Residual Control Chart for Binary Response with Multicollinearity Covariates by Neural Network Model

Jong-Min Kim

Ning Wang

Yumin Liu

Kayoung Park
Old Dominion University, kypark@odu.edu

Original Publication Citation
Kim, J. M., Wang, N., Liu, Y., & Park, K. (2020). Residual control chart for binary response with multicollinearity covariates by neural network model. *Symmetry, 12*(3), 381. doi:10.3390/sym12030381

symmetry

Article

Residual Control Chart for Binary Response with Multicollinearity Covariates by Neural Network Model

Jong-Min Kim [1], Ning Wang [2,*], Yumin Liu [2] and Kayoung Park [3]

[1] Division of Science and Mathematics, University of Minnesota-Morris, Morris, MN 56267, USA; jongmink@morris.umn.edu

[2] Business School, Zhengzhou University, Zhengzhou 450001, China; zz-wn@zzu.edu.cn (N.W.); yuminliu@zzu.edu.cn (Y.L.)

[3] Department of Mathematics and Statistics, Old Dominion University, Norfolk, Virginia, USA; kypark@odu.edu

* Correspondence: zz-wn@zzu.edu.cn

Received: 4 February 2020; Accepted: 26 February 2020; Published: 3 March 2020

Abstract: Quality control studies have dealt with symmetrical data having the same shape with respect to left and right. In this research, we propose the residual (r) control chart for binary asymmetrical (non-symmetric) data with multicollinearity between input variables via combining principal component analysis (PCA), functional PCA (FPCA) and the generalized linear model with probit and logit link functions, and neural network regression model. The motivation in this research is that the proposed control chart method can deal with both high-dimensional correlated multivariate data and high frequency functional multivariate data by neural network model and FPCA. We show that the neural network r control chart is relatively efficient to monitor the simulated and real binary response data with the narrow length of control limits.

Keywords: Residual Control Chart; binary data; PCA; FPCA; multicollinearity

1. Introduction

The current available quality control research has focused on symmetrical data having the same shape with respect to left and right. Data are getting bigger and highly correlated with each other and have asymmetric (non-symmetric) distributions. Therefore, the quality control is facing difficulty to handle highly correlated data so that we have a hard time to get accurate information from the current available control charts. In order to monitor a process mean vector, there have been a number of multivariate control charts including Hotelling T^2 distribution [1], mulvariate CUSUM [2] and multivariate EWMA [3]. These current available multivariate control charts have limitations to handle high-dimensional data because of the complexity of the covariance structure. Neural network based methods have been applied to quality control research areas, but there is no research available for residual (r) control charts for binary asymmetrical data with highly correlated multivariate covariates by using neural network regression model. Hence, this is a motivation to propose the r control chart for binary asymmetrical data with multicollinearity between input variables via principal component analysis (PCA), functional PCA (FPCA), and neural network model. To deal with high correlations among independent variables, [4] proposed Poisson, negative binomial, COM-Poisson-based principal component regression-based r-control charts for monitoring dispersed count data. More detailed information for diverse control charts can be found in [5,6]. The novelties with respect to existing strategies in this research have two things. The first one is that the proposed control chart method can deal with high-dimensional correlated multivariate data by neural network model and the second one is that the proposed control chart method can deal high frequency functional multivariate data by FPCA. One of the applications by the proposed statistical process control is monitoring clinical performance which measures binary asymmetrical data such as mortality with patient medical information.

2. Statistical Methods

In this research, we present regression-based r-control charts combining principal component analysis methods (PCA and FPCA) and binary response regression models (generalized linear model and neural network regression model) for binary asymmetrical data with multicollinearity among independent variables.

2.1. Generalized Linear Model and Neural Network Model for Binary Response Data

To introduce the binary response regression models, the generalized linear model (GLM) should be considered first because GLM is both a generalized and flexible model which can consider binary asymmetrical data. The GLM has the following probability density distribution which comes from the exponential family:

$$g(y|\lambda,\delta) = \exp\left[\frac{y(\lambda) - a_2(\lambda)}{a_1(\delta)} + a_3(y,\delta)\right] \tag{1}$$

where we denote the response variable to be y, the location parameter to be λ, the dispersion parameter to be δ, and arbitrary functions to be $a_1(\cdot)$, $a_2(\cdot)$, and $a_3(\cdot)$. We denote ζ to be the linear predictor for the response, $\mathbf{y}$ so that ζ is a linear combination of unknown parameters $\mathbf{b} = (\mathbf{b_0}, \mathbf{b_1}, \cdots, \mathbf{b_p})'$ and input variables $\mathbf{x} = (\mathbf{1}, \mathbf{x_1}, \cdots, \mathbf{x_p})'$. A link function f such that $E(\mathbf{y}) = \mathbf{f^{-1}}(\imath)$ provides the relationship between the linear predictor and the mean of the distribution function. The link function $f(\cdot)$ specified how to convert the expected value $\mu = E(\mathbf{y})$ to the linear predictor ζ: i.e.,

$$\zeta = f(\mu) = \mathbf{x}'\mathbf{b}. \tag{2}$$

Using a logit model as an example, we have

$$Logit[P(\mathbf{y} = \mathbf{1}|\mathbf{x})] = \mathbf{x}'\mathbf{b}, \tag{3}$$

where b is the column vector of the fixed-effects regression coefficients. For the (3), it can be written as

$$P(\mathbf{y} = \mathbf{1}|\mathbf{x}) = \frac{\mathbf{1}}{\mathbf{1} + \mathbf{e}^{-\mathbf{x}'\mathbf{b}}} = \mathbf{m}(\mathbf{x}) \tag{4}$$

With the (4), we can derive the likelihood function for GLMs as follows:

$$L(\mathbf{b}) = \prod_{\mathbf{i}=\mathbf{1}}^{\mathbf{n}} \mathbf{m}(\mathbf{x_i})^{\mathbf{y_i}}\{\mathbf{1} - \mathbf{m}(\mathbf{x_i})\}^{\mathbf{1}-\mathbf{y_i}}. \tag{5}$$

and so the log-likelihood function is given by

$$l(\mathbf{b}) = \log \mathbf{L}(\mathbf{b}) = \sum_{\mathbf{1}=\mathbf{1}}^{\mathbf{n}} [\mathbf{y_i} \log \mathbf{m}(\mathbf{x_i}) + (\mathbf{1} - \mathbf{y_i}) \log\{\mathbf{1} - \mathbf{m}(\mathbf{x_i})\}]. \tag{6}$$

Then the maximum-likelihood estimating equation for $\mathbf{b}$ is easily solved via standard softwares (e.g., SAS or R) using the Fisher scoring or Newton-Raphson method.

The GLM with probit link function requires the assumptions that the response is binary and that an underlying latent variable governing the binary process follows a normal distribution. In some cases the GLMs with probit link function can probably gives best goodness-of-fit of the test where response variables are assumed to have normal distributions [7] because the response probability distribution of the GLMs belongs to an exponential family of distributions which employ the methods analogous to the normal linear methods for the normal data [8,9]. Therefore, for asymmetrical (non-normal) distributed data, the GLM with probit link function may not be the best model. Hence, this is another motivation to propose a neural network model based on r control chart for the better predictive accuracy with the non-normal data.

An artificial neural network (ANN) is originally inspired from human brain and ANN resembles biological neural networks which imitate human brain activity through a computer simulations [10–12]. Physically, an ANN contains neurons connected by synapses that connected them. The ANN learning process heavily relies on both weights of the connections between the neurons specifying which variables involved in the network and activities of the neurons. The weights are computed by optimizing a learning algorithm. The ANN uses the concept of competition to select the highest probability of inhibiting all neurons [10–12]. Specifically, the most basic form of an ANN is a single layer feedforward type of connection among neurons. ANNs have input layers and multiple hidden layers. Lastly, the hidden layers are connected to the output layer, which produces the outputs. In the last decade, the ANN-based statistical process control research has been actively studied. Research papers such as [13] proposed the pattern recognition for bivariate process mean shifts using feature-based ANN and [14] proposed control chart pattern recognition using Radial Basis Function (RBF) neural networks. Recently, [15] proposed statistical process control with Intelligence Based on the Deep Learning Model and reviewed the neural network-based statistical process control. In this paper, we used 'nnet' R packge [16] for feed-forward neural networks with a single hidden layer, and for multinomial log-linear models. With repeated simulated data (each sample size 1,000 and 10,000 different replications for the case of PCA and each sample size 1000 and 30 different replications for the case of FPCA) and real data, we employed single layer and 30 neurons for the simulation study and real data analysis.

The r-control charts for binary response data uses the GLM models with logit and probit link functions and neural network models employ deviance residuals being independent and asymptotically normally distributed with zero mean and unit variance, i.e., $r_i \sim N(0,1)$ for $i = 1, \ldots, n$. In this research, we chose a deviance residual for the GLM models with logit and probit link functions and a neural network model because the R packages for the GLM models with logit and probit link functions and neural network model have a command for producing the deviance residual. It is easy to compare the residuals from both models which are the GLM-based model and neural network model. Ref. [17] proposed Shewhart control limits for the deviance residuals are

$$E(r_i) \pm k\sqrt{Var(r_i)} \approx \pm k \tag{7}$$

where k is defined by the false alarm probability, $\alpha = 1/ARL_0$, and ARL_0 is the average run length (ARL) under the process in-control. The ARL is a measure of the performance of control charts for monitoring a process.

2.2. Dimension Reduction by Principal Component Analysis

The principal component analysis (PCA) is a statistical orthogonal transformation method converting multivariate data set of correlated variables into a set of values of linearly uncorrelated variables called principal components. Hence, the PCA is the most common dimension reduction statistical method which can reduce the dimensionality of multivariate data to the smaller uncorrelated principal components which account for the variation of the original data. The r control chart for binary response regression model with primary principal components by the PCA is a new statistical process control which monitors the binary response variable as a function of uncorrelated PCs. In this paper, we propose a binary response regression model-based r control charts for binary response data overcoming a multicollinearity issue among independent variables.

To verify our proposed method in terms of the model flexibility and performance, we have run simulations for various circumstances: in-control, one inflated, or zero inflated binary data.

Through both simulation study and a real data example, we illustrated the good performance of our proposed method.

Ref. [4] denotes $\mathbf{X}$ to be an $n \times q$ matrix of independent variables and $\mathbf{W}$ be the standardized matrix of $\mathbf{X}$ so that each column has mean 0 and standard deviance 1. Let $\mathbf{Q}$ be the matrix of eigenvectors of $\mathbf{W}'\mathbf{W}$ and $\mathbf{D}$ be the diagonal matrix with eigenvalues of $\mathbf{W}'\mathbf{W}$ on the diagonal and values of zero everywhere else. Hence, we sort the eigenvalues from largest to smallest such as $\lambda_1 > \lambda_2 > \cdots > \lambda_q$ and sort the eigenvectors in $\mathbf{Q}$ accordingly, and let $\mathbf{Q}^*$ be this sorted matrix of eigenvectors. Then, the principal components, $\mathbf{W}^*$, are given by

$$\mathbf{W}^* = \mathbf{W}\mathbf{Q}^*. \tag{8}$$

We can perform a dimensional reduction to the uncorrelated variables $\mathbf{W}^*$ from the original multivariate high correlated data. Our proposed procedure uses these uncorrelated variables $\mathbf{W}^*$ to perform GLM with probit, GLM with logit, and neural network regression models.

2.3. Dimension Reduction by Functional Principal Component Analysis

Ref. [18] proposed a functional PCA (FPCA) for functional data, which is another dimensional reduction statistical method for explaining the variance of components by using non-liner eigenfunctions and for the multivariate highly correlated data, because FPCA overcomes the high-dimensionality difficulty and efficiently examines the sample covariance structure.

The functional form of $y_i(t)$ is given by the sum of the weighted basis functions, $\phi_p(t)$, across the set of times T.

$$y_i(t) = \sum_{k=1}^{P} c_{ip}\phi_p(t), \tag{9}$$

where P is a number of basis functions. In this study, a Fourier basis is used to represent smooth functions as a basis function due to its flexibility and computational advantages. Here, our goal is to obtain a smooth function which fits well into the observed time series, $y_i(t_j)$. For calculating functional PCA, we employ 'fdapace' R package [19]. This package is for functional principal component analysis (FPCA) via the principal analysis by the conditional estimation (PACE) algorithm which yields covariance and mean functions, eigenfunctions and principal component (scores). PACE provides fitted continuous trajectories with confidence bands [20,21].

2.4. New Binary response statistical process control Procedure

A new binary response statistical process control procedure for the deviance residuals, r, from binary response regression models that have high correlated multivariate covariates is proposed through the following steps:

1. Apply the (functional) principal component analysis in input variables $\mathbf{X}$ and obtain the principal components $\mathbf{w}^*$ from (8).
2. Fit the binary response regression model by using the binary response variable y and the (functional) principal components $\mathbf{w}^*$ through probit link function, logit link function, and neural network regression models, respectively.
3. Obtain the deviance residuals from each model.
4. Set k value and obtain the lower and upper control limits of the r-charts using (7).

3. Illustrated Examples

With the proposed method in Section 2, we perform the efficiency comparison among the proposed methods with simulated data and real data.

3.1. Simulation Study

In order to compare the r-charts based on binary regression models, we generate simulated data denoting $\mathbf{x} = (\mathbf{x_1}, \mathbf{x_2}, \mathbf{x_3}, \mathbf{x_4})'$ as input variables which are generated from the multivariate normal distribution with mean $(1,2\vec{,}3,4)^{'}$ and covariance matrix M as follows:

$$M = \begin{bmatrix} 1 & 0.9 & 0.2 & 0.1 \\ 0.9 & 1 & 0.4 & 0.3 \\ 0.2 & 0.4 & 1 & 0.7 \\ 0.1 & 0.3 & 0.7 & 1 \end{bmatrix}$$

With each simulated data $\mathbf{x}$, we define the coefficients of parameters (β's) to be $\beta_0 = 0.1$, $\beta_1 = 0.25$, $\beta_2 = -0.5$, $\beta_3 = 0.25$, $\beta_4 = 0.1$ so that $P(y = 1) = \frac{1}{1+\exp(-0.1-0.25x_1+0.5x_2-0.25x_2-0.1x_4))}$ which passes through an inverse logit function. Then, we generate response variable y randomly by using the Bernoulli distribution with the probability $P(y = 1)$ with sample size 1000. For the one ('1') inflated case of binary response data, we added 0.1 to the probability $P(y = 1)$ such as $P(y = 1) + 0.1$ and for the zero ('0') inflated case of binary response data, we subtracted 0.1 from the probability $P(y = 1)$ such as $P(y = 1) - 0.1$. Also, $P(y = 1)$ is used for the in-control dispersion case. In each setup, we perform $10,000$ different replications of sample size of 1000. Table 1 shows the simulation results. By using the deviance residuals for each model and (7) for $k = 1, 2, 3$, we compute the lower control limit (LCL) and upper control limit (UCL) for the process. The expected length of the confidence interval is computed by the average of the length of control limits. The coverage probability is the proportion of the deviance residuals contained in the control limits. The lower control limit and the upper control limit value for r-chart are calculated by means of y minus and plus its one, two and three standard deviations.

The summary statistics of $P(y = 1)$ in Figure 1 are that the minimum is 0.3450, the first quartile is 0.5723, the median is 0.6222, the mean is 0.6172, the third quartile is 0.6654 and the maximum is 0.8477. The skewness of $P(y = 1)$ is -0.3272 which proves the shape of the simulated data is asymmetry so that it can be more inclined to produce zero value data rather than one value data.

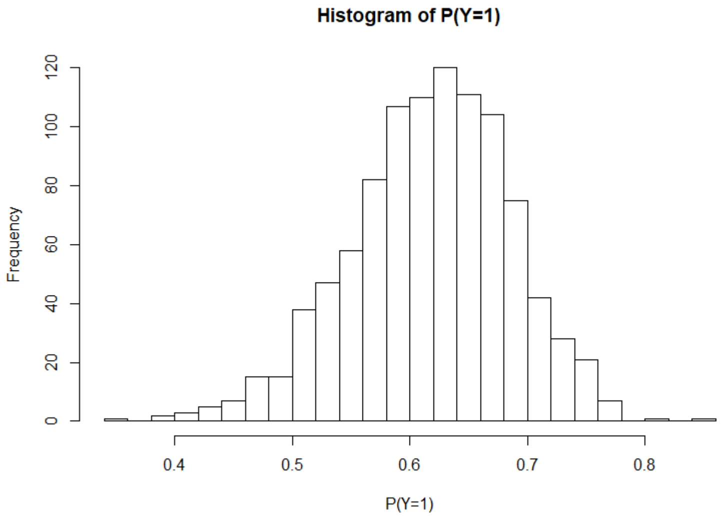


Figure 1. Histogram of $P(y = 1)$ with simulated asymmetrical data.

Based on PCA, Table 1 presents the average run length (ARL) results for simulated in-control, one inflated-, and zero inflated-dispersion data via r-charts based on the GLM with probit and logit link function and neural network models. Changes in w that resulted in one, two and three standard deviations from the mean of y are considered. From Table 1, and based on PCA, we can see that the coverage probabilities by the GLMs with probit and logit link functions are slightly greater or equal to the coverage probabilities by the neural network regression model, but the length of the confidence intervals (CIs) for the neural network regression model are much smaller than the the length of the confidence intervals (CIs) for the GLMs with probit and logit link functions and the ARLs of the neural network regression model are much smaller than the the ARLs for the GLMs with probit and logit link function because of the smaller length of CIs.

Table 1. Based on PCA, the coverage probability, expected confidence interval (CI) length, and control limits for the simulated in-control, one inflated-, and zero inflated-dispersion binary data via various r-charts based on GLM with probit, GLM with logit, and neural network models. Neural network model used single layer and 30 neurons. 'NA' in the table means that there is no points out of control limits and the number of simulations is $10,000$ different replications of sample size of 1000.

		Probit			**Logit**			**Neural Network**		
Case	$E(r_i) \pm w\sqrt{Var(r_i)}$	$k=1$	$k=2$	$k=3$	$k=1$	$k=2$	$k=3$	$k=1$	$k=2$	$k=3$
In-control	ARL	2.590	520.152	NA	2.586	536.938	NA	2.453	322.806	NA
	Center	0.013	0.013	0.013	0.013	0.013	0.013	0.000	0.000	0.000
	LCL	−1.063	−2.202	−3.340	−1.063	−2.201	−3.340	−0.455	−0.911	−1.366
	UCL	1.215	2.354	3.492	1.215	2.354	3.492	0.455	0.911	1.366
	CI Length	2.278	4.555	6.833	2.278	4.555	6.833	0.911	1.821	2.732
	Coverage	0.610	1.000	1.000	0.610	1.000	1.000	0.591	0.998	1.000
One Inflated	ARL	3.532	291.492	NA	3.528	302.546	NA	3.094	62.599	429.500
	Center	0.013	0.013	0.013	0.013	0.013	0.013	0.000	0.000	0.000
	LCL	−0.934	−2.002	−3.069	−0.934	−2.002	−3.069	−0.421	−0.842	−1.263
	UCL	1.200	2.268	3.335	1.200	2.268	3.335	0.421	0.842	1.263
	CI Length	2.135	4.269	6.404	2.135	4.269	6.404	0.842	1.684	2.527
	Coverage	0.717	0.997	1.000	0.717	0.998	1.000	0.677	0.981	1.000
Zero Inflated	ARL	2.177	NA	NA	2.178	NA	NA	2.197	429.347	NA
	Center	0.013	0.013	0.013	0.013	0.013	0.013	0.000	0.000	0.000
	LCL	−1.154	−2.320	−3.486	−1.154	−2.320	−3.486	−0.469	−0.938	−1.406
	UCL	1.179	2.345	3.512	1.179	2.345	3.512	0.469	0.938	1.406
	CI Length	2.333	4.665	6.998	2.333	4.665	6.998	0.938	1.875	2.813
	Coverage	0.536	1.000	1.000	0.536	1.000	1.000	0.555	0.999	1.000

From Figures 2–4 in case of the in-control dispersion based on PCA, we can observe that the residuals of a neural network regression model are much closer to zero than the residuals of the GLM with probit and logit model. Therefore, it is not a surprising result in Table 1 that the r-chart based on the neural network model shows a superiority in all cases in terms of the expected length of the confidence interval. We can say that the r-chart based on the neural network model for monitoring observations has the smallest expected length of the confidence interval with the reasonable coverage probability.

We also see that the r-charts based on the neural network model give the superior performance following the (7), $E(r_i) \pm k\sqrt{Var(r_i)} \approx \pm k$ but the r-charts based on the GLM with logit and probit link functions do not give the good performance following the (7). The reason for the difference is probably the tails of the distribution in the GLMs.

For calculating FPCA, we employ 'fdapace' R package [19] to represent smooth functions with Fourier basis. In order to generate the functional data, we set the number of subjects (N=1000) and the number of measurements per subjects (M=1000). We define the four covariates ($\mathbf{x}$) with four eigencomponents and we define the coefficients of parameters (β's) to be $\beta_0 = 0.1$, $\beta_1 = 0.25$, $\beta_2 = -0.5$, $\beta_3 = 0.25$, $\beta_4 = 0.1$ so that $P(y = 1) = \frac{1}{1+\exp(-0.1-0.25x_1+0.5x_2-0.25x_2-0.1x_4))}$ which passes through an inverse logit function. To apply the simulated data to the proposed methods, we generate 30 different functional simulated data replications of sample size of 1000 in this study. Based on FPCA, Table 2 presents the average run length (ARL) results for simulated in-control, one inflated-, and zero inflated-dispersion data via r-charts based on the GLM with probit and logit link function and neural network models. Changes in w that resulted in one, two and three standard deviations from the mean of y are considered.

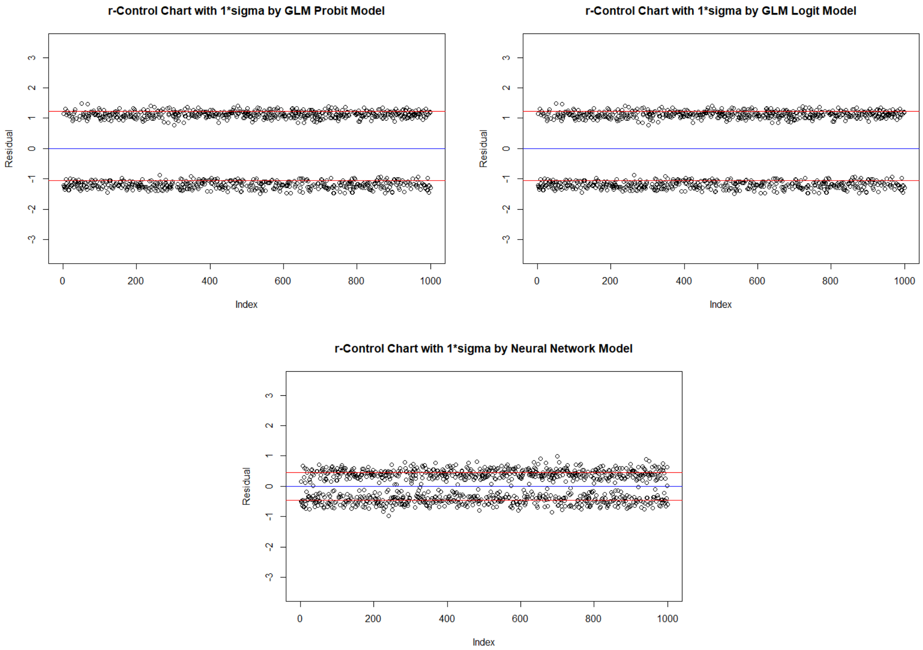


Figure 2. Based on PCA, r control charts $(E(r_i) \pm \sqrt{Var(r_i)})$ for probit, logit and neural network in the in-control case with the number of simulations is $10,000$ different replications of sample size of 1000.

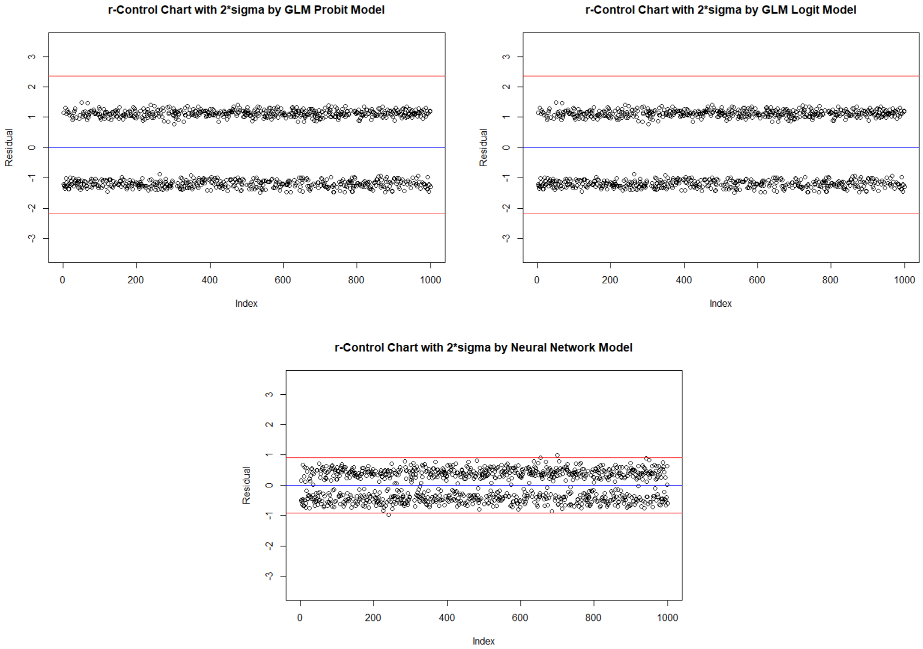


Figure 3. Based on PCA, r control charts $(E(r_i) \pm 2\sqrt{Var(r_i)})$ for probit, logit and neural network in the in-control case with the number of simulations is $10,000$ different replications of sample size of 1000.

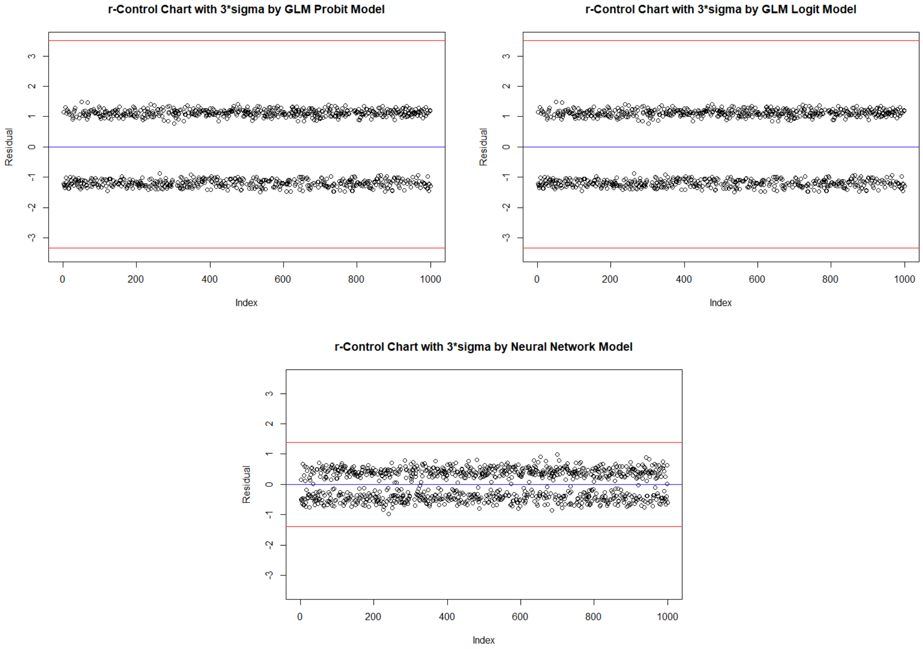


Figure 4. Based on PCA, r control charts $(E(r_i) \pm 3\sqrt{Var(r_i)})$ for probit, logit and neural network in the in-control case with the number of simulations is $10,000$ different replications of sample size of 1000.

Table 2. Based on FPCA, the coverage probability, expected confidence interval (CI) length, and control limits for the simulated in-control, one inflated-, and zero inflated-dispersion binary data via various r-charts based on GLM with probit, GLM with logit, and neural network models. Neural network model used single layer and 30 neurons. 'NA' in the table means that there is no points out of control limits and and the number of simulations is 30 different replications of sample size of 1000.

		Probit			**Logit**			**Neural Network**		
Case	$E(r_i) \pm w\sqrt{Var(r_i)}$	$k = 1$	$k = 2$	$k = 3$	$k = 1$	$k = 2$	$k = 3$	$k = 1$	$k = 2$	$k = 3$
In-control	ARL	2.9	NA	NA	2.9	NA	NA	3.0	NA	NA
	Center	0.076	0.076	0.076	0.076	0.076	0.076	0.000	0.000	0.000
	LCL	−1.074	−2.225	−3.375	−1.074	−2.225	−3.375	−0.482	−0.964	−1.446
	UCL	1.226	2.376	3.527	1.226	2.376	3.527	0.482	0.964	1.446
	CI Length	2.301	4.601	6.902	2.301	4.601	6.902	0.964	1.927	2.891
	Coverage	0.617	1.000	1.000	0.617	1.000	1.000	0.604	1.000	1.000
One Inflated	ARL	4.467	NA	NA	4.467	NA	NA	4.533	423.833	NA
	Center	0.139	0.139	0.139	0.139	0.139	0.139	0.001	0.001	0.001
	LCL	−0.935	−2.009	−3.084	−0.935	−2.009	−3.084	−0.442	−0.884	−1.326
	UCL	1.214	2.288	3.362	1.214	2.288	3.362	0.443	0.885	1.328
	CI Length	2.149	4.298	6.446	2.149	4.298	6.446	0.885	1.769	2.654
	Coverage	0.725	1.000	1.000	0.725	1.000	1.000	0.723	0.999	1.000
Zero Inflated	ARL	2.100	NA	NA	2.100	NA	NA	2.400	NA	NA
	Center	0.016	0.016	0.016	0.016	0.016	0.016	0.000	0.000	0.000
	LCL	−1.160	−2.335	−3.510	−1.160	−2.335	−3.510	−0.495	−0.990	−1.484
	UCL	1.191	2.366	3.542	1.191	2.366	3.542	0.495	0.989	1.484
	CI Length	2.351	4.701	7.052	2.351	4.701	7.052	0.990	1.979	2.969
	Coverage	0.527	1.000	1.000	0.527	1.000	1.000	0.520	1.000	1.000

From Table 2 which is based on FPCA, we can see the same results as the ones in Table 1 based on PCA. The coverage probabilities by the GLMs with probit and logit link functions are slightly greater or equal to the coverage probabilities by the neural network regression model. The length of the confidence intervals (CIs) for the neural network regression model are much smaller than the the length of the confidence intervals (CIs) for the GLMs with probit and logit link functions but the ARLs of the neural network regression model are not smaller than the the ARLs for the GLMs with probit and logit link function. This result based on FPCA is different from the result based on PCA.

From Figures 5–7, in case of the in-control dispersion based on FPCA, we can observe that the residuals of a neural network regression model are much closer to zero than the residuals of the GLM with probit and logit model, which are the same as the figures based on PCA.

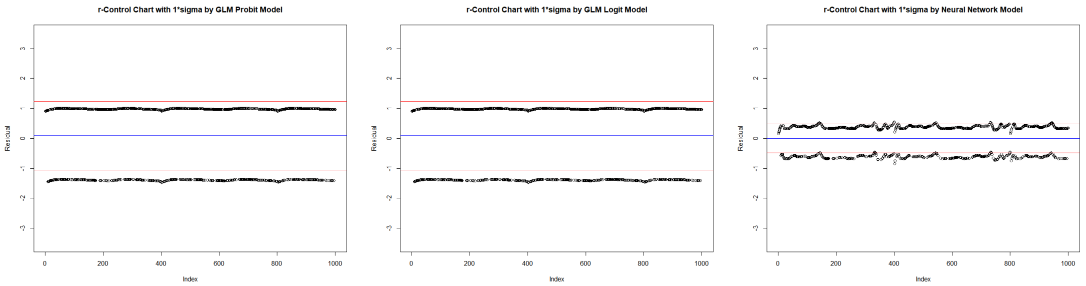


Figure 5. Based on FPCA, r control charts $(E(r_i) \pm \sqrt{Var(r_i)})$ for probit, logit and neural network in the in-control case with the number of simulations is 30 different replications of sample size of 1000.

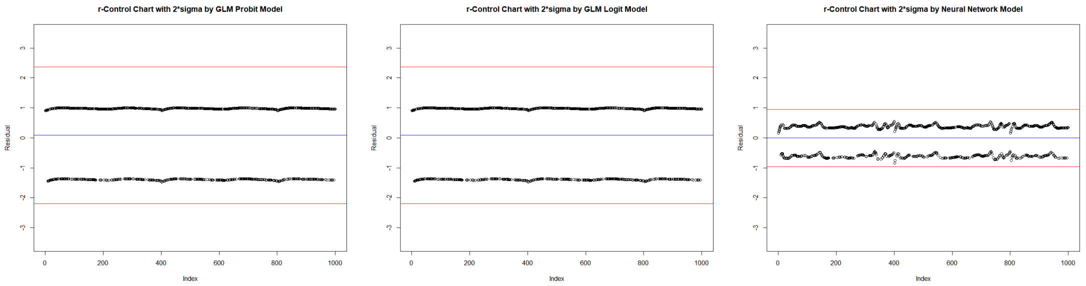


Figure 6. Based on FPCA, r control charts $(E(r_i) \pm 2\sqrt{Var(r_i)})$ for probit, logit and neural network in the in-control case with the number of simulations is 30 different replications of sample size of 1000.

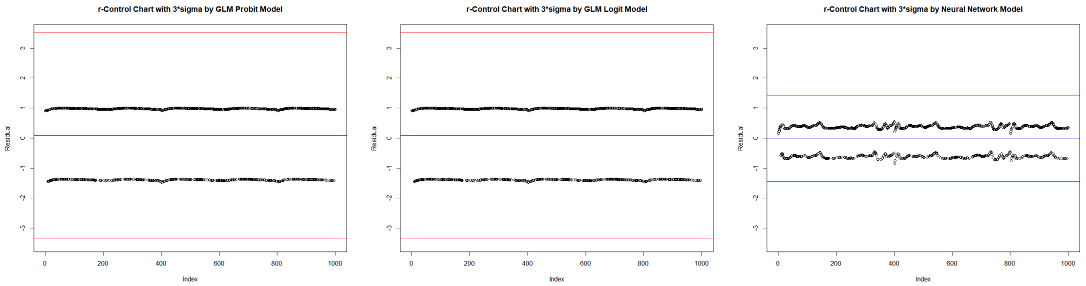


Figure 7. Based on FPCA, r control charts $(E(r_i) \pm 3\sqrt{Var(r_i)})$ for probit, logit and neural network in the in-control case with the number of simulations is 30 different replications of sample size of 1000.

From Tables 1 and 2, we can compare the results for the neural network regression model, the GLMs with probit and logit link functions based on PCA and FPCA. We found the that, for the in-control and the one-inflated cases, the ARLs based on PCA are smaller than the ARLs based on FPCA but, for zero-inflated case, the ARLs based on FPCA are smaller than the ARLs based on PCA. Another interesting result is that, in terms of the ARLs, the GLMs with probit and logit link functions based on FPCA is overall superior than the the neural network regression model. This result is an opposite result compared with the one based on PCA.

3.2. Real Data Analysis

Ref. [22] proposed R package “mlbench" which included Wisconsin breast cancer database named as Breast Cancer. We used Breast Cancer for the illustration of real data analysis in this paper.

The objective of the Wisconsin breast cancer database is to identify each of a number of benign or malignant classes which are binary data ('0' and '1'). Samples arrive periodically as Dr. Wolberg reports his clinical cases. The database is the chronological grouping of the data. A data frame with 699 observations on 11 variables, one being a character variable, nine being ordered or nominal, and 1 target class. In this paper, We used nine covariates and one target variable such as the Cl.thickness (Clump Thickness), Cell.size (Uniformity of Cell Size), Cell.shape (Uniformity of Cell Shape), Marg.adhesion (Marginal Adhesion), Epith.c.size (Single Epithelial Cell Size), Bare.nuclei (Bare Nuclei), Bl.cromatin (Bland Chromatin), Normal.nucleoli (Normal Nucleoli), Mitoses and Class which is the target binary variable (Y) ('0'=benigh and '1'=malignant).

Table 3 presents the Pearson correlation coefficients with the Breast Cancer real data. It shows that nine covariates have strong positive correlation coefficients It means that nine covariates in breast cancer real data positively correlated each other. Figure 8 also showed high correlated pairwise scatter plots of nine covariates in breast cancer real data.

Table 3. Pearson correlation coefficients of 9 covariates in Breast Cancer.

	Cl.thickness	**Cell.size**	**Cell.shape**	**Marg.adhesion**	**Epith.c.size**	**Bare.nuclei**	**Bl.cromatin**	**Normal.nucleoli**	**Mitoses**
Cl.thickness	1.000	0.642	0.653	0.488	0.524	0.593	0.554	0.534	0.355
Cell.size	0.642	1.000	0.907	0.707	0.754	0.692	0.756	0.719	0.465
Cell.shape	0.653	0.907	1.000	0.686	0.722	0.714	0.735	0.718	0.447
Marg.adhesion	0.488	0.707	0.686	1.000	0.595	0.671	0.669	0.603	0.425
Epith.c.size	0.524	0.754	0.722	0.595	1.000	0.586	0.618	0.629	0.481
Bare.nuclei	0.593	0.692	0.714	0.671	0.586	1.000	0.681	0.584	0.349
Bl.cromatin	0.554	0.756	0.735	0.669	0.618	0.681	1.000	0.666	0.354
Normal.nucleoli	0.534	0.719	0.718	0.603	0.629	0.584	0.666	1.000	0.437
Mitoses	0.355	0.465	0.447	0.425	0.481	0.349	0.354	0.437	1.000

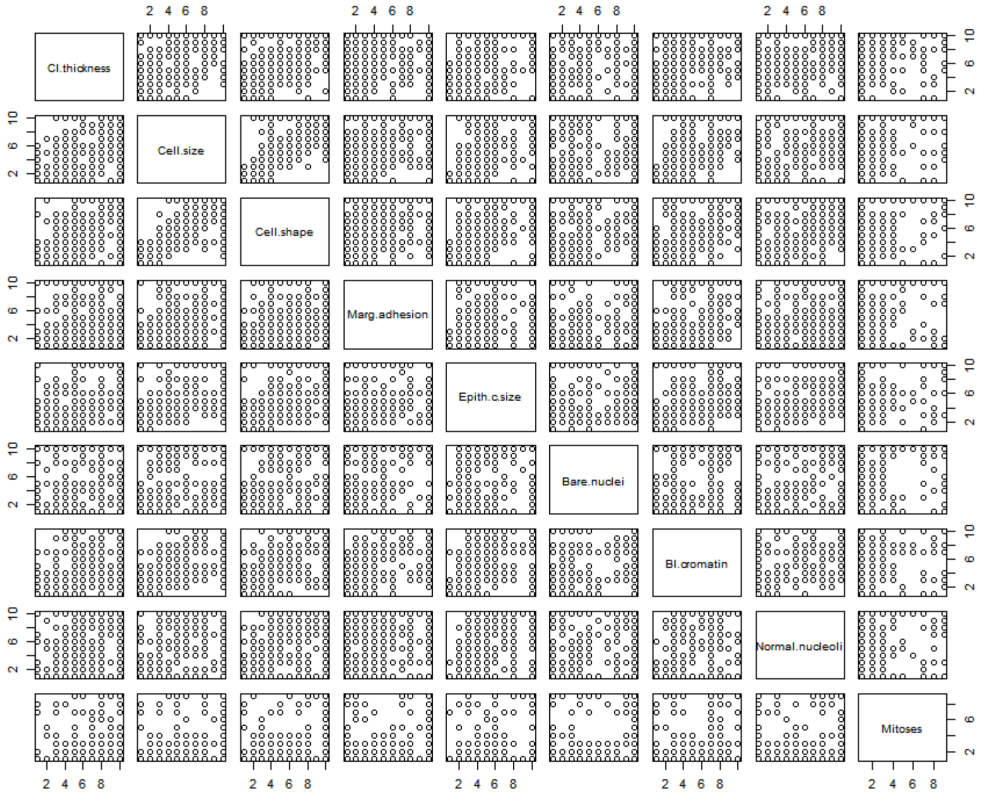


Figure 8. Pairwise scatter plots of nine covariates in Breast Cancer real data.

Table 4 showed the PCA summary with nine covariates in breast cancer real data. To avoid multicollinearity of the nine covariates, we used principal components for binary response data (Y=Class) via various *r*-charts based on GLM with probit, GLM with logit, and neural network models.

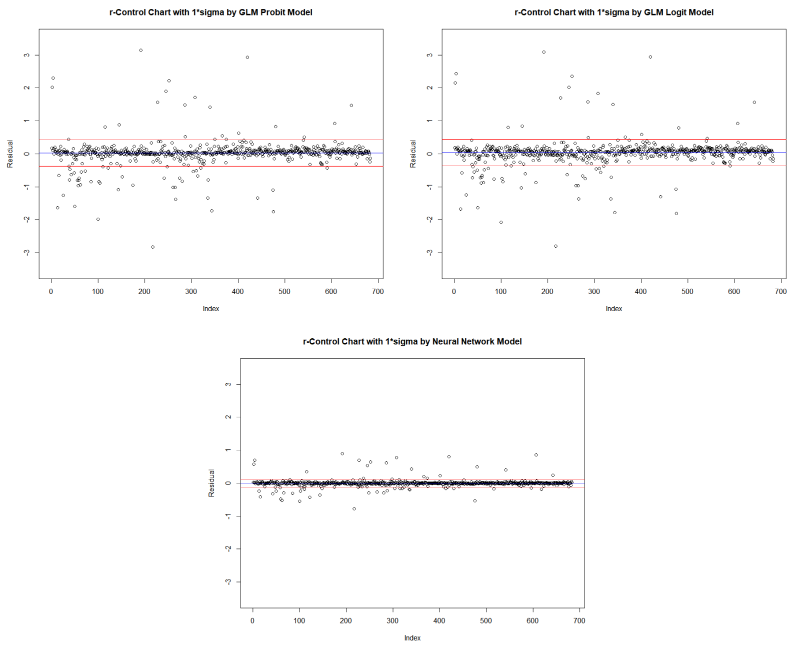
Table 4. PCA summary with nine covariates in Breast Cancer real data.

	Comp.1	**Comp.2**	**Comp.3**	**Comp.4**	**Comp.5**	**Comp.6**	**Comp.7**	**Comp.8**	**Comp.9**
Standard deviation	2.430	0.875	0.734	0.680	0.617	0.550	0.543	0.511	0.297
Proportion of Variance	0.656	0.085	0.060	0.051	0.042	0.034	0.033	0.029	0.010
Cumulative Proportion	0.656	0.741	0.801	0.853	0.895	0.928	0.961	0.990	1.000

Based on PCA, Table 5 shows that the *r*-chart based on neural network model has narrow control limits compared with the *r*-charts based on the GLM with probit and logit link function models. From Figures 9–11, we can observe that the residuals of a neural network regression model are much closer to zero than the residuals of the GLM with probit and logit model. With the narrow control limits of a neural network model, we can monitor the class of the patients with breast cancer by *r* control chart with important covaraites' information.

Table 5. Based on PCA, control limits for binary response data (Y=Class) via various *r*-charts based on GLM with probit, GLM with logit, and neural network models. Neural network model used single layer and 30 neurons.

	Probit			**Logit**			**Neural Network**		
$E(r_i) \pm w\sqrt{Var(r_i)}$	$k = 1$	$k = 2$	$k = 3$	$k = 1$	$k = 2$	$k = 3$	$k = 1$	$k = 2$	$k = 3$
Center	0.022	0.022	0.022	0.038	0.038	0.038	−0.002	−0.002	−0.002
LCL	−0.381	−0.785	−1.188	−0.368	−0.773	−1.178	−0.123	−0.244	−0.365
UCL	0.426	0.829	1.233	0.443	0.849	1.254	0.119	0.240	0.361
CL Length	0.807	1.614	2.421	0.811	1.622	2.433	0.242	0.485	0.727


r-Control Chart with 1*sigma by GLM Probit Model

r-Control Chart with 1*sigma by GLM Logit Model

r-Control Chart with 1*sigma by Neural Network Model

Figure 9. Based on PCA, *r* control charts ($E(r_i) \pm \sqrt{Var(r_i)}$) for probit, logit and neural network.

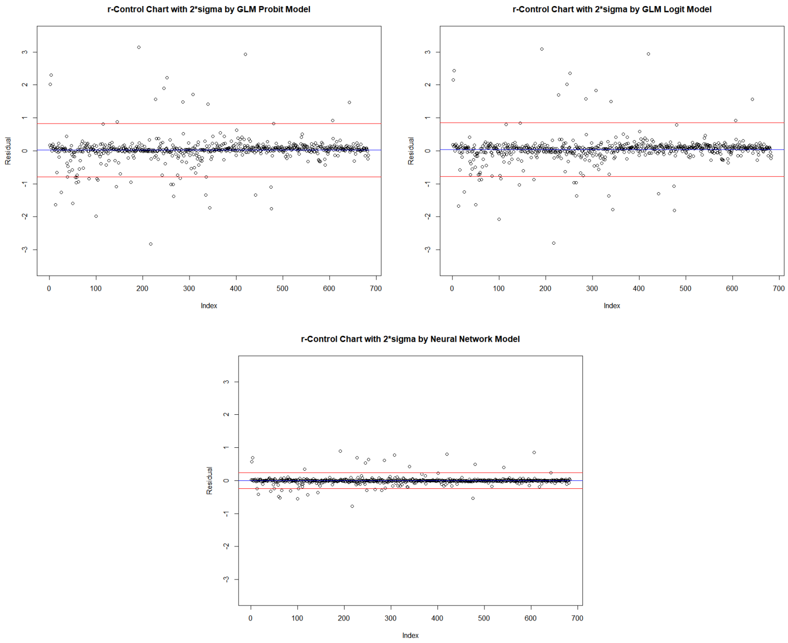


Figure 10. Based on PCA, r control charts $(E(r_i) \pm 2\sqrt{Var(r_i)})$ for probit, logit and neural network.

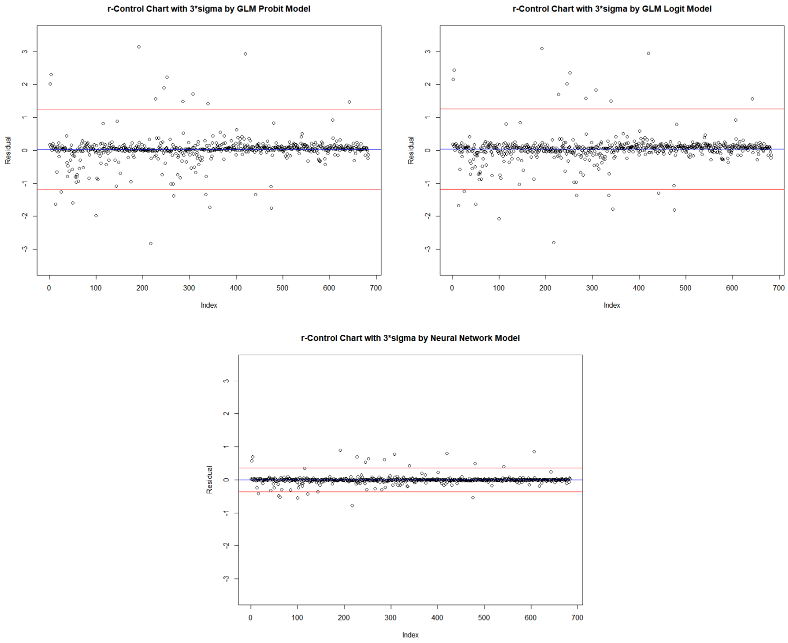


Figure 11. Based on PCA, r control charts $(E(r_i) \pm 3\sqrt{Var(r_i)})$ for probit, logit and neural network.

Figure 12 showed the plots of FPCA with nine covariates in Breast Cancer real data so that two main components explain the 94% proportion of variance by FPCA. Hence, we used two main components for the r-chart based regression models.

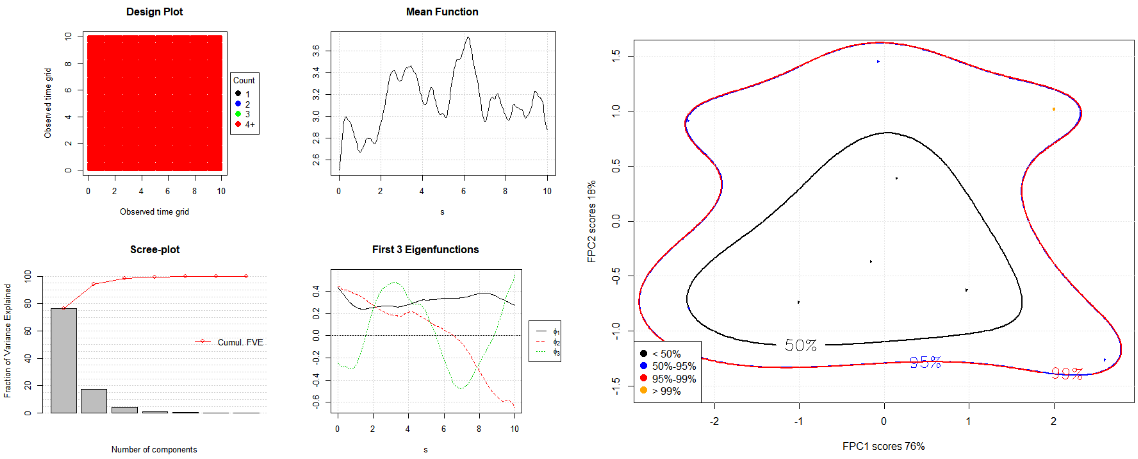


Figure 12. FPCA Plots with nine covariates in Breast Cancer real data.

Based on FPCA, Table 6 shows that the r-chart based on the neural network model has narrow control limits compared with the r-charts based on the GLM with probit and logit link function models. From Figures 13–15, we can observe that the residuals of a neural network regression model are much closer to zero than the residuals of the GLM with the probit and logit model which concur to the same result based on PCA. With the narrow control limits of a neural network model, we can monitor the class of the patients with breast cancer by r control chart with important covaraites' information.

Similar to the simulation data analysis, the r-chart based regression models based on PCA have the the narrower control limits than the r-chart based regression models based on FPCA.

Table 6. Based on FPCA, control limits for binary response data (Y=death) via various r-charts based on GLM with probit, GLM with logit, and neural network models. Neural network model used a single layer and 30 neurons.

	Probit			**Logit**			**Neural Network**		
$E(r_i) \pm w\sqrt{Var(r_i)}$	$k=1$	$k=2$	$k=3$	$k=1$	$k=2$	$k=3$	$k=1$	$k=2$	$k=3$
Center	0.095	0.095	0.095	0.095	0.095	0.095	−0.003	−0.003	−0.003
LCL	−1.034	−2.163	−3.293	−1.034	−2.163	−3.293	−0.470	−0.937	−1.403
UCL	1.225	2.354	3.484	1.225	2.354	3.483	0.464	0.930	1.397
CL Length	2.259	4.518	6.776	2.259	4.518	6.776	0.933	1.867	2.800

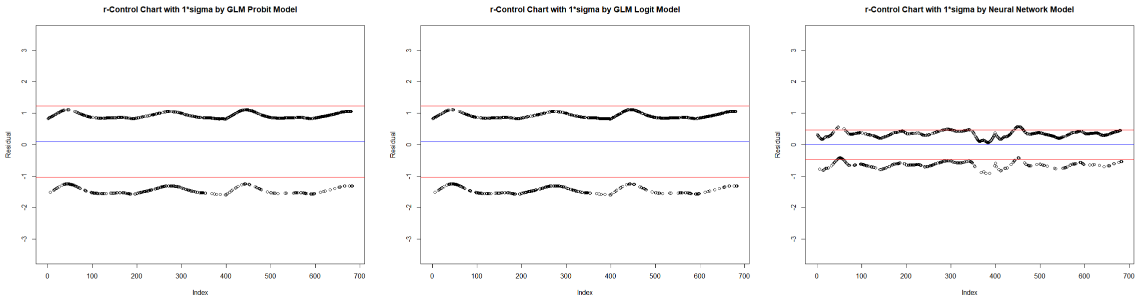


Figure 13. Based on FPCA, r control charts $(E(r_i) \pm \sqrt{Var(r_i)})$ for probit, logit and neural network.

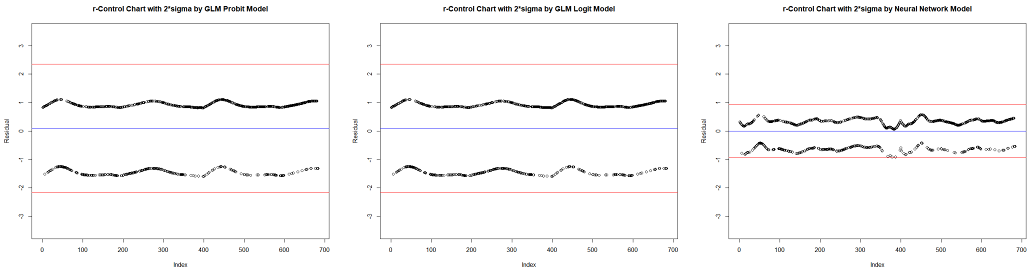


Figure 14. Based on FPCA, r control charts $(E(r_i) \pm 2\sqrt{Var(r_i)})$ for probit, logit and neural network.

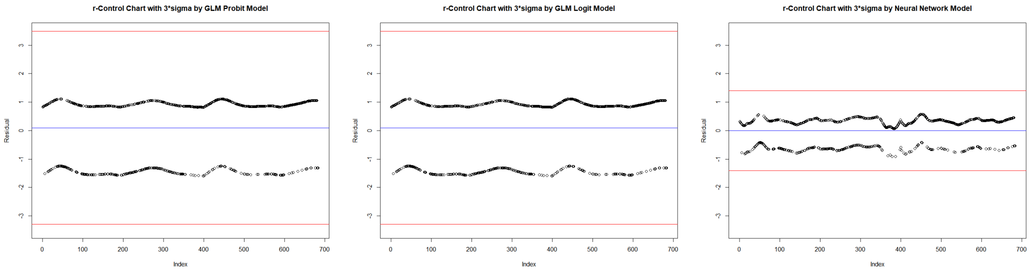


Figure 15. Based on FPCA, r control charts $(E(r_i) \pm 3\sqrt{Var(r_i)})$ for probit, logit and neural network.

4. Conclusions

In this research, we have presented the binary response regression model-based statistical process control r-charts for dispersed binary asymmetrical data with multicollinearity among input variables. We have demonstrated the proposed method in terms of the model flexibility and performance by running simulations for various circumstances: in-control, one inflated-, or zero inflated-dispersion data. With both simulated data and real data, our proposed method has shown a superiority of the performance. Furthermore, we compared PCA-based binary response regression model-based statistical control r-charts and FPCA-based binary response regression model-based statistical control r-charts with the GLM with probit and logit link function models and neural network model. In case of the dimension reduction by PCA, our proposed approach by a neural network is superior in handling cases of dispersed binary asymmetrical data with multicollinearity among explanatory variables. However, in case of the dimension reduction by FPCA, our proposed approach by neural network is superior in handling cases of dispersed binary asymmetrical data with multicollinearity among explanatory variables but it is not more efficient than the proposed method by the dimension reduction by PCA in this research.

The conclusion in this research is that for the high-dimensional correlated multivariate covariate data, the binary control chart by neural network model is a good statistical process control method and, for high-frequency functional multivariate data, the proposed GLM-based control charts by FPCA are good statistical process control methods. Hence, a cancer clinical study can be investigated by the proposed statistical process control.

Our future research will address the following topics. More general versions of binary asymmetric data will be considered with other machine learning models such as deep learning or multi-layer neural network model. Instead of the deviance residual, quantile residual can also be considered in the binary response regression control charts. Lastly, we need to consider other types of bases for constructing FPCA for further comparison studies with PCA-based models.

Author Contributions: J.-M.K. designed the model, analyzed the data and wrote the paper. N.W. formulated the conceptual framework, designed the model, obtained inference and wrote the paper. Y.L. formulated the conceptual framework, designed the model, obtained inference and wrote the paper. K.P. analyzed the data and provided editorial supports. All the authors cooperated to revise the paper. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by National Natural Science Foundation of China Grant (No. 71672182, No. U1604262 and No.U1904211).

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

References

1. Hotelling, H. *Multivariate Quality Control*; McGraw-Hill: New York, NY, USA, 1947.
2. Lowry, C. A.; Woodall, W. H.; Champ, C. W.; Rigdon, S. E. Multivariate exponentially weighted moving average control chart. *Technometrics* **1992**, *34*, 46–53. [CrossRef]
3. Crosier, R. B. Multivariate generalizations of cumulative sum qualitycontrol schemes. *Technometrics* **1988**, *30*, 291–303. [CrossRef]
4. Park, K.; Kim, J.-M.; Jung, D. GLM-based statistical control r-charts for dispersed count data with multicollinearity between input variables. *Qual. Reliab. Eng. Int.* **2018**, *34*, 1103–1109. [CrossRef]
5. Montgomery, D.C. *Statistical Quality Control*, 7th ed.; John Wiley and Sons Press: New York, NY, USA, 2012.
6. Qiu, P. *Introduction to Statistical Process Control*, 1st ed.; Chapman & Hall/CRC Texts in Statistical Science: Boca Raton, Florida, USA, 2013.
7. Bolker, B.M.; Brooks, M.E.; Clark, C.J.; Geange, S.W.; Poulsen, J.R.; Stevens, M.H.; White J.S. Generalized linear mixed models: a practical guide for ecology and evolution. *Trends Ecol Evol.* **2009**, *24*, 127–235. [CrossRef] [PubMed]
8. Myers, R. H., Montgomery, D.C.; Vining, G.G. *Generalized Linear Models, with Applications in Engineering and the Sciences*; John Wiley and Sons Press: New York, NY, USA, 2002.
9. Nelder, J.A.; Wendderburn, R.W.M. Generalized linear model. *J. R. Stat. Hence,c. A* **1972**, *35*, 370–384. [CrossRef]
10. Agatonovic-Kustrin, S.; Beresford, R. Basic concepts of artificial neural network (ANN) modeling and its application in pharmaceutical research. *J. Pharm. Biomed. Anal.* **2000**, *22*, 717–727. [CrossRef]
11. LeCun, Y.; Bengio, Y.; Hinton, G. Deep learning. *Nature* **2015**, 521, 436. [CrossRef] [PubMed]
12. Hassabis, D.; Kumaran, D.; Summerfield, C.; Botvinick, M. Neuroscience-inspired artificial intelligence. *Neuron* **2017**, 95, 245–258. [CrossRef] [PubMed]
13. Masood, I.; Hassan, A. Pattern Recognition for Bivariate Process Mean Shifts Using Feature-Based Artificial Neural Network. *Int. J. Adv. Manuf. Technol.* **2013**, *66*, 1201–1218. [CrossRef]
14. Addeh, A.; Khormali, A.; Golilarz, N. A. Control Chart Pattern Recognition Using RBF Neural Network with New Training Algorithm and Practical Features. *ISA Trans.* **2018**, *79*, 202–216. [CrossRef] [PubMed]
15. Zan, T.; Liu, Z.; Su, Z.; Wang, M.; Gao, X.; Chen, D. Statistical Process Control with Intelligence Based on the Deep Learning Model. *Appl. Sci.* **2020**, *10*, 308. [CrossRef]
16. Ripley, B.; Venables, W. *Feed-Forward Neural Networks and Multinomial Log-Linear Models*; R Package, mlbench; R Foundation for Statistical Computing: Vienna, Austria, 2016.
17. Skinner, K.R., Montgomery, D.C., Runger, G.C.: Process monitoring for multiple count data using generalized linear model-based control charts. *Int. J. Prod. Res.* **2003**, *41*, 1167–1180. [CrossRef]
18. Ramsay, J., and Silverman, B. *Functional Data Analysis*; Springer: New York, NY, USA, 2005.
19. Chen, Y.; Carroll, C.; Dai, X.; Fan, J.; Hadjipantelis, P. Z.; Han, K.; Ji, H.; Lin, S.-C.; Dubey, P.; Mueller, H.-G.; Wang, J.-L. *fdapace:Functional Data Analysis and Empirical Dynamics*; The R Project for Statistical Computing: Vienna, Austria, 2019.
20. Yao, F.; Múller, H.-G.; Wang, J.-L. Functional Data Analysis for Sparse Longitudinal Data. *J. Am. Assoc.* **2005**, *100*; 577–590. [CrossRef]
21. Liu, B, and Múller, H.-G. Estimating Derivatives for Samples of Sparsely Observed Functions, with Application to Online Auction Dynamics. *J. Am. Stat.* **2009**, *104*, 704–717. [CrossRef]
22. Leisch, F.; Dimitriadou, E. *Machine Learning Benchmark Problems*; R Package, mlbench; R Foundation for Statistical Computing: Vienna, Austria, 2015.