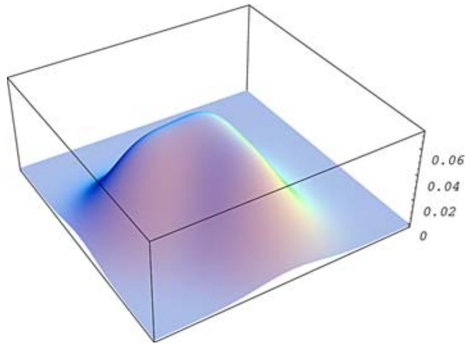

Fig. 2. Probability density function of data from example C

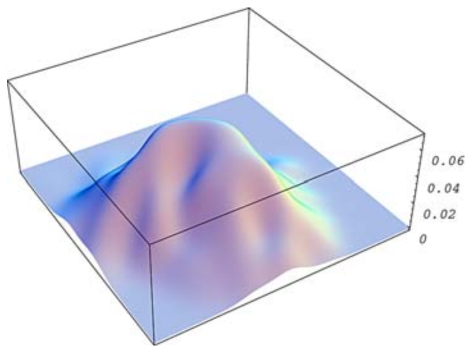

Fig. 3. Probability density function of data from example C estimated by RBF net

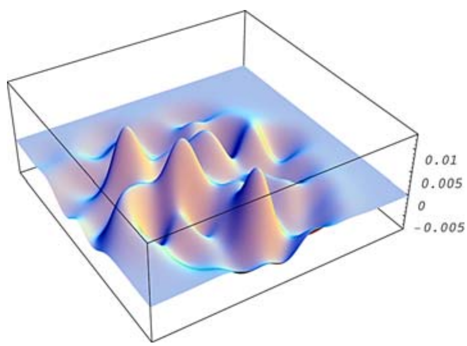

Fig. 4. Error of probability density estimation for example C

mixture of three normals distributions is not similar to any two-dimensional normal distribution. Figure 2 shows the true density function of data from example C. Figure 3 presents the probability density function of data under considerations estimated using the RBF neural network (obtained by formula 3). Error of this probability density estimation is given in Figure 4. The acceptance region obtained during experiments with in-control data is illustrated in Figure 5 as a

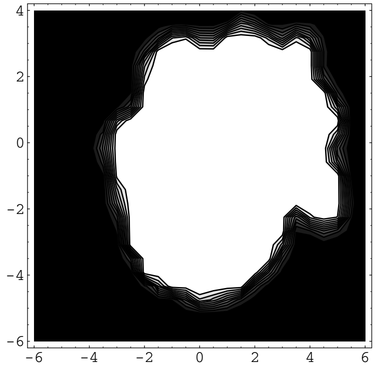

Fig. 5. Acceptance region obtained using a RBF net with $\lambda = 0.00029$

white shape. This is a set of points where the value of parameter λ is smaller than the probability density value estimated in those points by RBF network.

4 Concluding Remarks

The crucial problem faced in this paper is in designing the simple and robust nonparametric probability density function estimator for time independent multivariate processes. A new version of a RBF neural network allows the processing of large streams of data, memorizing only a small part of them. The network was successfully applied to the detection of changes in multivariate processes. The advantage of the proposed approach is in the short and reliable net training phase.

Acknowledgments

This paper was partially supported by a Polish Government Grant 2003–2005 and by the Polish Foundation for Science.

References

1. Bishop C.M.: Neural Networks for Pattern Recognition. Oxford University Press, Oxford (1995)
2. Chang S.I., Aw C.A.: A neural fuzzy control chart for detecting and classifying process mean shifts. International Journal of Production Research **34** (1996) 2265–2278
3. Chen L., Wang T.: Artificial neural networks to classify mean shifts from multivariate χ^2 charts signals. Computers and Industrial Engineering **47**(2004) 195–205
4. Cheng C.-S.: A neural network approach for the analysis of control charts patterns. International Journal of Production Reseach **235**(1997) 667–697

5. Cheng C.-S.: A multi-layer neural network model for detecting changes in the process mean. Computers and Industrial Engineering **28** (1995) 51–61
6. Cheng C.-S., Cheng S.: A neural network-based procedure for the monitoring of exponential mean. Computers and Industrial Engineering **40**(2001) 309–321
7. Chinnam R.B.: Support vector machines for recognizing shifts in correlated and other manufacturing processes. International Journal of Production Research **40**(2002) 4449–4466
8. Guh R.: A hybrid learning based model for on-line detection and analysis of control chart patterns. Computers and Industrial Engineering **49**(2005) 35–62
9. Guh R., Hsieh Y.: A neural network based model for abnormal pattern recognition of control charts. Computers and Industrial Engineering **36**(1999) 97–108
10. Ho E.S., Chang S.I.: An integrated neural network approach for simultaneously monitoring of process mean and variance shifts - a comparative study. International Journal of Production Research **37**(1999) 1881–1901
11. Krzyżak A., Skubalska-Rafajłowicz E.: Combining Space-filling Curves and Radial Basis Function Networks.In: Artificial Intelligence and Soft Computing ICAISC 2004. Lecture Notes in Artificial Intelligence Vol.3070. Springer-Verlag, Berlin Heidelberg New York (2004) 229–234
12. Low C., Hsu C., Yu F.:Analysis of variations in a multi-variate process using neural networks.Int. J Adv Manuf Technology **22** (2003) 911–921
13. Lowry C.A., Woodall W.H.: A Multivariate Exponentially Weighted Moving Average Control Chart. Technometrics **34** (1992)46–53
14. Montgomery D.C.:Introduction to statistical quality control.John Wiley and Sons, Inc.(2001)
15. Nadaraya A.: On the integral mean square error of some nonparametric estimates for the density function. Theory of Probability and its Applications **19** (1974) 133–141
16. Parzen E.:On estimation of a probability density function and mode.Annals of Mathematical Statistics **33** (1962) 1065–1076
17. Rafajłowicz E., Skubalska-Rafajłowicz E.: RBF nets based on equidistributed points. in Proc. of 9th IEEE International Conf. Methods and Models in Automation and Robotics MMAR 2003 **2** (2003) 921–926
18. Skubalska-Rafajłowicz E.: On using space-filling curves and vector quantization for constructing multidimensional control charts. Fifth Conf. "Neural Network and Soft Computing" Zakopane June 6-10, 2000 (2000) 162–167
19. Skubalska-Rafajłowicz E. : Pattern recognition algorithm based on space-filling curves and orthogonal expansion. IEEE Trans. on Information Theory **47** (2001) 1915–1927
20. Skubalska-Rafajłowicz E.: Space-filling Curves in Multivariate Decision Problems. (in Polish) Wrocław University of Technology Press, Wrocław (2001)
21. Xu L., Krzyżak A., Yuille A.: On Radial Basis Function Nets and Kernel Regression: Statistical Consistency, Convergence Rates, and Receptive Field Size. Neural Networks **7** (1994) 609–628
22. Zorriassatine F., Tannock J.D.T., O'Brien C.: Using novelty detection to identify abnormalities caused by mean shifts in bivariate processes.Computers and Industrial Engineering **44** (2003) 385–408