

Article

New Machine-Learning Control Charts for Simultaneous Monitoring of Multivariate Normal Process Parameters with Detection and Identification

Hamed Sabahno ^{1,*}  and Seyed Taghi Akhavan Niaki ² ¹ Department of Statistics, School of Business, Economics and Statistics, Umeå University, Umeå 901 87, Sweden² Department of Industrial Engineering, Sharif University of Technology, Tehran 145889694, Iran; niaki@sharif.edu

* Correspondence: hamed.sabahno@umu.se

Abstract: Simultaneous monitoring of the process parameters in a multivariate normal process has caught researchers' attention during the last two decades. However, only statistical control charts have been developed so far for this purpose. On the other hand, machine-learning (ML) techniques have rarely been developed to be used in control charts. In this paper, three ML control charts are proposed using the concepts of artificial neural networks, support vector machines, and random forests techniques. These ML techniques are trained to obtain linear outputs, and then based on the concepts of memory-less control charts, the process is classified into in-control or out-of-control states. Two different input scenarios and two different training methods are used for the proposed ML structures. In addition, two different process control scenarios are utilized. In one, the goal is only the detection of the out-of-control situation. In the other one, the identification of the responsible variable (s)/process parameter (s) for the out-of-control signal is also an aim (detection–identification). After developing the ML control charts for each scenario, we compare them to one another, as well as to the most recently developed statistical control charts. The results show significantly better performance of the proposed ML control charts against the traditional memory-less statistical control charts in most compared cases. Finally, an illustrative example is presented to show how the proposed scheme can be implemented in a healthcare process.

Keywords: process monitoring; machine-learning techniques; simultaneous process parameters monitoring; multivariate normal process; simulation

MSC: 62P30; 68T20; 68T09; 62P10; 62H10; 62H12; 62-08



Citation: Sabahno, H.; Niaki, S.T.A. New Machine-Learning Control Charts for Simultaneous Monitoring of Multivariate Normal Process Parameters with Detection and Identification. *Mathematics* **2023**, *11*, 3566. <https://doi.org/10.3390/math11163566>

Academic Editors: Arne Johannssen and Nataliya Chukhrova

Received: 21 July 2023

Revised: 14 August 2023

Accepted: 15 August 2023

Published: 17 August 2023



Copyright: © 2023 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

When there are more than two quality characteristics to be monitored in a process, multivariate control charts can be employed in the process monitoring context. In a multivariate normal process, there are two process parameters, i.e., the mean vector and the variance–covariance matrix. Simultaneous monitoring of these parameters is usually preferred over monitoring only one of them due to its better overall performance. We have two main simultaneous process parameters monitoring schemes: (i) a single-chart scheme and (ii) a double-chart scheme. In a single-chart scheme, only one chart for both process parameters is used, and in a double-chart scheme, one chart for each process parameter is employed. A single-chart scheme is usually preferred due to its simplicity, considering only one control chart should be administrated in a single-chart scheme in comparison to the double-chart scheme, in which two control charts should be administrated at the same time. Reynolds and Gyo-Young [1], Hawkins and Maboudou-Tchao [2], and Zhang and Chang [3] studied double-chart cases, and Khoo [4], Zhang et al. [5], Wang et al. [6], Sabahno et al. [7–9], Sabahno and Khoo [10], and Sabahno [11] investigated single-chart

cases. Sabahno et al. [9] proposed new memory-less statistical control charts with fixed and adaptive design parameters (sample size, sampling interval, and control limits) for simultaneous monitoring of multivariate normal process parameters. They used two statistics each for monitoring one process parameter, but in the end, they combined them into one statistic.

Machine-learning techniques have been extensively used for process monitoring with control charts for different reasons, such as dimension reduction, pattern recognition, change point estimation, signal detection, identification, and fault diagnosis. Artificial neural network (ANN) is the most used ML technique in this regard. Some of the most notable research that employed ML techniques in control charts are that conducted by Chang and Ho [12], Niaki and Abbasi [13,14], Cheng and Cheng [15], Abbasi [16], Salehi et al. [17], Hosseini-fard et al. [18], Weese et al. [19], Escobar and Morales-Menendez [20], Apsemidis et al. [21], Mohd Amiruddin et al. [22], Diren et al. [23], Yeganeh et al. [24], Mohammadzadeh et al. [25], and Sabahno and Amiri [26], and Yeganeh et al. [27–29].

ML structures have rarely been used to construct control charts in the literature. Niaki and Abbasi [14] developed a perceptron neural network for monitoring and classifying mean shifts in multi-attribute processes. Hosseini-fard et al. [18] developed three ANN control charts for monitoring simple linear profile parameters (the intercept, the slope, and the residual variance). One of their control charts was involved in detection and identification, while the other two were only involved in detection. Mohammadzadeh et al. [25] developed an SVR (support vector regression) control chart for monitoring a logistic profile by extending Hosseini-fard et al.'s [18] paper. Sabahno and Amiri [26] developed different statistical and machine-learning-based control charts with fixed and variable design parameters to monitor generalized linear regression profiles. Yeganeh et al. [27] extended Hosseini-fard et al.'s [18] work for social network surveillance. Yeganeh et al. [28] proposed an ANN-based control chart to monitor binary surgical outcomes, while in another study, Yeganeh et al. [29] proposed ML-based control charts for monitoring autocorrelated profiles.

The previously mentioned papers have utilized machine-learning (ML) techniques for regression to construct control charts. While our approach in this paper is similar to theirs, by proposing a special input set for ML structures (our first input scenario), we extend them to address multivariate processes and enable simultaneous monitoring of their parameters. Moreover, while they employed a single ML structure to build their control charts, we utilize multiple machine-learning techniques and extensively compare them in different scenarios. Additionally, all the ML control charts mentioned above solely trained their ML structures using a random shift size to represent out-of-control data. In contrast, we employ an alternative method in addition to this approach.

In this paper, for the first time in the literature, the ANN, SVM (support vector machine), and RF (random forest)-based control charts are proposed for simultaneous monitoring of multivariate normal process parameters. We use two different input scenarios (in one of them, for the first time the two statistics used by Sabahno et al. [9] are utilized as the inputs) and two different control scenarios (detection and detection–identification). We also use two training methods to see which types of datasets suit each ML structure better (in one we train the ML structures with small shifts and in the other one, for the first time, we train them with both small and large mean shifts). The ML control charts are developed in the cases of two, three, and four quality characteristics, based on which their performances with one another, and with the proposed statistical control charts by Sabahno et al. [9], are compared.

This paper is structured as follows: In Section 2, machine-learning control charts are developed. In Section 3, different models are developed, and then extensive numerical analyses in each scenario under different separate and simultaneous shift sizes are conducted. In Section 4, an illustrative example is presented. In Section 5, concluding remarks and suggestions for future research are discussed.

2. Machine-Learning Control Charts

In this section, three ML control charts are proposed. After obtaining the ML structure for each control chart, which is explained later in Section 3, the upper control limit (UCL) of each control chart is obtained using the after-mentioned algorithm. First note that the design parameters in a control chart are the sample size n , the sampling interval t , and the probability of type-I error α .

The monitoring strategy for each ML control chart is as follows:

- If at sample i , the ML structure's output \leq UCL, then the process is in control;
- If at sample i , the ML structure's output $>$ UCL, then the process is declared out-of-control.

The following algorithm can be used to compute the UCLs:

- Step 1. Choose a value for α and n ;
- Step 2. Choose and train an ML structure;
- Step 3. Obtain the initial value of UCL by generating and sorting 10,000 in-control samples in ascending order using the ML structure and choosing the $[10,000(1 - \alpha)]$ th value in the range;
- Step 4. Run 10,000 simulations and adjust the UCL so that the average of 10,000 run lengths (ARL) is $1/\alpha$.

The best way to estimate the UCL is to employ the above algorithm. However, it might be very time- and energy-consuming, especially if one is conducting many numerical analyses and should estimate many UCLs for different parameter settings. An easier way with almost the same accuracy is to obtain the value of the UCL by generating and sorting 10,000 in-control samples in ascending order and then choosing the $[10,000(1 - \alpha)]$ th value in the range, repeating this 10,000 times, and finally taking the average of these 10,000 values. This proposed approach works very well in memory-less schemes in most cases; however, it is better to confirm its result using step 4 of the above algorithm.

Instead of using a statistic and developing a control chart using it, one can use supervised ML techniques. As in this research, we only need linear (continuous) outputs from the ML techniques (because, as mentioned before, we then apply a classification method to their linear outputs); the types of ML techniques that can be used are limited because not all ML techniques are capable of regression. However, fortunately, the most popular ML techniques, namely ANN, SVM, and RF, are also capable of generating linear (regression) outputs.

In what follows in the next subsections, each of the ML techniques is shortly described. It should also be noted that after the ML structures are trained, they are tested and validated by achieving the desired in-control performance of the control chart. This is explained later in Section 3.

2.1. ANN Control Chart

ANN is one of the most popular ML techniques that mimics human brain activity. It has one input layer, a different number of hidden layers, and one output layer, and each layer contains at least one node. ANNs with more than one hidden layer are called deep-learning techniques. ANNs can be used for both classification and regression. Determining the number of nodes in each layer and the number of hidden layers is very important in ANNs, which can be carried out using the trial-and-error method. For simple problems, one hidden layer usually works best. Regarding the number of nodes in the hidden layer, although there are some rules of thumb, there is no solid rule in this regard, and the number of nodes should be determined to obtain the desired performance.

Another problem in ANNs is determining the optimal values of the connection weights and node biases. There are several optimization algorithms used in ANNs for this purpose, among which are gradient descent, stochastic gradient descent, mini-batch gradient descent, Broyden–Fletcher–Goldfarb–Shanno (BFGS), momentum, Nesterov accelerated gradient, Adagrad, AdaDelta, and Adam. In this research, we use the BFGS optimization method,

which is a variant of the gradient descent method. The reason for this is simply that it is the only optimization method that our selected R package ('nnet' package) uses (for more information about the computer package, refer to Section 3). The BFGS method overcomes some of the weaknesses of the gradient descent method by seeking a stationary point of the cost function; the cost function, in this case, is the mean-squared error (MSE) for regression problems and the cross-entropy (negative log-likelihood) for classification problems.

In this paper, the ANN technique is used for regression, and the trained ANN's structure's output for sample i is ANN_i . Different input scenarios, training methods, and output (control) scenarios that we use in this research for each ANN structure are explained in Section 3.

2.2. SVR Control Chart

SVM is a kernel-based method and is a powerful classification and regression technique. It can be used for classification, novelty detection (one-class classification), and regression (called SVR in this case). By ignoring errors that are smaller than a certain threshold, and creating a tube around the true output, SVM can perform regression.

SVMs usually work in two main steps: (i) transformation of the input space to a higher-dimensional feature space through a non-linear mapping function; and (ii) construction of the separating hyperplane with maximum distance from the closest points (called support vectors) of the training set. It has been shown that maximizing the margin of separation improves the generalization ability of the classifier/regressor. Training of an SVM is finding the solution for a quadratic optimization problem.

In SVMs, the calculation of dot products in a high-dimensional space can be avoided by introducing a kernel function, which allows all the necessary computations to be performed directly in the input space. The most popular kernel functions are linear, polynomial, radial basis, and sigmoid. In this research, we use an R package ('e1071' package, which is explained more in Section 3) for training and optimization, and we do not apply any optimization algorithms on our own to find the optimal values of the SVM structure's parameters. However, there are several optimization algorithms for SVMs, some of which are as follows: sequential minimization optimization (SMO), trust region Newton method (TRON), and chunking. The computer package we chose uses the SMO algorithm. The SMO algorithm solves the SVM's quadratic problem without using any numerical optimization steps. The adopted package also implements epsilon-support vector regression (ϵ -SVR) as the cost function for regression problems. The cost function in ϵ -SVR aims to minimize the deviations between the predicted values and the actual values using an ϵ -insensitive loss function. It also implements the hinge loss function, which penalizes misclassifications by introducing a linear error term, in classification problems. To construct a control chart, we use the SVM technique for regression; therefore, it is called SVR.

The same as the ANN case, the trained SVR structure's output at sample i is SVR_i , and the different input scenarios, training methods, and output (control) scenarios that we use in this research for each SVR structure are explained in Section 3.

2.3. RFR Control Chart

Decision trees are tree-structured classifiers/regressors with three types of nodes: (i) the root node, which is the initial node and represents the entire sample and may be split further into more nodes; (ii) the interior nodes that represent the features of a dataset that, with the branches that represent the decision rules, are connected to other nodes; and (iii) the leaf nodes, which represent the outcome. In the regression case (the case we use in this paper), they start with the root of the tree and follow splits based on variable outcomes until a leaf node is reached and a real number-type result is given. Although decision trees work best for classification, they work very well in regression cases as well. The most popular decision tree algorithms are Iterative Dichotomiser 3 (ID3), C4.5, and CART (classification and regression tree).

Random forest (RF) is an ML technique that uses ensemble learning methods for regression or classification. The ensemble learning method is a technique that combines predictions (in the case of the RF technique, by taking the average in a regression case (called RFR in the case of regression) or choosing the class with the maximum number of occurrences in a classification case) from multiple machine-learning algorithms to make a more accurate prediction than a single model (in the RF case, the single model would be a decision tree). An RF technique is powerful and accurate, and it overcomes the over-fitting issue of individual decision trees. It usually performs great on many problems, including features with non-linear relationships. The most popular RF algorithm was introduced by Leo Breiman. It builds a forest of uncorrelated trees using a CART-like procedure. The computer package we use ('randomForest' package) employs a CART algorithm and uses the mean-squared error (MSE) as the cost function for the regression problems and the Gini impurity or the cross-entropy as the cost function for the classification problems. The main problem in using RF is choosing the number of trees to be included in the model. Using more trees is not always better as it might unnecessarily and significantly increase the computational times. The best number of trees varies from problem to problem and should be determined to obtain the minimum errors, as well as the desired performance.

The same as before, the trained RFR structure's output at sample i is RFR_i .

3. Model Development and Analysis

In this section, different control–input–training scenarios are modeled for the above three ML control charts, based on which numerical analyses are conducted afterward. We, for the first time, consider several input sets, training methods, and control scenarios to see which works best for each ML structure for building a multivariate control chart for simultaneous monitoring of process parameters. As there are two input scenarios for each ML structure, two different training methods are used for each. The two training methods are (i) training the ML structures with only a certain small shift size to familiarize them with the out-of-control situations or, (ii) training them with a certain small and a certain large shift size. We have the same number of in-control and out-of-control data in both training cases. Note that in all training scenarios, training the ML structures is performed with an output of 0 representing an in-control situation and an output of 1 representing an out-of-control situation.

In the first input scenario, the two statistics employed by Sabahno et al. [9] are used. They used them in their statistical structure to develop a multivariate control chart for simultaneous monitoring of the process parameters. We, however, use these two statistics for the inputs of the ML structures. By doing so, we easily enable the ML structures to consider multivariate processes without adding complexity. These statistics are Hotelling T^2 and W .

When the in-control process parameters are known (μ_0 and Σ_0), the Hotelling T^2 statistic is evaluated for each sample i to monitor the process mean vector (μ_0) as follows:

$$T_i^2 = n(\bar{X}_i - \mu_0)'(\Sigma_0)^{-1}(\bar{X}_i - \mu_0), \quad (1)$$

where n is the sample size and \bar{X}_i is the sample's mean.

Regarding the process variability (Σ_0), they used the following statistic:

$$W_i = \frac{(n-1)|S_i|^{1/p}}{|\Sigma_i|^{1/p}}, \quad (2)$$

where S_i is the sample's variance–covariance matrix and p is the number of quality characteristics.

In the second input scenario, however, we use all the elements of the sample mean vector and variance–covariance matrix as the inputs.

For example, in the case of two quality characteristics, we have five inputs as follows: $\bar{x}_1, \bar{x}_2, s_1^2, s_2^2,$ and s_{12} (covariance), where $\bar{X} = (\bar{x}_1, \bar{x}_2)$ and $\mathbf{S} = \begin{bmatrix} s_1^2 & s_{12} \\ s_{12} & s_2^2 \end{bmatrix}$.

Note that according to the above paragraphs, as the process dimension (the number of quality characteristics) increases, the number of inputs in the second input scenario also increases. However, the number of inputs in the first input scenario is always two (T^2 and W). This makes it easier and more efficient to use the first input scenario in higher dimensions.

We also consider two types of control schemes in this paper. In the first type, the goal is only the detection of the out-of-control situation, and in the other type, the goal is to identify the responsible process parameters as well. In the first one, we only have one chart/output with which we determine whether the process is in or out of control. In the second one, we have several charts/outputs to identify the responsible variables, as well as process parameters, for the out-of-control situation. In summary, two input scenarios, two training scenarios, and two control scenarios are involved.

Moreover, three different numbers of quality characteristics are considered, i.e., $p = 2$, $p = 3$, and $p = 4$. However, to reduce the paper's size, we only consider the $p = 3$ case with the first input scenario and the $p = 4$ case with the first input scenario and the first control scenario (only detection). It should also be noted that as the process dimension (the number of variables) increases, the amount of data and diversity of the dataset increases as well in both training methods. In addition, since using the second input scenario is directly affected by the number of variables, increasing the number of variables increases the number of inputs in this scenario. In addition, as the dimension increases, the number of control charts using the second control scenario increases as well, and that could increase the false-alarm rates.

The ARL (average run length) and SDRL (standard deviation of run length) are used in this paper to measure each chart's performance, but in this section, we only make comparisons based on the ARL. The ARL is the average number of runs before an out-of-control signal is detected by a control chart. A larger ARL value is preferred when the process is in control and as low as possible when out of control.

The 'svm' function of the 'e1071' package in R is employed to train the SVR structures, the 'nnet' package for the ANN structures, and the 'randomForest' package for the RFR structures. In general, we try to use the default hyperparameters in each package, but those we had to change are explained separately in each scenario. However, one thing we had to change in all these packages is the output type, and we changed it from a classification output (the default set) to a linear (regression) output.

Note that most of the mean shifts and variation shifts in both $p = 2$ and 3 cases are chosen similarly to those of Sabahno et al. [9] to be able to compare the proposed control charts in each scenario with their proposed statistical charts. To avoid adding additional tables for the comparisons, we do not repeat their tables here and refer readers to see their work (the ARLs in their Table 4 for $p = 2$ case and in their Table 5 for $p = 3$ case are subjects of comparisons). Therefore, we only include the comparisons' results in this paper. Moreover, similar to Sabahno et al. [9], the sample size we use in this paper is 10 ($n = 10$).

3.1. Scenario a: Control Charts for Detection

In scenario a, the cases in which only signal detection is important are investigated. Since there is only one control chart involved, assuming that the probability of type-I error is equal to 0.005, the performance measure is computed as $ARL = 1/\alpha = 200$.

3.1.1. Scenario a1 (Control Type a, Input Set 1)

As mentioned before, only two inputs are considered for each ML control chart in this scenario, and they are T^2 and W statistics.

Scenario a11 (Control Type a, Input Set 1, Training Method 1)

In this section, the shift size we select for the mean shifts is 0.2, and for the variance and covariance shifts, it is 1.2 times the in-control values. We consider 250 data with only μ_1 shifted; 250 data with only μ_2 shifted; 250 data with both means shifted; 250 data with only σ_1^2 shifted; 250 data with only σ_2 shifted; 250 data with both σ_1^2 and σ_2^2 shifted; 250 data with only covariance shifted; 250 data with σ_1^2 , σ_2^2 , and covariance shifted; and 250 data with all the parameters shifted together. For the in-control dataset, the same amount of data, i.e., $9 \times 250 = 2250$ data, are included. We train each ML structure with these in and out-of-control datasets.

In this scenario, a linear kernel is used for the SVR structure and the RMSE is computed as 0.51. For the ANN structure, four nodes (twice the number of inputs) in the hidden layer are used and the trained ANN structure has an RMSE of 0.49. For the random forest package, 100 trees are used with an RMSE of 0.53. The UCLs of the ANN, SVR, and RFR control charts are computed as 0.7073, 1.016, and 0.920144, respectively. The result of this analysis, which is reported in Table 1, shows that the SVR and ANN charts perform better than the RFR chart. In general, as the mean shift increases, the SVR chart performs better, and as the variation shift increases, the ANN chart performs better. Another interesting result is that, as the shift size increases more than the values that the ML structure is trained with, the RFR chart performs worse, such that under the mean shift of size 2, it is not even able to detect the shift at all. However, this phenomenon does not happen in ANN and SVR charts.

Moreover, by comparing the results of Table 1 to Sabahno et al.'s [9] Table 4, one can see that in all cases, at least one of the ML control charts performs better than all their proposed control charts (fixed parameters and adaptive ones). Although we do not use any adaptive strategies, the proposed ML charts perform even better (much better in most cases) than their adaptive control charts.

In the case of three quality characteristics ($p = 3$), for the out-of-control dataset, we consider 150 data with only μ_1 shifted; 150 data with only μ_2 shifted; 150 data with only μ_3 shifted; 150 data with both μ_1 and μ_2 shifted; 150 data with both μ_1 and μ_3 shifted; 150 data with both μ_2 and μ_3 shifted; 150 data with only σ_1^2 shifted; 150 data with only σ_2^2 shifted; 150 data with only σ_3^2 shifted; 150 data with both σ_1^2 and σ_2^2 shifted; 150 data with both σ_1^2 and σ_3^2 shifted; 150 data with both σ_2^2 and σ_3^2 shifted; 150 data with only covariance shifted; 150 data with all three variances and covariance shifted; and 150 data with all the parameters shifted together. For the in-control dataset, we again include the same amount of data, which in this case is 2250 data. The same hyperparameters as in the case of $p = 2$ are used here as well for the ML structures. The RMSE of the ANN, SVR, and RFR structures are computed as 0.49, 0.51, and 0.53, respectively, and the computed UCLs are 0.724, 0.9955, and 0.9305, respectively. The results in Table 2 show that the RFR chart mostly performs better in higher dimensions (compared with Table 1), especially under large mean shifts, in which the deterioration in performance under shifts larger than the trained value is much less noticeable. For the ANN and SVR charts, it is kind of mediocre (in some cases they perform better, in some they perform worse); however, they perform better than the RFR chart even in higher dimensions. The other conclusions derived for the case of $p = 2$ are valid here as well.

Moreover, by comparing the results in Table 2 to Sabahno et al.'s [9] Table 5, it is evident that in most compared cases, at least one of the proposed ML control charts performs better than all their proposed control charts (fixed design parameters and adaptive ones). Only in the case of (0.5, 0.8, 0.5) mean shift, together with no/small variation shift, does at least one of their proposed charts perform a little bit better than the best of the three proposed charts in this paper.

Table 1. ARL, SDRL. Scenario a11, $p = 2$.

	ANN	SVR	RFR	ANN	SVR	RFR
(μ_1, μ_2)	$\sigma_1^2, \sigma_2^2, \sigma_{12} = 1, 1, 0.5$			$\sigma_1^2, \sigma_2^2, \sigma_{12} = 1.05, 1, 0.5$		
(0, 0)	200, 200	200, 200	200, 200	144.92, 145.2	154.96, 152.82	161.62, 158.7
(0.1, 0)	162, 160	138.66, 137.05	186.33, 189.21	122.4, 121.94	113.02, 116.26	152.91, 157.33
(0.1, 0.3)	62.19, 60.02	47.61, 48.63	140.32, 146.05	52.04, 51.61	41.22, 40.37	130.6, 125.98
(0.3, 0)	49.56, 49.55	35.99, 35.32	128.69, 125.66	40.8, 40.52	29.97, 30.13	116.94, 115.29
(0.5, 0.8)	5.12, 4.5	3.71, 3.18	60.35, 61.59	4.76, 4.3	3.46, 2.9	59.86, 58.94
(1, 1)	1.77, 1.17	1.54, 0.9	80.38, 78.19	1.83, 1.24	1.48, 0.82	74.51, 71.41
(2, 2)	1, 0	1, 0	>10,000	1, 0	1, 0	>10,000
(μ_1, μ_2)	$\sigma_1^2, \sigma_2^2, \sigma_{12} = 1.05, 1.05, 0.5$			$\sigma_1^2, \sigma_2^2, \sigma_{12} = 1.05, 1.3, 0.5$		
(0, 0)	107.32, 109.12	125.16, 121.37	146.96, 148.91	36.78, 36.65	51.55, 49.01	83.77, 81.68
(0.1, 0)	90.78, 89.31	95.82, 94.79	131.3, 129.98	32.13, 31.66	44.63, 44.43	84.89, 83.49
(0.1, 0.3)	43.42, 43.17	34.9, 34.19	113.39, 105.59	17.99, 17.23	19.29, 18.62	72.91, 73.01
(0.3, 0)	34.78, 33.34	29.68, 30.57	104.35, 99.9	16.6, 16.41	16.84, 16.49	68.34, 66.56
(0.5, 0.8)	4.38, 3.81	3.38, 2.95	56.93, 59.03	3.18, 2.56	2.87, 2.17	45.87, 46.56
(1, 1)	1.78, 1.17	1.48, 0.87	73.73, 72.28	1.57, 0.96	1.36, 0.7	64.34, 63.55
(2, 2)	1, 0	1, 0	>10,000	1, 0	1, 0	>10,000
(μ_1, μ_2)	$\sigma_1^2, \sigma_2^2, \sigma_{12} = 1.2, 1.2, 0.6$			$\sigma_1^2, \sigma_2^2, \sigma_{12} = 1.4, 1.4, 0.5$		
(0, 0)	43.95, 42.82	64.05, 62.16	90.6, 87.93	9.8, 9.55	19.51, 18.17	48.4, 45.87
(0.1, 0)	39.19, 37.37	52.2, 51.32	87.75, 90.61	9.13, 8.71	17.15, 16.7	47.2, 46.22
(0.1, 0.3)	20.18, 19.7	22.4, 22.59	71.62, 69.17	6.78, 6.28	10.3, 9.88	41.01, 39.85
(0.3, 0)	17.46, 16.44	17.58, 17.42	70.123, 69.91	5.88, 5.26	8.56, 8.3	40.36, 38.33
(0.5, 0.8)	3.37, 2.84	2.8, 2.26	47.63, 46.9	2.25, 1.65	2.29, 1.78	37.1, 34.78
(1, 1)	1.6, 0.97	1.38, 0.73	64.8, 70.47	1.35, 0.7	1.31, 0.64	58.83, 57.21
(2, 2)	1, 0	1, 0	>10,000	1, 0	1, 0	>10,000
(μ_1, μ_2)	$\sigma_1^2, \sigma_2^2, \sigma_{12} = 1.4, 1.4, 0.75$			$\sigma_1^2, \sigma_2^2, \sigma_{12} = 3, 3, 0.5$		
(0, 0)	17.5, 16.75	32.32, 31.66	61.28, 61.73	1.23, 0.53	2.79, 2.19	96.38, 94.14
(0.1, 0)	16.19, 15.61	27.22, 26.43	57.54, 56.92	1.22, 0.55	2.69, 2.12	94.52, 93.75
(0.1, 0.3)	10.5, 9.75	13.93, 13.28	53.38, 52.48	1.2, 0.5	2.46, 1.91	100.8, 94.77
(0.3, 0)	9.3, 9.08	12.07, 11.34	51.55, 49.99	1.21, 0.51	2.42, 1.85	96.16, 96.41
(0.5, 0.8)	2.75, 2.19	2.34, 2.02	40.78, 38.25	1.09, 0.33	1.51, 0.84	167.92, 165.58
(1, 1)	1.51, 0.876	1.37, 0.688	58.84, 60.1	1.04, 0.2	1.15, 0.41	325.44, 318.73
(2, 2)	1, 0	1, 0	>10,000	1, 0	1, 0	>10,000

In the case of four quality characteristics ($p = 4$), for the out-of-control dataset, we consider 125 data with only μ_1 shifted; 125 data with only μ_2 shifted; 125 data with only μ_3 shifted; 125 data with only μ_4 shifted; 125 data with μ_1 and μ_2 shifted; 125 data with μ_1 and μ_3 shifted; 125 data with μ_2 and μ_3 shifted; 125 data with μ_1 and μ_4 shifted; 125 data with μ_2 and μ_4 shifted; 125 data with μ_3 and μ_4 shifted; 125 data with μ_1, μ_2 and μ_3 shifted; 125 data with μ_1, μ_2 and μ_4 shifted; 125 data with μ_1, μ_3 and μ_4 shifted; 125 data with μ_2, μ_3 and μ_4 shifted; 125 data with only σ_1^2 shifted; 125 data with only σ_2^2 shifted; 125 data with only σ_3^2 shifted; 125 data with only σ_4^2 shifted; 125 data with σ_1^2 and σ_2^2 shifted; 125 data with σ_1^2 and σ_3^2 shifted; 125 data with σ_2^2 and σ_3^2 shifted; 125 data with σ_2^2 and σ_4^2 shifted; 125 data with σ_3^2 and σ_4^2 shifted; 125 data with σ_1^2, σ_2^2 and σ_4^2 shifted; 125 data with σ_1^2, σ_3^2 and σ_4^2 shifted; 125 data with σ_2^2, σ_3^2 and σ_4^2 shifted; 125 data with only covariance shifted; 125 data with all four variances and covariance shifted; and 125 data with all the parameters shifted together. For the in-control dataset, we again include the same amount of data, which in this case is 3875 data. The same hyperparameters as in the case of $p = 2$ are used here as well for the ML structures. The RMSE of the ANN, SVR, and RFR structures are computed as 0.49, 0.51, and 0.52, respectively, and the computed UCLs are 0.7164, 1.0439, and 0.9332, respectively.

Table 2. ARL, SDRL. Scenario a11, $p = 3$.

	ANN	SVR	RFR	ANN	SVR	RFR
(μ_1, μ_2, μ_3)	$\sigma_1^2, \sigma_2^2, \sigma_3^2, Cov = 1, 1, 1, 0.5$			$\sigma_1^2, \sigma_2^2, \sigma_3^2, Cov = 1.05, 1, 1, 0.5$		
(0, 0, 0)	200, 200	200, 200	200, 200	153.9, 156.8	158.9, 160	180.2, 179.7
(0.1, 0, 0.1)	150.8, 145.7	147.5, 147.7	181.6, 185.1	125.6, 124.1	113.3, 114.2	164.8, 170.4
(0.1, 0.3, 0.3)	53.19, 53.38	43.97, 43.62	96.25, 91.14	45.96, 44.98	37.58, 37.34	87.61, 86.93
(0.3, 0, 0)	53.66, 52.37	44.3, 44.42	95.67, 91.49	44.13, 42.38	35.45, 35.58	92.73, 93.9
(0.5, 0.8, 0.5)	5.51, 5.03	5.28, 4.65	27.19, 26.22	5.26, 4.745	4.81, 4.126	27.98, 27.18
(1, 1, 1)	1.73, 1.198	1.69, 1.06	21.56, 20.9	1.67, 1	1.65, 1.024	21.52, 19.81
(2, 2, 2)	1.02, 0.15	1, 0.03	33.14, 33.19	1.01, 0.11	1, 0	33.36, 33.13
(μ_1, μ_2, μ_3)	$\sigma_1^2, \sigma_2^2, \sigma_3^2, Cov = 1.05, 1.05, 1.05, 0.5$			$\sigma_1^2, \sigma_2^2, \sigma_3^2, Cov = 1.05, 1.3, 1.05, 0.5$		
(0, 0, 0)	94.39, 93.96	105.5, 106.1	135.2, 133.7	34.38, 34.84	43.38, 41.55	81.86, 83.72
(0.1, 0, 0.1)	79.17, 78.14	80.75, 78.77	133.9, 128.4	31.71, 30.83	36.21, 35.35	77.94, 71.35
(0.1, 0.3, 0.3)	33.21, 32.77	28.43, 28.67	75, 78.04	17.3, 16.7	15.98, 15.44	58.63, 59.42
(0.3, 0, 0)	33.71, 32.65	28.68, 27.97	83.53, 87.01	18.44, 18.62	16.73, 15.74	55.94, 54.5
(0.5, 0.8, 0.5)	4.85, 4.32	4.28, 3.8	28.38, 27.23	3.77, 3.2	3.4, 2.847	27.64, 26.99
(1, 1, 1)	1.598, 0.99	1.53, 0.89	22.56, 21.6	1.55, 0.93	1.38, 0.71	21.04, 20.5
(2, 2, 2)	1.01, 0.09	1, 0.02	27.36, 25.29	1, 0.07	1, 0.02	19.81, 19.74
(μ_1, μ_2, μ_3)	$\sigma_1^2, \sigma_2^2, \sigma_3^2, Cov = 1.2, 1.2, 1.2, 0.6$			$\sigma_1^2, \sigma_2^2, \sigma_3^2, Cov = 1.4, 1.4, 1.05, 0.5$		
(0, 0, 0)	34.11, 33.26	44.19, 43.62	79.7, 78.44	10.17, 9.71	15.38, 14.77	34.67, 35.8
(0.1, 0, 0.1)	28.58, 27.37	37.97, 38.48	73.85, 71.42	9.56, 9.17	14.02, 13.49	33.19, 31.55
(0.1, 0.3, 0.3)	16.5, 15.76	15.85, 14.85	59, 59.56	7.16, 6.69	8.18, 7.43	30.1, 29.57
(0.3, 0, 0)	16.31, 15.65	16.3, 15.62	57.71, 51.46	6.82, 6.53	7.86, 7.21	30.88, 31.9
(0.5, 0.8, 0.5)	3.72, 3.16	3.14, 2.56	27.61, 26.05	2.65, 2.07	2.44, 1.8	22.63, 21.41
(1, 1, 1)	1.51, 0.89	1.43, 0.77	21.26, 21.87	1.34, 0.68	0.68, 0.6	17.97, 18.27
(2, 2, 2)	1, 0.06	1, 0	20.33, 19.3	1, 0.02	1, 0	16.29, 15.34
(μ_1, μ_2, μ_3)	$\sigma_1^2, \sigma_2^2, \sigma_3^2, Cov = 1.4, 1.4, 1.05, 0.75$			$\sigma_1^2, \sigma_2^2, \sigma_3^2, Cov = 3, 3, 1.4, 0.5$		
(0, 0, 0)	45.86, 44.37	52.07, 50.04	104.4, 105.9	1.14, 0.41	1.92, 1.35	4.28, 3.87
(0.1, 0, 0.1)	42.02, 40.68	43.13, 42.54	99.6, 105.9	1.14, 0.41	1.87, 1.26	4.26, 3.69
(0.1, 0.3, 0.3)	20.61, 19.97	18.29, 17.97	66.77, 66.11	1.14, 0.4	1.67, 1.03	4.68, 4.35
(0.3, 0, 0)	20.79, 20.67	18.76, 18.69	67.56, 63.59	1.12, 0.36	1.77, 1.12	4.76, 4.25
(0.5, 0.8, 0.5)	3.97, 3.49	3.6, 3.04	28.77, 28.26	1.05, 0.23	1.27, 0.58	7.36, 6.835
(1, 1, 1)	1.6, 0.99	1.55, 0.9	21.1, 19.75	1.01, 0.13	1.05, 0.23	21.84, 20.87
(2, 2, 2)	1, 0.08	1, 0.02	23.68, 22.4	1, 0	1, 0	54.86, 55.44

The results in Table 3 show that the RFR chart mostly performs better again in higher dimensions (compared with both Tables 1 and 2). For the ANN and SVR charts, it is again kind of mediocre (in some cases they perform better, in some they perform worse); however, they still perform better than the RFR chart even in higher dimensions. The other conclusions derived for the case of $p = 2$ are valid here as well.

Scenario a12 (Control Type a, Input Set 1, Training Method 2)

In this scenario, we use the same input set as before, but with the second training method (trained with both small and large shifts). Here, for the out-of-control dataset, the shift size for the small mean shifts is 0.2; for the large mean shifts, it is 1; for the small variance and covariance shifts, it is 1.2 times the in-control values; and for the large variance and covariance shifts, it is 2 times the in-control values. For the small shifts, the following are considered: 150 data with only μ_1 shifted; 150 data with only μ_2 shifted; 150 data with both means shifted; 150 data with only σ_1^2 shifted; 150 data with only σ_2^2 shifted; 150 data with both variances shifted; 150 data with only covariance shifted; 150 data with both variances and covariance shifted; and 150 data with all the parameters shifted together. Similarly, for the large mean shifts, we have 150 data with only μ_1 shifted; 150 data with only μ_2 shifted; 150 data with both means shifted; 150 data with only σ_1^2 shifted; 150 data with only σ_2^2 shifted; 150 data with both variances shifted; 150 data with only covariance

shifted; 150 data with both varianbces and covariance shifted; and 150 data with all the parameters shifted together. The total number of out-of-control data in this scenario is 2700; therefore, the same number of in-control data is used. For all the ML structures, the same hyperparameters as in the previous case are used. The RMSEs for the ANN, SVR, and RFR structures are 0.44, 0.53, and 0.47, respectively. The UCLs of ANN, SVR, and RFR control charts are computed as 0.8658, 0.5958, and 0.9729, respectively. The results for this case are reported in Table 4.

Table 3. ARL, SDRL. Scenario a11, $p = 4$.

	ANN	SVR	RFR	ANN	SVR	RFR
$(\mu_1, \mu_2, \mu_3, \mu_4)$	$\sigma_1^2, \sigma_2^2, \sigma_3^2, \sigma_4^2, Cov = 1, 1, 1, 1, 0.5$			$\sigma_1^2, \sigma_2^2, \sigma_3^2, \sigma_4^2, Cov = 1.05, 1, 1, 1, 0.5$		
(0, 0, 0, 0)	200, 200	200, 200	200, 200	152.8, 154.7	175.2, 177	180.2, 174.3
(0.1, 0, 0.1, 0)	167.8, 173.3	138.6, 138	169.2, 171.5	132.8, 134.3	125, 120.6	152.7, 153.2
(0.1, 0.3, 0.3, 0.1)	79.52, 80.04	45.18, 42.7	101.4, 105.8	66.18, 65.77	40.66, 40.88	92.34, 92.61
(0.3, 0, 0, 0.3)	51.29, 50.1	27.61, 26.23	77.29, 73.17	43.4, 42.92	26.25, 25.07	71.88, 68.42
(0.5, 0.8, 0.5, 0.8)	5.4, 5.05	3.41, 2.83	19.91, 19.3	5.14, 4.65	3.31, 2.79	20.03, 19.75
(1, 1, 1, 1)	1.84, 1.26	1.5, 0.86	17.81, 17.81	1.83, 1.22	1.5, 0.88	17.47, 17.37
(2, 2, 2, 2)	1, 0	1, 0.02	>10,000	1, 0	1, 0	>10,000
$(\mu_1, \mu_2, \mu_3, \mu_4)$	$\sigma_1^2, \sigma_2^2, \sigma_3^2, \sigma_4^2, Cov = 1.05, 1.05, 1.05, 1.05, 0.5$			$\sigma_1^2, \sigma_2^2, \sigma_3^2, \sigma_4^2, Cov = 1.05, 1.3, 1.05, 1.3, 0.5$		
(0, 0, 0, 0)	80.29, 78.62	103.8, 104.2	118.4, 118.3	16.56, 15.96	29.58, 27.9	42.75, 42.33
(0.1, 0, 0.1, 0)	69.13, 69.04	78.11, 79.96	108.2, 117.8	15.13, 14.66	26.21, 26.3	41.4, 42.44
(0.1, 0.3, 0.3, 0.1)	40.03, 39.38	28.81, 27.94	71.83, 64.33	11.11, 10.62	13.11, 12.92	32.93, 31.47
(0.3, 0, 0, 0.3)	28.05, 28.22	18.5, 17.23	56.97, 56.89	9.82, 9.27	9.6, 9.15	29.23, 28.9
(0.5, 0.8, 0.5, 0.8)	4.57, 4.03	3.04, 2.52	21.23, 20.74	2.97, 2.37	2.38, 1.85	18.68, 18.81
(1, 1, 1, 1)	1.75, 1.12	1.4, 0.75	19.71, 19.52	1.53, 0.92	1.32, 0.66	24.26, 25.35
(2, 2, 2, 2)	1, 0	1, 0	>10,000	1, 0	1, 0	>10,000
$(\mu_1, \mu_2, \mu_3, \mu_4)$	$\sigma_1^2, \sigma_2^2, \sigma_3^2, \sigma_4^2, Cov = 1.2, 1.2, 1.2, 1.2, 0.6$			$\sigma_1^2, \sigma_2^2, \sigma_3^2, \sigma_4^2, Cov = 1.4, 1.4, 1.05, 1.05, 0.5$		
(0, 0, 0, 0)	25.26, 24.02	47.95, 46.88	62.73, 59.53	10.38, 9.69	20.91, 20.63	28.59, 28.24
(0.1, 0, 0.1, 0)	24.18, 23.71	38.09, 38.59	54.33, 55.11	10.39, 9.58	18.75, 18.68	29.28, 27.89
(0.1, 0.3, 0.3, 0.1)	16.12, 15.35	17.27, 16.97	42.41, 41.98	7.93, 7.43	10.35, 9.55	25.98, 25.84
(0.3, 0, 0, 0.3)	13.12, 12.61	11.73, 10.91	39.12, 42.09	6.86, 6.32	7.58, 6.97	22.48, 23.2
(0.5, 0.8, 0.5, 0.8)	3.44, 2.92	2.59, 2.05	21, 20.11	2.64, 2.08	2.21, 1.66	16.46, 16.63
(1, 1, 1, 1)	1.6, 0.95	1.37, 0.71	23.52, 21.43	1.47, 0.84	1.24, 0.56	23.3, 23.02
(2, 2, 2, 2)	1, 0	1, 0	>10,000	1, 0	1, 0	>10,000
$(\mu_1, \mu_2, \mu_3, \mu_4)$	$\sigma_1^2, \sigma_2^2, \sigma_3^2, \sigma_4^2, Cov = 1.4, 1.4, 1.05, 1.05, 0.75$			$\sigma_1^2, \sigma_2^2, \sigma_3^2, \sigma_4^2, Cov = 3, 3, 1.4, 1.4, 0.5$		
(0, 0, 0, 0)	135.3, 130.2	107.6, 108.6	166.1, 162.6	1.13, 0.39	2.28, 1.65	2.69, 2.12
(0.1, 0, 0.1, 0)	117.1, 118.2	79.76, 79.14	145.8, 141.2	1.14, 0.41	2.27, 1.67	2.82, 2.31
(0.1, 0.3, 0.3, 0.1)	55.12, 54.93	29.68, 29.4	83.9, 80.92	1.11, 0.36	2.12, 1.57	2.85, 2.23
(0.3, 0, 0, 0.3)	38.91, 38.39	22.2, 22.06	68.92, 69.37	1.12, 0.36	1.87, 1.32	2.98, 2.53
(0.5, 0.8, 0.5, 0.8)	4.78, 4.327	3.34, 2.78	22.08, 21.25	1.06, 0.26	1.3, 0.65	4.34, 3.7
(1, 1, 1, 1)	1.88, 1.32	1.54, 0.93	20.19, 20.95	1.02, 0.17	1.07, 0.28	9.06, 8.13
(2, 2, 2, 2)	1, 0	1, 0.02	>10,000	1, 0	1, 0	>10,000

The results in Table 4 show that the ANN and SVR charts perform mostly better with the second training method, only in the cases of no/small variation shifts. On the contrary, this training method suits the RFR chart the most, considering it performs better than the previous training method in all the cases. In terms of which chart among the three performs better in this scenario, despite the significant improvements in the performance of the RFR chart, it still cannot outperform the ANN and SVR charts, as the ANN chart performs better in most compared cases, and in the rest (no or small variation shifts), despite the loss of performance in this training method, the SVR chart still performs better than the others. In addition, unlike the previous training method, the RFR chart does not experience a deterioration in performance as the shift size becomes larger than the trained value.

Moreover, by comparing the results of Table 4 to Sabahno et al.'s [9] Table 4, one can again see that in all the cases, at least one of the proposed ML control charts performs better (mostly much better) than all their charts.

Table 4. ARL, SDRL. Scenario a12, $p = 2$.

	ANN	SVR	RFR	ANN	SVR	RFR
(μ_1, μ_2)	$\sigma_1^2, \sigma_2^2, \sigma_{12} = 1, 1, 0.5$			$\sigma_1^2, \sigma_2^2, \sigma_{12} = 1.05, 1, 0.5$		
(0, 0)	200, 200	200, 200	200, 200	155.48, 153.15	165.01, 161.56	156.21, 152.98
(0.1, 0)	154.13, 167.78	151.92, 149.84	158.22, 155.65	125.77, 118.97	119.68, 124.59	129.18, 121.49
(0.1, 0.3)	52.94, 50.03	43.71, 43.53	56.77, 56.35	46.37, 46.44	42.12, 41.52	56.99, 55.06
(0.3, 0)	37.56, 39.89	33.85, 32.85	47.98, 44.91	34.51, 34.49	28.73, 27.4	41.32, 39.63
(0.5, 0.8)	3.78, 3.23	3.38, 2.88	4.5, 3.95	3.69, 3.19	3.28, 2.72	4.55, 4.01
(1, 1)	1.53, 0.91	1.4, 0.76	1.75, 1.13	1.5, 0.87	1.43, 0.78	1.77, 1.2
(2, 2)	1, 0	1, 0	1.09, 0.3	1, 0	1, 0	1.09, 0.32
(μ_1, μ_2)	$\sigma_1^2, \sigma_2^2, \sigma_{12} = 1.05, 1.05, 0.5$			$\sigma_1^2, \sigma_2^2, \sigma_{12} = 1.05, 1.3, 0.5$		
(0, 0)	123.73, 124.92	141.1, 142.5	120.62, 118.66	45.01, 44.81	66.29, 65.08	47.93, 48.98
(0.1, 0)	102.23, 102.65	98.54, 95.45	104.1, 104.57	37.23, 36.58	56.49, 55.84	43.89, 43.52
(0.1, 0.3)	40.99, 41.03	35.92, 36.23	46.84, 47.33	20.49, 19.86	22.62, 21.77	22.76, 22.35
(0.3, 0)	30.97, 30.54	27.23, 26.92	37, 35.01	17.84, 17.01	20.25, 19.52	21.47, 20.87
(0.5, 0.8)	3.7, 3.15	3.15, 2.56	4.28, 3.82	3.25, 2.75	2.83, 2.19	3.94, 3.32
(1, 1)	1.47, 0.83	1.41, 0.76	1.74, 1.08	1.47, 0.8	1.39, 0.75	1.77, 1.18
(2, 2)	1, 0	1, 0	1.08, 0.3	1, 0	1, 0	1.08, 0.29
(μ_1, μ_2)	$\sigma_1^2, \sigma_2^2, \sigma_{12} = 1.2, 1.2, 0.6$			$\sigma_1^2, \sigma_2^2, \sigma_{12} = 1.4, 1.4, 0.5$		
(0, 0)	51.69, 50.41	80.46, 77.9	58.56, 56.6	11.5, 11.04	29.25, 29.02	13.79, 13.32
(0.1, 0)	46.38, 47.18	62.89, 60.51	48.64, 49.39	10.32, 9.458	24.92, 25.17	13.44, 12.6
(0.1, 0.3)	23.45, 23.36	25.63, 25.32	26.47, 25.15	7.87, 7.24	14.1, 13.59	9.72, 9.48
(0.3, 0)	20.17, 19.56	21.13, 19.76	21.94, 21.43	7.12, 6.69	11.42, 11.33	8.58, 7.85
(0.5, 0.8)	3.33, 2.8	3.05, 2.51	3.87, 3.39	2.52, 1.95	2.66, 2.06	3.08, 2.42
(1, 1)	1.53, 0.91	1.41, 0.75	1.74, 1.13	1.42, 0.76	1.33, 0.67	1.66, 1.05
(2, 2)	1, 0	1, 0	1.09, 0.31	1, 0	1, 0	1.08, 0.29
(μ_1, μ_2)	$\sigma_1^2, \sigma_2^2, \sigma_{12} = 1.4, 1.4, 0.75$			$\sigma_1^2, \sigma_2^2, \sigma_{12} = 3, 3, 0.5$		
(0, 0)	21.33, 20.72	46.28, 46.25	25.54, 24.71	1.2, 0.49	4.14, 3.6	1.49, 0.87
(0.1, 0)	19.5, 19.1	39.61, 38.78	24.42, 23.76	1.17, 0.46	4.08, 3.53	1.48, 0.84
(0.1, 0.3)	12.64, 11.76	18.49, 17.03	14.88, 14.3	1.16, 0.42	3.43, 2.9	1.47, 0.84
(0.3, 0)	11.19, 10.53	15.26, 14.42	13.72, 13.02	1.16, 0.45	3.36, 2.82	1.51, 0.89
(0.5, 0.8)	2.96, 2.36	2.79, 2.18	3.43, 2.97	1.08, 0.3	1.85, 1.2	1.27, 0.59
(1, 1)	1.49, 0.86	1.43, 0.76	1.77, 1.18	1.03, 0.18	1.31, 0.66	1.13, 0.39
(2, 2)	1, 0	1, 0	1.1, 0.32	1, 0	1, 0.3	1.02, 0.15

In the case of $p = 3$, for the out-of-control dataset, the following are considered: 125 data with μ_1 shifted 0.2 and 125 data shifted 1; 125 data with μ_2 shifted 0.2 and 125 data shifted 1; 125 data with only μ_3 shifted 0.2 and 125 data shifted 1; 125 data with μ_1 and μ_2 shifted 0.2 and 125 data shifted 1; 125 data with μ_1 and μ_3 shifted 0.2 and 125 data shifted 1; 125 data with μ_2 and μ_3 shifted 0.2 and 125 data shifted 1; 125 data with σ_1^2 shifted 1.2 times and 125 data shifted 2 times; 125 data with σ_2^2 shifted 1.2 times and 125 data shifted 2 times; 125 data with σ_3^2 shifted 1.2 times and 125 data shifted 2 times; 125 data with σ_1^2 and σ_2^2 shifted 1.2 times and 125 data shifted 2 times; 125 data with σ_1^2 and σ_3^2 shifted 1.2 times and 125 data shifted 2 times; 125 data with σ_2^2 and σ_3^2 shifted 1.2 times and 125 data shifted 2 times; 125 data with only covariance shifted 1.2 times and 125 data shifted 2 times; 125 data with all three variances and covariance shifted 1.2 times and 125 data shifted 2 times; 125 data with all the parameters shifted together with small shift sizes; and 125 data shifted together with large shift sizes. For the in-control dataset, we include the same amount of data, which in this scenario is 3750 data. By keeping the previous hyperparameters, the RMSE of the ANN, SVR, and RFR structures are 0.44, 0.52, and 0.5,

respectively. The UCLs are computed as 0.8635, 0.5781, and 0.9783, respectively. The results are presented in Table 5.

Table 5. ARL, SDRL. Scenario a12, $p = 3$.

	ANN	SVR	RFR	ANN	SVR	RFR
(μ_1, μ_2, μ_3)	$\sigma_1^2, \sigma_2^2, \sigma_3^2, Cov = 1, 1, 1, 0.5$			$\sigma_1^2, \sigma_2^2, \sigma_3^2, Cov = 1.05, 1, 1, 0.5$		
(0, 0, 0)	200, 200	200, 200	200, 200	162.9, 170.9	167.6, 163	177.2, 186
(0.1, 0, 0.1)	148.9, 145.8	141.3, 136.4	145.8, 148.7	123.2, 125.3	119.5, 118.7	132.2, 131.8
(0.1, 0.3, 0.3)	44.9, 45.35	38.54, 37.57	52.27, 52.87	41.05, 38.39	34.83, 34.83	44, 44.2
(0.3, 0, 0)	44.83, 44.09	36.98, 35.26	55.52, 53.86	38.93, 37.94	33.9, 32.46	43.5, 43.37
(0.5, 0.8, 0.5)	4.55, 3.9	3.93, 3.21	5.74, 4.92	4.44, 4.04	3.9, 3.55	5.53, 4.97
(1, 1, 1)	1.47, 0.81	1.42, 0.79	1.69, 1	1.51, 0.87	1.4, 0.73	1.78, 1.17
(2, 2, 2)	1, 0	1, 0	1.05, 0.23	1, 0	1, 0	1.06, 0.25
(μ_1, μ_2, μ_3)	$\sigma_1^2, \sigma_2^2, \sigma_3^2, Cov = 1.05, 1.05, 1.05, 0.5$			$\sigma_1^2, \sigma_2^2, \sigma_3^2, Cov = 1.05, 1.3, 1.05, 0.5$		
(0, 0, 0)	109.1, 110.3	117.8, 117.4	114.2, 120.1	46.17, 45.71	65.58, 67.24	48.03, 46.07
(0.1, 0, 0.1)	87.09, 88.44	93.44, 91.19	90.04, 94.33	39.46, 39.27	51.23, 49.16	40.36, 41.61
(0.1, 0.3, 0.3)	33.59, 32.73	30.85, 31.25	37.79, 36.54	19.72, 19.87	20.69, 21.13	21.6, 21.77
(0.3, 0, 0)	33.12, 33.06	29.23, 29.36	36.43, 35.96	20.39, 19.7	21.64, 21.35	21.92, 20.55
(0.5, 0.8, 0.5)	4.37, 3.84	3.81, 3.2	5.17, 4.93	3.76, 3.27	3.49, 2.82	4.42, 3.87
(1, 1, 1)	1.49, 0.88	1.35, 0.7	1.76, 1.16	1.47, 0.87	1.38, 0.71	1.7, 1.11
(2, 2, 2)	1, 0	1, 0	1.06, 0.24	1, 0	1, 0	1.07, 0.29
(μ_1, μ_2, μ_3)	$\sigma_1^2, \sigma_2^2, \sigma_3^2, Cov = 1.2, 1.2, 1.2, 0.6$			$\sigma_1^2, \sigma_2^2, \sigma_3^2, Cov = 1.4, 1.4, 1.05, 0.5$		
(0, 0, 0)	42.13, 41.58	68.06, 67.12	44.54, 42.65	12.74, 12.42	25.15, 24.26	13.02, 12.54
(0.1, 0, 0.1)	37.37, 35.82	54.46, 55.72	39, 39.25	12.14, 12.15	22.95, 22.95	12.32, 12.16
(0.1, 0.3, 0.3)	20, 19.04	20.41, 20.32	20.24, 19.76	8.43, 7.74	13.01, 12.13	9.54, 8.17
(0.3, 0, 0)	19.19, 19.2	20.66, 19.71	20.83, 20.14	8.34, 7.62	12.11, 11.96	8.66, 8.44
(0.5, 0.8, 0.5)	3.83, 3.29	3.32, 2.82	4.42, 4.03	3, 2.4	2.9, 2.38	3.36, 2.76
(1, 1, 1)	1.49, 0.9	1.38, 0.73	1.69, 1.1	1.37, 0.73	1.32, 0.63	1.52, 0.88
(2, 2, 2)	1, 0	1, 0	1.06, 0.25	1, 0	1, 0	1.06, 0.27
(μ_1, μ_2, μ_3)	$\sigma_1^2, \sigma_2^2, \sigma_3^2, Cov = 1.4, 1.4, 1.05, 0.75$			$\sigma_1^2, \sigma_2^2, \sigma_3^2, Cov = 3, 3, 1.4, 0.5$		
(0, 0, 0)	54.68, 54.76	69.61, 72.43	56.6, 55.83	1.2, 0.48	3.72, 3.25	1.36, 0.72
(0.1, 0, 0.1)	48.04, 48.68	52.62, 50.82	49.07, 49.57	1.19, 0.48	3.53, 3.1	1.36, 0.72
(0.1, 0.3, 0.3)	21.29, 21.31	20.82, 20.21	24.63, 23.86	1.17, 0.45	3.02, 2.5	1.3, 0.62
(0.3, 0, 0)	22.63, 22.02	19.92, 19.11	24.3, 24.31	1.19, 0.47	3.06, 2.42	1.34, 0.7
(0.5, 0.8, 0.5)	3.84, 3.32	3.39, 2.85	4.59, 4.16	1.1, 0.34	1.82, 1.23	1.25, 0.55
(1, 1, 1)	1.53, 0.89	1.47, 0.84	1.74, 1.2	1.03, 0.19	1.21, 0.52	1.08, 0.28
(2, 2, 2)	1, 0	1, 0, 02	1.08, 0.3	1, 0	1, 0	1.01, 0.11

This table shows that again under no/small variation shifts, the SVR chart mostly performs better than the other two, but as the variation shift increases, the other two begin to perform better than the SVR chart, with the ANN chart having the best performance. By comparing Tables 2 and 5 (the first and second training methods), the conclusions driven for the $p = 2$ case (comparing Tables 1 and 4) can be driven for the $p = 3$ case as well. By comparing the cases of $p = 3$ and $p = 2$ (Tables 4 and 5), we realize that the conclusions are similar to those of Scenario a11.

Moreover, by comparing the results of Table 5 to Sabahno et al.'s [9] Table 5, we can see that with the second training method, at least one of the ML control charts performs better than all their control charts, in all the shift cases.

In the case of four quality characteristics ($p = 4$), for the out-of-control dataset, we consider 100 data with only μ_1 shifted 0.2 and 100 data shifted 1; 100 data with only μ_2 shifted 0.2 and 100 data shifted 1; 100 data with only μ_3 shifted 0.2 and 100 data shifted 1; 100 data with only μ_4 shifted 0.2 and 100 data shifted 1; 100 data with μ_1 and μ_2 shifted 0.2 and 100 data shifted 1; 100 data with μ_1 and μ_3 shifted 0.2 and 100 data shifted 1; 100 data with μ_2 and μ_3 shifted 0.2 and 100 data shifted 1; 100 data with μ_1 and μ_4 shifted 0.2 and

100 data shifted 1; 100 data with μ_2 and μ_4 shifted 0.2 and 100 data shifted 1; 100 data with μ_3 and μ_4 shifted 0.2 and 100 data shifted 1; 100 data with μ_1 , μ_2 and μ_3 shifted 0.2 and 100 data shifted 1; 100 data with μ_1 , μ_2 and μ_4 shifted 0.2 and 100 data shifted 1; 100 data with μ_1 , μ_3 and μ_4 shifted 0.2 and 100 data shifted 1; 100 data μ_2 , μ_3 and μ_4 shifted 0.2 and 100 data shifted 1; 100 data with only σ_1^2 shifted 1.2 times and 100 data shifted 2 times; 100 data with only σ_2^2 shifted 1.2 times and 100 data shifted 2 times; 100 data with only σ_3^2 shifted 1.2 times and 100 data shifted 2 times; 100 data with only σ_4^2 shifted 1.2 times and 100 data shifted 2 times; 100 data with σ_1^2 and σ_2^2 shifted 1.2 times and 100 data shifted 2 times; 100 data with σ_1^2 and σ_3^2 shifted 1.2 times and 100 data shifted 2 times; 100 data with σ_1^2 and σ_4^2 shifted 1.2 times and 100 data shifted 2 times; 100 data with σ_2^2 and σ_3^2 shifted 1.2 times and 100 data shifted 2 times; 100 data with σ_2^2 and σ_4^2 shifted 1.2 times and 100 data shifted 2 times; 100 data with σ_3^2 and σ_4^2 shifted 1.2 times and 100 data shifted 2 times; 100 data with σ_1^2 , σ_2^2 and σ_3^2 shifted 1.2 times and 100 data shifted 2 times; 100 data with σ_1^2 , σ_2^2 and σ_4^2 shifted 1.2 times and 100 data shifted 2 times; 100 data with σ_1^2 , σ_3^2 and σ_4^2 shifted 1.2 times and 100 data shifted 2 times; 100 data with σ_2^2 , σ_3^2 and σ_4^2 shifted 1.2 times and 100 data shifted 2 times; 100 data with only covariance shifted 1.2 times and 100 data shifted 2 times; 100 data with all four variances and the covariance shifted 1.2 times and 100 data shifted 2 times; and 100 data with all the parameters shifted together with small shift sizes and 100 data shifted together with large shift sizes. For the in-control dataset, we include the same amount of data, which in this scenario is 6200 data.

By keeping the previous hyperparameters, the RMSE of the ANN, SVR, and RFR structures are 0.43, 0.53, and 0.46, respectively. The UCLs are computed as 0.856, 0.5108, and 0.9735, respectively. The results are presented in Table 6.

This table shows that again under no/small variation shifts, the SVR chart mostly performs better than the other two, but as the variation shift increases, the other two begin to perform better than the SVR chart, with the ANN chart having the best performance. By comparing Tables 3 and 6 (first and second training methods), the conclusions driven for the $p = 2$ case (comparing Tables 1 and 4) can be driven for the $p = 4$ case as well. The results in Table 6 show that all the charts mostly perform better in higher dimensions when the second training method is used.

3.1.2. Scenario a2 (Control Type a, Input Set 2)

As mentioned before, different inputs for the ML structures are considered in this scenario. The sample's mean vector and variance–covariance matrix are used and in the case of $p = 2$, there are going to be five inputs (individual variances, means, and covariance) as described in the following subsections.

Scenario a21 (Control Type a, Input Set 2, Training Method 1)

In-control and out-of-control datasets are the same as in Scenario a11. In this scenario, again, a linear kernel is used for the SVR structure, and the RMSE is computed as 0.51. For the ANN control chart, 10 nodes (twice the number of inputs) are used in the hidden layer and the trained ANN structure has an RMSE of 0.48. For the random forest structure, again 100 trees are used with an RMSE of 0.5. The UCLs of the ANN, SVR, and RFR control charts are computed as 0.8746, 1.0631, and 0.8143, respectively. The results of this analysis are reported in Table 7. Based on these results, the SVR chart performs better than the other two under all shift cases. By comparing their performance against the previous input scenario (comparing Tables 1 and 7), we realize that the ANN chart mostly performs worse when the second input scenario is used. The situation is kind of mediocre with the SVR chart, and it seems that the input method overall does not affect its performance. The RFR chart on the other hand, especially under moderate and large mean shifts, mostly performs better in this input scenario, and the deterioration in performance as the shift size becomes larger than the trained value is much less when this input set is used.

Table 6. ARL, SDRL. Scenario a12, $p = 4$.

	ANN	SVR	RFR	ANN	SVR	RFR
$(\mu_1, \mu_2, \mu_3, \mu_4)$	$\sigma_1^2, \sigma_2^2, \sigma_3^2, \sigma_4^2, Cov = 1, 1, 1, 1, 0.5$			$\sigma_1^2, \sigma_2^2, \sigma_3^2, \sigma_4^2, Cov = 1.05, 1, 1, 1, 0.5$		
(0, 0, 0, 0)	200, 200	200, 200	200, 200	163.9, 158.3	174.04, 169.22	160.2, 170.4
(0.1, 0, 0.1, 0)	148, 149.2	133.58, 131.92	152.2, 159.5	127.9, 126.5	121.35, 122.85	127.7, 126.7
(0.1, 0.3, 0.3, 0.1)	52.81, 51.7	44.38, 43.31	58.32, 56.33	46.43, 44.73	37.04, 36.51	54.4, 54.9
(0.3, 0, 0, 0.3)	34.3, 34.65	25.74, 25.18	40.84, 40.07	29.85, 29.68	23.79, 23.6	35.56, 36.57
(0.5, 0.8, 0.5, 0.8)	3.72, 3.2	3.21, 2.67	4.79, 4.23	3.73, 3.19	3.15, 2.57	4.47, 3.93
(1, 1, 1, 1)	1.58, 0.98	1.44, 0.78	1.83, 1.19	1.53, 0.89	1.4, 0.74	1.79, 1.203
(2, 2, 2, 2)	1, 0	1, 0	1.02, 0.15	1, 0	1, 0	1.02, 0.16
$(\mu_1, \mu_2, \mu_3, \mu_4)$	$\sigma_1^2, \sigma_2^2, \sigma_3^2, \sigma_4^2, Cov = 1.05, 1.05, 1.05, 1.05, 0.5$			$\sigma_1^2, \sigma_2^2, \sigma_3^2, \sigma_4^2, Cov = 1.05, 1.3, 1.05, 1.3, 0.5$		
(0, 0, 0, 0)	90.44, 90.41	113.47, 112.57	92.41, 100.2	19.97, 19.34	38.03, 37.17	22.14, 21.16
(0.1, 0, 0.1, 0)	73.3, 72.54	80.82, 80.86	75.77, 81.53	18.39, 18.98	31.29, 30.39	20.76, 20.77
(0.1, 0.3, 0.3, 0.1)	32.89, 34.2	29.7, 30.17	35.04, 36.03	11.77, 11.75	15.89, 15.8	13.84, 12.81
(0.3, 0, 0, 0.3)	21.97, 21.22	18.94, 18.27	26.83, 27.45	9.25, 8.63	10.53, 9.91	10.68, 10.29
(0.5, 0.8, 0.5, 0.8)	3.3, 2.62	2.9, 2.38	4.18, 3.6	2.51, 1.92	2.43, 1.88	3.2, 2.61
(1, 1, 1, 1)	1.47, 0.8	1.37, 0.72	1.8, 1.17	1.36, 0.7	1.31, 0.62	1.65, 1.05
(2, 2, 2, 2)	1, 0	1, 0	1.02, 0.17	1, 0	1, 0	1.03, 0.19
$(\mu_1, \mu_2, \mu_3, \mu_4)$	$\sigma_1^2, \sigma_2^2, \sigma_3^2, \sigma_4^2, Cov = 1.2, 1.2, 1.2, 1.2, 0.6$			$\sigma_1^2, \sigma_2^2, \sigma_3^2, \sigma_4^2, Cov = 1.4, 1.4, 1.05, 1.05, 0.5$		
(0, 0, 0, 0)	31.71, 31.05	55.85, 54.67	36.09, 33.88	13.98, 13.43	27.48, 27.52	15.11, 15.02
(0.1, 0, 0.1, 0)	29.45, 28.48	45.9, 47.48	28.89, 28.02	12.06, 11.21	22.92, 22.49	14.42, 13.81
(0.1, 0.3, 0.3, 0.1)	16.28, 15.78	19.25, 18.42	18.4, 17.29	8.66, 8.11	12.33, 12.17	10.21, 10.01
(0.3, 0, 0, 0.3)	11.75, 11.49	13.26, 12.59	14.95, 14.45	6.87, 6.63	9.14, 8.6	8.51, 7.64
(0.5, 0.8, 0.5, 0.8)	2.75, 2.21	2.59, 2.08	3.56, 3.1	2.28, 1.69	2.3, 1.69	2.97, 2.32
(1, 1, 1, 1)	1.46, 0.81	1.36, 0.69	1.68, 1.06	1.31, 0.65	1.3, 0.61	1.54, 0.88
(2, 2, 2, 2)	1, 0	1, 0	1.04, 0.21	1, 0	1, 0	1.05, 0.24
$(\mu_1, \mu_2, \mu_3, \mu_4)$	$\sigma_1^2, \sigma_2^2, \sigma_3^2, \sigma_4^2, Cov = 1.4, 1.4, 1.05, 1.05, 0.75$			$\sigma_1^2, \sigma_2^2, \sigma_3^2, \sigma_4^2, Cov = 3, 3, 1.4, 1.4, 0.5$		
(0, 0, 0, 0)	111.4, 112.1	96.52, 96.41	120.3, 118.1	1.15, 0.41	2.99, 2.39	1.3, 0.63
(0.1, 0, 0.1, 0)	92.3, 90.43	79.73, 80.99	106.6, 99.79	1.18, 0.46	2.95, 2.49	1.3, 0.63
(0.1, 0.3, 0.3, 0.1)	35.14, 33.45	28.81, 28.11	41.53, 40.01	1.16, 0.42	2.64, 2.01	1.27, 0.57
(0.3, 0, 0, 0.3)	25.72, 25.52	21.05, 20.43	30.23, 28.25	1.17, 0.45	2.37, 1.84	1.26, 0.59
(0.5, 0.8, 0.5, 0.8)	3.55, 3.06	3.11, 2.68	4.1, 3.57	1.06, 0.26	1.44, 0.80	1.17, 0.44
(1, 1, 1, 1)	1.59, 0.95	1.5, 0.84	1.86, 1.25	1.02, 0.16	1.12, 0.37	1.07, 0.27
(2, 2, 2, 2)	1, 0.02	1, 0	1.03, 0.17	1, 0	1, 0	1.03, 0.17

Moreover, by comparing the results in Table 7 to Sabahno et al.'s [9] Table 4, it can be seen that in most cases, at least one of the proposed ML control charts performs better (mostly much better) than all their proposed control charts. Only in the case of (0.5, 0.8) mean shift, together with no/small variation shift, at least one of their proposed charts performs a little bit better than the best of ours.

Scenario a22 (Control Type a, Input Set 2, Training Method 2)

In-control and out-of-control datasets are the same as in Scenario a12. The same as before, a linear kernel is used for the SVR structure, and the RMSE is computed as 0.48. For the ANN control chart, 10 nodes (twice the number of inputs) in the hidden layer are used and the trained ANN structure has an RMSE of 0.42. For the random forest structure, 100 trees are used again with an RMSE of 0.45. The UCLs of the ANN, SVR, and RFR control charts are computed as 0.8846, 0.8136, and 0.894, respectively.

Table 7. ARL, SDRL. Scenario a21, $p = 2$.

	ANN	SVR	RFR	ANN	SVR	RFR
(μ_1, μ_2)	$\sigma_1^2, \sigma_2^2, \sigma_{12} = 1, 1, 0.5$			$\sigma_1^2, \sigma_2^2, \sigma_{12} = 1.05, 1, 0.5$		
(0, 0)	200, 200	200, 200	200, 200	186.2, 187.6	142, 138.4	176.7, 177.4
(0.1, 0)	231.6, 222.2	162.3, 167.9	181.6, 174.8	212.4, 221.1	124.5, 121.7	181.7, 179.7
(0.1, 0.3)	39.04, 40.21	49.88, 49.47	50.44, 49.63	35.28, 34.55	40.77, 41.27	50.68, 48.63
(0.3, 0)	274.8, 282.8	116.7, 117.2	165.7, 171.1	225.2, 225.2	86.65, 84.26	168, 170.8
(0.5, 0.8)	5.17, 4.66	6.08, 5.38	12.31, 12.28	5, 4.38	5.52, 4.93	12.26, 11.75
(1, 1)	2.28, 1.74	2.39, 1.76	10.02, 8.69	2.17, 1.63	2.29, 1.69	10.06, 9.615
(2, 2)	1, 0.09	1, 0.08	11.04, 10.03	1, 0.08	1, 0.05	11.28, 10.79
(μ_1, μ_2)	$\sigma_1^2, \sigma_2^2, \sigma_{12} = 1.05, 1.05, 0.5$			$\sigma_1^2, \sigma_2^2, \sigma_{12} = 1.05, 1.3, 0.5$		
(0, 0)	150, 150.2	112.4, 111.3	178.2, 170.9	56.75, 53.19	35.34, 34.7	135.6, 133.3
(0.1, 0)	167.8, 165.7	91.67, 88.75	172.5, 167.6	63.89, 60	29.74, 30.07	108.6, 110.8
(0.1, 0.3)	31.64, 31.06	32.33, 32.36	46.03, 47.7	17.87, 17.33	13.91, 12.76	47.6, 45.41
(0.3, 0)	181.1, 185.6	66.24, 66.01	145, 139.8	70.16, 68.29	24.4, 23.79	98.41, 93.14
(0.5, 0.8)	4.89, 4.46	4.72, 4.41	12, 12.29	3.99, 3.49	3.11, 2.55	12.31, 11.62
(1, 1)	2.16, 1.6	2.18, 1.61	10.49, 9.53	2.01, 1.386	1.77, 1.172	11.4, 11.41
(2, 2)	1, 0.07	1, 0.05	11.06, 10.5	1, 0.08	1, 0.03	14.01, 13.45
(μ_1, μ_2)	$\sigma_1^2, \sigma_2^2, \sigma_{12} = 1.2, 1.2, 0.6$			$\sigma_1^2, \sigma_2^2, \sigma_{12} = 1.4, 1.4, 0.5$		
(0, 0)	82.78, 81.39	40.88, 41.09	112.5, 112.3	21.28, 20.96	9.24, 8.37	91.04, 88.54
(0.1, 0)	84.68, 82.83	35.39, 34.28	102, 101.2	22.17, 22.54	8.59, 7.91	86.13, 84.34
(0.1, 0.3)	22.31, 22.49	15.23, 14.85	35.87, 33.51	8.81, 8.45	5.02, 4.51	41.04, 42.17
(0.3, 0)	85.71, 88.17	27.79, 27.52	102, 106.5	21.26, 21.9	7.29, 6.61	74.15, 69.47
(0.5, 0.8)	3.84, 3.17	3.24, 2.67	9.72, 9.28	2.65, 2.13	1.91, 1.35	12.47, 12.18
(1, 1)	1.88, 1.29	1.79, 1.21	9.06, 8.25	1.59, 0.95	1.34, 0.66	12.14, 12.33
(2, 2)	1.01, 0.09	1, 0.06	10.77, 10.35	1, 0.06	1, 0.03	17.15, 16.48
(μ_1, μ_2)	$\sigma_1^2, \sigma_2^2, \sigma_{12} = 1.4, 1.4, 0.75$			$\sigma_1^2, \sigma_2^2, \sigma_{12} = 3, 3, 0.5$		
(0, 0)	42.26, 39.87	16.45, 15.6	75.73, 71.09	1.91, 1.3	1.2, 0.48	105.9, 108.5
(0.1, 0)	41.82, 39.94	15.16, 14.77	68.19, 70.49	1.98, 1.4	1.22, 0.53	92.67, 90.9
(0.1, 0.3)	13.78, 14.03	7.58, 7.1	26.01, 24.69	1.49, 0.86	1.13, 0.38	57.07, 56.54
(0.3, 0)	40.9, 40.55	12.26, 11.34	65.37, 64.64	1.97, 1.39	1.17, 0.45	84.41, 78.99
(0.5, 0.8)	3.02, 2.45	2.39, 1.84	8.55, 8.2	1.17, 0.47	1.03, 0.19	23.51, 23.63
(1, 1)	1.67, 1.04	1.54, 0.9	7.81, 7.57	1.11, 0.35	1.01, 0.12	22.48, 21.63
(2, 2)	1, 0.08	1, 0.07	10.28, 9.77	1, 0.08	1, 0	59.71, 60.83

According to the results of this analysis (reported in Table 8), the ANN chart is the worst-performing in all the compared cases. Under zero variation shift, the RFR chart performs better than the SVR chart. Other than that, under small variation shifts, together with small mean shifts, the SVR chart performs better, and together with moderate and large mean shifts, the RFR chart performs better. In large variation shift cases, the SVR chart performs better. By comparing the first and second training cases (Tables 7 and 8), it is evident that the ANN and RFR charts perform better with the second training method being applied, in all the shift cases. On the other hand, under no or small variation shifts, the SVR chart also performs better with the second training method, but under larger variation shifts, the chart performance is rather the same no matter which training method is used. In addition, no deterioration in performance as the shift size becomes larger than the trained value is noticeable in any of the charts in this scenario. By comparing Tables 4 and 8 (different input sets), we realize that all the charts mostly perform better with the second input scenario, with the RFR chart having the least improved cases.

Table 8. ARL, SDRL. Scenario a22, $p = 2$.

	ANN	SVR	RFR	ANN	SVR	RFR
(μ_1, μ_2)	$\sigma_1^2, \sigma_2^2, \sigma_{12} = 1, 1, 0.5$			$\sigma_1^2, \sigma_2^2, \sigma_{12} = 1.05, 1, 0.5$		
(0, 0)	200, 200	200, 200	200, 200	123.77, 120.89	112.98, 110.77	146.04, 137.69
(0.1, 0)	175.12, 169.66	155.3, 149.36	122.8, 117.12	107.61, 104.76	87.61, 86.69	100.7, 94.8
(0.1, 0.3)	55.5, 53.37	49.26, 48.86	37.66, 35.35	40.88, 40.46	32.22, 32.18	30.83, 29.54
(0.3, 0)	68.36, 69.04	93.47, 91.08	33.89, 32.93	49.6, 46.63	57.54, 59.13	30.19, 30.05
(0.5, 0.8)	4.73, 4.28	5.52, 5.12	3.47, 2.97	4.07, 3.59	4.19, 3.63	3.27, 2.64
(1, 1)	1.94, 1.35	2.1, 1.51	1.39, 0.72	1.77, 1.16	1.87, 1.26	1.36, 0.73
(2, 2)	1, 0.03	1, 0.03	1, 0.04	1, 0.05	1, 0.04	1, 0.05
(μ_1, μ_2)	$\sigma_1^2, \sigma_2^2, \sigma_{12} = 1.05, 1.05, 0.5$			$\sigma_1^2, \sigma_2^2, \sigma_{12} = 1.05, 1.3, 0.5$		
(0, 0)	119.7, 117.43	113.66, 111.26	145.8, 143.26	36.98, 36.29	35.36, 34.15	67.92, 71.98
(0.1, 0)	109.92, 110.25	91.58, 89.65	94.87, 95.08	32.44, 31.9	28.9, 29.51	57.81, 57.1
(0.1, 0.3)	39.17, 38.3	31.75, 30.94	32.75, 32.57	16.98, 16.67	13.64, 12.93	19.89, 19.42
(0.3, 0)	46.38, 46.47	57.45, 56.99	30.82, 28.69	21.45, 21.07	22.37, 21.61	26.26, 25.91
(0.5, 0.8)	4.2, 3.63	4.33, 3.76	3.36, 2.79	3.36, 2.75	3.03, 2.5	3.07, 2.52
(1, 1)	1.79, 1.18	1.9, 1.32	1.32, 0.68	1.63, 1.03	1.62, 0.98	1.36, 0.69
(2, 2)	1, 0.03	1, 0.02	1, 0.07	1, 0.04	1, 0.03	1, 0.07
(μ_1, μ_2)	$\sigma_1^2, \sigma_2^2, \sigma_{12} = 1.2, 1.2, 0.6$			$\sigma_1^2, \sigma_2^2, \sigma_{12} = 1.4, 1.4, 0.5$		
(0, 0)	49.03, 47.37	43.5, 43.13	70.52, 70.58	8.71, 8.04	9.21, 8.72	26.31, 26.03
(0.1, 0)	46.69, 46.07	35.71, 37.08	50.66, 50.53	7.93, 7.31	8.41, 7.83	22.27, 21.87
(0.1, 0.3)	21.07, 20.83	15.34, 14.99	20.19, 19.52	5.9, 5.29	5.08, 4.4	10.86, 10.34
(0.3, 0)	26.63, 25.92	24.93, 24.83	22.78, 23.16	6.36, 6.04	6.86, 6.31	12.83, 12.98
(0.5, 0.8)	3.61, 3.05	3.18, 2.74	2.86, 2.32	2.23, 1.56	1.89, 1.27	2.41, 1.76
(1, 1)	1.66, 1.05	1.64, 1	1.39, 0.75	1.39, 0.76	1.28, 0.59	1.3, 0.66
(2, 2)	1, 0.05	1, 0.05	1, 0.03	1.01, 0.12	1, 0	1, 0, 07
(μ_1, μ_2)	$\sigma_1^2, \sigma_2^2, \sigma_{12} = 1.4, 1.4, 0.75$			$\sigma_1^2, \sigma_2^2, \sigma_{12} = 3, 3, 0.5$		
(0, 0)	21.16, 20.8	17.8, 17.78	29.82, 28.27	1.26, 0.57	1.26, 0.58	2.01, 1.5
(0.1, 0)	19.72, 19.11	15.32, 14.3	25.04, 24.31	1.24, 0.53	1.2, 0.48	1.96, 1.32
(0.1, 0.3)	11.45, 10.87	7.954, 7.319	11.55, 10.3	1.26, 0.58	1.14, 0.39	1.67, 1.09
(0.3, 0)	14.49, 13.59	11.25, 10.4	13.41, 12.79	1.2, 0.47	1.18, 0.46	1.85, 1.26
(0.5, 0.8)	2.93, 2.43	2.26, 1.67	2.57, 2	1.23, 0.54	1.03, 0.2	1.26, 0.57
(1, 1)	1.59, 0.97	1.42, 0.76	1.34, 0.7	1.22, 0.54	1.01, 0.12	1.1, 0.35
(2, 2)	1, 0.09	1, 0.07	1, 0.04	1.345, 0.67	1, 0	1, 0, 05

Moreover, by comparing the results in Table 8 to Sabahno et al.’s [9] Table 4, we can again see that in all the cases, at least one of the proposed ML control charts performs better (mostly much better) than all their control charts. Therefore, when the second input scenario is used, if the second training method is utilized, the proposed scheme performs better under all shift sizes and types, unlike the first training method (Scenario a21).

3.2. Scenario b: Control Charts for Detection and Identification

In scenario b, several ML structures are involved, and consequently, so are several control charts in each control scheme. Each process parameter has its output (control chart), and if the corresponding chart signals, it means that this parameter (and consequently the variable associated with it) has shifted. However, as the number of quality characteristics increases, the number of control charts (outputs) also increases in this scenario. This might increase the false-alarm rates. As such, this scenario should be used with more caution, especially in larger dimensions. In addition, even before conducting any numerical analysis (next subsections), overall, worse performance compared to scenario a is expected in this scenario because more than one control chart is being monitored together. Having said that, the advantage of this scenario is not its performance, but helping one to identify the responsible variable parameter by allowing a little bit of performance to be sacrificed.

In the case of $p = 2$, since we have five process parameters, namely μ_1 , μ_2 , σ_1^2 , σ_2^2 and σ_{12} , five ML structures, each with its own control chart, are therefore required. These charts are monitored together and obtaining a signal from either one of them means that the process is out of control. Before we start developing control charts for each scenario, we should mention that it was more difficult to choose suitable hyperparameters for this scenario to obtain the desired ARL performance, especially in the case of $p = 3$. Therefore, we tried many combinations of hyperparameter values for the ML structures to obtain the desired overall performance. The desired performance, in this case, is computed (assuming the independency of control charts as well as equality of their type-I error probability) as follows: $\alpha_{overall} = 1 - (1 - \alpha)^m$, where m is the number of control charts, and $ARL = \frac{1}{\alpha_{overall}} = \frac{1}{0.005} = 200$. Consequently, for the case of $p = 2$, in which a maximum of five control charts is required ($m = 5$), by using the above formulae, we have $\alpha = 0.001$. This means that each control chart's performance is $\frac{1}{0.001} = 1000$, but they all together should have a performance of $\frac{1}{0.005} = 200$. Similarly, for the case of $p = 3$, individual α s are computed as 0.000716, which results in $ARL = \frac{1}{0.000716} = 1396.6$. Note that, for simplicity, we assume that all the covariances are equal in the case of $p = 3$. Therefore, we only have one control chart for monitoring the covariance, making a total of seven control charts ($m = 7$, i.e., three to monitor the means, three to monitor the variances, and one to monitor the covariance).

3.2.1. Scenario b1 (Control Type b, Input Set 1)

Similar to Scenario a1, the inputs, in this case, are T^2 and W statistics. Again, two different training methods are applied on each control chart (one trained with only small shifts and the other one trained with small and large shifts).

Scenario b11 (Control Type b, Input Set 1, Training Method 1)

In this scenario, the ML structures are trained with only small shift sizes. First, we generate 1000 in-control data that are going to be used in all control charts. Second, we generate 1000 out-of-control data with 0.2 shifts in μ_1 and use it for the control chart that monitors μ_1 ; 1000 out-of-control data with 0.2 shifts in μ_2 and use it for the control chart that monitors μ_2 ; 1000 out-of-control data with shifts of 1.2 times the in-control σ_1^2 and use it for the control chart that monitors σ_1^2 ; 1000 out-of-control data with shifts of 1.2 times the in-control σ_2^2 and use it for the control chart that monitors σ_2^2 ; and 1000 out-of-control data with shifts of 1.2 times the in-control σ_{12} and use it for the control chart that monitors σ_{12} . Now that we have five datasets, we need to train five different ML structures. For the ANN scheme, for monitoring μ_1 , μ_2 , σ_1^2 , σ_2^2 and σ_{12} , we use 5, 5, 5, 5, and 4 nodes in the hidden layers, respectively, to obtain an overall performance of 200. The RMSE of these ANN structures, respectively, are 0.49, 0.49, 0.48, 0.48, and 0.49. Finally, the UCLs are, respectively, computed as 1.0292, 1.1628, 0.8846, 1.048, and 0.8393.

For the SVR scheme, the major difference was that the linear kernel did not work for all the structures in this scenario, and we had to try the radial kernel as well (other kernel types did not work at all). The kernel types we used for μ_1 , μ_2 , σ_1^2 , σ_2^2 and σ_{12} control charts are, respectively, radial, radial, radial, radial, and linear. The RMSEs and UCLs are as follows. The RMSEs are 0.52, 0.52, 0.52, 0.53, and 0.51, and the UCLs are 1.1174, 1.1204, 1.1089, 1.0972, and 1.283. Regarding the RFR scheme, 100 trees still worked well for each structure in this scenario. The RMSEs and UCLs for the RFR charts are computed as follows. The RMSEs are 0.53, 0.54, 0.52, 0.52, and 0.52, and the UCLs are 0.9576, 0.9409, 0.9597, 0.9681, and 0.9433.

According to the reported results in Table 9, all the control charts mostly experience a deterioration in performance compared to when only one control chart is used (Table 1). However, the RFR scheme mostly performs better in this control case when the mean shift is large. The worst deterioration in performance can be seen in the SVR scheme. It even experiences a deterioration in performance under shifts larger than the trained value in this scenario (unlike Table 1). The RFR scheme, however, experiences that effect only when the variation shift size is very large in this scenario. Out of these three, the ANN is the

best-performing scheme. Its performance is even very close to that of being reported in Table 1 (used only for detection).

Table 9. ARL, SDRL. Scenario b11 (with identification), $p = 2$.

	ANN	SVR	RFR	ANN	SVR	RFR
(μ_1, μ_2)	$\sigma_1^2, \sigma_2^2, \sigma_{12} = 1, 1, 0.5$			$\sigma_1^2, \sigma_2^2, \sigma_{12} = 1.05, 1, 0.5$		
(0, 0)	200, 200	200, 200	200, 200	149.42, 151.52	201.87, 203.46	181.8, 177.7
(0.1, 0)	160.4, 155.4	173.9, 176.6	186.7, 191	124, 120	184.7, 191.4	180.1, 189
(0.1, 0.3)	62.01, 58.38	110.02, 110.37	155.2, 158.5	52.15, 50.89	109.1, 107.4	143, 151.4
(0.3, 0)	48.04, 46.43	93.51, 90	136.3, 138.5	40.94, 40.87	97.53, 97.58	139.4, 143.1
(0.5, 0.8)	4.75, 4.3	36.58, 36.75	49.6, 49.63	4.55, 4.16	38.09, 38.81	51.89, 50.52
(1, 1)	1.79, 1.15	51.36, 49.96	29.7, 30.33	1.77, 1.19	47.34, 46.34	31.93, 32.45
(2, 2)	1, 0.03	>10,000	19.38, 18.91	1, 0.03	>10,000	20.84, 20.32
(μ_1, μ_2)	$\sigma_1^2, \sigma_2^2, \sigma_{12} = 1.05, 1.05, 0.5$			$\sigma_1^2, \sigma_2^2, \sigma_{12} = 1.05, 1.3, 0.5$		
(0, 0)	118.52, 119.79	202.07, 207.02	180.8, 175.1	39.63, 39.97	168.7, 169	131.2, 131.3
(0.1, 0)	97.77, 99.47	172.9, 172.1	176.5, 178.1	36.89, 36.02	161.9, 158.9	130.8, 130.4
(0.1, 0.3)	43.95, 44.88	108.1, 106.8	138.2, 133	20.56, 20.27	102.3, 103.9	113.9, 116.9
(0.3, 0)	35.95, 35.47	95.47, 93.45	134.1, 128.2	19.35, 19.4	88.08, 86.51	122.5, 117.7
(0.5, 0.8)	4.422, 3.934	38.92, 37.7	56.43, 55.79	3.69, 3.19	39.36, 39.55	74.89, 70.93
(1, 1)	1.76, 1.16	47.73, 44.82	34.79, 32.97	1.77, 1.16	47.39, 47.61	52.86, 50.55
(2, 2)	1, 0.02	>10,000	22.8, 22.2	1, 0.05	>10,000	35.4, 34.89
(μ_1, μ_2)	$\sigma_1^2, \sigma_2^2, \sigma_{12} = 1.2, 1.2, 0.6$			$\sigma_1^2, \sigma_2^2, \sigma_{12} = 1.4, 1.4, 0.5$		
(0, 0)	47.96, 47.6	186.1, 183.4	132.9, 133.4	11.57, 11.02	146.5, 145.7	94.65, 99.79
(0.1, 0)	41.51, 40.24	157.3, 157.5	144.8, 156.5	11.2, 10.7	135.7, 139.9	97.2, 96.26
(0.1, 0.3)	23.57, 22.57	104.1, 106.2	126.2, 117.9	8.69, 8.15	103.9, 105.3	94.03, 94.87
(0.3, 0)	20.93, 20.87	88.51, 86.74	122.6, 121.7	7.41, 6.69	87.51, 88.02	96.93, 95.25
(0.5, 0.8)	3.79, 3.23	38.36, 38.36	69.34, 69.38	3, 2.4	43.3, 44.68	106.1, 105.8
(1, 1)	1.83, 1.24	48.31, 49.39	49.93, 52.86	1.65, 1.08	56.18, 54.86	109.8, 109.8
(2, 2)	1, 0.04	>10,000	33.44, 32.79	1, 0.07	>10,000	91.38, 85.05
(μ_1, μ_2)	$\sigma_1^2, \sigma_2^2, \sigma_{12} = 1.4, 1.4, 0.75$			$\sigma_1^2, \sigma_2^2, \sigma_{12} = 3, 3, 0.5$		
(0, 0)	19.79, 18.66	160.9, 161.4	108, 107.1	1.27, 0.6	725.8, 726.7	222.9, 221.5
(0.1, 0)	19.09, 18.97	141.7, 132.8	103.9, 106.7	1.27, 0.6	700.6, 692.5	206.9, 200.7
(0.1, 0.3)	13.15, 12.39	99.46, 98.47	104.4, 106	1.26, 0.58	735.2, 706.8	217.9, 228.2
(0.3, 0)	11.99, 10.93	88.22, 83.6	108.71, 104.8	1.28, 0.58	709.2, 712.3	221.7, 215.4
(0.5, 0.8)	3.46, 3.03	41.47, 41.09	94.66, 93.9	1.13, 0.38	643.3, 611.6	436.7, 432.3
(1, 1)	1.81, 1.18	53.56, 55.79	69.6, 75.21	1.08, 0.28	812.3, 578	1434, 1027
(2, 2)	1, 0.08	>10,000	59.42, 58.75	1, 0.02	>10,000	>10,000

Moreover, by comparing the results of Table 9 to Sabahno et al.'s [9] Table 4, one can still see that in most cases (except mostly in the case of (0.5, 0.8) mean shift together with no/small variation shift), at least one of the proposed ML control charts performs better than all their proposed control charts, even though their control charts are only designed for detection. In the case of three quality characteristics, the construction of the dataset is the same as the case of $p = 2$, except here we add the following: 1000 out-of-control data with 0.2 shifts in μ_3 and use it for the control chart that monitors μ_3 and 1000 out-of-control data with shifts 1.2 times the in-control σ_3^2 and use it for the control chart that monitors σ_3^2 . For the ANN scheme, for monitoring $\mu_1, \mu_2, \mu_3, \sigma_1^2, \sigma_2^2, \sigma_3^2$ and the covariance, we use 5, 5, 5, 5, 5, 5, and 4 nodes in the hidden layers, respectively, with RMSEs of 0.49, 0.49, 0.49, 0.49, 0.49, and 0.48, and UCLs of 0.9169, 1.2852, 0.9164, 0.822, 0.8787, 0.8193, and 0.7305. For the SVR scheme, the used kernels, respectively, are radial, radial, radial, radial, radial, radial, and linear. The RMSEs are 0.51, 0.51, 0.51, 0.52, 0.52, 0.52, and 0.52, and the UCLs are 1.1506, 1.0763, 1.0906, 1.0908, 1.1556, 1.0914, and 1.6437. Regarding the RFR scheme, the least numbers of trees that we had to use for each structure (to obtain the overall ARL of 200), respectively, 100, 100, 100, 500, 100, 300, and 100. The RMSEs are 0.53, 0.53, 0.53,

0.53, 0.52, 0.53, and 0.52, and the UCLs are 0.9485, 0.9585, 0.9474, 0.9577, 0.9647, 0.9647, and 0.9677. The results of this analysis are reported in Table 10.

Table 10. ARL, SDRL. Scenario b11 (with identification), $p = 3$.

	ANN	SVR	RFR	ANN	SVR	RFR
(μ_1, μ_2, μ_3)	$\sigma_1^2, \sigma_2^2, \sigma_3^2, Cov = 1, 1, 1, 0.5$			$\sigma_1^2, \sigma_2^2, \sigma_3^2, Cov = 1.05, 1, 1, 0.5$		
(0, 0, 0)	200, 200	200, 200	200, 200	173.2, 170	200, 197	180.7, 171.2
(0.1, 0, 0.1)	162.7, 158.5	191.7, 180.4	193.2, 182.2	140.1, 139.8	184.4, 186.3	169.1, 178.3
(0.1, 0.3, 0.3)	65.37, 63.74	112.4, 113.3	140.2, 136.4	62.19, 58.74	104.3, 106.4	137.1, 134.9
(0.3, 0, 0)	70, 68.85	112.1, 108.8	145.3, 147.1	62.41, 62.24	105.7, 102.8	136.1, 129.4
(0.5, 0.8, 0.5)	8.79, 8.26	39.13, 40.1	67.1, 65.66	8.88, 8.04	39.58, 38.35	64.91, 66.47
(1, 1, 1)	2.45, 1.88	41.89, 39.86	85.13, 79.04	2.41, 1.79	40.68, 41.82	81.15, 81.58
(2, 2, 2)	1, 0	>10,000	>10,000	1, 0	>10,000	>10,000
(μ_1, μ_2, μ_3)	$\sigma_1^2, \sigma_2^2, \sigma_3^2, Cov = 1.05, 1.05, 1.05, 0.5$			$\sigma_1^2, \sigma_2^2, \sigma_3^2, Cov = 1.05, 1.3, 1.05, 0.5$		
(0, 0, 0)	119.6, 120.1	166.9, 173.5	185.3, 174.1	58.1, 56.94	117, 121.3	122, 123.7
(0.1, 0, 0.1)	111.3, 112	161.5, 155.2	152, 144.9	53.26, 53.94	113.3, 114.5	114.6, 112.5
(0.1, 0.3, 0.3)	54.7, 54.6	98.81, 95.89	113.1, 106.3	34.07, 34.35	79.56, 82.71	98.41, 96.06
(0.3, 0, 0)	53.08, 51.83	96.81, 100.2	117.3, 121.1	32.76, 32.4	83.34, 78.16	106.1, 108
(0.5, 0.8, 0.5)	8.64, 7.86	39.58, 37.68	64.28, 62.58	6.88, 6.14	41.4, 40	70.57, 68.66
(1, 1, 1)	2.43, 1.86	41.77, 42.75	89.83, 86.32	2.27, 1.63	47.5, 45.62	99.91, 96.18
(2, 2, 2)	1, 0.04	>10,000	>10,000	1, 0.03	>10,000	>10,000
(μ_1, μ_2, μ_3)	$\sigma_1^2, \sigma_2^2, \sigma_3^2, Cov = 1.2, 1.2, 1.2, 0.6$			$\sigma_1^2, \sigma_2^2, \sigma_3^2, Cov = 1.4, 1.4, 1.05, 0.5$		
(0, 0, 0)	54.77, 54.89	115.1, 112	126.4, 129.6	17.75, 16.96	71.26, 68.07	74.38, 82.69
(0.1, 0, 0.1)	51.34, 48.68	108.7, 106.1	113.1, 105.3	17.79, 17.88	70.62, 71.74	79.58, 83.47
(0.1, 0.3, 0.3)	32.2, 32.72	81.05, 76.15	90.85, 86.53	13.11, 13.1	59.12, 58.77	71.82, 68.78
(0.3, 0, 0)	31.45, 30.61	77.98, 76.26	94.03, 86.58	12.84, 12.08	61.83, 58.73	80.5, 82.38
(0.5, 0.8, 0.5)	6.52, 5.72	40.93, 40.51	70.07, 71.17	4.77, 3.92	50.43, 49.86	76.7, 73.9
(1, 1, 1)	2.21, 1.58	47.82, 46.98	105.9, 98.5	1.96, 1.33	60.61, 61.3	138.3, 146.4
(2, 2, 2)	1, 0	>10,000	>10,000	1, 0.03	>10,000	>10,000
(μ_1, μ_2, μ_3)	$\sigma_1^2, \sigma_2^2, \sigma_3^2, Cov = 1.4, 1.4, 1.05, 0.75$			$\sigma_1^2, \sigma_2^2, \sigma_3^2, Cov = 3, 3, 1.4, 0.5$		
(0, 0, 0)	72.1, 76.37	130.7, 130	135.9, 136.5	1.24, 0.58	132.5, 133.2	66.75, 65.52
(0.1, 0, 0.1)	66.25, 67.53	129.3, 128.7	126.2, 122.1	1.25, 0.57	140.8, 132.4	72.06, 71.05
(0.1, 0.3, 0.3)	38.7, 39.57	90.35, 95.11	101.9, 105.1	1.22, 0.52	174.8, 182.1	80.7, 78.22
(0.3, 0, 0)	35.49, 36.39	84.25, 83.67	103.4, 106.2	1.2, 0.47	141.8, 138.7	72.15, 73.2
(0.5, 0.8, 0.5)	7.22, 6.9	43.91, 44.9	72.57, 70.33	1.16, 0.45	320.2, 319.3	167.3, 171.3
(1, 1, 1)	2.24, 1.62	43.59, 42.46	106.5, 100.2	1.06, 0.26	953.8, 947.7	944.5, 964.4
(2, 2, 2)	1, 0.05	>10,000	>10,000	1, 0	>10,000	>10,000

According to the results in Table 10, the ANN chart performs better than the others. By comparing Tables 2 and 10, one can conclude that when the identification is a goal as well, all the schemes perform worse. Also, both the SVR and RFR schemes experience significant deterioration in performance as the shift size becomes larger than the trained value in this scenario. By comparing Tables 9 and 10 ($p = 2$ and $p = 3$ cases), we realize that the ANN scheme mostly performs worse in higher dimensions and the SVR scheme mostly performs better (except under no variation shift). Regarding the RFR scheme, only under small mean shifts and very large variation shifts, it mostly performs better in higher dimensions.

Moreover, by comparing the results in Table 10 to Sabahno et al.'s [9] Table 5, we can see that the number of cases in which at least one of our control charts performs better than all their charts is almost equal to the cases that at least one of their proposed charts performs better than ours. Once again, their charts are designed for detection, and in this scenario, we are looking for both detection and identification; therefore, the performance deterioration is normal.

Scenario b12 (Control Type b, Input Set 1, Training Method 2)

In this scenario, the ML structures are trained with both small and large shift sizes. Here again, 1000 in-control data are generated to be used in all control charts. Regarding the out-of-control dataset, we generate 500 out-of-control data with 0.2 shifts and 500 out-of-control data with 1 shift in μ_1 and use it for the control chart that monitors μ_1 ; 500 out-of-control data with 0.2 shifts in μ_2 and 500 out-of-control data with 1 shift in μ_2 and use it for the control chart that monitors μ_2 ; 500 out-of-control data with shifts 1.2 times that of the in-control σ_1^2 and 500 out-of-control data with shifts 2 times that of the in-control σ_1^2 and use it for the control chart that monitors σ_1^2 ; 500 out-of-control data with shifts 1.2 times that of the in-control σ_2^2 and 500 out-of-control data with shifts 2 times that of the in-control σ_2^2 and use it for the control chart that monitors σ_2^2 ; and finally, 500 out-of-control data with shifts 1.2 times that of the in-control σ_{12} and 500 out-of-control data with shifts 2 times that of the in-control σ_{12} and use it for the control chart that monitors σ_{12} . As five datasets are considered, five different ML structures for each scheme are required.

For the ANN scheme, for μ_1 , μ_2 , σ_1^2 , σ_2^2 and σ_{12} , we use five nodes in all the hidden layers. The RMSE of these ANN structures, respectively, are 0.42, 0.42, 0.46, 0.46, and 0.43. Finally, the UCLs are computed as 1.1086, 0.9762, 0.9519, 0.985, and 1.007. For the SVR scheme, we use the radial kernel for all the structures. The RMSEs and UCLs are as follows. The RMSEs are 0.45, 0.44, 0.49, 0.49, and 0.46, and the UCLs are 1.0793, 1.1018, 1.0266, 1.0194, and 1.0103. Regarding the RFR scheme, 100 trees worked well for each structure. The RMSEs and UCLs for the RFR charts are as follows. The RMSEs are 0.45, 0.45, 0.49, 0.49, and 0.45, and the UCLs are 0.9998, 0.9998, 0.9937, 0.9961, and 0.999.

The results in Table 11 show that in most cases, the ANN scheme performs better than the other charts. However, under no variation shift, and small variation shifts when the mean shift is also small, the RFR scheme performs better. By comparing the results of this table with its equivalent scenario when only detection was the goal (comparing Tables 4 and 11), one can see the performance deterioration in all the schemes when identification is added to the goal as well. However, again same as in the previous case (Scenario b11), deterioration is at its lowest for the ANN chart. In addition, by comparing the results of two training methods (Scenario b11/ Table 9 and Scenario b12/ Table 11), one can see that the SVR and RFR schemes benefit from the second training method, with the RFR benefiting the most so that we do not see any deterioration in performance under shifts larger than the trained value. The ANN scheme, however, experiences a slight deterioration in performance in most cases.

Moreover, by comparing the results in Table 11 to Sabahno et al.'s [9] Table 4, we realize that although more cases are added to the ones in which at least one of their proposed control charts performs better, we can still see that the cases in which at least one of the proposed ML schemes performs better than all their proposed control charts are more, even though their control charts are only designed for detection.

In the case of three quality characteristics, the construction of the dataset is similar to the $p = 2$ case, with the only difference being that since two more process parameters (μ_3 and σ_3^2)/charts are added in the case of $p = 3$, 500 out-of-control data with 0.2 shifts and 500 out-of-control data with 1 shift in μ_3 are generated to be used for the control chart that monitors μ_3 , and 500 out-of-control data with shifts 1.2 times that of the in-control σ_3^2 and 500 out-of-control data with shifts 2 times that of the in-control σ_3^2 are considered to be used for the control chart that monitors σ_3^2 .

For the ANN control charts to monitor μ_1 , μ_2 , μ_3 , σ_1^2 , σ_2^2 , σ_3^2 , and the covariance, five nodes are utilized in each hidden layer, with RMSEs of 0.38, 0.38, 0.38, 0.45, 0.44, 0.44, and 0.27 as well as UCLs of 0.913, 0.9082, 0.96518, 0.9581, 0.9339, 0.9797, and 0.9951. For the SVR control charts, the used kernels are all radial. The RMSEs and UCLs are as follows. The RMSEs are 0.39, 0.39, 0.39, 0.48, 0.48, 0.48, and 0.29, and the UCLs are 1.0722, 1.0839, 1.1073, 1.0892, 1.0128, 1.0823, and 1.0281. Regarding the RFR control charts, the least numbers of trees that we had to use for each structure (to obtain the overall ARL of 200) are 100, 100, 100, 500, 300, 300, and 100. The RMSEs are 0.4, 0.4, 0.4, 0.45, 0.46, 0.46, and 0.29, and

the UCLs are 0.7026, 0.7052, 0.7254, 0.3872, 0.4453, 0.4077, and 0.8023. Note that for the RFR structures in this scenario, for the first time, we had to activate a feature that the ‘randomForest’ package offers, and it is called ‘corr.bias’, which performs bias correction for the regression model. Without that feature being activated, we were not able to obtain the desired performance, no matter how many trees we tried.

Table 11. ARL, SDRL. Scenario b12 (with identification), $p = 2$.

	ANN	SVR	RFR	ANN	SVR	RFR
(μ_1, μ_2)	$\sigma_1^2, \sigma_2^2, \sigma_{12} = 1, 1, 0.5$			$\sigma_1^2, \sigma_2^2, \sigma_{12} = 1.05, 1, 0.5$		
(0, 0)	200, 200	200, 200	200, 200	167.4, 170.5	193.6, 186.8	172.1, 177.2
(0.1, 0)	171.1, 171.3	181.2, 203.5	166.5, 145.6	139, 139.6	167.5, 170.9	157.8, 164
(0.1, 0.3)	79.9, 78.78	99.75, 105.6	63.04, 65.82	69.15, 70.55	83.67, 75.27	70.87, 68.67
(0.3, 0)	62.84, 59.85	75.31, 76.62	50.5, 52.81	53.81, 54.14	75.63, 74.51	44.89, 40.81
(0.5, 0.8)	5.51, 5.04	30.7, 35.05	5.29, 4.51	5.57, 5.1	24.71, 23	5.65, 4.97
(1, 1)	1.92, 1.27	31.48, 30.6	2.01, 1.36	1.89, 1.24	28.24, 25.56	2.01, 1.29
(2, 2)	1, 0.03	>10,000	1.22, 0.5414	1, 0.03	>10,000	1.26, 0.6
(μ_1, μ_2)	$\sigma_1^2, \sigma_2^2, \sigma_{12} = 1.05, 1.05, 0.5$			$\sigma_1^2, \sigma_2^2, \sigma_{12} = 1.05, 1.3, 0.5$		
(0, 0)	133.9, 131.2	157.5, 144.2	127.9, 123.6	52.48, 51.18	109.5, 115	55.23, 53.85
(0.1, 0)	116.3, 115.4	144.6, 145.5	119.3, 116.9	47.29, 46.04	102.5, 108.9	54.11, 51.69
(0.1, 0.3)	54.95, 57.15	83.44, 89.2	56.89, 56.27	26.13, 26.2	66.23, 71.56	29, 01, 30.95
(0.3, 0)	45.51, 45	74.85, 71.91	44.08, 41.1	23.33, 22.87	56.56, 56.35	28.85, 28.8
(0.5, 0.8)	5.32, 4.73	23.63, 21.63	4.93, 4.2	4.29, 3.84	18.89, 20	5.05, 4.82
(1, 1)	1.83, 1.22	27.98, 25.33	1.98, 1.28	1.73, 1.09	22.17, 19.72	2.07, 1.56
(2, 2)	1, 0	>10,000	1.26, 0.51	1, 0	>10,000	1.34, 0.71
(μ_1, μ_2)	$\sigma_1^2, \sigma_2^2, \sigma_{12} = 1.2, 1.2, 0.6$			$\sigma_1^2, \sigma_2^2, \sigma_{12} = 1.4, 1.4, 0.5$		
(0, 0)	60.34, 59.04	124.5, 132.4	75.91, 77.46	14.94, 14.66	58.04, 54.73	25.81, 25.53
(0.1, 0)	55.62, 53.89	107.8, 98.68	65.75, 63.07	14.1, 14.42	53.29, 48.72	21.36, 20.77
(0.1, 0.3)	31.88, 31.27	62.41, 56.23	34.39, 33.71	10.38, 9.79	36.14, 35.83	15.69, 15.49
(0.3, 0)	26.52, 25.58	55.24, 51.44	33.01, 31.47	9.14, 8.97	34.89, 34.84	14.04, 13.73
(0.5, 0.8)	4.41, 3.95	21.47, 22.07	5.49, 4.335	3.03, 2.45	15.77, 14.66	4.94, 4.5
(1, 1)	1.76, 1.18	24.05, 24.97	2.04, 1.35	1.55, 0.92	21.01, 21.06	2.38, 1.64
(2, 2)	1, 0.03	>10,000	1.29, 0.6	1, 0	3684.07, 2428.36	1.61, 0.98
(μ_1, μ_2)	$\sigma_1^2, \sigma_2^2, \sigma_{12} = 1.4, 1.4, 0.75$			$\sigma_1^2, \sigma_2^2, \sigma_{12} = 3, 3, 0.5$		
(0, 0)	27.24, 26.27	79.79, 76.59	35.43, 35.04	1.5, 0.85	64.87, 66.76	14.6, 14.12
(0.1, 0)	24.84, 23.87	70.3, 71.4	33.03, 32.2	1.49, 0.85	63.42, 63.09	14.04, 13.35
(0.1, 0.3)	16.94, 16.28	45.1, 47.75	21.66, 21.23	1.44, 0.8	62.29, 61.14	12.93, 11.99
(0.3, 0)	14.64, 13.86	42.77, 41.06	19.98, 19.13	1.46, 0.81	60.07, 59.96	12.91, 12.09
(0.5, 0.8)	3.5, 2.93	19.98, 19.23	4.85, 4.39	1.34, 0.64	55.18, 50.77	11.38, 10.41
(1, 1)	1.71, 1.12	22.8, 21.87	2.25, 1.657	1.16, 0.45	79.59, 74.07	11.11, 10.37
(2, 2)	1, 0.03	4761.09, 5179.76	1.4, 0.73	1, 0.04	4256.1, 4516.53	11.47, 10.73

The result of this analysis is reported in Table 12. This table shows that in this case, the RFR scheme performs better than the other two in all the shift cases. By comparing this case with the first training method (Tables 10 and 12), we realize that the RFR scheme performs better with the second training method, and on the contrary, the ANN and SVR schemes mostly perform worse. Comparing the $p = 3$ and $p = 2$ cases (Tables 11 and 12), the RFR scheme mostly performs better in higher dimensions. On the other hand, the ANN and SVR charts mostly perform worse in higher dimensions (however, under very large mean shifts, the SVR scheme performs better). The highest deterioration in performance belongs to the ANN scheme. By comparing two process control scenarios (Tables 5 and 12), one can see that all the schemes experience deterioration in performance, more so for the SVR scheme, as unlike the previous control case, it experiences a significant deterioration in performance as the shift size becomes larger than the trained value.

Table 12. ARL, SDRL. Scenario b12 (with identification), $p = 3$.

	ANN	SVR	RFR	ANN	SVR	RFR
(μ_1, μ_2, μ_3)	$\sigma_1^2, \sigma_2^2, \sigma_3^2, Cov = 1, 1, 1, 0.5$			$\sigma_1^2, \sigma_2^2, \sigma_3^2, Cov = 1.05, 1, 1, 0.5$		
(0, 0, 0)	200, 200	200, 200	200, 200	231.6, 237.3	222.7, 234.9	168.8, 170.3
(0.1, 0, 0.1)	190.1, 196.3	191.7, 190.8	160.7, 155	216.5, 214.1	212.96, 212.47	147.2, 157.1
(0.1, 0.3, 0.3)	145.1, 157.2	143.6, 145.7	53.17, 54.18	171, 171.5	153.5, 157.2	47.92, 49.33
(0.3, 0, 0)	155.2, 150.9	143.7, 148.2	52.73, 53.01	152.1, 143.3	157.2, 155.2	49.32, 51.13
(0.5, 0.8, 0.5)	18.42, 17.86	72.88, 72.84	6.11, 5.824	18.61, 18.54	80.7, 75.61	5.48, 4.97
(1, 1, 1)	3.56, 2.95	58.57, 58.1	1.97, 1.44	3.58, 3.03	61.34, 61.48	1.87, 1.27
(2, 2, 2)	1.11, 0.37	1494, 1397	1.2, 0.5	1.11, 0.34	1504, 1629	1.2, 0.4738
(μ_1, μ_2, μ_3)	$\sigma_1^2, \sigma_2^2, \sigma_3^2, Cov = 1.05, 1.05, 1.05, 0.5$			$\sigma_1^2, \sigma_2^2, \sigma_3^2, Cov = 1.05, 1.3, 1.05, 0.5$		
(0, 0, 0)	272, 274.4	277.6, 294.1	127.3, 127.9	389.4, 380.8	350.1, 358.3	57.92, 54.64
(0.1, 0, 0.1)	275.7, 287.7	268.9, 261.7	99.14, 94.62	361.7, 363.8	335.5, 331.6	54.21, 53.41
(0.1, 0.3, 0.3)	176.8, 173	184.2, 192.1	39.81, 39.34	191.4, 205.4	250.8, 270	27.95, 26.8
(0.3, 0, 0)	171.5, 174.2	172.6, 163.8	41.3, 39.46	224.7, 198.1	263.7, 257.4	30.73, 29.67
(0.5, 0.8, 0.5)	19.84, 19.12	88.31, 84.7	5.93, 4.911	23.47, 22.21	103.5, 104	5.5, 5.17
(1, 1, 1)	4.18, 3.65	62.5, 60.63	1.94, 1.35	4.89, 4.44	76.9, 73.98	1.93, 1.48
(2, 2, 2)	1.2, 0.5	1453.11, 1538	1.18, 0.45	1.32, 0.61	1120.58, 1064.46	1.28, 0.59
(μ_1, μ_2, μ_3)	$\sigma_1^2, \sigma_2^2, \sigma_3^2, Cov = 1.2, 1.2, 1.2, 0.6$			$\sigma_1^2, \sigma_2^2, \sigma_3^2, Cov = 1.4, 1.4, 1.05, 0.5$		
(0, 0, 0)	427.05, 391.7	382.5, 392.9	62.2, 68.76	522.8, 501.9	602.7, 5, 583	22.73, 20.64
(0.1, 0, 0.1)	400.4, 414.5	333, 328.5	54.46, 54.61	426.9, 392.3	496.7, 505.2	21.92, 21.14
(0.1, 0.3, 0.3)	215.4, 216.1	228.3, 233.4	27.86, 26.09	219.6, 217.7	354.7, 347.3	15.05, 14.52
(0.3, 0, 0)	234.6, 226.5	263.7, 278.7	27.52, 26.72	173.5, 167	384.7, 382.9	14.34, 13.9
(0.5, 0.8, 0.5)	22.8, 22.6	116.9, 126.7	5.2, 4.68	29.7, 28.48	139, 149.3	4.61, 4.04
(1, 1, 1)	5.22, 5.02	66.91, 65.02	2.02, 1.35	8.44, 7.63	68.62, 67.18	2.21, 1.58
(2, 2, 2)	1.33, 0.65	932.7, 871.2	1.28, 0.66	1.73, 1.12	974.3, 982.5	1.32, 0.6323
(μ_1, μ_2, μ_3)	$\sigma_1^2, \sigma_2^2, \sigma_3^2, Cov 1.4, 1.4, 1.05, 0.75$			$\sigma_1^2, \sigma_2^2, \sigma_3^2, Cov 3, 3, 1.4, 0.5$		
(0, 0, 0)	286.3, 291.4	285.6, 255.2	68.35, 64.36	63.12, 60.09	458.3, 432.8	2.47, 1.84
(0.1, 0, 0.1)	286.4, 284	271.5, 280	64.79, 74.13	63.15, 61.07	427.3, 411.2	2.5, 2.01
(0.1, 0.3, 0.3)	159.1, 150.6	189.4, 188.1	30.58, 30.93	55.69, 53.68	344, 340.6	2.51, 2.01
(0.3, 0, 0)	158.1, 165	219.4, 192.7	30.31, 31.45	50.76, 48.21	337.1, 351.6	2.45, 1.94
(0.5, 0.8, 0.5)	18.46, 16.9	106.8, 106.7	5.12, 4.67	29.64, 29.25	203.3, 191.9	1.97, 1.41
(1, 1, 1)	4.3, 4.01	70.44, 77.17	2.05, 1.33	19.25, 18.14	108, 110.8	1.48, 0.85
(2, 2, 2)	1.23, 0.51	803.2, 715.9	1.2, 0.49	13.82, 12.84	1979.95, 2028.43	1.13, 0.36

Moreover, by comparing the results in Table 12 to Sabahno et al.’s [9] Table 5, we can see that in the cases of large mean shifts and/or large variation shifts, at least one of their proposed charts performs better than all our proposed charts. Otherwise, at least one of our charts performs better than all their charts.

3.2.2. Scenario b2 (Control Type b, Input Set 2)

The individual elements of each sample’s mean vector and variance–covariance matrix are considered as the inputs in this scenario.

Scenario b21 (Control Type b, Input Set 2, Training Method 1)

The first training method is used in this scenario. The construction of in and out-of-control datasets are the same as in Scenario b11. For the ANN scheme, 11 nodes in each hidden layer are used. The obtained RMSEs are 0.45, 0.43, 0.46, 0.46, and 0.45. Also, the computed UCLs are 0.9644, 1.1054, 1.3556, 1.1599, and 1.177. For the SVR scheme, we use linear kernels in all the structures in this scenario. The RMSEs are 0.49, 0.47, 0.5, 0.51, and 0.5. The UCLs are 1.2462, 1.2957, 1.3706, 1.3492, and 1.1669. For the RFR scheme, we use 100 trees for each structure. The RMSEs are 0.49, 0.48, 0.49, 0.49, and 0.5. The UCLs are 0.8489, 0.9042, 0.88, 0.8534, and 0.8459. The result of this scenario is reported in Table 13.

Table 13. ARL, SDRL. Scenario b21 (with identification), $p = 2$.

	ANN	SVR	RFR	ANN	SVR	RFR
(μ_1, μ_2)	$\sigma_1^2, \sigma_2^2, \sigma_{12} = 1, 1, 0.5$			$\sigma_1^2, \sigma_2^2, \sigma_{12} = 1.05, 1, 0.5$		
(0, 0)	200, 200	200, 200	200, 200	176.7, 177.2	141.8, 144.9	173.2, 178.9
(0.1, 0)	179.64, 185.25	165.69, 162.43	180.9, 184.9	160.4, 164.7	118.7, 121.7	172.2, 171.5
(0.1, 0.3)	105.6, 109.4	55.46, 54.82	124.8, 125.5	111.2, 112.3	51.25, 50.59	129.1, 127.6
(0.3, 0)	141, 140.7	40.08, 38.88	153.3, 148.8	126, 123.9	32.26, 31.24	143, 142.8
(0.5, 0.8)	52.1, 51.68	9.35, 8.8	117.3, 118.2	52.31, 52.34	8.49, 7.61	118.2, 111.6
(1, 1)	57.59, 58.31	5.07, 4.43	403.6, 369.5	52.29, 50.81	4.74, 4.28	382.2, 413.6
(2, 2)	48.33, 47.02	1.03, 0.19	>10,000	43.19, 41.41	1.04, 0.22	>10,000
(μ_1, μ_2)	$\sigma_1^2, \sigma_2^2, \sigma_{12} = 1.05, 1.05, 0.5$			$\sigma_1^2, \sigma_2^2, \sigma_{12} = 1.05, 1.3, 0.5$		
(0, 0)	169.6, 167.2	111.5, 114	160.3, 155.6	99.24, 98.47	30.91, 31.46	111.5, 107.7
(0.1, 0)	144.2, 142.4	98.1, 100.21	157, 157.8	85.06, 84.01	33.06, 31.67	116, 125
(0.1, 0.3)	117.3, 116.6	42.04, 39.82	121.8, 125.3	123.7, 127.8	16.58, 17.14	109.6, 106.4
(0.3, 0)	106.5, 110.8	31.94, 29.97	145.3, 143.2	56.9, 56.08	20.03, 18.95	117, 119.1
(0.5, 0.8)	54.98, 51.53	7.92, 7.35	125.4, 126.1	73.16, 72.22	6.06, 5.25	134, 136.4
(1, 1)	50.73, 49.55	4.52, 3.93	370.3, 353.2	55.04, 56.33	3.83, 3.26	318.4, 337.9
(2, 2)	41.48, 43.25	1.03, 0.19	>10,000	45.47, 43.74	1.04, 0.21	>10,000
(μ_1, μ_2)	$\sigma_1^2, \sigma_2^2, \sigma_{12} = 1.2, 1.2, 0.6$			$\sigma_1^2, \sigma_2^2, \sigma_{12} = 1.4, 1.4, 0.5$		
(0, 0)	138.6, 134.6	54.05, 54.47	132, 131.3	38.51, 36.3	8.93, 9.49	73.89, 75.73
(0.1, 0)	127.4, 131	50.91, 49.72	125.2, 117.8	32.91, 32.83	8.87, 8.62	72.35, 75.25
(0.1, 0.3)	122, 118.4	25.85, 24.53	108.3, 104.1	54.57, 52.75	6.55, 6.16	83.54, 83.87
(0.3, 0)	83.93, 82.46	21.56, 19.51	131.3, 130	24.62, 24.27	6.96, 6.24	82.86, 79.47
(0.5, 0.8)	60.66, 62.22	7.25, 6.57	123.7, 128.5	46.98, 45.42	3.75, 3.27	151.9, 146.2
(1, 1)	52.75, 54.07	3.99, 3.56	342.6, 346.2	32.56, 31.69	2.57, 2.02	364.4, 311.5
(2, 2)	45.81, 48.53	1.05, 0.24	>10,000	35.74, 34.2	1.03, 0.2	>10,000
(μ_1, μ_2)	$\sigma_1^2, \sigma_2^2, \sigma_{12} = 1.4, 1.4, 0.75$			$\sigma_1^2, \sigma_2^2, \sigma_{12} = 3, 3, 0.5$		
(0, 0)	112.3, 112.9	24.84, 23.7	118.8, 120.6	2.53, 2.05	1.16, 0.43	58.39, 58.91
(0.1, 0)	90.95, 95.76	24.12, 23.53	114.7, 114.9	2.53, 1.98	1.13, 0.38	62.33, 63.42
(0.1, 0.3)	121.1, 119.5	15.19, 15.28	95.5, 89.7	2.8, 2.16	1.15, 0.43	72.95, 70.41
(0.3, 0)	67.03, 68.44	13.98, 13.48	122.5, 120.3	2.55, 1.97	1.13, 0.36	66.19, 65.73
(0.5, 0.8)	68.94, 66.51	5.32, 4.85	113.6, 110.5	3.36, 2.78	1.11, 0.36	224.6, 232.5
(1, 1)	55.07, 52.25	3.34, 2.84	260.3, 259.9	3.44, 2.74	1.07, 0.29	679.6, 665.7
(2, 2)	54.17, 54.49	1.05, 0.23	>10,000	6.23, 5.54	1, 0.06	>10,000

It is clear from the results in Table 13 that in all cases, the SVR scheme performs better than the other two in this scenario. By comparing Tables 7 and 13 (to its equivalent scenario with only detection as the goal), we realize that the ANN scheme mostly performs worse in this scenario. The RFR scheme mostly performs worse as well, but the deterioration is more severe, especially as the mean shifts increase. The SVR scheme mostly performs worse too, but the deterioration in performance in most cases is not that noticeable compared to the ANN and RFR schemes. By comparing Tables 9 and 13 (two different input methods, but the same training methods), we can see that the ANN scheme performs significantly worse with the second input set. On the contrary, the SVR scheme performs significantly better with this input set so that it does not experience any deterioration in performance under large shifts. Regarding the RFR scheme, it performs better under small mean shifts, but as the mean shift increases, its performance becomes significantly worse.

Moreover, by comparing the results in Table 13 to Sabahno et al.'s [9] Table 4, one can see that the situation is improved compared to the previous scenario, and in most cases (except in the case of (0.5, 0.8) mean shifts), at least one of the proposed ML control charts performs better than all their proposed control charts again, even though their control charts are only designed for detection.

Scenario b22 (Control Type b, Input Set 2, Training Method 2)

The construction of in-control and out-of-control datasets is the same as in Scenario b12. For the ANN scheme, 10 nodes in each hidden layer are used. The obtained RMSEs are 0.38, 0.36, 0.41, 0.42, and 0.38. The UCLs are computed as 0.9637, 1.2445, 1.7237, 1.034, and 1.061. For the SVR scheme, again linear kernels are used in all the structures, with the RMSEs equal to 0.41, 0.4, 0.47, 0.47, and 0.43 and the UCLs computed as 0.8145, 0.8271, 0.9899, 0.9878, and 0.9457. Regarding the RFR scheme, 100 trees for each ML structure are used. The obtained RMSEs are 0.42, 0.4, 0.45, 0.45, and 0.43. The computed UCLs are 0.9401, 0.9528, 0.9458, 0.9284, and 0.894. The results of this analysis are reported in Table 14.

Table 14. ARL, SDRL. Scenario b22 (with identification), $p = 2$.

	ANN	SVR	RFR	ANN	SVR	RFR
(μ_1, μ_2)	$\sigma_1^2, \sigma_2^2, \sigma_{12} = 1, 1, 0.5$			$\sigma_1^2, \sigma_2^2, \sigma_{12} = 1.05, 1, 0.5$		
(0, 0)	200, 200	200, 200	200, 200	178.1, 179.7	148.2, 152	153.3, 149.7
(0.1, 0)	153.9, 151.9	155.4, 152	148, 142	132.5, 132	117.4, 121.1	129.3, 133.3
(0.1, 0.3)	108.7, 107.5	54.17, 56.74	69.72, 70.86	118.4, 120.6	44.3, 44.38	65.13, 63.27
(0.3, 0)	44.82, 42.93	40.26, 39.92	47.45, 47.45	38.06, 36.72	32.54, 32.8	44.68, 47.01
(0.5, 0.8)	28.11, 29.32	8.69, 7.84	23.04, 21.48	30.53, 29.86	8.05, 7.17	22.21, 21.46
(1, 1)	16.27, 15.87	4.88, 4.32	60.94, 56.68	16.11, 14.79	4.57, 4.01	54.22, 54.52
(2, 2)	8.55, 8.31	1.02, 0.15	>10,000	9.41, 8.35	1.03, 0.18	>10,000
(μ_1, μ_2)	$\sigma_1^2, \sigma_2^2, \sigma_{12} = 1.05, 1.05, 0.5$			$\sigma_1^2, \sigma_2^2, \sigma_{12} = 1.05, 1.3, 0.5$		
(0, 0)	160.7, 165.5	117.5, 119.7	140.7, 146.2	68.12, 68.09	31.45, 31.49	65.31, 66.4
(0.1, 0)	115.2, 113.9	97.86, 100.5	110, 116.6	54.93, 53.79	30.67, 30.35	65.18, 64
(0.1, 0.3)	103.9, 106.1	36.53, 35.09	63.48, 61.96	65.09, 65.87	16.59, 15.88	44.09, 42.66
(0.3, 0)	35.67, 35.27	32.35, 31.18	42.7, 39.96	26.15, 26.13	20.27, 19.6	34.88, 33.61
(0.5, 0.8)	26.22, 26	7.21, 6.7	21.78, 21.91	16.03, 15.26	5.81, 5.3	22.63, 22.88
(1, 1)	14.59, 14.39	4.46, 3.8	54.27, 54.18	8.78, 7.95	3.76, 3.33	41.85, 42.24
(2, 2)	9.35, 8.75	1.04, 0.2	>10,000	8.85, 8.08	1.04, 0.21	>10,000
(μ_1, μ_2)	$\sigma_1^2, \sigma_2^2, \sigma_{12} = 1.2, 1.2, 0.6$			$\sigma_1^2, \sigma_2^2, \sigma_{12} = 1.4, 1.4, 0.5$		
(0, 0)	121.5, 126.7	55.63, 54.19	94.1, 97.06	26.39, 25.6	8.66, 8.15	28.33, 27.67
(0.1, 0)	84.17, 84.25	49.78, 50.41	75.83, 76.77	22.12, 21.59	8.36, 7.67	27.04, 25.97
(0.1, 0.3)	93.44, 92.86	24.24, 23.77	55.67, 56.03	33.73, 34.4	6.8, 6.24	26.5, 26.05
(0.3, 0)	28.67, 28.06	22.83, 22.92	34.83, 34.89	12.12, 11.71	7.4, 7.06	21.43, 20.75
(0.5, 0.8)	21.62, 20.51	6.47, 5.76	22.83, 22.38	15.73, 14.94	3.42, 2.81	21.8, 20.96
(1, 1)	11.13, 10.06	4, 3.5	48.59, 44.41	7.24, 6.82	2.66, 2.1	40.76, 38.22
(2, 2)	7.87, 7.28	1.04, 0.2	>10,000	8.43, 7.65	1.02, 0.16	>10,000
(μ_1, μ_2)	$\sigma_1^2, \sigma_2^2, \sigma_{12} = 1.4, 1.4, 0.75$			$\sigma_1^2, \sigma_2^2, \sigma_{12} = 3, 3, 0.5$		
(0, 0)	79.65, 74.89	26.41, 74.89	26.41, 26.34	2.99, 2.39	1.13, 0.37	12.99, 12.41
(0.1, 0)	55.06, 55.22	26.12, 25.51	64.21, 59.84	2.74, 2.13	1.15, 0.41	13.86, 13.97
(0.1, 0.3)	68.62, 71.7	15.83, 15.82	46.4, 44.21	3.19, 2.55	1.12, 0.35	14.38, 13.47
(0.3, 0)	22.16, 21.4	14.23, 13.54	33.38, 31.51	2.46, 1.87	1.13, 0.4	16.09, 15.11
(0.5, 0.8)	15.89, 15.82	5.39, 4.8	25.09, 24.34	2.82, 2.27	1.09, 0.31	23.9, 23.52
(1, 1)	8.03, 7.9	3.44, 2.83	51.26, 51.36	2.35, 1.74	1.08, 0.3	61.49, 58.96
(2, 2)	5.75, 5.25	1.03, 0.19	>10,000	1.85, 1.3	1, 0.08	>10,000

According to the results in Table 14, in most cases, the SVR scheme performs better than the other two. By comparing Table 14 to Table 13 (two different training methods), we realize that the ANN and RFR schemes perform better with the second training method. On the other hand, on average, the performance of the SVR scheme is very similar in the two training methods. By comparing Table 14 to its equivalent in scenario a (which is Table 8), we can conclude that all the schemes mostly perform worse when the identification is a goal and when the second training method is used. Having said that, the RFR scheme is the most affected one, while the SVR scheme is the least affected. By comparing Table 14 to Table 11 (different input sets), we realize that the ANN scheme mostly performs worse

with this input set. The RFR scheme mostly performs worse too, but it even experiences a significant deterioration in performance as the shifts become larger than the training value in this input set. The SVR scheme performs better with this input method such that there is no deterioration in performance anymore as the shift size becomes larger than the trained value (unlike the RFR scheme).

Moreover, by comparing the results in Table 14 to Sabahno et al.’s [9] Table 4, one can see that similar to the previous scenario, except in the case of (0.5, 0.8) mean shift, at least one of the proposed ML control charts performs better than all their proposed control charts.

4. An Illustrative Example

A real case originally discussed by Hawkins and Maboudou-Tchao [2] regarding a healthcare process for monitoring blood pressure and heart rate is used in this section to illustrate the application of the proposed ML schemes. The main indicators, in this case, are heart attack and stroke. The quality characteristics are x_1 = systolic blood pressure, x_2 = diastolic blood pressure, and x_3 = heart rate. They follow a multivariate normal distribution with the following in-control parameter values.

$$\mu_0 = (126.61, 77.48, 80.95)' \text{ and } \Sigma_0 = \begin{bmatrix} 15.04 & 8.66 & 10.51 \\ 8.66 & 5.83 & 5.56 \\ 10.51 & 5.56 & 15.17 \end{bmatrix}.$$

To identify the quality characteristic and the process parameter responsible for the chart signal, the second control scenario is used for this practical case. Based on the results of the numerical analyses section, the first training method for the ANN and SVR schemes and the second training method for the RFR scheme (because the results showed that the RFR scheme performs better with the second training method) are utilized. The mean shift size used for training is $0.2 \times \sigma$ (note that in the numerical analyses section, since all the standard deviations were equal to 1, we simply used $0.2 \times 1 = 0.2$) for small shifts, and similarly, it is $1 \times \sigma$ for large shifts. The shifted variances used for training are as they were in Section 3 (because in both cases, they are multiplied by a coefficient). In addition, we assume that detection of the covariance shift is not a priority for the quality system; therefore, we only consider six control charts for monitoring $\mu_1, \mu_2, \mu_3, \sigma_1^2, \sigma_2^2,$ and σ_3^2 . The UCLs are computed using the proposed algorithm in Section 2 with $\alpha = 0.005$ and $n = 10$. Also, the same R packages as in the simulations study section are used in this section. Similarly, the only changes we carried out in those packages’ default settings were changing the output type to regression and changing the number of trees in the RFR scheme, the number of nodes in the hidden layer in the ANN scheme, and the kernel type in the SVR scheme.

For the SVR scheme, radial kernels are used in all the structures. The RMSEs are 0.47, 0.47, 0.51, 0.44, 0.46, and 0.52. The UCLs 1.0714, 1.0792, 1.0735, 1.0744, 1.0033, and 1.0932. For the ANN scheme, five nodes are used in the hidden layers. The RMSEs are 0.45, 0.45, 0.49, 0.42, 0.44, and 0.49. The UCLs are 0.9934, 1.7028, 1.0451, 0.9771, 0.9736, and 0.8285. For the RFR scheme, each structure’s number of trees is as follows: 100, 100, 100, 300, 100, and 500, respectively. The bias correction feature is only turned on for the last structure. The RMSEs are 0.4, 0.4, 0.43, 0.38, 0.41, and 0.47. The UCLs are 0.9858, 0.98, 0.9999, 0.9986, 0.9993, and 0.9914.

To see how each control chart would react in the case of an out-of-control situation, we make an artificial shift to the process and take ten consecutive samples from the process. To do so, we shift the third mean by 1.1 ($\mu_3 = 80.95 + 1.1$) and the third variance by 1.4 ($\sigma_3^2 = 1.4 \times 15.17$). It should be noted that we use the detection plus identification scenario; therefore, we have six control charts for each of the proposed ML schemes, and if any of those six charts signal in each scheme, we call the process out of control.

The results of ten consecutive random samplings from the process for each control scheme, i.e., SVR, ANN, and RFR, are reported in Tables 15–17, respectively. Note that, since there are six control charts involved in each scheme (making it a total of 18 control charts), to reduce the paper size, they are presented in tables.

Table 16. The ANN scheme for the illustrative example.

Sample		1	2	3	4	5	6	7	8	9	10
input	T^2	6.5128	15.7031	1.3442	1.3567	3.2905	1.2372	6.0249	10.778	4.049	14.8453
	W	7.2728	8.1192	6.4587	9.7029	5.818	6.3745	9.3089	7.5685	14.8656	16.7297
ANN1	output	0.5011	0.8582	0.2728	0.1333	0.1906	0.2616	0.4963	0.7294	0.3312	0.8917
	UCL	0.9934	0.9934	0.9934	0.9934	0.9934	0.9934	0.9934	0.9934	0.9934	0.9934
	status	In-control	In-control	In-control	In-control	In-control	In-control	In-control	In-control	In-control	In-control
ANN2	output	0.4603	0.8571	0.0971	0.0984	0.298	0.0894	0.4286	0.7975	0.3115	0.778
	UCL	1.7028	1.7028	1.7028	1.7028	1.7028	1.7028	1.7028	1.7028	1.7028	1.7028
	status	In-control	In-control	In-control	In-control	In-control	In-control	In-control	In-control	In-control	In-control
ANN3	output	0.5155	0.9792	0.1965	0.3162	0.2041	0.1703	0.4309	0.7775	0.3236	1.1416
	UCL	1.0451	1.0451	1.0451	1.0451	1.0451	1.0451	1.0451	1.0451	1.0451	1.0451
	status	In-control	In-control	In-control	In-control	In-control	In-control	In-control	In-control	In-control	Out-of-control
ANN4	output	0.7431	0.9343	0.6765	0.2614	0.8417	0.6917	0.4771	0.8653	0.0965	0.2469
	UCL	0.9771	0.9771	0.9771	0.9771	0.9771	0.9771	0.9771	0.9771	0.9771	0.9771
	status	In-control	In-control	In-control	In-control	In-control	In-control	In-control	In-control	In-control	In-control
ANN5	output	0.7848	0.9268	0.668	0.3124	0.8712	0.683	0.4604	0.8637	0.1478	0.535
	UCL	0.9736	0.9736	0.9736	0.9736	0.9736	0.9736	0.9736	0.9736	0.9736	0.9736
	status	In-control	In-control	In-control	In-control	In-control	In-control	In-control	In-control	In-control	In-control
ANN6	output	0.7737	0.8694	0.793	0.4344	0.7838	0.7943	0.6009	0.8315	0.3094	0.4433
	UCL	0.8285	0.8285	0.8285	0.8285	0.8285	0.8285	0.8285	0.8285	0.8285	0.8285
	status	In-control	Out-of-control	In-control	In-control	In-control	In-control	In-control	Out-of-control	In-control	In-control

Regarding the SVM scheme (Table 15), none of its control charts signal during the first ten consecutive samplings. Regarding the ANN scheme (Table 16), the ANN6 chart, which is responsible for the detection of σ_3^2 shifts, signals at samples no. 2 and no. 8, and the ANN3 chart, which is responsible for the detection of μ_3 shifts, signals at sample no. 10. Regarding the RFR scheme (Table 17), its RFR1 and RFR3 charts, which are responsible for the detection of μ_1 and μ_3 shifts, respectively, both signal at sample no. 10. Clearly in the case of RFR1, we received a false alarm from the RFR scheme, considering there is no shift in the first mean.

Note that, this was only a simple example to show how each proposed control scheme can be implemented in practice, and based on only 10 samples, no comparisons can be made. Performance comparisons were the purpose of the previous section.

5. Concluding Remarks

This paper proposed new control charts for simultaneous monitoring of multivariate normal process mean vectors and the variance–covariance matrix. For the first time, machine-learning techniques were used for this purpose. Three used ML techniques are ANN, SVM, and RF. We received linear outputs from these ML structures and then applied control chart rules to decide whether the process is in control or out of control. Two different input sets and two different training methods were employed for the proposed ML structures. In the first input set, two statistics (one representing the process mean vector and the other for the process variability) were employed, and for the second input set, we used each process parameter for each quality characteristic separately as inputs. In the first training method, we only trained the ML structures with a small shift size, and in the other method, a small and a large shift size were considered. We also used two different process control scenarios. In the first scenario, the only goal was the detection of the out-of-control situation, regardless of which variable and process parameter is responsible for it. In the second scenario, on the other hand, other than detection, identifying which variable (s)/process parameter (s) is responsible for the signal was also a goal, which involved several control charts to be monitored together.

For each of these control–input–training scenarios, the ML structures were trained, and control charts were developed. Numerical analyses were performed for the cases of two, three, and four quality characteristics. The results, in general, showed that depending on which control–input–training scenario is used, as well as the number of variables, each of these ML control charts performed better in some cases, and there is no absolute winner among them. However, considering how its decision-making procedure works based on dividing tree branches, the RFR scheme tended to mostly perform better when there were more inputs, more diverse training, and more quality characteristics (higher dimension). However, this did not happen in all cases (except for the diverse training part), and most importantly, even if its performance was improved by all these diversities, it did mean that it should perform better than the ANN and SVR charts (which actually in most cases it did not). It was also concluded that when identification was also a goal, the charts performed worse. However, this deterioration in performance was mostly at its lowest for one ML scheme (it differed based on the scenario).

We also compared the proposed ML charts with some recently developed multivariate statistical control charts with fixed and adaptive chart parameters (designed only for detection). For the case of $p = 2$, the results showed that in the detection-only scenario and with the first input set, at least one of the proposed ML charts performed better than all their proposed charts in all the shift cases, even though our proposed schemes are all fixed parameters. With the second input set together with the first training method, our proposed charts performed better in most cases, and together with the second training method, they performed better in all cases. Regarding the detection–identification scenario, our proposed ML charts still performed better in more cases, even though their charts have only been designed for detection and that usually by default means better performance. For the case of $p = 3$, the results showed that in the detection-only scenario, at least one of the ML charts

performed better than all their proposed charts in most shift cases, and with the second training method, they performed better in all the shift cases. In the detection–identification scenario, one can say that the cases in which at least one of our proposed schemes performed better than all their charts and vice versa were almost the same. However, keep in mind that, unlike their charts, our proposed charts are also capable of identification. Lastly, an illustrative example based on a healthcare-related practical case was presented to show how the proposed schemes can be implemented in practice.

Highlighting the primary focus of this paper, our investigation centered the utilization of diverse machine-learning techniques in constructing control charts, effectively substituting traditional statistical methods. This exploration involved rigorous testing of different input sets and training methodologies to surpass the performance of statistical control charts. While our study concentrated on specific control charts, along with a limited selection of input sets and training approaches, it presents an opportunity for further exploration of a wide range of control charts, as well as diverse input sets and training methods, thus broadening the horizons of research in this field.

For future developments, one might be interested in trying different input sets, training methods, and even output sets for ML structures. Adding adaptive features to the proposed control charts would also be a major improvement. Since ML control charts have rarely been developed, developing them for many other applications and comparing them to traditional statistical control charts might also be interesting. In particular, since all the developed ML control charts are so far memory-less, developing memory-type ML control charts and comparing them to memory-type statistical control charts might also be interesting for some researchers. In addition, how to train the ML structures in the case of unknown distributions is another challenge that might be worth investigating by some other researchers.

Author Contributions: Conceptualization, H.S.; methodology, H.S. and S.T.A.N.; software, H.S.; validation, H.S.; formal analysis, H.S.; investigation, H.S. and S.T.A.N.; resources, H.S.; data curation, H.S.; writing—original draft preparation, H.S.; writing—review and editing, S.T.A.N.; visualization, H.S.; supervision, S.T.A.N.; project administration, H.S. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: All data are available in the manuscript.

Acknowledgments: The authors thank the journal’s editorial board and appreciate the esteemed reviewers for their constructive comments, which led to significant improvements in the quality of the paper.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Reynolds, M.R., Jr.; Gyo-Young, C. Multivariate control charts for monitoring the mean vector and covariance matrix. *J. Qual. Technol.* **2006**, *38*, 230–253.
2. Hawkins, D.M.; Maboudou-Tchao, E.M. Multivariate exponentially weighted moving covariance matrix. *Technometrics* **2008**, *50*, 155–166. [[CrossRef](#)]
3. Zhang, G.; Chang, S.I. Multivariate EWMA control charts using individual observations for process mean and variance monitoring and diagnosis. *Int. J. Prod. Res.* **2008**, *46*, 6855–6881. [[CrossRef](#)]
4. Khoo, M.B.C. A new bivariate control chart to monitor the multivariate process mean and variance simultaneously. *Qual. Eng.* **2004**, *17*, 109–118. [[CrossRef](#)]
5. Zhang, J.; Li, Z.; Wang, Z. A multivariate control chart for simultaneously monitoring process mean and variability. *Comput. Stat. Data Anal.* **2010**, *54*, 2244–2252. [[CrossRef](#)]
6. Wang, K.; Yeh, A.B.; Li, B. Simultaneous monitoring of process mean vector and covariance matrix via penalized likelihood estimation. *Comput. Stat. Data Anal.* **2014**, *78*, 206–217. [[CrossRef](#)]
7. Sabahno, H.; Castagliola, P.; Amiri, A. A variable parameters multivariate control chart for simultaneous monitoring of the process mean and variability with measurement errors. *Qual. Reliab. Eng. Int.* **2020**, *36*, 1161–1196. [[CrossRef](#)]

8. Sabahno, H.; Castagliola, P.; Amiri, A. An adaptive variable-parameters scheme for the simultaneous monitoring of the mean and variability of an autocorrelated multivariate normal process. *J. Stat. Comput. Simul.* **2020**, *90*, 1430–1465. [[CrossRef](#)]
9. Sabahno, H.; Amiri, A.; Castagliola, P. A new adaptive control chart for the simultaneous monitoring of the mean and variability of multivariate normal processes. *Comput. Ind. Eng.* **2021**, *151*, 106524. [[CrossRef](#)]
10. Sabahno, H.; Khoo, M.B.C. A multivariate adaptive control chart for simultaneously monitoring of the process parameters. *Commun. Stat. Simul. Comput.* **2022**, 1–19. [[CrossRef](#)]
11. Sabahno, H. An adaptive max-type multivariate control chart by considering measurement errors and autocorrelation. *J. Stat. Comput. Simul.* **2023**, 1–26. [[CrossRef](#)]
12. Chang, S.I.; Ho, E.S. A two-stage network approach for process variance change detection and classification. *Int. J. Prod. Res.* **1999**, *37*, 1581–1599. [[CrossRef](#)]
13. Niaki, S.T.A.; Abbasi, B. Fault diagnosis in multivariate control charts using artificial neural networks. *Qual. Reliab. Eng. Int.* **2005**, *21*, 825–840. [[CrossRef](#)]
14. Niaki, S.T.A.; Abbasi, B. Detection and classification mean-shifts in multiattribute processes by artificial neural networks. *Int. J. Prod. Res.* **2008**, *46*, 2945–2963. [[CrossRef](#)]
15. Cheng, C.-S.; Cheng, H.-P. Identifying the source of variance shifts in the multivariate process using neural networks and support vector machines. *Expert Syst. Appl.* **2008**, *35*, 198–206. [[CrossRef](#)]
16. Abbasi, B. A neural network applied to estimate process capability of nonnormal processes. *Expert Syst. Appl.* **2009**, *36*, 3093–3100. [[CrossRef](#)]
17. Salehi, M.; Bahreininejad, A.; Nakhai, I. On-line analysis of out-of-control signals in multivariate manufacturing processes using a hybrid learning-based model. *Neurocomputing* **2011**, *74*, 2083–2095. [[CrossRef](#)]
18. Hosseini-fard, S.Z.; Abdollahian, M.; Zeepongsekul, P. Application of artificial neural networks in linear profile monitoring. *Expert Syst. Appl.* **2011**, *38*, 4920–4928. [[CrossRef](#)]
19. Weese, M.; Martinez, W.; Megahed, F.M.; Jones-Farmer, L.A. Statistical learning methods applied to process monitoring: An overview and perspective. *J. Qual. Technol.* **2016**, *48*, 4–24. [[CrossRef](#)]
20. Escobar, C.A.; Morales-Menendez, R. Machine learning techniques for quality control in high conformance manufacturing environment. *Adv. Mech. Eng.* **2018**, *10*, 1–16. [[CrossRef](#)]
21. Apsemidis, A.; Psarakis, S.; Moguerza, J.M. A review of machine learning kernel methods in statistical process monitoring. *Comput. Ind. Eng.* **2020**, *142*, 106376. [[CrossRef](#)]
22. Mohd Amiruddin, A.A.A.; Zabiri, H.; Taqvi, S.A.A.; Dendena Tufa, L. Neural network applications in fault diagnosis and detection: An overview of implementations in engineering-related systems. *Neural Comput. Appl.* **2020**, *32*, 447–472. [[CrossRef](#)]
23. Demircioglu Diren, D.; Boran, S.; Cil, I. Integration of machine learning techniques and control charts in multivariate processes. *Sci. Iran.* **2020**, *27*, 3233–3241. [[CrossRef](#)]
24. Yeganeh, A.; Pourpanah, F.; Shadman, A. An ANN-based ensemble model for change point estimation in control charts. *Appl. Soft Comput.* **2021**, *110*, 107604. [[CrossRef](#)]
25. Mohammadzadeh, M.; Yeganeh, A.; Shadman, A. Monitoring logistic profiles using variable sample interval approach. *Comput. Ind. Eng.* **2021**, *158*, 107438. [[CrossRef](#)]
26. Sabahno, H.; Amiri, A. New statistical and machine learning based control charts with variable parameters for monitoring generalized linear model profiles. *Comput. Ind. Eng.* **2023**; *submitted*.
27. Yeganeh, A.; Chukhrova, N.; Johannssen, A.; Fotuhi, H. A network surveillance approach using machine learning based control charts. *Expert Syst. Appl.* **2023**, *219*, 119660. [[CrossRef](#)]
28. Yeganeh, A.; Shadman, A.; Shongwe, S.C.; Abbasi, S.A. Employing evolutionary artificial neural network in risk-adjusted monitoring of surgical performance. *Neural Comput. Appl.* **2023**, *35*, 10677–10693. [[CrossRef](#)]
29. Yeganeh, A.; Johannssen, A.; Chukhrova, N.; Abbasi, S.A.; Pourpanah, F. Employing machine learning techniques in monitoring autocorrelated profiles. *Neural Comput. Appl.* **2023**, *35*, 16321–16340. [[CrossRef](#)]

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.