Computers, Materials & Continua
DOI:10.32604/cmc.2022.023119
Article

Tech Science Press

Estimating Weibull Parameters Using Least Squares and Multilayer Perceptron *vs.* Bayes Estimation

Walid Aydi[1,3,*] and Fuad S. Alduais[2,4]

[1]Department of Computer Science, College of Humanities and Science in Al Aflaj, Prince Sattam Bin Abdulaziz University, Al-Aflaj, Saudi Arabia
[2]Department of Mathematics, College of Humanities and Science in Al Aflaj, Prince Sattam Bin Abdulaziz University, Al-Aflaj, Saudi Arabia
[3]Laboratory of Electronics & Information Technologies, Sfax University, Sfax, Tunisia
[4]Department of Administration, Administrative Science College, Thamar University, Yemen
*Corresponding Author: Walid Aydi. Email: w.aydi@psau.edu.sa

Received: 28 August 2021; Accepted: 25 October 2021

Abstract: The Weibull distribution is regarded as among the finest in the family of failure distributions. One of the most commonly used parameters of the Weibull distribution (WD) is the ordinary least squares (OLS) technique, which is useful in reliability and lifetime modeling. In this study, we propose an approach based on the ordinary least squares and the multilayer perceptron (MLP) neural network called the OLSMLP that is based on the resilience of the OLS method. The MLP solves the problem of heteroscedasticity that distorts the estimation of the parameters of the WD due to the presence of outliers, and eases the difficulty of determining weights in case of the weighted least square (WLS). Another method is proposed by incorporating a weight into the general entropy (GE) loss function to estimate the parameters of the WD to obtain a modified loss function (WGE). Furthermore, a Monte Carlo simulation is performed to examine the performance of the proposed OLSMLP method in comparison with approximate Bayesian estimation (BLWGE) by using a weighted GE loss function. The results of the simulation showed that the two proposed methods produced good estimates even for small sample sizes. In addition, the techniques proposed here are typically the preferred options when estimating parameters compared with other available methods, in terms of the mean squared error and requirements related to time.

Keywords: Weibull distribution; maximum likelihood; ordinary least squares; MLP neural network; weighted general entropy loss function

1 Introduction

The parameters of the Weibull distribution are widely used in reliability studies and many engineering applications, such as the lifetime analysis of material strength [1], estimation of rainfall [2], hydrology [3], predictions of material and structural failure [4], renewable and alternative energies [5–8], power electronic systems [9], and many other fields [10–12].

The form of the probability density function (PDF) of two parameters of WD is given by:

$$\mathrm{f}(x;\ \vartheta,\lambda) = \frac{\lambda}{\vartheta}\left(\frac{x}{\vartheta}\right)^{(\lambda-1)} \exp\left[-\left(\frac{x}{\vartheta}\right)^{\lambda}\right],\quad x > 0\quad \vartheta,\lambda > 0 \tag{1}$$

The cumulative distribution function (CDF) and the survival function S of the WD can be expressed as

$$F(x;\ \vartheta,\lambda) = 1 - \exp\left[-\left(\frac{x}{\vartheta}\right)^{\lambda}\right],\quad x > 0\quad \vartheta,\lambda > 0 \tag{2}$$

$$S(t;\ \vartheta,\lambda) = \exp\left[-\left(\frac{t}{\vartheta}\right)^{\lambda}\right],\quad t > 0\quad \vartheta,\lambda > 0 \tag{3}$$

where the parameters ϑ and λ represent the scale and the shape of the distribution, respectively.

Several approaches to estimating the parameters of the WD have been proposed [13]. They can generally be classified as manual or numerical [14].

Manual approaches include the ordinary least squares [15,16], unbiased good linear estimators [17], and weighted least squares [18]. Computational methods include maximum likelihood estimation [19], the moments estimation method [20], Bayesian approach [21], and least-squares estimation with particle swarm optimization [22].

In addition to computational methods, many studies in the literature have attempted to use the neural network (NN) to anticipate the parameters of the WD in many areas, such as the method developed by Jesus that applies the Weibull and ANN analysis to anticipate the shelf life and acidity of vacuum-packed fresh cheese [23]. In survival analysis, Achraf constructed a deep neural network model called DeepWeiSurv. It was assumed that the distribution of survival times follows a finite mixture of a two-parameter WD [24]. In another work in the field of electric power generation, an artificial NN (ANN) and q-Weibull were applied to the survival function of brushes in hydroelectric generators [25].

Recently, a few methods have been attempted to combine the robustness of the ANN and some of the above statistical methods. Maria modeled the distribution of tree diameters using the OLS and the ANN [26]. In the same way and based on the ability of the OLS, in its simplest form, which assumes a linear relationship between the predictor and the unreliability function on one hand and the robustness and rapidness of the single-hidden-layer networks to handle the linear functions compared with multiple-hidden-layer [27] on the other hand, we will propose to combine OLS and a neural network to predict the two-parameter WD.

In the proposed method, we solve the problem whereby the reliability of the OLS method is compromised by outliers through the introduction of a pre-trained neural network after the linearization of the CDF. The remaining sections of this paper are organized as follows: Section 2 provides a review of different numerical and graphical methods for estimating the parameters of the WD, such as the MLE, OLS, WLS, and BLGE. In Section 3 we present the proposed methods. To evaluate their appropriateness in comparison with competing methods, the relevant performance metrics are covered in Section 4. The results are discussed in Section 5. Finally, the conclusions of this study are provided in Section 6.

2 Review of Numerical and Graphical Methods for Estimating Parameters of WD

The most commonly used approaches to estimate the parameters λ and ϑ of the WD are described below.

2.1 Maximum Likelihood Estimator (MLE)

Let the set $(x_1, x_2, x_3, \ldots x_n)$ of n random lifetimes from the WD be defined by Eq. (1). Then, the likelihood function Lf and its corresponding logarithm ℓ for the given sample observations are shown in Eqs. (4) and (5), respectively [28]:

$$Lf(\underline{x};\ \vartheta,\lambda)=\prod_{i=1}^{n}\frac{\lambda}{\vartheta}\left(\frac{x_i}{\vartheta}\right)^{\lambda-1}\exp\left[-\left(\frac{x}{\vartheta}\right)^{\lambda}\right], \tag{4}$$

$$\ell\ = nln\lambda - n\,\lambda ln\vartheta + (\lambda-1)\sum_{i=1}^{n} lnx_i\ - \sum_{i=1}^{n}\left(\frac{x_i}{\vartheta}\right)^{\lambda} \tag{5}$$

The partial derivatives of the equation for ℓ with respect to the variables ϑ and λ are given by:

$$\frac{\partial\ell}{\partial\ \vartheta} = -n\ \left(\frac{\lambda}{\hat{\vartheta}}\right) + \frac{\lambda}{\hat{\vartheta}}\sum_{i=1}^{n}\left(\frac{x_i}{\hat{\vartheta}}\right) = 0, \tag{6}$$

$$\frac{\partial\ell}{\partial\ \lambda} = \frac{n}{\hat{\lambda}} + \sum_{i=1}^{n}\left(\frac{x_i}{\vartheta}\right) - \sum_{i=1}^{n}\left(\frac{x_i}{\vartheta}\right)^{\lambda} ln\left(\frac{x_i}{\vartheta}\right) = 0 \tag{7}$$

The MLE estimator $\hat{\vartheta}_{MLE}$ of ϑ is:

$$\hat{\vartheta}_{MLE} = \left[\frac{1}{n}\sum_{i=1}^{n}(x_i)^{\hat{\lambda}}\right]^{\frac{1}{\hat{\lambda}}} \tag{8}$$

The parameter λ can be obtained by using any numerical method, such as the Newton–Raphson.

2.2 Ordinary Least Squares Method (OLS)

To estimate the parameters of the WD, the OLS method is extensively used in mathematics and engineering problems [16]. We can obtain a linear relationship between parameters by taking the logarithm of Eq. (2) as follows:

$$ln[-ln(1-F(x;\ \vartheta,\lambda)] = -\lambda ln\vartheta + \lambda lnx \tag{9}$$

Let $Y_i = ln[-ln(1-F(x_i;\ \vartheta,\lambda))]$, $X_i = lnx_{(i)}$, $\alpha_0 = -\lambda ln\vartheta$, and $\beta = \lambda$. Then, Eq. (9) can be written as $\mathrm{Y}_i = \alpha_0 + \beta X_i + \epsilon_i$

Let $X_{(1)},\ X_{(2)},\ X_{(3)}, \ldots X_{(n)}$ be order statistics of $X_1,\ X_2,\ X_3, \ldots X_n$, and let $x_{(1)} <\ x_{(2)}\ < x_{(3)} < \ldots < x_{(\mathrm{n})}$ be the ordered observations in a random sample of size n. To estimate the values of the cumulative distribution function $F(x_{(i)}\vartheta,\lambda)$, we use the mean rank method as follows:

$$\hat{F}(x_i) = (i-0.3)/(n+0.4),\quad i = 1,2,3,\ldots,n \tag{10}$$

The estimates $\hat{\alpha}_0$ and $\hat{\beta}$ of the regression parameters α_0 and β minimize the function

$$\mathcal{Q}(a,\ b\ ) = \sum_{i=1}^{n} (\,Y_i - \alpha_0 - \beta lnx_{(i)}\ )^2 \tag{11}$$

Therefore, the estimates $\hat{\alpha}_0$ and $\hat{\beta}$ of the parameters α_0 and β are given by

$$\hat{\beta}_{OLS} = \frac{n\sum_{i=1}^{n} lnx_{(i)}\ \ \ln[-ln(1-\hat{F}(x_i)))] - \sum_{i=1}^{n} lnx_i\ \ \sum_{i=1}^{n}\ln[-ln(1-\hat{F}(x_i)))]}{n\ \sum_{i=1}^{n}\ln^2(x_i) - \left(\sum_{i=1}^{n}\ln x_{(i)}\right)^2} \tag{12}$$

$$\hat{\alpha}_{0_{OLS}} = \frac{1}{n}\sum_{i=1}^{n}\ln[-ln(1-\hat{F}(x_i)))] - \hat{\beta}_{OLS}\frac{1}{n}\sum_{i=1}^{n}\ln x_{(i)} \tag{13}$$

The estimates $\hat{\lambda}_{OLS}$ and $\hat{\vartheta}_{OLS}$ of the parameters λ and ϑ are given by

$$\hat{\lambda}_{OLS} = \frac{n\sum_{i=1}^{n} lnx_i\ \ ln[-ln(1-\hat{F}_i)] - \sum_{i=1}^{n} lnx_i\ \ \sum_{i=1}^{n} ln[-ln(1-\hat{F}_i))]}{n\ \ \sum_{i=1}^{n} ln^2(x_i) - \left(\sum_{i=1}^{n} lnx_{(i)}\right)^2} \tag{14}$$

$$\hat{\vartheta}_{OLS} = exp\left[\ -\frac{\sum_{i=1}^{n} ln[-ln(1-\hat{F}_i))] - \hat{\beta}_{OLS}\ \sum_{i=1}^{n} lnx_{(i)}}{n\hat{\beta}_{OLS}}\right] \tag{15}$$

2.3 *Weighted Least Squares Method (WLS)*

In the WLS estimate, the parameters λ and ϑ are the values of the parameters that minimize the function:

$$\mathcal{Q}^*{}_{\mathcal{W}}(\lambda\ ,\ \vartheta\ ) = \sum_{i=1}^{n}\mathcal{W}_i(ln[-ln\ \ (1-\hat{F}(x_i))] + ln\vartheta - \lambda lnx_{(i)})^2 \tag{16}$$

The biggest challenge in the application of the WLS is in finding the weights W_i in Eq. (15). We use the delta method [29] to find them:

$$Var\ (ln[-ln\ \ (1-\hat{F}(x_i))]\ ) \propto \frac{1}{[(1-\hat{F}(x_i))ln(1-\hat{F}(x_i))]^2} \tag{17}$$

Hence, the weights can be written as follows:

$$\mathcal{W}_i = \frac{[(1-\hat{F}(x_i))ln(1-\hat{F}(x_i))]^2}{\sum_{i=1}^{n}[(1-\hat{F}(x_i))ln(1-\hat{F}(x_i))]^2},\qquad i = 1\ ,2,\ 3,\ \ldots, n. \tag{18}$$

Minimizing $\mathcal{Q}^*{}_{\mathcal{W}}(\lambda\ ,\vartheta)$ we obtain the WLS estimates of λ and ϑ as

$$\hat{\lambda}_{WOLS} = \frac{\sum_{i=1}^{n}\mathcal{W}_i\,\mathcal{D}_i\mathcal{A}_i - \left(\sum_{i=1}^{n}\mathcal{W}_i\,\mathcal{D}_i\right)\left(\sum_{i=1}^{n}\mathcal{W}_i\,\mathcal{A}_i\right)}{\sum_{i=1}^{n}\mathcal{W}_i\,\mathcal{D}_i^2 - \left(\sum_{i=1}^{n}\mathcal{W}_i\,\mathcal{D}_i\right)^2} \tag{19}$$

$$\hat{\vartheta}_{WOLS} = \exp\left[\frac{\hat{\psi}_{W}}{\hat{\lambda}_{WOLS}}\right], \tag{20}$$

where

$$\hat{\psi}_{\mathrm{W}} = \hat{\lambda}_{WOLS}\sum_{i=1}^{n}\mathrm{W}_i\,\mathrm{D}_i - \sum_{i=1}^{n}\mathrm{W}_i\,\mathrm{A}_i, \text{ with } \mathrm{D}_i = lnx_{(i)} \text{ and } \mathrm{A}_i = ln[-ln(1-\hat{F}(x_i))]$$

2.4 Approximate Bayes Estimator

In this section, the approximate Bayesian estimator under a GE loss function of the parameters λ and ϑ of the WD is discussed. We assume a non-informative (vague) prior according to [30] as

$$\pi(\lambda\ ,\ \varphi) \propto \frac{1}{\lambda\vartheta} \tag{21}$$

The parameters λ and φ are estimated using Lindley's approximation technique. The posterior expectation E is given by Eq. (22) [31]:

$$E[u(\phi)|x] = \frac{\int_{\Omega} u(\phi)\pi(\phi)L(\phi|x)\,d\phi}{\int_{\Omega}\pi(\phi)L(\phi|x)\,d\phi} \tag{22}$$

Moreover, it can be asymptotically estimated by:

$$E[u(\phi)|x] = u + \frac{1}{2}\sum_{i}\sum_{j}(u_{ij} + 2u_i\rho_j)\sigma_{ij} + \frac{1}{2}\sum_{i}\sum_{j}\sum_{k}\sum_{l}L_{ijk}\,\sigma_{ij}\sigma_{kl}u_i \tag{23}$$

where $i,j,k,l = 1,2,\ldots m$, $\phi = (\phi_1,\ \phi_1,\ldots\phi_m)$, $\pi(\phi)$ represents the prior distribution of ϕ, $u = u(\phi)$, $L = L(\phi)$ is the likelihood function, $\rho \equiv \rho(\phi) = \ln(\pi(\phi))$, $\rho_i = \frac{\partial\rho}{\partial\phi_i}$, $u_i = \frac{\partial u}{\partial\phi_i}$, $u_{ij} = \frac{\partial^2 u}{\partial\phi_i\partial\phi_j}$, $L_{ijk} = \frac{\partial^3 L}{\partial\phi_i\partial\phi_j\partial\phi_k}$, and $\sigma_{ij} = element\ (\ i,\ j)$ of the covariance matrix of the parameter estimators.

For the two-parameter case $\phi = (\lambda,\ \varphi)$, Eq. (22) reduces to:

$$E\ [(\phi|x)] = u + \frac{1}{2}[u_{11}\sigma_{11} + u_{22}\sigma_{22}] + u_1\rho_1\sigma_{11} + u_2\rho_2\sigma_{22} + \frac{1}{2}[L_{30}u_1\sigma_{11}^2 + L_{03}u_2\sigma_{22}^2] \tag{24}$$

The functions in Eq. (24) are computed using MLEs with respect to λ and φ.

To apply the Lindley model of Eq. (24) to estimate the parameters of the WD, the following are obtained from Eq. (23):

$$\rho(\varphi,\lambda) = ln\frac{1}{\lambda\vartheta} = -ln(\varphi) - ln(\lambda)$$

$$\rho_1 = \frac{\partial\rho}{\partial\varphi} = -\left(\frac{1}{\varphi}\right)$$

$$\rho_2 = \frac{\partial\rho}{\partial\lambda} = -\left(\frac{1}{\lambda}\right)$$

The elements σ_{ij} of the covariance matrix are expressed by

$$L_{20} = n\left(\frac{\lambda}{\varphi^2}\right) - \left(\frac{\lambda^2}{\varphi^2}\right)\sum_{i=1}^{n}\left(\frac{x_i}{\varphi}\right)^{\lambda} - \left(\frac{\lambda}{\varphi^2}\right)\sum_{i=1}^{n}\left(\frac{x_i}{\varphi}\right)^{\lambda}$$

$$\sigma_{11} = (-L_{20})^{-1}$$

$$L_{30} = -2n\left(\frac{\lambda}{\varphi^3}\right) + 2\left(\frac{\lambda^2}{\varphi^3}\right)\sum_{i=1}^{n}\left(\frac{x_i}{\varphi}\right)^{\lambda} + \left(\frac{\lambda^3}{\varphi^3}\right)\sum_{i=1}^{n}\left(\frac{x_i}{\varphi}\right)^{\lambda} + 2\left(\frac{\lambda}{\varphi^3}\right)\sum_{i=1}^{n}\left(\frac{x_i}{\varphi}\right)^{\lambda} + \left(\frac{\lambda^2}{\varphi^3}\right)\sum_{i=1}^{n}\left(\frac{x_i}{\varphi}\right)^{\lambda}$$

$$L_{02} = -\left(\frac{n}{\varphi^2}\right) - \sum_{i=1}^{n}\left(\frac{x_i}{\varphi}\right)^{\lambda} ln^2\left(\frac{x_i}{\varphi}\right)$$

$$\sigma_{22} = (-L_{02})^{-1}$$

$$L_{03} = 2\left(\frac{n}{\lambda^3}\right) - \sum_{i=1}^{n}\left(\frac{x_i}{\varphi}\right)^{\lambda} ln^3\left(\frac{x_i}{\varphi}\right).$$

2.4.1 Estimates Based on General Entropy Loss Function

The general entropy loss function L for ϕ, shown in Eq. (24), is expressed by the following form [32]:

$$L(\hat{\phi},\ \phi\ ) \propto \left(\frac{\hat{\phi}}{\phi}\right)^{q} - q ln\left(\frac{\hat{\phi}}{\phi}\right) - 1, \qquad q \neq 0, \tag{25}$$

where $\hat{\phi}$ is an estimate of ϕ. The Bayes estimator of ϕ, denoted by $\hat{\phi}_{GE}$, is the value $\hat{\phi}$ that minimizes Eq. (26):

$$\hat{\phi}_{GE} = [E_{\vartheta}(\phi^{-q})]^{-\frac{1}{q}} \tag{26}$$

The BLGE of $\hat{\lambda}_{BLGE}$ for λ from Eq. (24) is found by the following expressions:

$$u = (\lambda)^{-q}, \quad u_2 = \frac{\partial u}{\partial \lambda} = -q(\lambda)^{-q-1}, \quad u_{22} = \frac{\partial^2 u}{\partial (\lambda)^2} = -(-q^2 - q)(\lambda)^{-q-2}$$

$$u_1 = 0, \quad u_{11} = 0$$

In the same way, the BLGE of $\hat{\vartheta}_{BLGE}$ for ϑ is found by the following expressions:

$$u = (\vartheta)^{-q}, \quad u_1 = \frac{\partial u}{\partial \vartheta} = -q(\vartheta)^{-q-1}, \quad u_{11} = \frac{\partial^2 u}{\partial (\vartheta)^2} = -(-q^2 - q)(\vartheta)^{-q-2}$$

$$u_2 = 0, \quad u_{22} = 0.$$

3 Proposed Methods

In the following sections, we describe the proposed BLWGE and OLSMLP methods.

3.1 Weighted General Entropy Loss Function

The WGE loss function was proposed as dependent on the weighted loss GE function as follows:

$$L_w(\hat{\phi},\ \phi\ ) \propto\ w(\phi)\ \left[\left(\frac{\hat{\phi}}{\phi}\right)^q - qln\left(\frac{\hat{\phi}}{\phi}\right) - 1\quad\right], \qquad q \neq 0 \tag{27}$$

where ϕ represents the estimated parameters that minimize the expectation of the loss function (Eq. (27)), and $w(\phi)$ represents the proposed weighted function as expressed by Eq. (28):

$$w(\phi) = \frac{1}{\phi^z} \tag{28}$$

Based on the posterior distribution of the parameter ϕ, and by using the WGE function given in Eq. (28), we obtain the estimated BLWGE of the parameter ϑ as follows:

$$\begin{aligned}
\mathrm{E}[L_w(\hat{\phi},\ \phi)] &= \int_{\forall\phi} L_w(\hat{\phi},\ \phi) f(\phi|\underline{x})\ d\phi \\
&= \int_{\forall\phi} w(\phi)[(\hat{\phi}/\ \phi)^q - qln(\hat{\phi}/\ \phi) - 1] f(\phi|\underline{x})\ d\phi \\
&= \int_{\forall\phi} \tfrac{1}{\phi^z}\ [(\hat{\phi}/\ \phi)^q - qln(\hat{\phi}/\ \phi) - 1] f(\phi|\underline{x})\ d\phi \\
&= \int_{\forall\phi} \tfrac{1}{\phi^z}\ (\hat{\phi}/\ \phi)^q\ f(\phi|\underline{x})\ d\phi - \int_{\forall\phi} \frac{qln\hat{\phi}}{\phi^z}\ f(\phi|\underline{x})\ d\phi + \int_{\forall\phi} \frac{qln\phi}{\phi^z} f(\phi|\ \underline{x})\ d\phi - \int_{\forall\phi} \frac{1}{\phi^z} f(\phi|\underline{x})\ d\phi \\
&= \hat{\phi}^q \int_{\forall\phi} \frac{1}{\phi^{z+q}} f(\phi|\underline{x})\ d\phi - qln\hat{\phi} \int_{\forall\phi} \frac{1}{\phi^z} f(\phi|\underline{x})\ d\phi + q \int_{\forall\phi} \frac{ln\phi}{\phi^z} f(\phi|\underline{x})\ d\phi - \int_{\forall\phi} \frac{1}{\phi^z} f(\phi|\ \underline{x})\ d\phi
\end{aligned}$$

$$\mathrm{E}[L_w(\hat{\phi},\ \phi)] = \hat{\phi}^q\ E\ (\phi^{-(z+q)}|\ \underline{x}) - qln\hat{\phi}\ E\ (\phi^{-z)}|\ \underline{x}) + qE\left(\frac{ln\phi}{\phi}|\ \underline{x}\right) - E\ (\phi^{-z)}|\ \underline{x})$$

$$\frac{\partial L_w(\hat{\phi},\ \phi)}{\partial\hat{\phi}} = q\ \hat{\phi}^{q-1}\ E\ (\phi^{-(z+q)}|\ \underline{x}) -\ \frac{q}{\hat{\phi}}\ E\ (\phi^{-z)}|\ \underline{x}) = 0$$

Thus, we can find that

$$q\ \hat{\phi}^{q-1}\ E\ (\phi^{-(z+q)}|\ \underline{x}) = \frac{q}{\hat{\phi}}\ E\ (\phi^{-z)}|\ \underline{x})$$

Consequently, the BLWGE of parameter ϕ, obtained by using the WGE loss function, is $\hat{\phi}_{BLWGE}$ as presented in Eq. (29):

$$\hat{\phi}_{WGE} = \frac{E\ (\phi^{-(z+q)}|\ \underline{x})}{E\ (\phi^{-z)}|\ \underline{x})} \qquad \text{where,}\ \ \mathrm{z} + \mathrm{q} \neq 0 \tag{29}$$

provided that $E_\phi(\phi^{-z})$ and $E_\phi(\phi^{-(z+q)})$ exist and are finite, where E_ϕ represents the expected value.

We note that the GE is a special case of the WGE when $z = 0$ in Eq. (29).

3.1.1 Estimates of Parameters of WD Based on Weighted General Entropy Loss Function

Based on the WGE and by using Eq. (29), the approximate Bayes estimator $\hat{\lambda}_{BLWGE}$ for λ is shown as:

$$\hat{\lambda}_{BLWGE} = \left[\frac{E(\lambda^{\ -(z+q)} |\ \underline{x})}{E(\lambda^{\ -z)} |\ \underline{x})}\right]^{-\frac{1}{q}} = \left[\frac{I_1}{I_2}\right]^{-\frac{1}{q}} \tag{30}$$

where

$$I_1 = u^* + \frac{1}{2}[u^*_{11}\sigma_{11} + u^*_{22}\sigma_{22}] + u^*_1\rho_1\sigma_{11} + u^*_2\rho_2\sigma_{22} + \frac{1}{2}[L_{30}\ u^*_1\ \ \sigma^2_{11} + L_{03}\ u^*_2\ \ \sigma^2_{22}] \tag{31}$$

$$u^* = (\lambda)^{-(Z+q)}, \quad u^*_2 = \frac{\partial u}{\partial \beta} = -(z+q)(\lambda)^{-(q+z)-1}, \quad u^*_{22} = \frac{\partial^2 u}{\partial (\lambda)^2} = -(-(z+q)^2 - (q+z))(\lambda)^{-(q+z)-2}$$

$$u^*_1 = 0, \quad u^*_{11} = 0$$

and

$$I_2 = u^{**} + \frac{1}{2}[u^{**}_{11}\sigma_{11} + u^{**}_{22}\sigma_{22}] + u^{**}_1\rho_1\sigma_{11} + u^{**}_2\rho_2\sigma_{22} + \frac{1}{2}[L_{30}\ u^{**}_1\ \ \sigma^2_{11} + L_{03}u^{**}_2\ \ \sigma^2_{22}] \tag{32}$$

Thus, the BLWGE $\hat{\lambda}_{BLWGE}$ for the shape parameter λ is

$$\hat{\lambda}_{\mathrm{WGE}}\left[\frac{\mathrm{I}_1}{\mathrm{I}_2}\right]^{-\frac{1}{\mathrm{q}}} = \left[\frac{u^* + \frac{1}{2}[u^*_{11}\sigma_{11} + u^*_{22}\sigma_{22}] + u^*_1\rho_1\sigma_{11} + u^*_2\rho_2\sigma_{22} + \frac{1}{2}[L_{30}\ u^*_1\ \ \sigma^2_{11} + L_{03}\ u^*_2\ \ \sigma^2_{22}]}{u^{**} + \frac{1}{2}[u^{**}_{11}\sigma_{11} + u^{**}_{22}\sigma_{22}] + u^{**}_1\rho_1\sigma_{11} + u^{**}_2\rho_2\sigma_{22} + \frac{1}{2}[L_{30}\ u^{**}_1\ \ \sigma^2_{11} + L_{03}u^{**}_2\ \ \sigma^2_{22}]}\right]^{-\frac{1}{\mathrm{q}}} \tag{33}$$

Similarly, the BLWGE $\hat{\vartheta}_{BLWGE}$ for ϑ, is given by Eq. (34):

$$\hat{\vartheta}_{WGE} = \left[\frac{E(\vartheta^{\ -(z+q)} |\ \underline{x})}{E(\vartheta^{\ -z)} |\ \underline{x})}\right]^{-\frac{1}{q}} = \left[\frac{I_3}{I_4}\right]^{-\frac{1}{q}} \tag{34}$$

where

$$I_3 = \nu + \frac{1}{2}[\nu_{11}\sigma_{11} + \nu_{22}\sigma_{22}] + \nu_1\rho_1\sigma_{11} + \nu_2\rho_2\sigma_{22} + \frac{1}{2}[L_{30}\nu_1\sigma^2_{11} + L_{03}\nu_2\sigma^2_{22}] \tag{35}$$

$$\nu = (\vartheta)^{-(z+q)}, \quad \nu_1 = \frac{\partial u}{\partial \vartheta} = -(z+q)(\vartheta)^{-(q+z)-1}, \quad \nu_{11} = \frac{\partial^2 u}{\partial (\vartheta)^2} = -(-(z+q)^2 - (z+q))(\vartheta)^{-(z+q)-2}$$

$$\nu_2 = 0, \quad \nu_{22} = 0,$$

and

$$I_4 = \nu^* + \frac{1}{2}[\nu^*_{11}\sigma_{11} + \nu^*_{22}\sigma_{22}] + \nu^*_1\rho_1\sigma_{11} + \nu^*_2\rho_2\sigma_{11} + \frac{1}{2}[L_{30}\ \nu^*_1\ \ \sigma^2_{11} + L_{03}\nu^*_2\ \ \ \sigma^2_{22}] \tag{36}$$

$$\nu^* = (\vartheta)^{-z}, \quad \nu^*_1 = \frac{\partial u}{\partial \vartheta} = -z(\vartheta)^{-z-1}, \quad \nu^*_{11} = \frac{\partial^2 u}{\partial (\vartheta)^2} = -(z^2 - z)(\vartheta)^{-z-2}$$

$$\nu^*_2 = 0, \quad \nu^*_{22} = 0.$$

Thus, the weighted Bayes estimator for the shape parameter ϑ is

$$\hat{\vartheta}_{\mathrm{BLWGE}}\left[\frac{\mathrm{I}_3}{\mathrm{I}_4}\right]^{-\frac{1}{q}} = \left[\frac{\nu + \frac{1}{2}[\nu_{11}\sigma_{11} + \nu_{22}\sigma_{22}] + \nu_1\rho_1\sigma_{11} + \nu_2\rho_2\sigma_{22} + \frac{1}{2}[\mathrm{L}_{30}\nu_1\sigma_{11}^2 + \mathrm{L}_{03}\nu_2\sigma_{22}^2]}{\nu^* + \frac{1}{2}[\nu_{11}^*\sigma_{11} + \nu_{22}^*\sigma_{22}] + \nu_1^*\rho_1\sigma_{11} + \nu_2^*\rho_2\sigma_{11} + \frac{1}{2}[\mathrm{L}_{30}\ \nu_1^*\ \ \sigma_{11}^2 + \mathrm{L}_{03}\nu_2^*\ \ \sigma_{22}^2]}\right]^{-\frac{1}{q}} \tag{37}$$

3.2 Ordinary Least Squares and the Multilayer Perceptron Neural Network (OLSMLP)

As previous studies have shown [14,33], manual calculations yield the smallest standard deviation (STD) in the parameter λ, and are consequently more accurate than computational methods. Moreover, methods of manual estimation are more accurate for small sample sizes [14]. However, these computational methods, especially the OLS, are sensitive to outliers and specific residual behavior [34]. To solve these problems, many studies have proposed different methods, such as the iterative weighting method based on the modified OLS [34], the WLS, and many other methods based on the WLS [35]. A major challenge in these methods is determining the weights.

3.2.1 Proposed Method to Estimate Parameters of WD

We now describe the proposed method, which is divided into two main parts: the linearization of the CDF, and the application of a feedforward network with backpropagation to estimate the values of λ and ϑ of the WD.

The OLS method takes the CDF defined in Eq. (2) and linearizes it as described in Eq. (10). It then determines the coefficients α_0 and β via linear regression by using the slope and the intercept. The principle of the method used by the OLS to compute α_0 and β can be violated even with a few outliers.

Therefore, instead of using the slope and the intercept, we propose applying Algorithm 1 as described below.

- **Application of Proposed Model to Estimate Parameters of WD**

The steps used to evaluate the parameters of the WD from the input csv file are described by Algorithm 1.

Input: Three comma separated value file (CSV) files containing the matrices X_i and Y_i, and their corresponding parameters (shape and scale) *SCi*.
Output: The predicted shape $\hat{\lambda}_{OLSMLP}$ and scale $\hat{\vartheta}_{OLSMLP}$ for the test set.
1: Normalize the inputs matrices X_i, Y_i, and *SCi* separately to unit norm using RobustScaler followed by MinMaxScaler norm.
2: Split the normalized X_i, Y_i, and *SCi* into random training and test subsets.
3: Create the neural network model (define the input layer, hidden layer, and output layer).
4: Compile the model and fit it to the data.
5: Predict $\hat{\lambda}_{OLSMLP}$ and $\hat{\vartheta}_{OLSMLP}$ for the test set.
6: Evaluate the performance of the proposed model.
Steps **2**, **3**, **4,** and **5** are explained in more detail in the following subsections:

- **Data Normalization**

Normalization is an essential preprocessing tool for a neural network [36,37]. Before training a neural network model, the input data are scaled using the RobustScaler norm in a preliminary phase, where each sample with at least one non-zero component is rescaled using the median and quartile range as described by Eq. (38). The RobustScaler norm is used to remove the influence of outliers. Following this, the MinMaxScaler, defined by Eq. (39), is applied to the output of the RobustScaler. The MinMaxScaler scales all the data features to the range [0, 1]:

$$X_i^{sm} = \frac{X_i - Q_1(\boldsymbol{X})}{Q_3(\boldsymbol{X}) - Q_1(\boldsymbol{X})} \tag{38}$$

where X is a feature vector, X_i is an element of feature X, X_i^{sm} is the rescaled element obtained by using MinMaxScaler, and $\mathrm{X}_{\mathrm{i}}^{\mathrm{sr}}$ is the rescaled element obtained by using RobustScaler.

$$X_i^{sr} = \frac{X_i^{sr} - \min(X^{sr})}{\max(X^{sr}) - \min(X^{sr})} \tag{39}$$

- **Structure of the Proposed Neural Network**

To estimate the parameters of the WD, we propose using a multilayer perceptron (MLP), which is a feedforward network with backpropagation [38]. According to the structure of the MLP, the proposed network, as shown in Fig. 1, consists of an input layer (with *n* neurons), a hidden layer (with *k* neurons), and an output layer (with *m* neurons that yield the Weibull parameters as the output of the network).

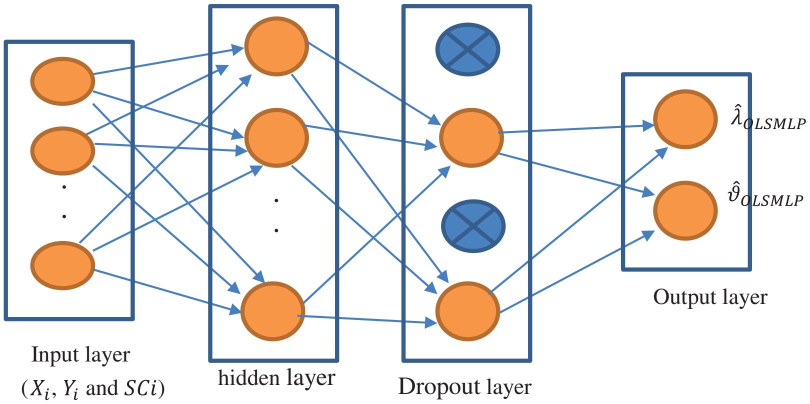

Figure 1: Topology of the proposed MLP

Various criteria have been proposed in the literature to fix the number of hidden neurons [39]. In our architecture, we use the rule whereby "the number of hidden neurons *k* should be 2/3 times the size of the input layer, plus the size of the output layer" [38–40].

The hyperbolic tangent activation function (*tanh*) is proposed here in the input layer, and the sigmoid function in the output layer. They are used frequently in feedforward nets, and are suitable for shallow networks as well as applications of prediction and mapping [38,41].

The objective of our neural network is a model that performs well on the data used in both the training and the test datasets. For this reason, we add a well-known regularization layer as described in the next section.

- **Regularization**

Regularization is a technique that can prevent overfitting [37,38]. A number of regularization techniques have been develop in the literature, such as L1 and L2 regularizations, bagging, and dropout. In the proposed structure, we use dropout, a well-known technique that randomly "drops out" or omits hidden neurons of the neural network to make them unavailable during part of the training [38,42]. This reduces the co-adaption between neurons, which results in less overfitting [38].

- **Optimization Algorithm**

The optimization of deep networks is an active area of research [43]. The most popular gradient-based optimization algorithms are Adagrad, Momentum, RMSProp, Adam, AdaDelta, AdaMax, Nadam, and AMSGrad [38,43,44]. We chose Nadam due to its superiority in supervised machine learning over the other techniques, especially for a deep network [43]. Moreover, it combines the strengths of the Nesterov acceleration gradient (NAG) and the adaptive estimation (Adam) algorithms as described in [44]:

$$w_{t+1} = w_t - \frac{\alpha_{nad}}{\sqrt{\widehat{v}_t} + \varepsilon}\left(\beta_1 \hat{m}_t + \frac{1-\beta_1}{1-\beta_1^t} * \frac{\partial L}{\partial w_t}\right) \tag{40}$$

where

$$\hat{m}_t = \frac{m_t}{1-\beta_1^t}$$

$$\widehat{v}_t = \frac{v_t}{1-\beta_2^t}$$

$$m_t = \beta_1 m_{t-1} + (1-\beta_1)\frac{\partial L}{\partial w_t}$$

$$v_t = \beta_2 v_{t-1} + (1-\beta_2)\left[\frac{\partial L}{\partial w_t}\right]^2$$

t: time step

α_{nad} : learning rate

v_t : the exponential average square of gradients

m_t : momentum vector

w_t : the weight that we want to update

ε : smoothing term

$\frac{\partial L}{\partial w_t}$: gradient of L; the loss function to minimize.

β_1, β_2: momentum decay and scaling decay, respectively

4 Performance Metrics

To evaluate the proposed methods with respect to other methods, we used two statistical tools, the mean squared error (MSE) and the mean absolute percentage error (MAPE) [5], in addition to the computation time.

5 Results and Discussion

5.1 Dataset Description

We generated 250,000 random data points from the WD for different parameters and different values of ϑ ranging from 1 to 299, and those of λ ranging from 0.5 to 100. For each shape/scale pair, we generated 10,000 samples of different sizes $n = 10,\ 20,\ 30,\ 40,\ \text{and } 50$.

We used the same dataset for the neural network in the training phase, but applied one sample to each shape/scale pair. This was unlike in the other methods (MLE, OLS, WLS, BLGE, and BLWGE), which used 10,000 samples to estimate the parameters of the WD. This dataset was divided into two subsets. The first subset was used to fit the model, and is referred to as the training dataset; it was characterized by known inputs and outputs. The second subset is referred to as the test dataset, and was used to evaluate the fitted machine learning model and make predictions on the new subset, for which we did not have the expected output. We chose the train–test procedure for our experiments because we guessed that we had a sufficiently large dataset available.

5.2 Experimental Setting

5.2.1 Parameter Selection for OLSMLP

In all experiments, we trained the model with Google Collaboratory (GPU) for 25 epochs. We used the Nadam optimizer with learning rate of $\alpha_{nad} = 0.001$; terms representing the momentum decay, scaling decay, and smoothing were kept at their default values: $\beta_1 = 0.9$, $\beta_2 = 0.999$, and $\varepsilon = 10^{-7}$. A dropout with a ratio of 0.6 was applied to the hidden layer. As described in Section 3, the hidden and output layers used the *tanh* and *sigmoid* activation functions, respectively. The error function or loss function was the mean squared error, and was used to estimate the loss of the model.

5.2.2 Parameter Selection of BLGE and BLWGE

In all experiments, the parameters of the BLWGE and BLGE were empirically determined. The values of the weights q and z of the BLWGE were -3 and 6, respectively. For the BLGE, the parameter $q = 1.5$.

5.3 Estimating Parameters of Weibull Distribution

5.3.1 Effect of Sample Size on Estimation of WD Parameters Using Prevalent Methods

Fig. 2 shows the evolution of the average MSE as a function of the sample size n. The MSE decreased quasi-linearly from $n = 10$ to $n = 40$ for all methods. Fig. 2 shows that the BLWGE, WLS, BLGE, and MLE had the lower MSE values for the different sample sizes compared with the OLS. We can deduce also that the WLS, GE, and MLE gave similar results with a slightly better start for the MLE at $n = 10$.

5.3.2 Effect of Sample Size on Estimation of WD Parameters Using Proposed Method

To illustrate how the sample size affects the calculation of the MSE, Fig. 3 shows the evolution of the latter as a function of the sample size n from 10 to 50.

From Fig. 3, we can deduce that as the sample size increased, the estimate of the MSE by the proposed method decreased and fluctuated. This fluctuation was due to the random nature of the information used and the limited number of samples (one sample) for each pair of shapes/scales.

Tabs. 1 and 2 show the results of the simulation of the proposed method and the other methods considered above. The results show the following:

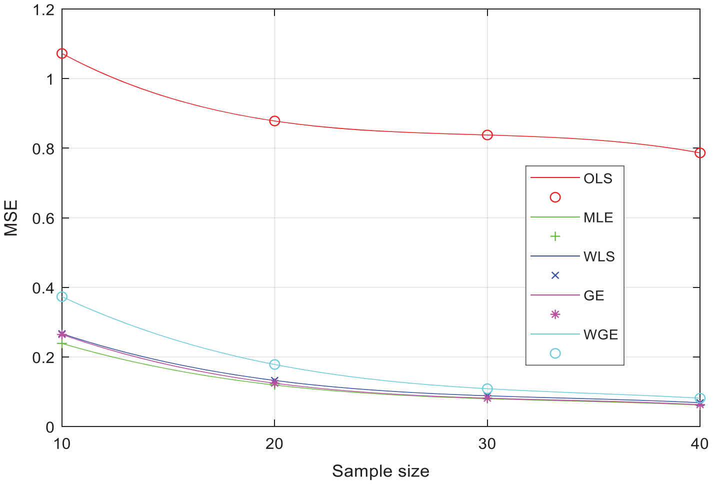

Figure 2: The evolution of the MSE using the parameters $\vartheta = 2.5$ and $\lambda = 1.685$ as a function of $n = [10 - 40]$ for the MLE, OLS, WLS, BLGE, and BLGWGE

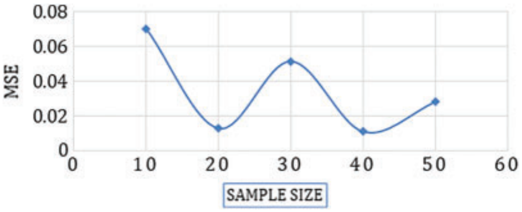

Figure 3: The evolution of the MSE using the parameters $\vartheta = 0.75$ and $\lambda = 1.75$ as a function of $n = [10 - 50]$

1. The MLE and WLS behaved similarly as shown in Tab. 1: Their MSE values decreased gradually when their shape values increased at a fixed scale. Conversely, when the scale value increased with a fixed shape, the MSE increased.

2. The behavior of the OLS and GE was the opposite of that of the MLE and WLS. As depicted in Tab. 1, the MSE increased when the shape increased (at a fixed scale), and decreases when the scale increased (with a fixed shape).

3. The BLWGE and the OLSMLP behaved similarly in terms of scale estimation, as shown in Tab. 1.

4. All methods had the same global variation function, as shown in Fig. 4 and Tab. 2.

5. The MLE was slightly superior globally in terms of scale estimation to the other methods, but had the worst estimation of shape, as shown in Tab. 2.

6. The proposed MLP neural network acceptably estimated the scale, better than some methods. By contrast, it outperformed all other methods in terms of shape estimations most of the time.

Table 1: MSEs of the estimated ϑ with varying values of the parameters λ and ϑ for different parameters of WD estimation methods

Scale (ϑ)	Shape (λ)	$\hat{\vartheta}_{\mathrm{MLE}}$	$\hat{\vartheta}_{\mathrm{OLS}}$	$\hat{\vartheta}_{WLS}$	$\hat{\vartheta}_{\mathrm{BLGE}}$	$\hat{\vartheta}_{\mathrm{BLWGE}}$	$\hat{\vartheta}_{\mathrm{OLSMLP}}$
1	1.5	0.049387	0.056885	0.056176	0.085488	0.096340	0.066142
1	1.75	0.034172	0.555655	0.038032	0.088573	0.094040	0.027932
1	2	0.027430	0.982985	0.030333	0.103455	0.095413	0.014994
2.5	**4**	0.043991	2.225489	0.047444	0.310981	0.419640	0.643763
3.25	**4**	0.072562	0.587675	0.078860	0.277142	0.538829	0.904675
4	**4**	0.110517	0.116685	0.118746	0.269783	0.636383	1.331979

Table 2: MSEs of the estimated λ with varying values of the parameters λ and ϑ using different methods to estimate the parameters of the WD

Scale (ϑ)	Shape (λ)	$\hat{\lambda}_{\mathrm{MLE}}$	$\hat{\lambda}_{\mathrm{OLS}}$	$\hat{\lambda}_{WLS}$	$\hat{\lambda}_{\mathrm{BLGE}}$	$\hat{\lambda}_{\mathrm{BLWGE}}$	$\hat{\lambda}_{\mathrm{OLSMLP}}$
1	1.5	0.348991	0.236889	0.209179	0.243708	0.199329	0.042465
1	1.75	0.457484	0.314301	0.284707	0.320195	0.263958	0.026617
1	2	0.604420	0.420835	0.376867	0.423471	0.348965	0.178389
2.5	**4**	2.429318	1.643576	1.499686	1.693472	1.387491	3.685218
3.25	**4**	2.407258	1.636167	1.479661	1.679463	1.378634	1.997234
4	**4**	2.461472	1.658105	1.496513	1.726041	1.418305	0.925600

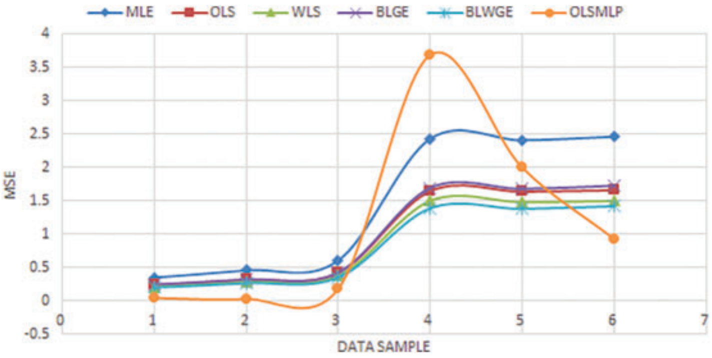

Figure 4: MSEs of $\hat{\lambda}$ with varying values of the parameters $\vartheta = [1\ 1\ 1\ 2.5\ 3.25\ 4]$ and $\lambda = [1.5\ 1.75\ 2\ 4\ 4\ 4]$ for the MLE, OLS, BLGE, WLS, and the proposed methods

From Tab. 3, we see that both statistical indicators, MSE and MAPE, yielded different values. The global rank was calculated to evaluate the best method. The results in the table indicate that the proposed method offered the best compromise between shape and scale estimation, as indicated by the global rank. Moreover, it retained the speed of the OLS and enhanced the accuracy of estimation of the parameters of the WD compared with the MLE, BLGE, and BLWGE.

Table 3: Performance evaluation of the MLE, OLS, WLS, BLGE, BLWGE, and OLSMLP methods using different statistical indicators

Parameters of WD $\vartheta = 3,\ \lambda = 8.5$ and $n = 10$													
Methods	Estimated shape ($\hat{\lambda}$)	Estimated scale ($\hat{\vartheta}$)	Statistical analysis										
			MSE		Ranking		MAPE (%)		Ranking		Global rank		Time Computation
			$\hat{\lambda}_{(.)}$	$\hat{\vartheta}_{(.)}$	$\hat{\lambda}_{(.)}$	$\hat{\vartheta}_{(.)}$	$\hat{\lambda}_{(.)}$	$\hat{\vartheta}_{(.)}$	$\hat{\lambda}_{(.)}$	$\hat{\vartheta}_{(.)}$	$\hat{\lambda}_{(.)}$	$\hat{\vartheta}_{(.)}$	
MLE	9.95111	2.98894	11.05354	0.01459097	6	1	0.27084	0.0319	6	1	6	1	2.87 ms
OLS	8.23364	3.00974	7.47628	30.15793	4	6	0.24563	0.64591	5	6	5	6	0.9 us
WLS	8.01792	3.00197	6.871836	0.01563281	3	2	0.2394	0.033138	4	2	4	2	0.98 us
BLGE	9.06909	1.8185	7.744257	1.509358	5	5	0.23293	0.39383	3	5	3	5	2.81 ms
BLWGE	8.40374	2.09433	6.369233	0.8291446	2	4	0.22206	0.30189	2	4	2	4	2.77 ms
OLSMLP	8.1827	2.95646	0.424712	0.0380619	1	3	0.061342	0.051703	1	3	1	3	17.61 us

6 Conclusion

This study proposed a method to estimate the parameters of the WD. This method is based on the OLS graphical method and the MLP neural network. The MLP solves the problems caused by the presence of outliers and eases the difficulty of determining the weights in the WLS method. It yielded acceptable results in simulations, especially in terms of shape estimation. It is also faster than the MLE, BLGE, and BLWGE.

We also proposed a second method (BLWGE), in which we introduced weight to the GE loss function. The results of simulations showed that BLWGE yields good results, especially in terms of shape estimation, compared with the other methods.

Acknowledgement: This project was supported by the Deanship of Scientific Research at Prince Sattam bin Abdulaziz University under Research Project No. 2020/01/16725.

Funding Statement: The authors are grateful to the Deanship of Scientific Research at Prince Sattam bin Abdulaziz University Supporting Project Number (2020/01/16725), Prince Sattam bin Abdulaziz University, Saudi Arabia.

Conflicts of Interest: The authors declare that they have no conflicts of interest to report regarding this study.

References

[1] M. R. Piña-Monarrez, “Weibull stress distribution for static mechanical stress and its stress/strength analysis,” *Quality and Reliability Engineering International*, vol. 34, no. 2, pp. 229–244, 2018.

[2] A. Alonge and T. Afullo, “Rainfall drop-size estimators for Weibull probability distribution using method of moments technique,” *SAIEE Africa Research Journal*, vol. 103, no. 2, pp. 83–93, 2012.

[3] F. Ashkar, I. Ba and B. B. Dieng, "Hydrological frequency analysis: Some results on discriminating between the Gumbel or Weibull probability distributions and other competing models," in *World Environmental and Water Resources Congress: Watershed Management, Irrigation and Drainage, and Water Resources Planning and Management*, Reston, VA: American Society of Civil Engineers, pp. 374–387, 2019.

[4] C. W. Yang and S. J. Jiang, "Weibull statistical analysis of strength fluctuation for failure prediction and structural durability of friction stir welded Al–Cu dissimilar joints correlated to metallurgical bonded characteristics," *Materials*, vol. 12, no. 2, pp. 205, 2019.

[5] P. K. Chaurasiya, S. Ahmed and V. Warudkar, "Study of different parameters estimation methods of Weibull distribution to determine wind power density using ground based Doppler SODAR instrument," *Alexandria Engineering Journal*, vol. 57, no. 4, pp. 2299–2311, 2018.

[6] H. H. Surendra, D. Seshachalam and K. R. Sudhindra, "Reliability analysis of solar energy resources using Weibull distribution for a standalone system in Indian context," *International Journal of Scientific Research in Mathematical and Statistical Sciences*, vol. 7, pp. 64–68, 2020.

[7] M. Bassyouni, S. A. Gutub, U. Javaid, M. Awais, S. Rehman *et al.*, "Assessment and analysis of wind power resource using Weibull parameters," *Energy Exploration & Exploitation*, vol. 33, no. 1, pp. 105–122, 2015.

[8] M. Sumair, T. Aized, S. A. R. Gardezi, S. U. Ur Rehman and S. M. S. Rehman, "Wind potential estimation and proposed energy production in Southern Punjab using Weibull probability density function and surface measured data," *Energy Exploration & Exploitation*, vol. 39, pp. 2150–2168, 2020.

[9] B. Rackauskas, M. J. Uren, T. Kachi and M. Kuball, "Reliability and lifetime estimations of GaN-on-GaN vertical pn diodes," *Microelectronics Reliability*, vol. 95, pp. 48–51, 2019.

[10] B. B. Sagar, R. K. Saket and C. G. Singh, "Exponentiated Weibull distribution approach-based inflection S-shaped software reliability growth model," *Ain Shams Engineering Journal*, vol. 7, no. 3, pp. 973–991, 2016.

[11] E. J. Tuegel, R. P. Bell, A. P. Berens, T. Brussat, J. W. Cardinal *et al.*, "Aircraft structural reliability and risk analysis handbook." *Air Force Research Lab*. Wright-Patterson Air Force Base, 2013.

[12] Q. Fu, H. Wang and X. Yan, "Evaluation of the aeroengine performance reliability based on generative adversarial networks and Weibull distribution," *Proceedings of the Institution of Mechanical Engineers, Part G: Journal of Aerospace Engineering*, vol. 233, no. 15, pp. 5717–5728, 2019.

[13] M. Sumair, T. Aized, S. A. R. Gardezi and M. Waqas Aslam, "Efficiency comparison of historical and newly developed Weibull parameters estimation methods," *Energy Exploration & Exploitation*, vol. 39, pp. 1–22, 2020.

[14] K. C. Datsiou and M. Overend, "Weibull parameter estimation and goodness-of-fit for glass strength data," *Structural Safety*, vol. 73, pp. 29–41, 2018.

[15] J. Maroco, "Consistency and efficiency of ordinary least squares, maximum likelihood, and three type II linear regression models: A monte carlo simulation study," *Methodology: European Journal of Research Methods for the Behavioral and Social Sciences*, vol. 3, no. 2, pp. 81, 2007.

[16] J. Cohen, P. Cohen, S. G. West and L. S. Aiken, in *Applied Multiple Regression/Correlation Analysis for the Behavioral Sciences*, Mahwah, N.J: Lawrence Erlbaum Associates, Routledge, New York, 2002.

[17] M. Engelhardt and L. J. Bain, "Simplified statistical procedures for the Weibull or extreme-value distribution," *Technometrics*, vol. 19, no. 3, pp. 323–331, 1977.

[18] K. Jabłońska, "Dealing with heteroskedasticity giving the example of modelling quality of life of older people," *Statistics in Transition, New Series*, vol. 19, no. 3, pp. 433–452, 2018.

[19] H. Saleh, A. E. A. Aly and S. Abdel-Hady, "Assessment of different methods used to estimate Weibull distribution parameters for wind speed in Zafarana wind farm, Suez Gulf, Egypt," *Energy*, vol. 44, no. 1, pp. 710–719, 2012.

[20] R. B. Abernethy, in *the New Weibull Handbook: Reliability and Statistical Analysis for Predicting Life, Safety, Supportability, Risk, Cost and Warranty Claims*, 5th edition, Hickory: Barringer & Associates, 2006.

[21] K. Ullah, M. Aslam and T. N. Sindhu, "Bayesian analysis of the Weibull paired comparison model using informative prior," *Alexandria Engineering Journal*, vol. 59, no. 4, pp. 2371–2378, 2020.

[22] N. Qiu, Q. Liu and Z. Zeng, "Particle swarm optimization and least squares method for geophysical parameter inversion from magnetic anomalies data," in *2010 IEEE Int. Conf. on Intelligent Computing and Intelligent Systems*, Xiamen, China, pp. 879–881, 2010.

[23] J. A. Sánchez-González and J. F. Oblitas-Cruz, "Application of Weibull analysis and artificial neural networks to predict the useful life of the vacuum-packed soft cheese," *Revista Facultad de Ingeniería Universidad de Antioquia*, vol. 82, pp. 53–59, 2017.

[24] A. Bennis, S. Mouysset and M. Serrurier, "Estimation of conditional mixture Weibull distribution with right censored data using neural network for time-to-event analysis," in *2020 Pacific-Asia Conf. on Knowledge Discovery and Data Mining*, Singapore, pp. 687–698, 2020.

[25] E. M. De Assis, C. L. S. Figueirôa Filho, G. A. D. C. Lima, L. A. N. Costa and G. M. D. O. Salles, "Machine learning and q-Weibull applied to reliability analysis in hydropower sector," *IEEE Access*, vol. 8, pp. 203331–203346, 2020.

[26] M. J. Diamantopoulou, R. Özçelik, F. Crecente-Campo and Ü. Eler, "Estimation of Weibull function parameters for modelling tree diameter distribution using least squares and artificial neural networks methods," *Biosystems Engineering*, vol. 133, pp. 33–45, 2015.

[27] T. Nakama, "Comparisons of single-and multiple-hidden-layer neural networks," in *2011 Conf. Advances in Neural Networks*, Guilin, China, vol. 6675, pp. 270–279, 2011.

[28] S. Abdulah, H. Ltaief, Y. Sun, M. G. Genton and D. E. Keyes, "Parallel approximation of the maximum likelihood estimation for the prediction of large-scale geostatistics simulations," in *2018 IEEE Conf. on Cluster Computing (CLUSTER)*, Belfast, UK, pp. 98–108, 2018.

[29] W. L. Hung and Y. C. Liu, "Estimation of Weibull parameters using a fuzzy least-squares method," *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems*, vol. 12, no. 5, pp. 701–711, 2004.

[30] S. K. Sinha and J. A. Sloan, "Bayes estimation of the parameters and reliability function of the 3-parameter Weibull distribution," *IEEE Transactions on Reliability*, vol. 37, pp. 364–369, 1988.

[31] L. M. Lye, K. P. Hapuarachchi and S. Ryan, "Bayes estimation of the extreme-value reliability function," *IEEE Transactions on Reliability*, vol. 42, no. 4, pp. 641–644, 1993.

[32] R. Calabria and G. Pulcini, "Point estimation under asymmetric loss functions for left-truncated exponential samples," *Communications in Statistics-Theory and Methods*, vol. 25, no. 3, pp. 585–600, 1996.

[33] F. N. Nwobi and C. A. Ugomma, "A comparison of methods for the estimation of Weibull distribution parameters," *Metodoloski Zvezki*, vol. 11, no. 1, pp. 65, 2014.

[34] M. Bashiri and A. Moslemi, "The analysis of residuals variation and outliers to obtain robust response surface," *Journal of Industrial Engineering International*, vol. 9, no. 1, pp. 1–10, 2013.

[35] L. F. Zhang, M. Xie and L. C. Tang, "On weighted least squares estimation for the parameters of Weibull distribution," in *Recent Advances in Reliability and Quality in Design*, London, UK: Springer, pp. 57–84, 2008.

[36] E. Hoffer, R. Banner, I. Golan and D. Soudry, "Norm matters: Efficient and accurate normalization schemes in deep networks," in *2018 32nd Conf. on Neural Information Processing Systems*, Montréal, Canada, 2018.

[37] G. Abosamara and H. Oqaibi, "An optimized deep residual network with a depth concatenated block for handwritten characters classification," *Computers Materials & Continua*, vol. 68, no. 1, pp. 1–28, 2021.

[38] J. Heaton, in *Artificial Intelligence for Humans*, 3rd edition, vol. 1, St. Louis: Charleston Createspace, 2015.

[39] K. G. Sheela and S. N. Deepa, "Review on methods to fix number of hidden neurons in neural networks," *Mathematical Problems in Engineering*, vol. 2013, pp. 1–11, 2013.

[40] J. Heaton, "The number of hidden layers," 2021, [online]. Available: https://www.heatonresearch.com/2017/06/01/hidden-layers.html [Accessed 19 April 2021].

[41] T. Szandała, "Review and comparison of commonly used activation functions for deep neural networks," in *Bio-inspired Neurocomputing*, Singapore: Springer, pp. 203–224, 2021.

[42] N. Srivastava, G. Hinton, A. Krizhevsky, L. Sutskever and R. Salakhutdinov, “Dropout: A simple way to prevent neural networks from overfitting,” *Journal of Machine Learning Research*, vol. 15, no. 1, pp. 1929–1958, 2014.

[43] D. Soydaner, “A comparison of optimization algorithms for deep learning,” *International Journal of Pattern Recognition and Artificial Intelligence*, vol. 34, no. 13, pp. 2052013, 2020.

[44] E. M. Dogo, O. J. Afolabi, N. I. Nwulu, B. Twala and C. O. Aigbavboa, “A comparative analysis of gradient descent-based optimization algorithms on convolutional neural networks,” in *2018 Conf. on Computational Techniques, Electronics and Mechanical Systems*, Belgaum, India, pp. 92–99, 2018.