Determination of the Optimal Neural Network Transfer Function for Response Surface Methodology and Robust Design

Tuan-Ho Le ¹, Hyeonae Jang ²,* and Sangmun Shin ³,*

¹ Department of Electrical Engineering, Faculty of Engineering and Technology, Quy Nhon University, Binh Dinh 591417, Vietnam; tuanhole@qnu.edu.vn
² Department of Technology Management Engineering, Jeonju University, Jeonju 55069, Korea
³ Department of Industrial & Management Systems Engineering, Dong-A University, Busan 49315, Korea
* Correspondence: hajang@jj.ac.kr (H.J.); sshin@dau.ac.kr (S.S.)

Abstract: Response surface methodology (RSM) has been widely recognized as an essential estimation tool in many robust design studies investigating the second-order polynomial functional relationship between the responses of interest and their associated input variables. However, there is scope for improvement in the flexibility of estimation models and the accuracy of their results. Although many NN-based estimations and optimization approaches have been reported in the literature, a closed functional form is not readily available. To address this limitation, a maximum-likelihood estimation approach for an NN-based response function estimation (NRFE) is used to obtain the functional forms of the process mean and standard deviation. While the estimation results of most existing NN-based approaches depend primarily on their transfer functions, this approach often requires a screening procedure for various transfer functions. In this study, the proposed NRFE identifies a new screening procedure to obtain the best transfer function in an NN structure using a desirability function family while determining its associated weight parameters. A statistical simulation was performed to evaluate the efficiency of the proposed NRFE method. In this particular simulation, the proposed NRFE method provided significantly better results than conventional RSM. Finally, a numerical example is used for validating the proposed method.

Keywords: response surface methodology; neural network; desirability function; maximum-likelihood estimation; robust design

1. Introduction

Among the many quality engineering methodologies, robust design (RD) based on statistical design and analysis methods and optimization methods have contributed significantly to the improvement of product/process quality for more than 20 years. The main objective of RD is to identify the optimal factor settings that can minimize both variability and bias (namely, the deviation between the process mean and desired target value) in a process/product. To demonstrate this RD principle, Taguchi introduced a two-step model based on new design and analysis approaches, particularly orthogonal array and signal-to-noise ratios [1]. However, orthogonal arrays, statistical analysis, and signal-to-noise ratios are highly controversial [2–5]. Consequently, Vining and Myers [6] improved Taguchi’s model by proposing dual-response (DR) estimation and optimization approaches, in which the separate functions of the mean and variance responses are estimated using response surface methodology (RSM) based on the least-squares method (LSM). Their proposed optimization approach can calculate robust factor settings by minimizing process variability while maintaining the process mean at the target value. This results in the primary procedure of RD being identified in three sequential steps: design...
of experiment (DoE), response function estimation, and optimization. However, Lin and Tu [7] pointed out that process bias and variability are not considered simultaneously in the DR model and proposed a mean square error (MSE) model as an alternative approach providing more model flexibility when compared with that provided by the DR approach. These DR and MSE models were further extended by Del Castillo and Montgomery [8], Copeland and Nelson [9], Cho et al. [10], Ames et al. [11], Borror [12], Kim and Lin [13], Koksoy and Doganaksoy [14], Ding et al. [15], Shin and Cho [16–18], Robinson et al. [19], Fogliatto [20], Goethals and Cho [21], Truong and Shin [22,23], Nha et al. [24], Baba et al. [25], and Yankoğlu et al. [26].

In the response function estimation step of the RD procedure, RSM is used extensively to obtain the estimated function associated with an output response and its associated input factors based on the DoE results. RSM is applied extensively in real-world industrial situations, mainly where several input factors potentially affect a product or process’s performance measures or quality characteristics [27]. The relationship between several input factors and one or more associated output responses can be estimated using RSM. Conventionally, LSM can estimate the unknown coefficients of a regression model with many specific error assumptions, such as the errors in the output responses should be followed by a normal distribution (i.e., zero mean and constant variance) identically and independently. However, these assumptions are often violated in many practical problems. Several alternative approaches, such as transformation, weighted least squares (WLS), maximum-likelihood estimation (MLE), Bayesian, and inverse problems, can be used to address this problem. For example, the WLS method has been applied to RSM to estimate the model parameters for unbalanced data [28,29], and Lee and Park [30] and Cho et al. [31] have integrated an MLE method and an expectation–maximization algorithm for unknown parameter estimations using incomplete data. In addition, Goegebeur et al. [32] and Chen and Ye [33,34] proposed a Bayesian approach to estimate the coefficients of response functions where the variances follow a log-normal distribution. Truong and Shin [22,23] proposed a new estimation method that used an inverse problem approach based on Bayesian perspectives. Most of the existing estimation methods in the RSM and RD literature attempt to generate functional forms in input factors for output responses (namely, the process mean, process variance, and quality characteristic). However, relaxing the error assumptions and increasing the estimation precision would result in further improvements.

Artificial neural networks (ANNs) or neural networks (NNs) with nonlinear mapping structures using the human brain function have been widely used as powerful tools in data classification, forecasting, clustering, function approximation, and optimization in recent decades. Irie and Miyake [35], Hornik et al. [36], Cybenko [37], and Funahashi [38] recently addressed the question of approximation using feed-forward NNs and proved that NNs are universal functional approximators with the desired accuracy. ANNs belong to a class of self-adaptive and data-driven techniques, and therefore the undefined relationships between input factors and outputs of a process/product can be ascertained effectively. ANNs can provide a linear or nonlinear relationship between inputs and several outputs without any assumptions based on generalizing capacities of the activation function. Consequently, the functional relationships between input factors and their related output responses in an RD procedure can be estimated without any specific assumptions effectively. Most of the literature on NN function approximation has concentrated on their structure, hidden layers, hidden neurons, and training algorithms. On the contrary, less of them focus on the transfer or activation function, which can strongly affect the complexity and performance of NNs. In addition, most of the transfer functions reported in the literature can provide only a particular configuration to transfer the information from inputs to the associated outputs of a neuron.

The desirability function (DF) approach is one of the most commonly utilized techniques for multiple response optimization problems, in which each quality characteristic is converted into an individual DF that takes a value ranging from 0 to 1. The DF concept
was introduced by Harrington [39] for the simultaneous optimization of multiple response problems in the industry field. This approach can be used to determine those optimal factors settings of input variables that result in the most desirable output values, while none of these response values are beyond the specified boundaries. Derringer and Suich [40] implemented a further extension of Harrington’s DF. Depending on whether the response will minimize, maximize, or specify the desired target value, there are three associated DFs: STB (smaller is better), NTB (nominal is better), and LTB (larger is better). Based on the weight parameters, the DF family may provide a flexible transfer function configuration.

Rowlands et al. [41] integrated an NN approach into RD and used the NN to perform the DoE. Su and Hsieh [42], Cook et al. [43], Chow et al. [44], Chang [45], and Chang and Chen [46] combined an NN approach as an estimation method with a genetic algorithm to determine the optimal process parameter settings or the optimal costs in both static and dynamic characteristics in Taguchi’s formula without considering the process mean and variance functions. Recently, Arungpadang and Kim [47] presented a feed-forward NN structure based on RSM in the RD concept.

In the robust design (RD) concept, the process mean and standard deviation of an output response in experiment results can be estimated as a function by many different statistical approaches (i.e., LSM, MLE, and so on). These processes mean and standard deviation functions can be optimized simultaneously based on their desired targets using existing optimization methods. The significant objective of this paper is to propose an alternative estimation method to DR functions (i.e., mean and standard deviation functions) by integrating NN approaches for RSM and RD modeling and then to compare this method to the conventional LSM-based estimation method. First, a feed-forward NN is integrated into the RD procedure as a new DR estimation approach to estimate the process mean and standard deviation response functions. In order to develop the new NN-based estimation method, many different types of transfer functions that can affect the quality of estimation results should be considered.

For this reason, DFs can be utilized as a transfer function family because DFs can represent all three different types (i.e., L-type, S-type, and N-type) of quality characteristics. Among these transfer functions based on DFs, the best transfer function for experimental data using the identified learning and validation approaches is then investigated in this proposed approach. This proposed best transfer function determination procedure for a given NN structure based on DF is derived for the proposed DR estimation method. DFs with their associated weight parameters can provide a variety of transfer function families. In addition, the process mean is often kept at the target value, and the process standard deviation is minimized to the least possible in the RD methodology. Therefore, the “nominal is best” and “smaller is better” DFs are proposed as new NN transfer functions. Next, the best transfer functions for the process mean response function and standard deviation response function are determined based on the optimal weight parameters of the proposed transfer functions using MLE. The associated confidence intervals for the optimal weight parameters are also determined. The results of a case study are presented to support the view that better solutions are obtained from the proposed feed-forward NN-based DR estimation method compared with those from the conventional LSM method and the feed-forward NN estimation approach by using the conventional log-sigmoid transfer function. An overview of the proposed NN-based DR function estimation method is presented in Figure 1.
Figure 1. Overview of the proposed NN-based DR function estimation method.

2. RD Estimation Method Based on RSM

RSM, developed by Box and Wilson [48], consists of mathematical and statistical techniques based on the fittings of empirical models obtained from an experimental design. RSM can estimate the functional empirical relationship between the input factors and their related output responses. The basic theory, estimation methods, and analytical techniques of RSM can be found in the study by Myers and Montgomery [27]. Myers [49] and Khuri and Mukhopadhyay [50] provide insights into the various developmental stages and future directions of RSM. In general, the output response $y$ can be identified as a function of the input factor $\mathbf{x}$ as follows:

$$ y = \mathbf{x}\beta + \epsilon, \quad (1) $$

where $\mathbf{x}$, $\beta$, and $\epsilon$ denote the vector of control factors, column vector of model parameters, and random error. The estimated second-order models for the process mean and standard deviation functions are represented as follows:
\[ \hat{\beta}_{LSM}(x) = \hat{\beta}_0 + \sum_{i=1}^{p} \hat{\beta}_i x_i + \sum_{i=1}^{p} \hat{\beta}_i x_i^2 + \sum_{i<j}^{p} \hat{\delta}_{ij} x_i x_j, \]

\[ \hat{\delta}_{LSM}(x) = \hat{\delta}_0 + \sum_{i=1}^{p} \hat{\delta}_i x_i + \sum_{i<j}^{p} \hat{\delta}_{ij} x_i x_j, \]

where \( \hat{\beta} \) and \( \hat{\delta} \) are the estimators of the unknown parameters in the process mean and standard deviation functions, respectively. These coefficients are estimated using the conventional LSM, as follows:

\[ \hat{\beta} = (X^T X)^{-1} X^T \overline{y}_{obs} \text{ and } \hat{\delta} = (X^T X)^{-1} X^T s_{obs}, \]

where \( \overline{y}_{obs} \), \( s_{obs} \), and \( X \) are the observation average, the standard deviation of replicated responses, and design matrix, respectively.

### 3. Proposed Feed-Forward NN-Based DR Estimation Method

#### 3.1. Feed-Forward NN Structure

NNs may include a broad class of flexible nonlinear regression, discriminant and data reduction models, and nonlinear dynamical systems [51]. An NN consists of simple, highly interconnected computational units called artificial neurons. Zainuddin and Pauline [52] demonstrated that feed-forward NNs from the input to the output are the most popular NNs for use in function approximation. In addition, Cybenko [37], Hornik et al. [36], Funahashi [38], and Hartman et al. [53] demonstrated that a feed-forward NN including one hidden layer could approximate a continuous, arbitrary nonlinear, and multidimensional function. Both processes mean and standard deviation can be two output response characteristics of interest in the RD methodology. The proposed feed-forward NN-based method for estimating the functional relationship associated with input factors and output responses in the RD is depicted in Figure 2.

![Figure 2. Proposed feed-forward NN-based RD estimation method.](image)

A vector \( x \) comprising \( k \) control factors \( x_1, \ldots, x_i, \ldots, x_k \), named as the input layer, is the input to the hidden neuron in the hidden layer. The summation of \( k \) inputs of neurons with identified weights (i.e., coefficients) and bias \( a \) is transformed using the transfer function \( f \) to generate the associated output of the hidden neuron. The outputs of the hidden neuron are used as inputs to the neuron in the output layer. Assume that there are \( h \) hidden nodes \( 1, \ldots, j, \ldots, h \), \( w_{ij} \), \( v_j \), \( a_j \), and \( b \) denotes the weight connecting input factor \( x_i \) to hidden node \( j \), the weight connecting hidden node \( j \) to the final output, the bias at the hidden node \( j \), and the bias at the output, respectively. Then, the general form of the response function obtained using the proposed NN-based estimation method is as follows:

\[ \hat{y}_{NN} = g\{\sum_{j=1}^{h} v_j y_j^{hid} + b_{out}\} = g\{\sum_{i=1}^{h} v_j [f(\sum_{i=1}^{k} w_{ij} x_i + a_j)] + b_{out}\}. \]
3.2. Back-Propagation Learning Algorithm

Among the learning algorithms reported in the literature for performing NN estimation, the back-propagation algorithm is commonly used for training a network owing to its simplicity and applicability [54]. This algorithm comprises two sequential steps: first, the weights of the NN are initialized randomly, and then, the output of the NN is computed and compared with the actual output. The error at the output layer, that is, the network error, is calculated by comparing the actual output $\hat{y}_{out}$ to the desired value $y_{out}$ to iteratively adjust the weights of the hidden and output layers in accordance with the following function:

$$ E = \frac{1}{2} \sum (y_{out} - \hat{y}_{out})^2. \quad (6) $$

Epoch is a hyperparameter that defines the number of times the entire training vectors are used to update the weights. During the training process, the error is optimally minimized. The iterative step of the gradient descent algorithm changes weights $w_j$ based on the following relationship:

$$ w_j \rightarrow w_j + \Delta w_j, \text{ where } \Delta w_j = -\eta \frac{\partial E(w)}{\partial w_j}. \quad (7) $$

The parameter $\eta (> 0)$ is called the learning rate. There are numerous variations in the basic algorithm based on other optimization techniques (namely, conjugate gradient and Newton methods).

The Marquardt-Levenberg algorithm (trainlm) approximates Newton’s method, which is designed to approach the second-order training speed without applying the Hessian matrix.

The resilient back-propagation training algorithm (trainrp) is a local adaptive learning technique, which performs supervised batch learning in a feed-forward NN. Trainrp removes the negative impact of the dimension of the partial derivative. Therefore, only the sign of the derivative implies the direction of a weight update process.

4. Proposed Transfer Functions Based on DFs

4.1. Integration of DFs into Transfer Functions

DF $d(y)$ scales the possible response ($y$) value to the range of $[0, 1]$, where $d(y) = 0$ and $d(y) = 1$ represent completely undesirable and desirable values, respectively. Among the three DFs proposed by Derringer and Suich [40] for three situations of a particular response, the “nominal is best” and “smaller is better” DFs are utilized as new transfer functions in the hidden NN layers in order to estimate the process mean and standard deviation functions. Assume that there are $q$ runs, and hence, the summation at each hidden neuron represents vector $y_j$ consisting of $\{y_1, ..., y_p, ..., y_q\}$ values, where the input of each hidden neuron can then be calculated as $y_p(x) = \sum_{i=1}^{q} x_i w_{ij} + a_j$ at each run order. Therefore, the associated DF output can be represented as $d_j = \{d_1, ..., d_p, ..., d_q\}$. When a response belongs to the “nominal is best” case, the individual DF is defined as

$$ d_p(y_p) = \begin{cases} 
0 & \text{if } y_p(x) < L_p \\
\left(\frac{y_p(x) - L_p}{T_p - L_p}\right)^s & \text{if } L_p \leq y_p(x) < T_p \\
\left(\frac{y_p(x) - U_p}{T_p - U_p}\right)^t & \text{if } T_p \leq y_p(x) \leq U_p \\
0 & \text{if } y_p(x) > U_p 
\end{cases} \quad (8) $$

When a response needs to be minimized, the individual DF is
\[ d_p(y_p) = \begin{cases} 
1 & \text{if } y_p(x) < L_p \\
\left(\frac{y_p(x) - U_p}{L_p - U_p}\right)^r & \text{if } L_p \leq y_p(x) \leq U_p \\
0 & \text{if } y_p(x) > U_p 
\end{cases} \quad (9)\]

where \( L_p, T_p, \text{ and } U_p \) denote the lower specification, target, and upper specification values, respectively. Superscripts \( r \) and \( s \) in Equation (8) and superscript \( t \) in Equation (9) represent weight parameters of the “nominal is best” and the “smaller is better” DFs, respectively. When the values of \( s, t, \text{ and } r \) are set to 1, the DFs become linear. For \( s < 1, t < 1, \text{ and } r < 1 \) the functions are convex, and for \( s > 1, t > 1, \text{ and } r > 1 \) the functions become concave. The integration of the “nominal is best” and “smaller is better” DFs as new transfer functions in the hidden layers to estimate the process mean and standard deviation is depicted in Figure 3.

**Figure 3.** Integration of DFs as transfer functions in hidden layers: (a) “nominal is best” transfer function and (b) “smaller is better” transfer function.

From Equation (5), the proposed estimated process mean and standard deviation functions (namely, \( \hat{\mu}_{pro}(x) \) and \( \hat{\sigma}_{pro}(x) \)) are defined, respectively, as

\[
\hat{\mu}_{pro}(x) = \sum_{j=1}^{h_{mean\_pro}} \left( \nu_{j, mean\_pro} \left( d_{mean}^{\_pro} \left( \sum_{i=1}^{k} w_{ij}^{\_pro} x_i + a_{j, mean\_pro} \right) \right) \right) + b_{mean}^{\_pro} \quad (10)
\]

\[
\hat{\sigma}_{pro}(x) = \sum_{j=1}^{h_{std\_pro}} \left( \nu_{j, std\_pro} \left( d_{std}^{\_pro} \left( \sum_{i=1}^{k} w_{ij}^{\_pro} x_i + a_{j, std\_pro} \right) \right) \right) + b_{std}^{\_pro}, \quad (11)
\]

where \( h_{mean\_pro}, h_{std\_pro}, a_{j, mean\_pro}, a_{j, std\_pro}, d_{j, mean\_pro}, d_{j, std\_pro}, b_{mean\_pro}, b_{std\_pro}, \nu_{j, mean\_pro}, \nu_{j, std\_pro}, w_{ij}^{\_pro}, w_{ij}^{\_std\_pro}, d_{mean}^{\_pro}, \text{ and } d_{std}^{\_pro} \) denote the number of hidden neurons, bias at the hidden node \( j \), bias at the output neuron, weights connecting hidden node \( j \) to the output responses, weights connecting input factors \( x_i \) to hidden node \( j \) of the proposed feed-forward NN, and the optimal DFs for the process mean and standard deviation functions, respectively. In addition, the log-sigmoid and linear functions are usually used as conventional transfer functions in hidden and output layers, respectively. The conventional log-sigmoid function \( f \) and linear function \( g \) with input layer \( x \) are given as

\[
f = \frac{1}{1 + \exp(-x)} \quad \text{and} \quad g = x. \quad (12)
\]

Similarly, the estimated process mean and standard deviation functions are given, respectively, as follows:

\[
\hat{\mu}_{NN}(x) = \sum_{j=1}^{h_{mean}} \left( \nu_{j, mean} \left( \frac{1}{1 + \exp(-\left(\sum_{i=1}^{k} w_{ij}^{\_mean} x_i + a_{j, mean} \right))} \right) \right) + b_{mean}^{\_NN} \quad (13)
\]

\[
\hat{\sigma}_{NN}(x) = \sum_{j=1}^{h_{std}} \left( \nu_{j, std} \left( \frac{1}{1 + \exp(-\left(\sum_{i=1}^{k} w_{ij}^{\_std} x_i + a_{j, std} \right))} \right) \right) + b_{std}^{\_NN}, \quad (14)
\]
where \( h_{\text{mean}}, h_{\text{std}}, a^\text{mean}_j, a^\text{std}_j, h^\text{NN}_j, v^\text{mean}_j, v^\text{std}_j, \) \( w^\text{mean}_{ij} \) and \( w^\text{std}_{ij} \) denote the number of hidden neurons, bias at hidden node \( j \), bias at the output neuron, weight connecting hidden node \( j \) to the final output, and the weight connecting input factor \( x_i \) to hidden node \( j \) of the feed-forward NNs for the process mean and standard deviation functions, respectively.

### 4.2. Estimation of Weight Parameters for NN Transfer Functions

The DF can provide a flexible NN transfer function family based on various values of the weight parameter. Among the likelihood-based estimation methods, the MLE method was proposed to estimate the optimal weight parameters for NN transfer functions. On the premise that DF values \( d_j \) follow a normal distribution \((\mu, \sigma^2)\), the distribution of summation \( y_j \) can be inferred.

#### 4.2.1. Estimation of Parameter \( s \)

Using Equation (8), the parametric family in this case can be given as \((\mu, \sigma^2, s)\). Based on the principles of MLE, parameters \((\mu, \sigma^2, s)\) can be identified when log-likelihood functions \( \ell(d_j, \mu, \sigma^2, s) \) or \( \ell(d_j, s) \) are the highest. The log-likelihood function in relation to summation \( y \) is obtained by multiplying the normal probability density function by the Jacobian of the DF, \( J(d_j, s) \):

\[
\ell(d_j, s) = \log \left[ \prod_{p=1}^{q} \frac{1}{\sigma \sqrt{2\pi}} e^{-\frac{(d_p - y)^2}{2\sigma^2}} \times J(d_j, s) \right],
\]

(15)

where

\[
J(d_j, s) = \prod_{p=1}^{q} \left( \frac{d_p}{\sigma^2} \right)^q = \left( \frac{s}{\sigma^2} \right)^q \prod_{p=1}^{q} \frac{y_p - L_p}{\sigma^2}.
\]

(16)

Therefore, the log-likelihood function is

\[
\ell(d_j, s) = -\frac{q}{2} \log \sigma^2 - \frac{q}{2} \log 2\pi - \frac{\sum_{p=1}^{q} (d_p - \mu)^2}{2\sigma^2} + q \log \left( \frac{s}{\sigma^2} \right) + \sum_{p=1}^{q} \log \left( \frac{y_p - L_p}{\sigma^2} \right)
\]

(17)

For a fixed value \( s \), obtaining the derivative of \( \ell(d_j, s) \) with respect to \( \sigma^2 \), setting this derivative function equal to zero, and solving this function by \( \sigma^2 \) yields

\[
\sigma^2 = \frac{\sum_{p=1}^{q} (d_p - \mu)^2}{q}.
\]

(18)

Further, obtaining the derivative of \( \ell(d_j, s) \) with respect to \( \mu \), assuming it equal to zero, and solving for \( \mu \) yields

\[
\mu = \frac{1}{q} \sum_{p=1}^{q} d_p.
\]

(19)

Accordingly, the log-likelihood function can be represented as follows:

\[
\ell(d_j, s) = -\frac{q}{2} \log 2\pi - \frac{q}{2} \log 2\pi - \frac{\sum_{p=1}^{q} (d_p - \mu)^2}{2\sigma^2} + q \log \left( \frac{s}{\sigma^2} \right) + \sum_{p=1}^{q} \log \left( \frac{y_p - L_p}{\sigma^2} \right)
\]

(20)

Optimal value \( s \) is specified using the statistical simulation method.

#### 4.2.2. Estimation of Parameter \( t \)

Similarly, from Equation (8), the log-likelihood function for parameter \( t \) is

\[
\ell(d_j, t) = -\frac{q}{2} \log 2\pi - \frac{q}{2} \log 2\pi - \frac{\sum_{p=1}^{q} (d_p - \mu)^2}{2\sigma^2} + q \log \left( \frac{t}{\sigma^2} \right) + (t - 1) \sum_{p=1}^{q} \log \left( \frac{y_p - U_p}{\sigma^2} \right)
\]

(21)
The profile log-likelihood function used to estimate parameter $t$ can be defined as follows:

$$
\ell(d,t) = -\frac{q}{2} \log 2\pi - \frac{q}{2} \log \frac{\sum_{p=1}^{q}(d_p - \mu)^2}{2\sigma^2} - \frac{q}{2} + q \log \left( \frac{t}{t_p - U_p} \right) + (t - 1) \sum_{p=1}^{q} \log \left( \frac{y_p - U_p}{t_p - U_p} \right) \tag{22}
$$

Optimal value $\hat{t}$ is also identified using the statistical simulation method.

4.2.3. Estimation of Parameter $r$

Using Equation (9), the log-likelihood function for parameter $r$ is

$$
\ell(d,r) = -\frac{q}{2} \log 2\pi - \frac{q}{2} \log 2\pi - \frac{\sum_{p=1}^{q}(d_p - \mu)^2}{2\sigma^2} + q \log \left( \frac{r}{L_p - U_p} \right) + (r - 1) \sum_{p=1}^{q} \log \left( \frac{y_p - U_p}{L_p - U_p} \right) \tag{23}
$$

The profile log-likelihood function can then be defined as follows:

$$
\ell(d,r) = -\frac{q}{2} \log 2\pi - \frac{q}{2} \log 2\pi - \frac{\sum_{p=1}^{q}(d_p - \mu)^2}{2\sigma^2} - \frac{q}{2} + q \log \left( \frac{r}{L_p - U_p} \right) + (r - 1) \sum_{p=1}^{q} \log \left( \frac{y_p - U_p}{L_p - U_p} \right) \tag{24}
$$

Optimal value $\hat{r}$ is also determined using the statistical simulation method.

4.3. Confidence Intervals for Weight Parameters

The asymptotic distribution of a maximum likelihood estimator is proposed to identify confidence intervals for weight parameters because the distribution of parameter $\theta$ estimated using MLE is asymptotically normal [55]. An approximate $(1 - \alpha)100\%$ confidence interval for a parameter $\theta$ can be given as follows:

$$
\hat{\theta}_i \pm z^* \hat{SE}(\hat{\theta}_i), \tag{25}
$$

where $z^*$ refers to the $(1 - \alpha)\frac{1}{2}$th percentile of the standard normal distribution and $\hat{SE}(\hat{\theta}_i)$ represents the estimated value of the standard error of $\hat{\theta}_i$. Based on the estimates of the parameters $\theta$, the standard errors of the parameters can be defined by the square root of the diagonal elements of the estimated covariance matrix. First, the distribution of $\theta$ is stated, followed by the derivation of the approximate asymptotic distribution of parameters $\hat{s}$, $\hat{t}$, or $\hat{r}$.

The first derivative of log-likelihood function $\ell(\theta|d_j)$ can be given as

$$
V = \frac{\partial\ell(\theta|d_j)}{\partial \theta}. \tag{26}
$$

The observed Fisher’s total information matrix is given as

$$
J(\theta) = -\frac{\partial V}{\partial \theta} = -\frac{\partial^2 \ell(\theta|d_j)}{\partial \theta^2}. \tag{27}
$$

Therefore, Fisher’s total information matrix $J$ can then be defined as follows:

$$
J(\theta) = E[J(\theta)]. \tag{28}
$$

The distribution of $\sqrt{q}(\hat{\theta} - \theta)$ as $q \to \infty$ converges to a normal distribution. In particular,

$$
\frac{J(\theta)^{1/2}}{q} \sqrt{q}(\hat{\theta} - \theta) \xrightarrow{dist} N(0,1) \tag{29}
$$

where $J(\theta)$ is Fisher’s total information matrix [55]. $J(\theta)$ is typically not known, but by substituting a $\frac{1}{q}$ consistent estimator, $\frac{\hat{J}(\theta)}{q}$ for $\frac{J(\theta)}{q}$ in Equation (29), the same result can be obtained. Accordingly,
\[ J(\theta)^{-1}(\theta - \theta) \overset{\text{dist}}{\rightarrow} N(0, I) \] and \( \theta \sim N(\theta, J(\theta)^{-1}) \). (30)

4.3.1. Confidence Intervals for Parameter \( s \)

The general parameter \( \theta \) can be determined as \( \theta = (\mu \, \sigma^2)^T \). From Equation (27), \( -\hat{J}(\theta) \) can be given as

\[
-\hat{J}(\theta) = \begin{pmatrix}
\frac{\partial^2 \ell(\theta | d_j)}{\partial \mu^2} & \frac{\partial^2 \ell(\theta | d_j)}{\partial \mu \partial \sigma^2} & \frac{\partial^2 \ell(\theta | d_j)}{\partial \mu \partial \sigma^2} \\
\frac{\partial^2 \ell(\theta | d_j)}{\partial \sigma^2 \partial \mu} & \frac{\partial^2 \ell(\theta | d_j)}{\partial \sigma^2 \partial \sigma^2} & \frac{\partial^2 \ell(\theta | d_j)}{\partial \sigma^2 \partial \sigma^2} \\
\frac{\partial^2 \ell(\theta | d_j)}{\partial \sigma^2 \partial \mu} & \frac{\partial^2 \ell(\theta | d_j)}{\partial \sigma^2 \partial \sigma^2} & \frac{\partial^2 \ell(\theta | d_j)}{\partial \sigma^2 \partial \sigma^2}
\end{pmatrix}.
\] (31)

From the log-likelihood function in Equation (17), \( -\hat{J}(\theta) \) can be calculated as follows (see Appendix A.1):

\[
-\hat{J}(\theta) = \begin{pmatrix}
-\frac{q^2}{\sigma^2} & -\sum_{p=1}^{q}(d_p - \mu)^2 \\
\sum_{p=1}^{q}(d_p - \mu) & 2q \left( -\frac{1}{\sigma^2} \right) & \frac{1}{(\sigma^2)^2} - \frac{1}{\sigma^2} \\
0 & 0 & 0
\end{pmatrix} = \begin{pmatrix}
\hat{J}(\theta)_{11} & \hat{J}(\theta)_{12} & \hat{J}(\theta)_{13} \\
\hat{J}(\theta)_{21} & \hat{J}(\theta)_{22} & \hat{J}(\theta)_{23} \\
\hat{J}(\theta)_{31} & \hat{J}(\theta)_{32} & \hat{J}(\theta)_{33}
\end{pmatrix}
\] (32)

The square root of the diagonal elements of \( \hat{J}(\theta)^{-1} \) represents an estimate of the standard error for each estimator. An estimator for the standard error of \( \hat{s} \) is of particular interest and can be found using the inverses of the partitioned matrices. First, the partition \( -\hat{J}(\theta) \) is as follows:

\[
-\hat{J}(\theta) = \begin{pmatrix}
\hat{J}(\theta)_{11} & \hat{J}(\theta)_{12} \\
\hat{J}(\theta)_{21} & \hat{J}(\theta)_{22}
\end{pmatrix}.
\] (33)

Next, one estimator for \( \text{Var}(\hat{s}) \) can be obtained as follows:

\[
\text{Var}(\hat{s}) = \left( \hat{J}(\theta)_{21} \hat{J}(\theta)_{22} - \hat{J}(\theta)_{22} \hat{J}(\theta)_{12} \right)^{-1} = - \left[ \frac{q^2}{\sigma^2} \left( \frac{1}{(\sigma^2)^2} \right) \sum_{p=1}^{q}(d_p - \mu)^2 \left( \frac{1}{(\sigma^2)^2} \right) \right]^{-1}.
\] (34)

The associated standard error of \( \hat{s} \) can be calculated by

\[
\text{SE}(\hat{s}) = \left[ \frac{q^2}{\sigma^2} \left( \frac{1}{(\sigma^2)^2} \right) \sum_{p=1}^{q}(d_p - \mu)^2 \left( \frac{1}{(\sigma^2)^2} \right) \right]^{-1/2}.
\] (35)

Therefore, an approximation \((1 - \alpha)100\%\) confidence interval of \( s \) can be obtained by using

\[
\hat{s} \pm z^* \text{SE}(\hat{s}).
\] (36)

4.3.2. Confidence Intervals for Parameter \( t \)

Similarly, a confidence interval of \( t \) can be obtained by using

\[
\hat{t} \pm z^* \text{SE}(\hat{t}).
\] (37)

The standard error of \( \hat{t} \) can be defined as follows (see Appendix A.2):

\[ ... \]
Appl. Sci. 2021, 11, 6768

4.3.3. Confidence Intervals for Parameter $r$

Similarly, an confidence interval of $r$ can be obtained by using

$$
\hat{r} \pm z^*SE(\hat{r}).
$$

The standard error of $\hat{r}$ can be defined as follows (see Appendix A.3):

$$
SE(\hat{r}) = \frac{1}{\sqrt{\frac{q}{\sigma^2} \left( \frac{1}{2} \left( \frac{\sum_{p=1}^{q} (d_p - \mu)^2}{\sigma^2} \right) + \left( \frac{\sum_{p=1}^{q} (d_p - \mu)}{\sigma^2} \right)^2 \right)}}.
$$

5. Case Study

This case study was performed using the printing data example proposed by Vining and Myers [6] and Lin and Tu [7]. The impacts of three input factors, that is, speed ($x_1$), pressure ($x_2$), and distance ($x_3$), on the capabilities of a printing machine that applies colored inks to package levels ($y$) were investigated. Each input factor had three levels, resulting in a total of 27 runs. Three replications were performed for each data combination.

In comparative studies between methods, the expected quality loss (EQL) is usually applied as a critical optimization criterion in RD. The expectation of the loss function is defined as

$$
EQL = \theta[(\hat{\mu}(x) - \tau)^2 + \hat{\sigma}^2(x)].
$$

where $\theta$ represents a positive loss coefficient. $\hat{\mu}(x)$, $\tau$, and $\hat{\sigma}(x)$ are the estimated process mean function, desirable target value, and estimated standard deviation function, respectively. In this case study, the target value was $\tau = 500$. Using MATLAB, the process-estimated mean and standard deviation functions obtained using the LSM based on RSM were as follows:

$$
\hat{\mu}_{LSM}(x) = 327.6296 + x^T m_{LSM} + x^T M_{LSM} x,
$$

$$
\hat{\sigma}_{LSM}(x) = 34.8832 + x^T n_{LSM} + x^T N_{LSM} x
$$

where

$$
x = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix},
\begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix},
\begin{bmatrix} 177.000 \\ 109.426 \\ 131.463 \end{bmatrix},
\begin{bmatrix} 32.000 & 66.028 & 75.472 \\ 66.028 & -22.389 & 43.583 \\ 75.472 & 43.583 & -29.056 \end{bmatrix},
\begin{bmatrix} 11.527 \\ 15.323 \\ 29.190 \end{bmatrix},
\begin{bmatrix} 4.204 & 7.7120 & 5.1093 \\ 7.7120 & -1.316 & 14.082 \\ 5.109 & 14.082 & 16.778 \end{bmatrix}.
$$

For comparison, the criteria used in the NN (mainly, the number of hidden neurons, training algorithms, and the number of epochs of NNs) for the conventional log-sigmoid transfer function in the hidden layer were the same as those of the proposed NNs. The NN information for the conventional log-sigmoid transfer function and the proposed DR estimation method are presented in Table 1. The NN architecture consists of several input factors, hidden neurons, and output responses. The weight and bias matrix values of the NNs using the conventional log-sigmoid transfer function for the process mean and standard deviation functions are presented in Table 2. Based on the structures, training algorithms, transfer functions, and the number of epochs of the proposed feed-forward NNs used to estimate the functions of process mean and standard deviation, optimal weight
parameters $\delta$, $\hat{t}$, and $\hat{r}$ were calculated using Equations (20), (22), and (24), respectively. As shown in Table 3, twenty hidden neurons in the hidden layer are used to estimate mean function, and twenty hidden neurons in the hidden layer are used to estimate standard deviation function. For each hidden neuron, a desirability function is proposed as a transfer function. In addition, weight parameters (i.e., $s$, $t$, and $r$) for DFs identified in Equations (8) and (9) and associated confidence intervals are demonstrated in Table 3. The respective 95% confidence intervals for these weight parameters calculated by using Equations (35), (38), and (40) are presented in Table 3. The weight and bias matrix values of the feed-forward NNs using the proposed transfer functions for the process mean and standard deviation functions are presented in Table 4.

Table 1. Feed-forward NNs information used to estimate DR functions.

| Model       | Transfer Functions | Training Function | Architecture | Number of Epochs |
|-------------|--------------------|-------------------|--------------|-----------------|
| Conventional| Mean               | Logsig-Purelin    | Trainlm      | 3-20-1          | 5               |
|             | Standard deviation | Logsig-Purelin    | Trainrp      | 3-20-1          | 83              |
| Proposed    | Mean               | “nominal is best”-Purelin | Trainlm      | 3-20-1          | 5               |
|             | Standard deviation | “smaller is better”-Purelin | Trainrp      | 3-20-1          | 83              |

Table 2. Weight and bias values from the NNs approach by using the conventional log-sigmoid transfer function.

(a) Estimated Process Mean 

| Weights | Biases | Weights | Biases |
|---------|--------|---------|--------|
| $W_{ij}^{\text{mean}}$ | $(V_{ij}^{\text{mean}})^{\prime}$ | $a_{ij}^{\text{mean}}$ | $b_{\text{NN}}^{\text{mean}}$ | $W_{ij}^{\text{std}}$ | $(V_{ij}^{\text{std}})^{\prime}$ | $a_{ij}^{\text{std}}$ | $b_{\text{NN}}^{\text{std}}$ |
| 7.048   | 3.571  | -1.123  | 0.134  | -7.259  | 6.693  | 3.771  | -0.855  | 0.0516  | -7.531  |
| 4.726   | -5.71  | -1.371  | -0.212 | -6.831  | 4.928  | -5.169 | -1.884  | -0.038 | -7.045  |
| -4.981  | 4.672  | 3.442   | -0.789 | 6.003   | -4.907 | 4.677  | 3.435   | -0.147 | 5.755   |
| 4.459   | 3.990  | 3.498   | 1.079  | -6.208  | 4.251  | 4.486  | 3.809   | 0.467  | -5.651  |
| 2.679   | 3.682  | -6.266  | -0.199 | -4.201  | 2.864  | 3.086  | -5.745  | 0.330  | -4.851  |
| -6.436  | 4.389  | 0.793   | 0.209  | 3.175   | -6.849 | 3.647  | 0.288   | 0.467  | 3.440   |
| -4.908  | 5.623  | -1.339  | 0.124  | 2.863   | -5.123 | 5.469  | -1.166  | -0.356 | 2.874   |
| 1.562   | -4.993 | 5.438   | 0.190  | -2.048  | 1.447  | -4.815 | 5.474   | -0.271 | -2.451  |
| 6.692   | 2.115  | 3.035   | -0.139 | -0.859  | 6.152  | 2.694  | 2.570   | -0.250 | -1.183  |
| 5.521   | -4.127 | 3.079   | 0.099  | -0.243  | 6.115  | -3.555 | 2.652   | -0.034 | 0.050   |
| -5.977  | 3.126  | -3.407  | 0.325  | -0.669  | -6.131 | 2.967  | -3.265  | 0.230  | -0.850  |
| 5.256   | -5.103 | 1.936   | 0.633  | 1.256   | 5.651  | -4.723 | 1.535   | -0.040 | 1.651   |
| 6.483   | -3.230 | 2.033   | -0.294 | 2.105   | 6.082  | -3.433 | 2.667   | 0.697  | 1.926   |
| -0.589  | -5.911 | -4.442  | -0.462 | -3.229  | -0.441 | -5.646 | -4.983  | -0.529 | -2.783  |
| 3.818   | -4.951 | -4.368  | -0.443 | 3.550   | 4.070  | -5.306 | -4.141  | 0.407  | 3.526   |
| -6.018  | 4.983  | -0.135  | -0.385 | -4.140  | -6.091 | 4.643  | -0.476  | -0.545 | -4.469  |
| -1.158  | 2.826  | 6.948   | -0.294 | -5.321  | -1.107 | 3.380  | 6.462   | -0.155 | -5.651  |
| 6.528   | -3.113 | -2.385  | 0.419  | 6.011   | 6.112  | -2.436 | -2.970  | -0.514 | 6.074   |
| 4.157   | 6.574  | 1.314   | 0.425  | 6.397   | 4.160  | 5.845  | 1.643   | 0.072  | 7.251   |
| 4.931   | -5.012 | -2.952  | -0.233 | 7.576   | 4.467  | -4.909 | -3.407  | -0.070 | 7.674   |
### Table 3. Estimated weight parameters and associated confidence intervals.

| Hidden Neurons | $\hat{\delta}$ | $\hat{\delta} - \frac{z^*SE(\delta)}{\sqrt{N}}$ | $\hat{\delta} + \frac{z^*SE(\delta)}{\sqrt{N}}$ | $\hat{\tau}$ | $\hat{\tau} - \frac{z^*SE(\tau)}{\sqrt{N}}$ | $\hat{\tau} + \frac{z^*SE(\tau)}{\sqrt{N}}$ |
|----------------|----------------|-----------------------------------------------|-----------------------------------------------|----------------|-----------------------------------------------|-----------------------------------------------|
| 1st            | 1.150          | 1.024                                         | 1.276                                         | 0.100          | -0.007                                         | 0.207                                         | 0.180                                         | 0.118                                         | 0.242                                         |
| 2nd            | 0.100          | -0.006                                        | 0.206                                         | 0.180          | 0.095                                          | 0.265                                         | 0.160                                         | 0.128                                         | 0.192                                         |
| 3rd            | 0.120          | 0.008                                         | 0.232                                         | 0.100          | 0.019                                          | 0.181                                         | 0.190                                         | 0.139                                         | 0.241                                         |
| 4th            | 0.100          | -0.022                                        | 0.222                                         | 0.150          | 0.056                                          | 0.244                                         | 0.091                                         | 0.177                                         | 1.043                                         |
| 5th            | 0.100          | -0.038                                        | 0.238                                         | 0.110          | 0.017                                          | 0.203                                         | 0.017                                         | 0.170                                         | 0.270                                         |
| 6th            | 0.100          | -0.056                                        | 0.256                                         | 0.650          | 0.469                                          | 0.831                                         | 0.310                                         | 0.156                                         | 0.464                                         |
| 7th            | 0.100          | -0.076                                        | 0.276                                         | 0.100          | -0.056                                         | 0.256                                         | 0.120                                         | -0.236                                       | 0.476                                         |
| 8th            | 0.100          | -0.108                                        | 0.308                                         | 0.140          | 0.005                                          | 0.275                                         | 0.100                                         | -0.115                                       | 0.315                                         |
| 9th            | 0.100          | -0.187                                        | 0.387                                         | 0.150          | 0.023                                          | 0.277                                         | 0.160                                         | 0.029                                         | 0.291                                         |
| 10th           | 0.820          | 0.667                                         | 0.973                                         | 0.370          | 0.245                                          | 0.495                                         | 0.110                                         | 0.012                                         | 0.208                                         |
| 11th           | 0.110          | -0.063                                        | 0.283                                         | 1.050          | 0.852                                          | 1.248                                         | 0.100                                         | 0.029                                         | 0.171                                         |
| 12th           | 0.130          | -0.073                                        | 0.333                                         | 0.100          | -0.137                                         | 0.337                                         | 0.150                                         | 0.095                                         | 0.205                                         |
| 13th           | 0.100          | -0.123                                        | 0.323                                         | 0.110          | -0.173                                         | 0.393                                         | 2.370                                         | 2.249                                         | 2.491                                         |
| 14th           | 0.120          | -0.139                                        | 0.379                                         | 0.120          | -0.283                                         | 0.523                                         | 0.390                                         | 0.267                                         | 0.513                                         |
| 15th           | 0.310          | 0.019                                         | 0.601                                         | 0.190          | -0.424                                         | 0.804                                         | 0.150                                         | 0.013                                         | 0.287                                         |
| 16th           | 0.480          | 0.313                                         | 0.647                                         | 0.100          | -0.324                                         | 0.524                                         | 0.230                                         | 0.071                                         | 0.389                                         |
| 17th           | 0.110          | -0.069                                        | 0.289                                         | 3.200          | 3.085                                          | 3.315                                         | 0.120                                         | -0.079                                       | 0.319                                         |
| 18th           | 0.240          | 0.042                                         | 0.438                                         | 0.100          | -0.017                                         | 0.217                                         | 0.100                                         | -0.261                                       | 0.461                                         |
| 19th           | 0.180          | -0.046                                        | 0.406                                         | 0.120          | 0.001                                          | 0.029                                         | 0.100                                         | -0.167                                       | 0.367                                         |
| 20th           | 0.100          | -0.154                                        | 0.354                                         | 0.100          | -0.021                                         | 0.221                                         | 0.400                                         | 0.204                                         | 0.596                                         |

### Table 4. Weight and bias values by using the NN-based DR estimation method.

| (a) Estimated Mean Response | (b) Estimated Standard Deviation Response |
|-----------------------------|------------------------------------------|
| **Weights**                 | **Biases**                               | **Weights**                 | **Biases**                               |
| $W_{ij}^{\text{mean,pro}}$  | $(v_j^{\text{mean,pro}})$               | $a_i^{\text{mean,pro}}$  | $b_i^{\text{pro}}$                      |
| $W_{ij}^{\text{std,pro}}$   | $(v_j^{\text{std,pro}})$               | $a_i^{\text{std,pro}}$  | $b_i^{\text{pro}}$                      |
| 7.048                       | 3.571                                    | -1.123                     | 0.134                                    |
| 4.726                       | -5.71                                    | -1.371                     | -0.212                                   |
| -4.981                      | 4.672                                    | 3.442                      | -0.789                                   |
| 4.459                       | 3.990                                    | 3.498                      | 1.079                                    |
| 2.679                       | 3.682                                    | -6.266                     | -0.199                                   |
| -6.436                      | 4.389                                    | 0.793                      | 0.209                                    |
| -4.908                      | 5.623                                    | -1.339                     | 0.124                                    |
| 1.562                       | -4.993                                   | 5.438                      | 0.190                                    |
| 6.692                       | 2.115                                    | 3.035                      | -0.139                                   |
| 5.521                       | -4.127                                   | 3.079                      | 0.099                                    |
| -5.977                      | 3.126                                    | -3.407                     | 0.325                                    |
| 5.256                       | -5.103                                   | 1.936                      | 0.633                                    |
| 6.483                       | -3.230                                   | 2.033                      | -0.294                                   |
| -0.589                      | -5.911                                   | -4.442                     | -0.462                                   |
| 3.818                       | -4.951                                   | -4.368                     | -0.443                                   |
| -6.018                      | 4.983                                    | -0.135                     | -0.385                                   |
| -1.158                      | 2.826                                    | 6.948                      | -0.294                                   |
| 6.528                       | -3.113                                   | -2.385                     | 0.419                                    |
| 4.157                       | 6.574                                    | 1.314                      | 0.425                                    |
| 4.931                       | -5.012                                   | -2.952                     | -0.233                                   |
The estimated process mean and standard deviation functions obtained by applying the conventional LSM-RSM, the NN-based estimation method using the conventional log-sigmoid transfer function, and the proposed DR approach with the estimated optimal weight parameters listed in Table 3, along with the associated coefficients of determination $R^2$ of each estimated response function, are illustrated in Figures 4–6 as contour plots, respectively.

**Figure 4.** Estimated process mean and standard deviation response functions from the conventional LSM approach based on RSM: (a) mean ($R^2 = 92.68\%$); (b) standard deviation ($R^2 = 45.42\%$).

**Figure 5.** Estimated process mean and standard deviation response functions from the feed-forward NN approach with conventional log-sigmoid transfer function: (a) mean ($R^2 = 93.44\%$); (b) standard deviation ($R^2 = 51.48\%$).

**Figure 6.** Estimated process mean and standard deviation response functions from the feed-forward NN approach with the proposed transfer functions: (a) mean ($R^2 = 67.17\%$); (b) standard deviation ($R^2 = 53.49\%$).
The optimal control factor settings, the optimal process parameters (i.e., process mean, bias, and variance), and their associated EQL values obtained by the three different approaches (i.e., the conventional LSM based on RSM, the NN-based estimation method using the conventional log-sigmoid transfer functions, and the proposed NN-based DR estimation method) are presented in Table 5.

**Table 5. Comparative results of various methods.**

| Estimation Model | Optimal Factor Settings | Process Mean | Process Bias | Process Variance | EQL |
|------------------|-------------------------|--------------|--------------|------------------|-----|
| LSM              | 1.000 0.072 -0.250      | 494.672      | 5.328        | 1977.533         | 2005.917 |
| Feed-forward NN  | -0.067 0.170 1.000      | 499.321      | 0.679        | 397.507          | 397.967 |
| Proposed method  | 0.133 -0.014 0.096      | 499.885      | 0.115        | 103.900          | 103.913 |

As demonstrated in Table 5, the process variance obtained from the NN-based estimation methods is remarkably smaller than that of the conventional statistical LSM approach. Therefore, NN-based estimation methods can provide considerably smaller EQL values (i.e., significant criteria to evaluate the level of quality) than those of the conventional LSM approach in this particular example, although the same RD optimization model is utilized. Of the two NN-based estimation methods, the proposed DR approach provides better optimal solutions when compared with that of the NN-based estimation method that uses the conventional log-sigmoid transfer function because the variance obtained from the proposed DR approach is almost zero in this particular example. Therefore, in terms of EQL, the best optimal solutions can be obtained using the NN-based DR estimation method. The criterion spaces (squared bias vs. variance) of the three estimated functions by using the conventional LSM approach based on RSM, the N-based estimation method using the log-sigmoid transfer function, and the proposed DR approach are illustrated in Figure 7. A green cross marks the optimal settings in each figure.

**Figure 7.** Criterion spaces of the three estimated functions: (a) LSM based on RSM; (b) log-sigmoid transfer function-based NN; (c) proposed NN-based DR method.

**6. Conclusions and Further Studies**

This study proposes a feed-forward NN-based DR estimation method to estimate the process mean and standard deviation functions for RSM and RD modeling. The proposed method can avoid basic error assumptions of the conventional LSM approach in this particular situation. Further, integrating the DFs into the NN structures in the proposed method results in a general and flexible transfer function family for transforming data.
based on the proposed DFs. The optimal weight parameters of the DF family can be determined using the MLE and the confidence intervals of the optimal weight parameters using the proposed asymptotic distribution of maximum-likelihood estimators. The best transfer function in the NN structure was achieved using the proposed method. The proposed NN-based DR estimation method provides better solutions for EQL than those of the conventional LSM approach and the NN-based estimation method that uses the conventional log-sigmoid transfer function in the particular case study example. In order to improve the reliability of the proposed methods, different types of experimental data (i.e., small and large data sets, different DoE results, and small and large sets of replications) should be considered.

In further studies, the assumption that the error follows a normal distribution and whether the DF values can follow any distribution will be investigated. Additionally, the optimal weight parameters of the DF-based transfer functions were estimated using a Bayesian approach or the Newton–Raphson algorithm. We then plan to determine the confidence intervals of the optimal weight parameters using a t-distribution. In addition, a more comprehensive comparative study between the proposed NN models and higher-order models of conventional RSM can be a significant further research issue. Based on this comparative study and a Bayesian approach, we may develop a new optimal estimation system by integrating the proposed NN models and conventional RSM models.

Author Contributions: Conceptualization, T.-H.L. and S.S.; Methodology, T.-H.L. and S.S.; Modeling, T.-H.L.; Validation T.-H.L. and S.S.; Writing—original draft preparation, T.-H.L. and S.S.; Writing—Review and Editing, H.J. and S.S.; Funding Acquisition, H.J. and S.S. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported by the National Research Foundation of Korea (NRF) (No. NRF-2019R1F1A1060067 and No. NRF-2019R1G1A1010335).

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

Appendix A. Confidence Intervals for Weight Parameters

Appendix A.1. Parameter $s$

Fisher’s total information matrix $J(\theta)$ for the general parameter $\theta$ is defined as follows:

$$
J(\theta) = \begin{pmatrix}
\sum_{i=1}^{q} \frac{\partial^2 \ell(\theta|d_i)}{\partial \mu^2} & \sum_{i=1}^{q} \frac{\partial^2 \ell(\theta|d_i)}{\partial \mu \partial \sigma^2} & \sum_{i=1}^{q} \frac{\partial^2 \ell(\theta|d_i)}{\partial \mu \partial \sigma^2} \\
\sum_{i=1}^{q} \frac{\partial^2 \ell(\theta|d_i)}{\partial \mu \partial \sigma^2} & \sum_{i=1}^{q} \frac{\partial^2 \ell(\theta|d_i)}{\partial \sigma^2} & \sum_{i=1}^{q} \frac{\partial^2 \ell(\theta|d_i)}{\partial \sigma^2 \partial \mu} \\
\sum_{i=1}^{q} \frac{\partial^2 \ell(\theta|d_i)}{\partial \mu \partial \sigma^2} & \sum_{i=1}^{q} \frac{\partial^2 \ell(\theta|d_i)}{\partial \sigma^2 \partial \mu} & \sum_{i=1}^{q} \frac{\partial^2 \ell(\theta|d_i)}{\partial \sigma^2}
\end{pmatrix}.
$$

(A1)

The first derivatives of the log-likelihood function $\ell(\theta|d_i)$ for the mean, variance, and parameter $s$ can be expressed as follows:

$$
\frac{\partial \ell(\theta|d_i)}{\partial \mu} = \frac{\sum_{j=0}^{q} \frac{d_p - \mu}{\sigma^2}}{\sum_{j=0}^{q} \frac{d_p - \mu}{\sigma^2}}, \quad \frac{\partial \ell(\theta|d_i)}{\partial \sigma^2} = -\frac{q}{2} \frac{1}{\sigma^2} + \frac{\sum_{j=0}^{q} (d_p - \mu)^2}{2 (\sigma^2)^2} \quad \text{and} \quad \frac{\partial \ell(\theta|d_i)}{\partial s} = \frac{q}{2} + \sum_{j=0}^{q} \log \left( \frac{\gamma_p - \gamma_p}{\gamma_p - \gamma_p} \right).
$$

(A2)

The second derivatives of the log-likelihood function $\ell(\theta|d_i)$ for the mean, variance, and parameter $s$ with mixed partials are given as follows:

$$
\frac{\partial^2 \ell(\theta|d_i)}{\partial \mu^2} = -\frac{q}{\sigma^2}, \quad \frac{\partial^2 \ell(\theta|d_i)}{\partial \mu \partial \sigma^2} = -\frac{\sum_{j=0}^{q} (d_p - \mu)^2}{(\sigma^2)^2}, \quad \frac{\partial^2 \ell(\theta|d_i)}{\partial \sigma^2} = 0, \quad \frac{\partial^2 \ell(\theta|d_i)}{\partial \sigma^2 \partial \mu} = -\frac{\sum_{j=0}^{q} (d_p - \mu)^2}{(\sigma^2)^2}.
$$

(A3)

Consequently, Fisher’s total information matrix $J(\theta)$ can be expressed as
\[-\mathbf{J}(\mathbf{\theta}) = \begin{pmatrix} \frac{-q}{\sigma^2} & \frac{-\sum_{p=1}^{q}(d_p - \mu)}{(\sigma^2)^2} & 0 \\ \frac{q}{2(\sigma^2)^2} & \frac{\sum_{p=1}^{q}(d_p - \mu)^2}{(\sigma^2)^3} & 0 \\ 0 & 0 & \frac{-q}{t^2} \end{pmatrix}. \quad (A4)\]

Appendix A.2. Parameter $t$

The general parameter $\mathbf{\theta}$ can be determined as $\mathbf{\theta} = (\mu, \sigma^2, t)^T$. Based on Equation (27), $-\mathbf{J}(\mathbf{\theta})$ can be expressed as

\[-\mathbf{J}(\mathbf{\theta}) = \begin{pmatrix} \frac{\partial^2 \ell(\mathbf{\theta}|\mathbf{d}_j)}{\partial \mu^2} & \frac{\partial^2 \ell(\mathbf{\theta}|\mathbf{d}_j)}{\partial \mu \partial \sigma^2} & \frac{\partial^2 \ell(\mathbf{\theta}|\mathbf{d}_j)}{\partial \mu \partial t} \\ \frac{\partial^2 \ell(\mathbf{\theta}|\mathbf{d}_j)}{\partial \sigma^2 \partial \mu} & \frac{\partial^2 \ell(\mathbf{\theta}|\mathbf{d}_j)}{\partial (\sigma^2)^2} & \frac{\partial^2 \ell(\mathbf{\theta}|\mathbf{d}_j)}{\partial \sigma^2 \partial t} \\ \frac{\partial^2 \ell(\mathbf{\theta}|\mathbf{d}_j)}{\partial \partial \mu} & \frac{\partial^2 \ell(\mathbf{\theta}|\mathbf{d}_j)}{\partial \partial \sigma^2} & \frac{\partial^2 \ell(\mathbf{\theta}|\mathbf{d}_j)}{\partial \partial t} \end{pmatrix}. \quad (A5)\]

Based on the log-likelihood function in Equation (21), the first derivatives of the log-likelihood function $\ell(\mathbf{\theta}|\mathbf{d}_j)$ for the mean, variance, and parameter $t$ can be expressed as follows:

\[
\begin{align*}
\frac{\partial \ell(\mathbf{\theta}|\mathbf{d}_j)}{\partial \mu} &= \frac{\sum_{p=1}^{q}(d_p - \mu)}{\sigma^2}, \quad \frac{\partial \ell(\mathbf{\theta}|\mathbf{d}_j)}{\partial \sigma^2} = -\frac{q}{2\sigma^2} + \frac{\sum_{p=1}^{q}(d_p - \mu)^2}{2(\sigma^2)^2} \quad \text{and} \quad \frac{\partial \ell(\mathbf{\theta}|\mathbf{d}_j)}{\partial t} = \frac{q}{t} + \sum_{p=1}^{q} \log \left( \frac{y_p - \mu}{\bar{y}_p - \bar{y}} \right). \\
\end{align*} \quad (A6)\]

The second derivatives of log-likelihood function $\ell(\mathbf{\theta}|\mathbf{d}_j)$ for the mean, variance, and parameter $t$ with mixed partials are given as follows:

\[
\begin{align*}
\frac{\partial^2 \ell(\mathbf{\theta}|\mathbf{d}_j)}{\partial \mu^2} &= \frac{-q}{\sigma^2}, \quad \frac{\partial^2 \ell(\mathbf{\theta}|\mathbf{d}_j)}{\partial \mu \partial \sigma^2} = -\frac{\sum_{p=1}^{q}(d_p - \mu)^2}{(\sigma^2)^2}, \quad \frac{\partial^2 \ell(\mathbf{\theta}|\mathbf{d}_j)}{\partial \mu \partial t} = 0, \quad \frac{\partial^2 \ell(\mathbf{\theta}|\mathbf{d}_j)}{\partial \sigma^2 \partial \mu} = -\frac{\sum_{p=1}^{q}(d_p - \mu)}{(\sigma^2)^2}, \\
\frac{\partial^2 \ell(\mathbf{\theta}|\mathbf{d}_j)}{\partial (\sigma^2)^2} &= \frac{-q}{2(\sigma^2)^2}, \quad \frac{\partial^2 \ell(\mathbf{\theta}|\mathbf{d}_j)}{\partial \sigma^2 \partial t} = 0, \quad \frac{\partial^2 \ell(\mathbf{\theta}|\mathbf{d}_j)}{\partial t \partial \mu} = 0, \quad \frac{\partial^2 \ell(\mathbf{\theta}|\mathbf{d}_j)}{\partial t \partial \sigma^2} = 0 \quad \text{and} \quad \frac{\partial^2 \ell(\mathbf{\theta}|\mathbf{d}_j)}{\partial t^2} = -\frac{q}{t^2}. \\
\end{align*} \quad (A7)\]

Therefore,

\[-\mathbf{J}(\mathbf{\theta}) = \begin{pmatrix} \frac{-q}{\sigma^2} & \frac{-\sum_{p=1}^{q}(d_p - \mu)}{(\sigma^2)^2} & 0 \\ \frac{q}{2(\sigma^2)^2} & \frac{\sum_{p=1}^{q}(d_p - \mu)^2}{(\sigma^2)^3} & 0 \\ 0 & 0 & \frac{-q}{t^2} \end{pmatrix} \begin{pmatrix} \hat{\mathbf{\theta}}_{11} & \hat{\mathbf{\theta}}_{12} & \hat{\mathbf{\theta}}_{13} \\ \hat{\mathbf{\theta}}_{21} & \hat{\mathbf{\theta}}_{22} & \hat{\mathbf{\theta}}_{23} \\ \hat{\mathbf{\theta}}_{31} & \hat{\mathbf{\theta}}_{32} & \hat{\mathbf{\theta}}_{33} \end{pmatrix}. \quad (A8)\]

The square root of the diagonal elements of $\mathbf{J}(\mathbf{\theta})^{-1}$ represents the standard error for each estimator. An estimator for the standard error of $\hat{\mathbf{\theta}}$ is of particular interest and can be obtained using the inverses of the partitioned matrices. First, the partition $-\mathbf{J}(\mathbf{\theta})$ is

\[-\mathbf{J}(\mathbf{\theta}) = \begin{pmatrix} \hat{\mathbf{\theta}}_{11} & \hat{\mathbf{\theta}}_{12} \\ \hat{\mathbf{\theta}}_{21} & \hat{\mathbf{\theta}}_{22} \end{pmatrix}. \quad (A9)\]

Next, one estimator for $\text{Var}(\hat{\mathbf{\theta}})$ can be obtained as follows:

\[
\text{Var}(\hat{\mathbf{\theta}}) = (\mathbf{J}(\mathbf{\theta})_{11} - \mathbf{J}(\mathbf{\theta})_{12} - \mathbf{J}(\mathbf{\theta})_{21} \mathbf{J}(\mathbf{\theta})_{22})^{-1} \\
= \begin{pmatrix} \frac{q}{\sigma^2} \left( \frac{\sum_{p=1}^{q}(d_p - \mu)^2}{(\sigma^2)^3} + \left( \frac{\sum_{p=1}^{q}(d_p - \mu)}{(\sigma^2)^2} \right)^2 \right) & 0 \\ 0 & \frac{q}{t^2} \end{pmatrix}^{-1}. \quad (A10)\]

The associated standard error of $\hat{\mathbf{\theta}}$ can be defined as
\[ \text{SE}(\hat{t}) = \sqrt{\frac{1}{q \sigma^2} \left[ \frac{q}{2} \left( \frac{1}{\sigma^2} - \frac{\sum_{p=1}^{q} (d_p - \mu)^2}{(\sigma^2)^2} \right) + \left( \frac{\sum_{p=1}^{q} (d_p - \mu)}{\sigma^2} \right)^2 \right]} \]  
\[ (A11) \]

Appendix A.3. Parameter \( r \)

To estimate parameter \( r \), general parameter \( \theta \) can be determined as \( \theta = (\mu \sigma^2 r)^T \).

From Equation (27),

\[ -\hat{J}(\theta) = \begin{pmatrix} \frac{\partial^2 \ell(\theta|d)}{\partial \mu^2} & \frac{\partial^2 \ell(\theta|d)}{\partial \mu \partial \sigma^2} & \frac{\partial^2 \ell(\theta|d)}{\partial \mu \partial r} \\ \frac{\partial^2 \ell(\theta|d)}{\partial \sigma^2 \partial \mu} & \frac{\partial^2 \ell(\theta|d)}{\partial (\sigma^2)^2} & \frac{\partial^2 \ell(\theta|d)}{\partial \sigma^2 \partial r} \\ \frac{\partial^2 \ell(\theta|d)}{\partial \sigma^2 \partial r} & \frac{\partial^2 \ell(\theta|d)}{\partial \sigma^2 \partial \mu} & \frac{\partial^2 \ell(\theta|d)}{\partial (\sigma^2)^2} \end{pmatrix} \]  
\[ (A12) \]

Based on the log-likelihood function in Equation (23), the first derivatives of the log-likelihood function \( \ell(\theta|d) \) for the mean, variance, and parameter \( r \) can be expressed as follows:

\[ \frac{\partial \ell(\theta|d)}{\partial \mu} = \frac{\sum_{p=1}^{q} (d_p - \mu)}{\sigma^2}, \quad \frac{\partial \ell(\theta|d)}{\partial \sigma^2} = -\frac{1}{2 \sigma^2} + \frac{1}{2} \frac{\sum_{p=1}^{q} (d_p - \mu)^2}{(\sigma^2)^2} \quad \text{and} \quad \frac{\partial \ell(\theta|d)}{\partial r} = \frac{q}{r} + \sum_{p=1}^{q} \log \left( \frac{y_p - \mu}{\lambda_p - \mu} \right) \]  
\[ (A13) \]

The second derivatives of log-likelihood function \( \ell(\theta|d) \) for the mean, variance, and parameter \( r \) with mixed partials are given as follows:

\[ \frac{\partial^2 \ell(\theta|d)}{\partial \mu^2} = \frac{q}{\sigma^2}, \quad \frac{\partial^2 \ell(\theta|d)}{\partial \mu \partial \sigma^2} = -\frac{\sum_{p=1}^{q} (d_p - \mu)}{(\sigma^2)^2}, \quad \frac{\partial^2 \ell(\theta|d)}{\partial \mu \partial r} = 0, \quad \frac{\partial^2 \ell(\theta|d)}{\partial \sigma^2 \partial \mu} = 0, \quad \frac{\partial^2 \ell(\theta|d)}{\partial \sigma^2 \partial r} = 0 \quad \text{and} \quad \frac{\partial^2 \ell(\theta|d)}{\partial r^2} = \frac{-q}{r^2}. \]  
\[ (A14) \]

Therefore,

\[ -\hat{J}(\theta) = \begin{pmatrix} \frac{q}{\sigma^2} & -\frac{\sum_{p=1}^{q} (d_p - \mu)}{(\sigma^2)^2} & 0 \\ -\frac{\sum_{p=1}^{q} (d_p - \mu)}{(\sigma^2)^2} & \frac{q}{2 (\sigma^2)^2} & -\frac{\sum_{p=1}^{q} (d_p - \mu)^2}{(\sigma^2)^3} \\ 0 & -\frac{\sum_{p=1}^{q} (d_p - \mu)}{(\sigma^2)^2} & \frac{q}{(\sigma^2)^2} \end{pmatrix} = \begin{pmatrix} \hat{J}(\theta)_{11} & \hat{J}(\theta)_{12} & \hat{J}(\theta)_{13} \\ \hat{J}(\theta)_{21} & \hat{J}(\theta)_{22} & \hat{J}(\theta)_{23} \\ \hat{J}(\theta)_{31} & \hat{J}(\theta)_{32} & \hat{J}(\theta)_{33} \end{pmatrix} \]  
\[ (A15) \]

The squared root of the diagonal entries of \( \hat{J}(\theta)^{-1} \) represents the standard error for each estimator. An estimator for the standard error of \( \hat{r} \) is of particular interest and can be obtained using the inverses of the partitioned matrices. First, partition \( \hat{J}(\theta) \) is

\[ -\hat{J}(\theta) = \begin{pmatrix} \hat{J}(\theta)_{11} & \hat{J}(\theta)_{12} \\ \hat{J}(\theta)_{21} & \hat{J}(\theta)_{22} \end{pmatrix} \]  
\[ (A16) \]

Next, one estimator for \( \text{Var}(\hat{r}) \) can be obtained as follows:

\[ \text{Var}(\hat{r}) = (\hat{J}(\theta)_{12} \hat{J}(\theta)_{22} - \hat{J}(\theta)_{12} \hat{J}(\theta)_{22})^{-1} = \left[ -\frac{q}{\sigma^2} \left( \frac{q}{2 (\sigma^2)^2} - \frac{\sum_{p=1}^{q} (d_p - \mu)^2}{(\sigma^2)^3} \right) + \left( \frac{\sum_{p=1}^{q} (d_p - \mu)^2}{(\sigma^2)^2} \right)^2 \right]^{-1} \]  
\[ (A17) \]

Finally, the associated standard error of \( \hat{r} \) can be defined as
\[
\text{SE}^2(\hat{r}) = \frac{1}{\sqrt{\frac{q}{\sigma^2} \left( \frac{1}{2} \left( \frac{1}{\mathcal{O}} \right)^2 - \frac{\sum_{p=1}^{q} (d_p - \mu)^2}{\sigma^2} \right) + \left( \frac{\sum_{p=1}^{q} (d_p - \mu)}{\sigma^2} \right)^2}} 
\]

(A18)

References

1. Taguchi, G. Introduction to Quality Engineering; UNIPUB/Kraus International: White Plains, NY, USA, 1986.
2. Leon, R.V.; Shoemaker, A.C.; Kackar, R.N. Performance measures independent of adjustment: An explanation and extension of Taguchi signal-to-noise ratio. Technometrics 1987, 29, 253–285.
3. Box, G.E.P. Signal-to-noise ratios, performance criteria, and transformations. Technometrics 1988, 30, 1–17.
4. Box, G.; Bisgaard, S.; Fung, C. An explanation and critique of Taguchi’s contribution to quality engineering. Qual. Reliab. Eng. Int. 1988, 4, 123–131.
5. Nair, V.N. Taguchi’s parameter design: A panel discussion. Technometrics 1992, 34, 127–161.
6. Vining, G.G.; Myers, R.H. Combining Taguchi and response surface philosophies: A dual response approach. J. Qual. Technol. 1990, 22, 38–45.
7. Lin, D.K.J.; Tu, W. Dual response surface optimization. J. Qual. Technol. 1995, 27, 34–39.
8. Del Castillo, E.; Montgomery, D.C. A nonlinear programming solution to the dual response problem. J. Qual. Technol. 1993, 25, 199–204.
9. Copeland, K.A.F.; Nelson, P.R. Dual response optimization via direct function minimization. J. Qual. Technol. 1996, 28, 331–336.
10. Cho, B.R.; Philips, M.D.; Kapur, K.C. Quality improvement by RSM modeling for robust design. In Proceedings of the 5th Industrial Engineering Research Conference, Minneapolis, MN, USA, 18–20 May 1996; pp. 650–655.
11. Ames, A.E.; Mattucci, N.; Macdonald, S.; Szonyi, G.; Hawkins, D.M. Quality loss functions for optimization across multiple response surfaces. J. Qual. Technol. 1997, 29, 339–346.
12. Borror, C.M. Mean and variance modeling with qualitative responses: A case study. Qual. Eng. 1998, 11, 141–148.
13. Kim, K.J.; Lin, D.K.J. Dual response surface optimization: A fuzzy modeling approach. J. Qual. Technol. 1998, 30, 1–10.
14. Koksoy, O.; Doganaksoy, N. Joint optimization of mean and standard deviation using response surface methods. J. Qual. Technol. 2003, 35, 239–252.
15. Ding, R.; Lin, D.K.J.; Wei, D. Dual response surface optimization: A weighted MSE approach. Qual. Eng. 2004, 16, 377–385.
16. Shin, S.; Cho, B.R. Bias-specified robust design optimization and an analytical solutions. Comput. Ind. Eng. 2005, 48, 129–148.
17. Shin, S.; Cho, B.R. Robust design models for customer specified bounds on process parameters. J. Syst. Sci. Syst. Eng. 2006, 15, 2–18.
18. Shin, S.; Cho, B.R. Studies on a bi-objective robust design optimization problem. IIE Trans. 2009, 41, 957–968.
19. Robinson, T.J.; Wulff, S.S.; Montgomery, D.S.; Khuri, A.I. Robust parameter design using generalized linear mixed models. J. Qual. Technol. 2006, 38, 65–75.
20. Fogliatto, S.F. Multiresponse optimization of products with functional quality characteristics. Qual. Reliab. Eng. Int. 2008, 24, 927–939.
21. Goethals, P.L.; Cho, B.R. The development of a robust design methodology for time-oriented dynamic quality characteristics with a target profile. Qual. Reliab. Eng. Int. 2010, 27, 403–414.
22. Truong, N.K.V.; Shin, S. Development of a new robust design method based on Bayesian perspectives. Int. J. Qual. Eng. Technol. 2012, 3, 50–78.
23. Truong, N.K.V.; Shin, S. A new robust design method from an inverse-problem perspective. Int. J. Qual. Eng. Technol. 2013, 3, 243–271.
24. Nha, V.T.; Shin, S.; Jeong, S.H. Lexicographical dynamic goal programming approach to a robust design optimization within the pharmaceutical environment. Eur. J. Oper. Res. 2013, 229, 505–517.
25. Baba, I.; Midi, H.; Rana, S.; Ibragimov, G. An alternative approach of dual response surface optimization based on penalty function method. Math. Probl. Eng. 2015, 2015, 450131.
26. Yankılıç, I.; den Hertog, D.; Kleijnen, J.P. Robust dual-response optimization. IIE Trans. 2016, 48, 298–312.
27. Myers, R.H.; Montgomery, D.C. Response Surface Methodology: Process and Product Optimization Using Designed Experiments, 1st ed. John Wiley & Sons: New York, NY, USA, 1995.
28. Lunter, J.J. Achieving continuous improvement with the dual response approach: A demonstration of the Roman catapult. Qual. Eng. 1994, 6, 691–705.
29. Cho, B.R.; Park, C.S. Robust design modeling and optimization with unbalanced data. Comput. Ind. Eng. 2005, 48, 173–180.
30. Lee, S.B.; Park, C.S. Development of robust design optimization using incomplete data. Comput. Ind. Eng. 2006, 50, 345–356.
31. Cho, B.R.; Choi, Y.; Shin, S. Development of censored data-based robust design for pharmaceutical quality by design. Int. J. Adv. Manuf. Technol. 2010, 49, 839–851.
32. Googebeur, Y.; Goos, P.; Vandebroek, M. A hierarchical Bayesian approach to robust parameter design. SSRN 2007, 1–23.
33. Chen, Y.; Ye, K. Bayesian hierarchical modeling on dual response surfaces in partially replicated designs. Qual. Technol. Quant. Manag. 2009, 6, 371–389.
34. Chen, Y.; Ye, K. A Bayesian hierarchical approach to dual response surface modeling. *J. Appl. Stat.* 2011, 38, 1963–1975.
35. Irie, B.; Miyake, S. Capabilities of three-layered perceptrons. In Proceedings of the International Conference Neural Networks, San Diego, CA, USA, 24–27 July 1988; pp. 641–648.
36. Hornik, K.; Stinchcombe, M.; White, H. Multilayer feedforward networks are universal approximators. *Neural Netw.* 1989, 2, 359–366.
37. Cybenko, G. Approximation by superpositions of a sigmoidal function. *Math. Control Signals Syst.* 1989, 2, 303–314.
38. Funahashi, K. On the approximate realization of continuous mappings by neural networks. *Neural Netw.* 1989, 2, 183–192.
39. Harrington, E.C. The desirability function. *Ind. Qual. Control.* 1965, 21, 494–498.
40. Derringer, G.; Suich, R. Simultaneous optimization of several response variables. *J. Qual. Technol.* 1980, 12, 214–219.
41. Rowlands, H.; Packianather, M.S.; Oztemel, E. Using artificial neural networks for experimental design in off-line quality. *J. Syst. Eng.* 1996, 6, 46–59.
42. Su, C.T.; Hsieh, K.L. Applying neural network approach to achieve robust design for dynamic quality characteristics. *Int. J. Qual. Reliab. Manag.* 1998, 15, 509–519.
43. Cook, D.F.; Ragsdale, C.T.; Major, R.L. Combining a neural network with a genetic algorithm for process parameter optimization. *Eng. Appl. Artif. Intell.* 2000, 13, 391–396.
44. Chow, T.T.; Zhang, G.Q.; Lin, Z.; Song, C.L. Global optimization of absorption chiller system by genetic algorithm and neural network. *Energy Build.* 2002, 34, 103–109.
45. Chang, H.H. Applications of neural networks and genetic algorithms to Taguchi’s robust design. *Int. J. Electron. Commer.* 2005, 3, 90–96.
46. Chang, H.H.; Chen, Y.K. Neuro-genetic approach to optimize parameter design of dynamic multiresponse experiments. *Appl. Soft. Comput.* 2011, 11, 436–442.
47. Arungpadang, R.T.; Kim, J.Y. Robust parameter design based on back propagation neural network. *Korean Manag. Sci. Rev.* 2012, 29, 81–89.
48. Box, G.E.P.; Wilson, K.B. On the experimental attainment of optimum conditions (with discussion). *J. R. Stat. Soc. B* 1951, 13, 270–310.
49. Myers, R.H. Response surface methodology—Current status and future directions. *J. Qual. Technol.* 1999, 31, 54–57.
50. Khuri, A.I.; Mukhopadhyay, S. Response surface methodology. *Wiley Interdiscip. Rev. Comput Stat.* 2010, 2, 128–149.
51. Sarle, W.S. Neural networks and statistical models. In Proceeding of the 19th Annual SAS User Group International Conference, Dallas, TX, USA, 10–13 April 1994; pp. 1–13.
52. Zainuddin, Z.; Pauline, O. Function approximation using artificial neural networks. *WSEAS Trans. Math.* 2008, 7, 333–338.
53. Hartman, E.J.; Keeler, J.D.; Kowalski, J.M. Layered neural networks with Gaussian hidden units as universal approximations. *Neural Comput.* 1990, 2, 210–215.
54. Zilouchian, A.; Jamshidi, M. *Intelligent Control. Systems Using Soft Computing Methodologies*; CRC Press: Boca Raton, FL, USA, 2000.
55. Lindsey, J.K. *Parametric Statistical Inference*; Oxford University Press: New York, NY, USA, 1996.