Hindawi
International Journal of Mathematics and Mathematical Sciences
Volume 2020, Article ID 2412857, 13 pages
https://doi.org/10.1155/2020/2412857

Research Article

Inference of Process Capability Index C_{py} for 3-Burr-XII Distribution Based on Progressive Type-II Censoring

Rashad M. EL-Sagheer and Mustafa M. Hasaballah

Mathematics Department, Faculty of Science, Al-Azhar University, Nasr City, Cairo 11884, Egypt

Correspondence should be addressed to Rashad M. EL-Sagheer; rashadmath@yahoo.com

Received 26 June 2020; Accepted 9 September 2020; Published 5 October 2020

Academic Editor: Andrei I. Volodin

In this paper, we discussed the estimation of the index C_{py} for a 3-Burr-XII distribution based on Progressive Type-II censoring. The maximum likelihood and Bayes method have been used to obtain the estimating of the index C_{py}. The Fisher information matrix has been used to construct approximate confidence intervals. Also, bootstrap confidence intervals (CIs) of the estimators have been obtained. The Bayesian estimates for the index C_{py} have been obtained by the Markov Chain Monte Carlo method. Also, the credible intervals are constructed by using MCMC samples. Two real-datasets have been discussed using the proposed index.

1. Introduction

Statistician and quality control engineers in manufacturing industries often employ varied statistical process techniques to measure the capability of a manufacturing process and quantify the process behavior to identify contradictions between the actual process performance and the desired specifications. These techniques include the process capability index (PCI), and the PCI compares the output of the process to customer's specification. The objective of the PCI is to provide a numerical indicator of whether or not a production process is able to produce products within the specification limits. These specifications are determined through the lower specification limit (L), the upper specification limit (U), and the target value (t) The most commonly used PCIs C_p, C_{pk}, C_{pmk}, and C_{pm} are based on the assumption that a given process may be described by a normal probability model with process mean and process standard deviation. For more information, see Juran [1], Kane [2], Chan et al. [3], and Pearn et al. [4] are based on the assumption that a given process may be described by a normal probability model with process mean μ and process standard deviation σ. However, the assumption of normality is largely a simplifying assumption in different manufacturing and service processes, and often invalid. For more details, see Gunter [5]. In fact, there are several PCIs and their study for different conditions is valid for both typical and nonnormal output characteristics of processes in the literature for more information, see Clements [6], Rodriguez [7], Polansky [8], Yeh and Bhattarchya [9], and Perakis and Xekalaki [10]. In the recent past, Maiti et al. [11] have established a generalized PCI C_{py} which is directly or indirectly connected to most of the PCIs described in the literature. Furthermore, it includes both normal and nonnormal and continuous as well as discrete random variables and is defined as follows:

$$C_{py} = \frac{F(U) - F(L)}{F(UDL) - F(LDL)} = \frac{p}{p_0}, \qquad (1)$$

where $F(t) = P[X \leq t]$ is the CDF of X, U is the upper specification limit, L is the lower specification limit, LDL is the lower desirable limit, UDL is the upper desirable limit, p is the process yield, and p_0 is the desirable yield. If the process distribution is normal with $\text{LDL} = \mu - 3\sigma$ and $\text{UDL} = \mu + 3\sigma$, then the generalized PCI $C_p y$ can be written as $(p/0.9973)$. Huiming et al. [12] proposed Bayesian approach for the problem of estimation and testing PCI depending on subsamples obtained over time from an in-control process. Miao et al. [13] discussed Bayesian approach under SE loss function for computing PCIs. Wu and Lin [14] suggested one-sided lower Bayesian estimation of C_{pmk}. Recently, Kargar et al. [15] studied the Bayesian approach with normal prior depending on subsamples to check process capability via capability index C_{pk}. Maiti and Saha [16] obtained the Bayesian estimation of the index C_{py} based on SE loss function for normal, exponential, and Poisson process distributions. Mahmoud et al.[17] studied the inferences of the lifetime performance index with Lomax distribution based on progressive type-IIcensored data. Ali and Riaz [18] discussed the generalized PCIs from the Bayesian view point under symmetric and asymmetric loss functions for the simple and mixture of generalized lifetime models. Saha et al. [19] studied the classical and Bayesian inference of the index C_{py} for generalized Lindley distributed quality characteristic. The rest of this paper is organized as follows. In Section 2, we developed C_{py} for 3-Burr-XII distribution (TPBXIID). In Section 3, the maximum likelihood estimators (MLEs) of the unknown parameters of TPBXIID as well as C_{py} are studied. In Section 4, deals with approximate confidence intervals (ACIs) based on the MLEs. Bootstrap confidence intervals are discussed in Section 5. In Section 6, the MCMC techniques have been used to get the Bayes estimates and construct credible intervals (CRIs) of the index C_{py} based on squared error (SE) loss functions for the TPBXIID. Two real-datasets are analyzed to illustrative purposes in Section 7. In Section 8, Monte Carlo simulation is performed to compare the efficiency of the proposed classical estimators and Bayes estimators of the index C_{py} in terms of their MSEs. Finally, Section 9 contains conclusions.

2. The Index C_{py} for 3-Burr-XII Distribution

Burr [20] introduced the Burr XII distribution, and this distribution is popularly used in reliability analysis as a more flexible alternative to Weibull distribution, see Wingo [21, 22] and Zimmer et al. [23], and its 3-Burr XII distribution (TPBXIID) form is a generalisation of the log-logistic distribution, see Shao [24]. The TPBXIID has the following CDF:

$$F(x;\alpha,\theta,\gamma) = 1 - \left[1 + \left(\frac{x}{\alpha}\right)^{\theta}\right]^{-\gamma}, \quad x > 0, \alpha, \theta, \gamma > 0. \tag{2}$$

The PDF

$$f(x;\alpha,\theta,\gamma) = \theta\gamma\alpha^{-\theta}x^{\theta-1}\left[1 + \left(\frac{x}{\alpha}\right)^{\theta}\right]^{-(\gamma+1)}, \quad x > 0, \alpha, \theta, \gamma > 0. \tag{3}$$

Here, γ and θ are the shape parameters and α is a scale parameter. It is important to note that when $\theta = 1$, TPBXIID reduces to the Lomax distribution, when $\theta > 1$, the density function is upside-down bathtub shaped with mode at $x = \alpha[(\theta-1)/(\theta\gamma+1)]^{(1/\theta)}$ and is L-shaped when $\theta = 1$.

Substituting from (2) and (3) into (1), the index C_{py} can be written as

$$C_{py} = \frac{1}{p_0}\left[\left[1 + \left(\frac{L}{\alpha}\right)^{\theta}\right]^{-\gamma} - \left[1 + \left(\frac{U}{\alpha}\right)^{\theta}\right]^{-\gamma}\right]. \tag{4}$$

3. ML Inference

Let $X_{1:m:m}^{R_1,\ldots,R_m}, X_{2:m:m}^{R_1,\ldots,R_m}, \ldots, X_{m:m:m}^{R_1,\ldots,R_m}$ be a progressive type-II censored scheme from TPBXIID. To obtain the maximum likelihood estimators of the unknown location and scale parameters, the likelihood function is written as

$$L(\underline{x};\alpha,\theta,\gamma) = A\alpha^{-m\theta}\theta^m\gamma^m\prod_{i=1}^{m}x_i^{\theta-1}\prod_{i=1}^{m}\left[1 + \left(\frac{x_i}{\alpha}\right)^{\theta}\right]^{-(\gamma+1)} \cdot \prod_{i=1}^{m}\left[1 + \left(\frac{x_i}{\alpha}\right)^{\theta}\right]^{-\gamma(R_i)}, \tag{5}$$

where $A = n(n-1-R_1)(n-2-R_1-R_2)\ldots(n-\sum_{i=1}^{m-1}(R_1+1))$. The log-likelihood function for the 3-Burr-XII distribution

$$\ell(\alpha,\theta,\gamma) = \ln(A) + m\ln\theta - m\theta\ln\alpha + m\ln\gamma + (\theta-1)\sum_{i=1}^{m}\ln(x_i) - (\gamma+1)\sum_{i=1}^{m}\ln\left[1 + \left(\frac{x_i}{\alpha}\right)^{\theta}\right] - \gamma\sum_{i=1}^{m}R_i\ln\left[1 + \left(\frac{x_i}{\alpha}\right)^{\theta}\right]. \tag{6}$$

Taking the first derivatives of equation (6) with reference to α, θ, and γ and setting each of them equal to zero, we obtain

$$\frac{-m\widehat{\theta}}{\widehat{\alpha}} + (\widehat{\gamma}+1)\sum_{i=1}^{m}\frac{\widehat{\theta}x_i(x_i/\widehat{\alpha})^{\widehat{\theta}-1}}{\alpha^2\left[1 + (x_i/\widehat{\alpha})^{\widehat{\theta}}\right]} + \widehat{\gamma}\sum_{i=1}^{m}\frac{\widehat{\theta}R_ix_i(x_i/\widehat{\alpha})^{\widehat{\theta}-1}}{\alpha^2\left[1 + (x_i/\widehat{\alpha})^{\widehat{\theta}}\right]} = 0, \tag{7}$$

$$\frac{m}{\widehat{\theta}} - m\ln\widehat{\alpha} + \sum_{i=1}^{m}\ln x_i - (\widehat{\gamma}+1)\sum_{i=1}^{m}\frac{(x_i/\widehat{\alpha})^{\widehat{\theta}}\ln(x_i/\widehat{\alpha})}{1 + (x_i/\widehat{\alpha})^{\widehat{\theta}}} - \widehat{\gamma}\sum_{i=1}^{m}\frac{(x_i/\widehat{\alpha})^{\widehat{\theta}}\ln(x_i/\widehat{\alpha})}{1 + (x_i/\widehat{\alpha})^{\widehat{\theta}}} = 0, \tag{8}$$

$$\frac{m}{\widehat{\gamma}} - \sum_{i=1}^{m}\ln\left[1 + \left(\frac{x_i}{\widehat{\alpha}}\right)^{\widehat{\theta}}\right] - \sum_{i=1}^{m}R_i\ln\left[1 + \left(\frac{x_i}{\widehat{\alpha}}\right)^{\widehat{\theta}}\right] = 0. \tag{9}$$

From (6), we obtain the MLE $\widehat{\gamma}$ as

$$\widehat{\gamma} = \left[\sum_{i=1}^{m}\ln\left[1 + \left(\frac{x_i}{\widehat{\alpha}}\right)^{\widehat{\theta}}\right] + \sum_{i=1}^{m}R_i\ln\left[1 + \left(\frac{x_i}{\widehat{\alpha}}\right)^{\widehat{\theta}}\right]\right]^{-1}. \tag{10}$$

Since it is difficult to express equations (7) and (8) in closed forms, the Newton–Raphson iteration process was used to generate the estimates. For more information, see EL-Sagheer [25]. In addition, after replacing α, θ, and γ by their MLEs $\widehat{\alpha}$, $\widehat{\theta}$, and $\widehat{\gamma}$, we can get the estimator of C_{py} as follows:

$$\widehat{C}_{\mathrm{py}} = \frac{1}{p_0}\left[\left[1+\left(\frac{L}{\widehat{\alpha}}\right)^{\widehat{\theta}}\right]^{-\widehat{\gamma}} - \left[1+\left(\frac{U}{\widehat{\alpha}}\right)^{\widehat{\theta}}\right]^{-\widehat{\gamma}}\right]. \tag{11}$$

3.1. Approximate Confidence Interval. The asymptotic variance-covariance of the MLEs for parameters α, θ, and γ are given by elements of the negative of the Fisher information matrix are defined as follows:

$$I_{ij} = -E\left(\frac{\partial^2 \ell}{\partial\psi_i \partial\psi_j}\right), \tag{12}$$

where $i, j = 1, 2, 3$ and $(\psi_1, \psi_2, \psi_3) = (\alpha, \theta, \gamma)$.

However, the exact mathematical expressions for the above expectations are very hard to obtain. Hence, the asymptotic variance-covariance matrix is obtained as follows:

$$\begin{aligned} I^{-1}(\alpha,\theta,\gamma) &= \begin{pmatrix} -\dfrac{\partial^2\ell}{\partial\alpha^2} & -\dfrac{\partial^2\ell}{\partial\,\partial\theta} & -\dfrac{\partial^2\ell}{\partial\alpha\partial\gamma} \\ -\dfrac{\partial^2\ell}{\partial\theta\,\partial\alpha} & -\dfrac{\partial^2\ell}{\partial\theta^2} & -\dfrac{\partial^2\ell}{\partial\theta\partial\gamma} \\ -\dfrac{\partial^2\ell}{\partial\gamma\partial\alpha} & -\dfrac{\partial^2\ell}{\partial\gamma\partial\theta} & -\dfrac{\partial^2\ell}{\partial\gamma^2} \end{pmatrix}^{-1}_{\downarrow(\alpha,\theta,\gamma)=(\widehat{\alpha},\widehat{\theta},\widehat{\gamma})} \\ &= \begin{pmatrix} \widehat{\mathrm{var}}(\widehat{\alpha}) & \mathrm{cov}(\widehat{\alpha},\widehat{\theta}) & \mathrm{cov}(\widehat{\alpha},\widehat{\gamma}) \\ \mathrm{cov}(\widehat{\theta},\widehat{\alpha}) & \widehat{\mathrm{var}}(\widehat{\theta}) & \mathrm{cov}(\widehat{\theta},\widehat{\gamma}) \\ \mathrm{cov}(\widehat{\gamma},\widehat{\alpha}) & \mathrm{cov}(\widehat{\gamma},\widehat{\theta}) & \widehat{\mathrm{var}}(\widehat{\gamma}) \end{pmatrix}, \end{aligned} \tag{13}$$

with

$$\begin{aligned}
\frac{\partial^2\ell}{\partial\alpha^2} &= \frac{m\theta}{\alpha^2} + (\gamma+1)\sum_{i=1}^{m}\frac{\theta(x_i/\alpha)^{\theta}\left[-\alpha^2(\theta-1)\left[1+(x_i/\alpha)^{\theta}\right] - 2\alpha x_i(x_i/\alpha)^{-1}\left[1+(x_i/\alpha)^{\theta}\right] + \theta x_i^2(x_i/\alpha)^{\theta-2}\right]}{\left[\alpha^2\left[1+(x_i/\alpha)^{\theta}\right]\right]^2} \\
&\quad + \gamma\sum_{i=1}^{m}\frac{R_i\theta(x_i/\alpha)^{\theta}\left[-\alpha^2(\theta-1)\left[1+(x_i/\alpha)^{\theta}\right] - 2\alpha x_i(x_i/\alpha)^{-1}\left[1+(x_i/\alpha)^{\theta}\right] + \theta x_i^2(x_i/\alpha)^{\theta-2}\right]}{\left[\alpha^2\left[1+(x_i/\alpha)^{\theta}\right]\right]^2}, \\
\frac{\partial^2\ell}{\partial\theta^2} &= \frac{-m}{\theta^2} - (\gamma+1)\sum_{i=1}^{m}\frac{(x_i/\alpha)^{\theta}(\ln[x_i/\alpha])^2}{\left[1+(x_i/\alpha)^{\theta}\right]^2} - \gamma\sum_{i=1}^{m}\frac{R_i(x_i/\alpha)^{\theta}(\ln[x_i/\alpha])^2}{\left[1+(x_i/\alpha)^{\theta}\right]^2}, \\
\frac{\partial^2\ell}{\partial\gamma^2} &= \frac{-m}{\gamma^2}, \\
\frac{\partial^2\ell}{\partial\alpha\,\partial\theta} = \frac{\partial^2\ell}{\partial\theta\,\partial\alpha} &= \frac{-m}{\alpha} + (\gamma+1)\sum_{i=1}^{m}\frac{(1/\alpha)(x_i/\alpha)^{\theta}\left[\ln[x_i/\alpha] + (x_i/\alpha)^{\theta} + 1\right]}{\left[1+(x_i/\alpha)^{\theta}\right]^2} + \gamma\sum_{i=1}^{m}\frac{R_i(1/\alpha)(x_i/\alpha)^{\theta}\left[\ln[x_i/\alpha] + (x_i/\alpha)^{\theta} + 1\right]}{\left[1+(x_i/\alpha)^{\theta}\right]^2}, \\
\frac{\partial^2\ell}{\partial\gamma\,\partial\alpha} = \frac{\partial^2\ell}{\partial\alpha\,\partial\gamma} &= \sum_{i=1}^{m}\frac{\theta x_i(x_i/\alpha)^{\theta-1}}{\alpha^2\left[1+(x_i/\alpha)^{\widehat{\theta}}\right]} + \sum_{i=1}^{m}\frac{R_i\theta x_i(x_i/\alpha)^{\theta-1}}{\alpha^2\left[1+(x_i/\alpha)^{\widehat{\theta}}\right]}, \\
\frac{\partial^2\ell}{\partial\gamma\,\partial\theta} = \frac{\partial^2\ell}{\partial\theta\,\partial\gamma} &= -\sum_{i=1}^{m}\frac{(x_i/\alpha)^{\theta}\ln(x_i/\alpha)}{1+(x_i/\alpha)^{\theta}} - \sum_{i=1}^{m}\frac{R_i(x_i/\alpha)^{\theta}\ln(x_i/\alpha)}{1+(x_i/\alpha)^{\theta}}.
\end{aligned} \tag{14}$$

Then, $(1-\eta)100\%$ CIs for parameters α, θ, and γ are, respectively, given as

$$\left(\widehat{\alpha} \pm Z_{(\eta/2)}\sqrt{\widehat{\operatorname{var}}(\widehat{\alpha})}\right), \left(\widehat{\theta} \pm Z_{\eta/2}\sqrt{\widehat{\operatorname{var}}(\widehat{\theta})}\right) \text{ and } \cdot \left(\widehat{\gamma} \pm Z\gamma_{(\eta/2)}\sqrt{\widehat{\operatorname{var}}(\widehat{\gamma})}\right), \quad (15)$$

where $Z_{(\eta/2)}$ is the percentile of the standard normal distribution with right-tail probability $(\eta/2)$. Furthermore, to construct the asymptotic confidence interval of the C_{py}, which is function of the parameters α, θ, and γ, we need to find the variances of it. In order to find the approximate estimates of the variance of $\widehat{C}_{\text{py}}$, we use the delta method referred to in Green [26] to compute ACIs for C_{py}. Based on this method, the variance of $\widehat{C}_{\text{py}}$ can be approximated by $\widehat{\sigma}^2_{\widehat{C}_{\text{py}}} = [\nabla\widehat{C}_{\text{py}}]^T[\widehat{V}][\nabla\widehat{C}_{\text{py}}]$, where $\nabla\widehat{C}_{\text{py}}$ is the gradient of $\widehat{C}_{\text{py}}$ with respect to α, θ, and γ and $\widehat{V} = I^{-1}(\alpha,\theta,\gamma)$. Thus, the $(1-\eta)100\%$ ACIs for C_{py} can be given by $(\widehat{C}_{\text{py}} \pm Z_{(\eta/2)}\sqrt{\widehat{\sigma}^2_{\widehat{C}_{\text{py}}}})$.

4. Bootstrap Confidence Intervals

In this section, we propose two confidence intervals' dependent bootstrapping. The two methods of bootstrap which are commonly used in practice are as follows:

(1) The percentile bootstrap (Boot-p) proposed by Efron [27]

(2) The bootstrap-t method (Boot-t) proposed by Hall [28]

4.1. Boot-p Method

(1) Depending on the original sample $\underline{x} = (x_1, x_2, \ldots, x_n)$, compute the MLEs of the parameters α, θ, and γ from equations (7)–(8) and (10).

(2) Using the values of $\widehat{\alpha}$, $\widehat{\theta}$ and $\widehat{\gamma}$ to generate a bootstrap sample X^* with the same values of R_i, $i = 1, 2, \ldots, r$, using algorithm presented in Balakrishnan and Sandhu [29].

(3) Get a bootstrap sample $\underline{x^*} = (x_1^*, x_2^*, \ldots, x_n^*)$ by resampling with replacement.

(4) As in Step 1, based on $\underline{x^*}$, compute the bootstrap sample estimates of $\widehat{\varphi}$, where $\widehat{\varphi} = [\widehat{\alpha}, \widehat{\theta}, \widehat{\gamma}, \widehat{C}_{\text{py}}]$, say $\widehat{\varphi}^* = [\widehat{\alpha}^*, \widehat{\theta}^*, \widehat{\gamma}^*, \widehat{C}_{\text{py}}{}^*]$.

(5) Repeat Steps 3 and 4 N Boot times, and obtain $\widehat{\varphi}_1^*, \widehat{\varphi}_2^*, \ldots, \widehat{\varphi}^*_{\text{NBoot}}$.

(6) Arrange $\widehat{\varphi}_i^*$, $i = 1, 2, \ldots, N$ Boot in an ascending order to obtain the bootstrap sample $(\widehat{\varphi}^*_{(1)}, \widehat{\varphi}^*_{(2)}, \ldots, \widehat{\varphi}^*_{(\text{NBoot})})$.

(7) Let $G_1(z) = p(\widehat{\varphi}_i^* \le z)$ be the cdf of $\widehat{\varphi}_i^*$. Define $\widehat{\varphi}^*_{\text{iBoot}} = G_1^{-1}(z)$ for given z. The approximate Boot-p $100(1-\eta)\%$ CI of $\widehat{\varphi}$ is given by $[\varphi^*_{\text{iBoot}-p}(\eta/2), \varphi^*_{\text{iBoot}-p}(1-(\eta/2))]$.

4.2. Boot-t Method

(1) From (1) to (4) is the same steps in Boot-p.

(2) Compute the $T^{*\varphi}$ statistic defined as $T^{*\varphi} = (\sqrt{N}(\widehat{\varphi}^* - \widehat{\varphi})/\sqrt{\widehat{\operatorname{var}}(\widehat{\varphi}^*)})$, where $\operatorname{var}(\widehat{\varphi}^*)$ are obtained by using Fisher information matrix.

(3) Repeat Step 1 and 2 N Boot times and obtain $T_1^{*\varphi}, T_2^{*\varphi}, \ldots, T^{*\varphi}_{\text{NBoot}}$.

(4) Arrange $T_1^{*\varphi}, T_2^{*\varphi}, \ldots, T^{*\varphi}_{\text{NBoot}}$, in an ascending orders and obtain the ordered sequences $(T^{*\varphi}_{(1)}, T^{*\varphi}_{(2)}, \ldots, T^{*\varphi}_{(\text{NBoot})})$.

(5) Let $G_2(z) = p(T^* \le z)$ be the cdf of T^*. For a given z, define $\widehat{\varphi}_{\text{Boot}-t}(z) = \widehat{\varphi} + N^{-(1/2)}\sqrt{\widehat{\operatorname{var}}(\widehat{\varphi}^*)}G_2^{-1}(z)$. Then, the approximate Boot-t $100(1-\eta)\%$ CI of $\widehat{\varphi}$ is given by $[\widehat{\varphi}_{\text{Boot}-t}(\eta/2), \widehat{\varphi}_{\text{Boot}-t}(1-(\eta/2))]$.

5. Bayes Estimation

In this section, we present the posterior densities of the parameters α, θ, and γ based on progressive type-II censored data and then obtain the corresponding Bayes estimates of these parameters. In order to obtain the joint posterior density of α, θ, and γ, we suppose that α, θ, and γ are independently distributed as gamma (a_1, b_1), gamma (a_2, b_2), and gamma (a_3, b_3) priors, respectively. Consequently, the prior density functions of α, θ, and γ becomes

$$\begin{aligned}
\pi_1(\alpha) &\propto \alpha^{a_1-1}e^{-b_1\alpha}, \quad \alpha > 0,\\
\pi_2(\theta) &\propto \theta^{a_2-1}e^{-b_2\theta}, \quad \theta > 0,\\
\pi_3(\gamma) &\propto \gamma^{a_3-1}e^{-b_3\gamma}, \quad \gamma > 0,
\end{aligned} \quad (16)$$

where all the hyperparameters a_i and b_i, where $i = 1, 2, 3$, are chosen to reflect prior knowledge about α, θ, and γ. The joint prior distribution for α, θ, and γ is

$$\pi(\alpha,\theta,\gamma) \propto \alpha^{a_1-1}\theta^{a_2-1}\gamma^{a_3-1}e^{-b_1\alpha-b_2\theta-b_3\gamma}. \quad (17)$$

The posterior distribution of the parameters α, θ, and γ up to proportionality can be obtained by combining the likelihood function (5) with the joint prior (17) via Bayes' theorem, and it can be written as

$$\pi^*(\alpha,\theta,\gamma\mid\underline{x}) = \frac{L(\alpha,\theta,\gamma\mid\underline{x})\pi(\alpha,\theta,\gamma)}{\int_0^\infty\int_0^\infty\int_0^\infty L(\alpha,\theta,\gamma\mid\underline{x})\pi(\alpha,\theta,\gamma)\mathrm{d}\alpha\,\mathrm{d}\theta\mathrm{d}\gamma}$$
$$\propto \alpha^{a_1-m\theta-1}\theta^{a_2+m-1}\gamma^{a_3+m-1}e^{-\theta\left\{b_2-\sum_{i=1}^m\ln(x_i)\right\}}e^{-\gamma\left\{b_3+\sum_{i=1}^m R_i\ln\left[1+(x_i/\alpha)^\theta\right]\right\}}\times e^{-b_1\alpha-(\gamma+1)\sum_{i=1}^m\ln\left[1+(x_i/\alpha)^\theta\right]}. \tag{18}$$

From equation (18), it may be observed that explicit forms for the marginal posterior distributions for each parameter are difficult to obtain. For this reason, we assume to use MCMC approximation method to produce samples from the joint posterior density function in (18) and to use these samples to calculate the Bayes estimate of α, θ, and γ and any function of them such as C_{py} as well as to construct associated credible intervals. We consider the Gibbs within Metropolis sampler to implement the MCMC technique, which requires derivation of the complete set of conditional posterior distribution. A lot of papers dealt with MCMC technique such as Chen and Shao [30] and EL-Sagheer [25]. It can be shown that the conditional posterior density function of α, θ, and γ can be written, up to proportionality, as follows:

$$\pi_1^*(\alpha\mid\theta,\gamma,\underline{x}) \propto \alpha^{a_1-m\theta-1}e^{-b_1\alpha-\sum_{i=1}^m\ln\left[1+(x_i/\alpha)^\theta\right]}, \tag{19}$$

$$\pi_2^*(\theta\mid\alpha,\gamma,\underline{x}) \propto \theta^{a_2+m-1}\alpha^{a_1-m\theta-1}e^{-\theta\left\{b_2-\sum_{i=1}^m\ln(x_i)\right\}}e^{-\sum_{i=1}^m\ln\left[1+(x_i/\alpha)^\theta\right]}, \tag{20}$$

$$\pi_3^*(\gamma\mid\alpha,\theta,\underline{x}) \propto \gamma^{a_3+m-1}e^{-\gamma\left\{b_3+\sum_{i=1}^m R_i\ln\left[1+(x_i/\alpha)^\theta\right]\sum_{i=1}^m\ln\left[1+(x_i/\alpha)^\theta\right]\right\}}. \tag{21}$$

In this representation, the full conditional forms given in (21) is gamma density with parameter of shape (a_3+m) and parameter of scale $\left\{b_3+\sum_{i=1}^m R_i\ln[1+(x_i/\alpha)^\theta]+\sum_{i=1}^m\ln[1+(x_i/\alpha)^\theta]\right\}$. So, samples of γ can be easily generated using any gamma-generating routine. In addition, since the conditional posteriors of α and θ in (19) and (20), respectively, do not give standard forms, and therefore Gibbs sampling is not a straightforward choice, and it is appropriate to use the Metropolis–Hastings sampler to implement MCMC technique, see Metropolis et al. [31]. Because of these conditional distributions in (19) and (20), the following is a hybrid algorithm with Gibbs sampling steps to update parameter γ and Metropolis–Hastings sampler steps to update α and θ.

5.1. Metropolis-Hastings Algorithm

(1) Start with initial guess of α, θ, and γ, say $\alpha^{(0)}$, $\theta^{(0)}$, and $\gamma^{(0)}$, respectively, $M=$ burn-in.

(2) Set $j=1$.

(3) Generate $\gamma^{(j)}$ from Gamma $\left\{m+a_3, b_3+\sum_{i=1}^m R_i\ln[1+(x_i/\alpha^{(j-1)})^{\theta(j-1)}]+\sum_{i=1}^m\ln[1+(x_i/\alpha^{(j-1)})^{\theta(j-1)}]\right\}$.

(4) Using Metropolis–Hastings, generate $\alpha^{(j)}$ and $\theta^{(j)}$ from $\pi_1^*(\alpha^{(j-1)}\mid\theta^{(j-1)},\gamma,\underline{x})$ and $\pi_2^*(\theta^{(j-1)}\mid\alpha,\gamma,\underline{x})$ with normal proposal distribution, $N(\alpha^{(j-1)},\mathrm{var}(\alpha))$ and $N(\theta^{(j-1)},\mathrm{var}(\theta))$, where $\mathrm{var}(\alpha)$ and $\mathrm{var}(\theta)$ are obtained from the variance-covariance matrix.

(i) Calculate the acceptance probability:

$$r_1 = \min\left[1, \frac{\pi_1^*\left(\alpha^*\mid\theta^{(j-1)},\gamma^{(j)},\underline{x}\right)}{\pi_1^*\left(\alpha^{(j-1)}\mid\theta^{(j-1)},\gamma^{(j)},\underline{x}\right)}\right],$$
$$r_2 = \min\left[1, \frac{\pi_2^*\left(\theta^*\mid\alpha^{(j)},\gamma^{(j)},\underline{x}\right)}{\pi_2^*\left(\theta^{(j-1)}\mid\alpha^{(j)},\gamma^{(j)},\underline{x}\right)}\right]. \tag{22}$$

(ii) Generate u_1 and u_2 from a uniform $(0,1)$ distribution.

(iii) If $u_1\le r_1$, accept the proposal and set $\alpha^{(i)}=\alpha^*$, else set $\alpha^{(i)}=\alpha^{(i-1)}$.

(iv) If $u_2\le r_2$, accept the proposal and set $\theta^{(i)}=\theta^*$, else set $\theta^{(i)}=\theta^{(i-1)}$.

(5) Calculate C_{py} as

$$\widehat{C}_{\mathrm{py}}^{(i)} = \frac{1}{p_0}\left[\left[1+\left(\frac{L}{\widehat{\alpha}^{(i)}}\right)^{\widehat{\theta}^{(i)}}\right]^{-\widehat{\gamma}^{(i)}} - \left[1+\left(\frac{U}{\widehat{\alpha}^{(i)}}\right)^{\widehat{\theta}^{(i)}}\right]^{-\widehat{\gamma}^{(i)}}\right]. \tag{23}$$

(6) Set $j = j + 1$.

(7) Repeat Steps $3-6$ N times and obtain $\alpha^{(i)}, \theta^{(i)}, \gamma^{(i)}$, and $C_{\mathrm{py}}^{(i)}$, $i = 1, \ldots, N$. In order to guarantee the convergence and to remove the affection of selecting of initial values, the first M simulated varieties are discarded. Then, the chosen samples are $\alpha^{(j)}, \theta^{(j)}, \gamma^{(j)}$, and $C_{\mathrm{py}}^{(j)}$, $j = M+1, \ldots, N$, for sufficiently large N forms an approximate posterior samples which can be used to develop the Bayesian inferences. The approximate Bayes estimate of C_{py} under SE loss function is given by

$$\widehat{C}_{\mathrm{py}}^{\mathrm{MCMC}} = E\left[C_{\mathrm{py}} \mid \underline{x}\right] = \frac{1}{N-M}\sum_{i=M+1}^{N} C_{\mathrm{py}}^{(i)}. \tag{24}$$

(8) To calculate the CRIs of C_{py}, order $C_{\mathrm{py}}^{(i)}$, $i = 1, \ldots, N$, as $(C_{\mathrm{py}}^{(1)} < \ldots < C_{\mathrm{py}}^{(N-M)})$. Then, the $100(1-\eta)\%$ CRIs of C_{py} become $(C_{\mathrm{py}(N-M)(\eta/2)}, C_{\mathrm{py}(N-M)(1-(\eta/2))})$.

6. Applications to Real Life Data

In this section, we present two examples to illustrate the computations of the methods proposed in this article using two different real-datasets.

Dataset I. We chose the real-dataset from Leiva et al. [32], and we added (2) to this data, the quality characteristic in this dataset is ball size (in millimeters) and the process has been monitored with USL and LSL for this quality characteristic is $L = 0.80$ mil and $U = 10.0$ mil $(1 \cdot \mathrm{mil} = (1/1000)\,\mathrm{in} = 0.00254\,\mathrm{mm})$, respectively. The data are given as follows:

2.619	2.665	2.68	2.889	2.921	2.923	2.94	3.05	3.066	3.083	3.102	3.131	3.175
3.295	3.301	3.311	3.38	3.383	3.508	3.516	3.59	3.6	3.618	3.636	3.694	3.694
3.77	3.811	3.842	3.845	3.954	3.982	3.992	4.093	4.111	4.207	4.227	4.26	4.26
4.312	4.333	4.336	4.366	4.381	4.413	4.472	4.495	4.573	4.582	4.752	4.788	4.797
4.803	4.89	4.891	4.891	4.892	4.899	5.055	5.158	5.177	5.222	5.234	5.334	5.36
5.492	5.508	5.591	5.696	5.722	5.75	5.782	5.842	5.982	6.035	6.061	6.101	6.281
6.417	6.419	6.489	6.494	6.495	6.542	6.62	6.805	6.882	7.228	7.48	7.639	7.706
7.896	7.954	8.123	8.221	8.229	8.549	8.644	8.659	11.725				

Data set II. In this set of data, the first failure times (in months) of 20 electric carts are used in a large manufacturing facility for internal transport and distribution. Here, we have set the hypothetical LSL and the hypothetical USL, respectively, are $L = 0.60$ and $U = 5.00$ and the details are given in Zimmer et al. [33]. The data are as follows:

0.9	1.5	2.3	3.2	3.9	5.0	6.2	7.5	8.3	10.4	11.1	12.6	15.0	16.3
19.3	22.6	24.8	31.1	38.1	53.0								

We used Kolmogorov–Smirnov (K-S) test to fit whether the data distribution as TPBXIID or not. The calculated value of the K-S test of dataset I and dataset II are 0.0517172 and 0.0527746, respectively, for the TPBXIID and these values are smaller than their corresponding values expected at 5% significance level which is 0.13403 and P value equal 0.939171 at $n = 100$ and 0.29408 and P value equal 1.0 at $n = 20$. So, it can be observed that the TPBXIID fits these data very well and also we have just plotted the empirical $S(t)$ and the fitted $S(t)$ for dataset I and dataset II in Figures 1 and 2, respectively. Note that the TPBXIID can be a good fitting model for these data. According to the dataset I presented by Leiva et al. [32], we can generate the progressive type-II censored scheme sample of size $r = 20$ taken from sample size $n = 100$ with censoring scheme $R = (0_{(19)}, 80)$. A progressive type-II censored scheme sample generated from the real-dataset I is given as follows.

2.619	2.665	2.68	3.301	3.311	3.38	3.516	3.59	3.992	4.093	4.207	4.381
4.582	4.752	4.788	4.891	5.334	5.508	5.591	5.75.				

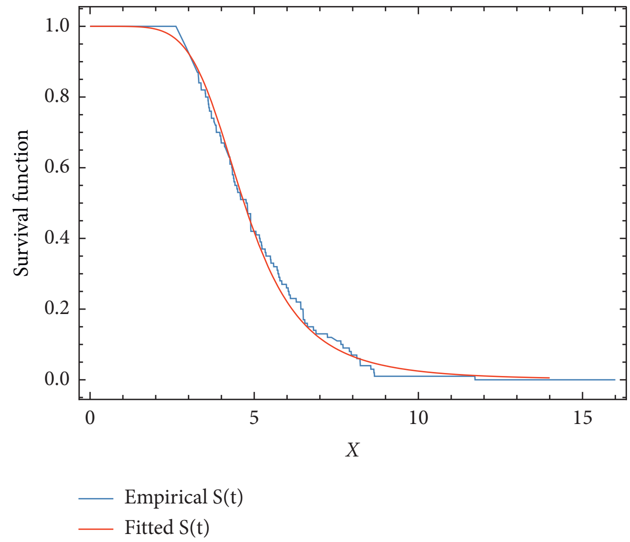

Figure 1: Empirical and fitted survival functions.

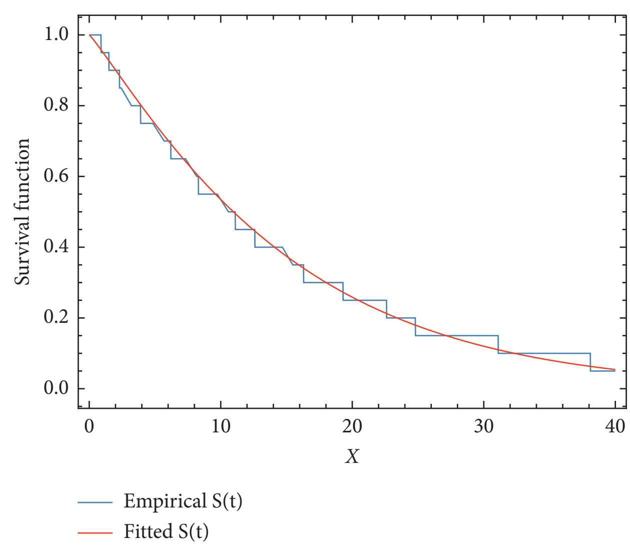

Figure 2: Empirical and fitted survival functions.

Table 1: Descriptive statistics for the considered datasets.

Data	n	Minimum	Q_1	Median	Q_3	Maximum	Mean	SD	Kurtosis	Skewness
I	100	2.619	3.694	4.77	6.048	11.725	5.03589	1.71543	4.13006	0.99858
II	20	0.9	4.45	10.75	20.95	53.0	14.655	13.638	4.30858	1.35378

Table 2: Different point estimates of C_{py}.

Dataset	$C_{py\,(Exact)}$	Classical estimates of C_{py}			Bayes estimates of C_{py}
		MLE	Boot-p	Boot-t	MCMC
I	0.4441	0.3755	0.4249	0.2966	0.3748
II	0.1820	0.1655	0.1803	0.1413	0.1615

Table 3: 95% CIs/CRIs of C_{py}.

Dataset	MLE			Boot-p			Boot-t			MCMC		
	Lower	Upper	Length	Lower	Upper	Length	Lower	Upper	Length	Lower	Upper	Length
I	0.2053	0.5456	0.3403	0.3369	0.8527	0.5158	0.0336	0.4282	0.3946	0.2404	0.5115	0.2711
II	−0.0055	0.3365	0.342	0.0561	0.3391	0.2830	0.0041	0.3645	0.3604	0.0581	0.2890	0.2309

Table 4: MCMC results of C_{py} for dataset I and dataset II.

Data	n	Mean	Median	Mode	SD	Skewness
I	100	0.3748	0.3698	0.3578	0.0664	0.1688
II	20	0.1615	0.1470	0.1368	0.0625	0.7295

Table 5: True value of C_{py} and its classical and the Bayes estimates using different methods of estimation along with their MSEs (in parentheses) when $\alpha = 7.0, \theta = 4.0$, and $\gamma = 0.50$.

n	m	$(R_1, \cdots, R_m)$	C_{py}	Classical estimates of C_{py}			Bayes estimates of C_{py}	
				MLE	Boot-p	Boot-t	Prior-0	Prior-I
20	10	$(0_{(9)}, 10)$	0.2043	0.2942	0.5139	0.2854	0.2871	0.1548
				(0.0081)	(0.0066)	(0.0959)	(0.0065)	(0.0024)
		$(10, 0_{(9)})$	0.2043	0.1241	0.1598	0.1269	0.1233	0.1851
				(0.0064)	(0.0060)	(0.0020)	(0.0016)	(0.0004)
20	15	$(0_{(14)}, 5)$	0.2043	0.2781	0.2881	0.2581	0.2737	0.1770
				(0.0054)	(0.0127)	(0.0117)	(0.0048)	(0.0007)
		$(5, 0_{(14)})$	0.2043	0.1319	0.1390	0.1290	0.1283	0.2177
				(0.0052)	(0.0054)	(0.0052)	(0.0050)	(0.0002)
40	15	$(0_{(14)}, 25)$	0.2043	0.1967	0.1955	0.1920	0.1953	0.2376
				(0.0001)	(0.0002)	(0.0001)	(0.0001)	(0.0001)
		$(25, 0_{(14)})$	0.2043	0.2804	0.2469	0.2678	0.2769	0.1705
				(0.0058)	(0.0056)	(0.0055)	(0.0054)	(0.0011)
40	30	$(0_{(29)}, 10)$	0.2043	0.2059	0.2141	0.2140	0.2044	0.2975
				(0.0003)	(0.0002)	(0.0001)	(0.0001)	(0.0001)
		$(10, 0_{(29)})$	0.2043	0.2984	0.2136	0.2993	0.2962	0.2184
				(0.0089)	(0.0087)	(0.0080)	(0.0084)	(0.0002)
60	30	$(0_{(29)}, 30)$	0.2043	0.1785	0.1780	0.1770	0.1782	0.1920
				(0.0007)	(0.0006)	(0.0005)	(0.0004)	(0.0002)
		$(30, 0_{(29)})$	0.2043	0.1609	0.1663	0.1661	0.1608	0.2234
				(0.0009)	(0.0007)	(0.0006)	(0.0004)	(0.0002)
60	40	$(0_{(39)}, 20)$	0.2043	0.2377	0.2322	0.2327	0.2373	0.1821
				(0.0006)	(0.0005)	(0.0004)	(0.0003)	(0.0002)
		$(20, 0_{(39)})$	0.2043	0.2239	0.1704	0.2228	0.2237	0.2465
				(0.0002)	(0.0002)	(0.0002)	(0.0001)	(0.0001)
80	40	$(0_{(39)}, 40)$	0.2043	0.1645	0.1635	0.1597	0.1638	0.1772
				(0.0008)	(0.0007)	(0.0005)	(0.0004)	(0.0002)
		$(40, 0_{(39)})$	0.2043	0.2000	0.1990	0.1989	0.1983	0.2800
				(0.0038)	(0.0030)	(0.0028)	(0.0018)	(0.0015)
80	50	$(0_{(49)}, 30)$	0.2043	0.2331	0.2311	0.2312	0.2324	0.2606
				(0.0004)	(0.0004)	(0.0003)	(0.0002)	(0.0001)
		$(30, 0_{(49)})$	0.2043	0.1938	0.1924	0.1880	0.1935	0.2257
				(0.0003)	(0.0002)	(0.0001)	(0.0001)	(0.0001)
100	60	$(0_{(59)}, 40)$	0.2043	0.2304	0.2335	0.2346	0.2297	0.1756
				(0.0007)	(0.0005)	(0.0004)	(0.0002)	(0.0001)
		$(40, 0_{(59)})$	0.2043	0.2653	0.1915	0.2687	0.2649	0.2370
				(0.0037)	(0.0035)	(0.0033)	(0.0031)	(0.0011)
100	80	$(0_{(79)}, 20)$	0.2043	0.1700	0.1695	0.1739	0.1694	0.1898
				(0.0012)	(0.0011)	(0.0010)	(0.0009)	(0.0002)
		$(20, 0_{(79)})$	0.2043	0.2147	0.2080	0.2173	0.2141	0.2523
				(0.0006)	(0.0005)	(0.0004)	(0.0003)	(0.0001)

TABLE 6: True value of C_{py} and its classical and the Bayes estimates using different methods of estimation along with their MSEs (in parentheses) when $\alpha = 4.40, \theta = 5.79$, and $\gamma = 0.77$.

n	m	$(R_1, \cdots, R_m)$	C_{py}	Classical estimates of C_{py}			Bayes estimates of C_{py}	
				MLE	Boot-p	Boot-t	Prior-0	Prior-I
20	10	$(0_{(9)}, 10)$	0.8180	0.9869	0.9920	0.9922	0.9602	0.8216
				(0.0285)	(0.0275)	(0.0265)	(0.0202)	(0.0200)
		$(10, 0_{(9)})$	0.8180	0.9028	0.8365	0.8365	0.8784	0.8418
				(0.0072)	(0.0065)	(0.0060)	(0.0036)	(0.0006)
20	15	$(0_{(14)}, 5)$	0.8180	0.9806	0.9554	0.9905	0.9644	0.7805
				(0.0264)	(0.0250)	(0.0241)	(0.0214)	(0.0014)
		$(5, 0_{(14)})$	0.8180	0.9368	0.9262	0.9262	0.9200	0.9306
				(0.0141)	(0.0117)	(0.0112)	(0.0104)	(0.0101)
40	15	$(0_{(14)}, 25)$	0.8180	0.8236	0.8365	0.8382	0.8110	0.8348
				(0.0009)	(0.0007)	(0.0006)	(0.0003)	(0.0001)
		$(25, 0_{(14)})$	0.8180	0.8208	0.8030	0.8035	0.8022	0.7999
				(0.0007)	(0.0006)	(0.0004)	(0.0002)	(0.0001)
40	30	$(0_{(29)}, 10)$	0.8180	0.9870	0.9812	0.9795	0.9575	0.8167
				(0.0228)	(0.0222)	(0.0218)	(0.0195)	0.0002
		$(10, 0_{(29)})$	0.8180	0.8870	0.8750	0.8751	0.8791	0.9731
				(0.0048)	(0.0043)	(0.0040)	(0.0037)	(0.0027)
60	30	$(0_{(29)}, 30)$	0.8180	0.8922	0.9100	0.9129	0.8813	0.7453
				(0.0055)	(0.0045)	(0.0042)	(0.0040)	(0.0053)
		$(30, 0_{(29)})$	0.8180	0.8873	0.8860	0.8862	0.8794	0.8321
				(0.0024)	(0.0023)	(0.0020)	(0.0019)	(0.0002)
60	40	$(0_{(39)}, 20)$	0.8180	0.8161	0.8255	0.8252	0.8102	0.8248
				(0.0006)	(0.0005)	(0.0003)	(0.0002)	(0.0001)
		$(20, 0_{(39)})$	0.8180	0.7428	0.7360	0.7358	0.7359	0.8374
				(0.0028)	(0.0024)	(0.0021)	(0.0018)	(0.0002)
80	40	$(0_{(39)}, 40)$	0.8180	0.9559	0.9599	0.9598	0.9500	0.8430
				(0.0095)	(0.0092)	(0.0089)	(0.0087)	(0.0003)
		$(40, 0_{(39)})$	0.8180	0.7865	0.7768	0.7760	0.7829	0.7459
				(0.0005)	(0.0004)	(0.0003)	(0.0002)	(0.0001)
80	50	$(0_{(49)}, 30)$	0.8180	0.8543	0.8744	0.8733	0.8469	0.8165
				(0.0013)	(0.0011)	(0.0010)	(0.0008)	(0.0005)
		$(30, 0_{(49)})$	0.8180	0.8848	0.8788	0.8784	0.8803	0.8433
				(0.0022)	(0.0020)	(0.0017)	(0.0014)	(0.0006)
100	60	$(0_{(59)}, 40)$	0.8180	0.8094	0.8189	0.8185	0.8053	0.7262
				(0.0009)	(0.0007)	(0.0006)	(0.0004)	(0.0002)
		$(40, 0_{(59)})$	0.8180	0.8789	0.8771	0.8763	0.8745	0.8107
				(0.0037)	(0.0034)	(0.0033)	(0.0032)	(0.0001)
100	80	$(0_{(79)}, 20)$	0.8180	0.8815	0.8820	0.8817	0.8778	0.7649
				(0.0041)	(0.0038)	(0.0035)	(0.0031)	(0.0028)
		$(20, 0_{(79)})$	0.8180	0.8103	0.7284	0.8042	0.8081	0.8103
				(0.0016)	(0.0012)	(0.0010)	(0.0008)	(0.0001)

TABLE 7: True value of C_{py} and 95% CIs of MLE, Boot-p and Boot-t when $\alpha = 7.0, \theta = 4.0$, and $\gamma = 0.50$.

n	m	$(R_1, \cdots, R_m)$	C_{py}	MLE			Boot-p			Boot-t		
				Lower	Upper	Length	Lower	Upper	Length	Lower	Upper	Length
20	10	$(0_{(9)}, 10)$	0.2043	0.1190	0.4694	0.3504	0.1335	1.0526	0.9192	0.1391	0.4191	0.2799
		$(10, 0_{(9)})$	0.2043	−0.0173	0.2654	0.2828	0.0198	0.7191	0.6993	0.0168	0.2903	0.2735
20	15	$(0_{(14)}, 5)$	0.2043	0.1023	0.4539	0.3516	0.0656	0.7463	0.6807	0.0508	0.4914	0.4406
		$(5, 0_{(14)})$	0.2043	−0.0108	0.2746	0.2854	0.0168	0.3044	0.2876	0.0015	0.3347	0.3332
40	15	$(0_{(14)}, 25)$	0.2043	0.0887	0.3046	0.2159	0.1032	0.2565	0.1533	0.1041	0.2475	0.1434
		$(25, 0_{(14)})$	0.2043	0.0664	0.4943	0.4279	0.0765	0.6025	0.5260	0.0814	0.5007	0.4193
40	30	$(0_{(29)}, 10)$	0.2043	0.0946	0.3171	0.2226	0.1253	0.4068	0.2815	0.1025	0.3746	0.2721
		$(10, 0_{(29)})$	0.2043	0.1527	0.4441	0.2914	−0.3530	0.4342	0.7872	0.1462	0.4911	0.3449
60	30	$(0_{(29)}, 30)$	0.2043	0.0932	0.2639	0.1707	−0.0132	0.4681	0.4813	0.0976	0.2538	0.1563
		$(30, 0_{(29)})$	0.2043	0.0521	0.2696	0.2174	0.0438	0.3277	0.2839	0.0611	0.3039	0.2427

Table 7: Continued.

n	m	$(R_1, \cdots, R_m)$	C_{py}	MLE			Boot-p			Boot-t		
				Lower	Upper	Length	Lower	Upper	Length	Lower	Upper	Length
60	40	$(0_{(39)}, 20)$	0.2043	0.1306	0.3448	0.2142	−0.5665	0.2945	0.8610	0.1390	0.3554	0.2164
		$(20, 0_{(39)})$	0.2043	0.1080	0.3398	0.2318	−0.2273	0.3294	0.5567	0.1144	0.3811	0.2667
80	40	$(0_{(39)}, 40)$	0.2043	0.0864	0.2426	0.1562	−0.5435	0.2003	0.7438	0.0814	0.2337	0.1523
		$(40, 0_{(39)})$	0.2043	0.0860	0.3141	0.2281	−0.4838	0.3025	0.7863	0.0973	0.3739	0.2766
80	50	$(0_{(49)}, 30)$	0.2043	0.1489	0.3172	0.1683	0.0442	0.3163	0.2721	0.1577	0.3154	0.1576
		$(30, 0_{(49)})$	0.2043	0.0987	0.2888	0.1901	0.0114	0.3254	0.3140	0.0920	0.3269	0.2349
100	60	$(0_{(59)}, 40)$	0.2043	0.1473	0.3134	0.1661	0.0221	0.2847	0.2626	0.1605	0.3389	0.1784
		$(40, 0_{(59)})$	0.2043	0.1551	0.3755	0.2204	−0.3582	0.3424	0.7006	0.1566	0.3766	0.2200
100	80	$(0_{(79)}, 20)$	0.2043	0.1033	0.2367	0.1333	0.1004	0.2354	0.1350	0.1093	0.3034	0.1941
		$(20, 0_{(79)})$	0.2043	0.1365	0.2928	0.1563	0.0934	0.2970	0.2036	0.1378	0.2989	0.1611

Table 8: True value of C_{py} and 95% CIs of MLE, Boot-p and Boot-t when $\alpha = 4.40, \theta = 5.79$, and $\gamma = 0.77$.

n	m	$(R_1, \cdots, R_m)$	C_{py}	MLE			Boot-p			Boot-t		
				Lower	Upper	Length	Lower	Upper	Length	Lower	Upper	Length
20	10	$(0_{(9)}, 10)$	0.8180	0.7741	1.1998	0.4258	0.5354	1.0526	0.5172	0.7693	1.0526	0.2833
		$(10, 0_{(9)})$	0.8180	0.7131	1.0925	0.3793	0.1354	1.0107	0.8753	-0.0778	1.0344	1.1122
20	15	$(0_{(14)}, 5)$	0.8180	0.8467	1.1145	0.2679	0.6988	1.0520	0.3532	0.9158	1.0458	0.1301
		$(5, 0_{(14)})$	0.8180	0.8006	1.0729	0.2723	0.4764	1.0418	0.5653	0.7925	1.0391	0.2465
40	15	$(0_{(14)}, 25)$	0.8180	0.5041	1.1430	0.6389	0.3706	1.0526	0.6820	0.5780	1.0332	0.4552
		$(25, 0_{(14)})$	0.8180	0.6395	1.0021	0.3626	0.1145	0.9203	0.8049	0.4262	1.0229	0.5967
40	30	$(0_{(29)}, 10)$	0.8180	0.8677	1.0702	0.2026	0.8195	1.0526	0.2331	0.9280	1.0301	0.1021
		$(10, 0_{(29)})$	0.8180	0.7740	1.0001	0.2260	0.4504	0.9534	0.5030	0.7252	0.9873	0.2621
60	30	$(0_{(29)}, 30)$	0.8180	0.7302	1.0541	0.3239	0.5126	1.0512	0.5386	0.7901	1.0289	0.2388
		$(30, 0_{(29)})$	0.8180	0.7735	1.001	0.2275	0.6012	0.9419	0.3407	0.6192	1.0244	0.4052
60	40	$(0_{(39)}, 20)$	0.8180	0.6483	0.9839	0.3356	0.5042	0.8365	0.3323	0.7648	0.8921	0.1272
		$(20, 0_{(39)})$	0.8180	0.6129	0.8726	0.2597	0.5210	0.8358	0.3148	0.5437	0.8632	0.3195
80	40	$(0_{(39)}, 40)$	0.8180	0.7932	1.1186	0.3254	0.7126	1.0445	0.3319	0.8904	1.0159	0.1255
		$(40, 0_{(39)})$	0.8180	0.6681	0.9048	0.2366	0.5686	0.8883	0.3197	0.5432	0.9168	0.3736
80	50	$(0_{(49)}, 30)$	0.8180	0.7086	1.0000	0.2913	0.5234	0.8712	0.3478	0.7975	1.0158	0.2183
		$(30, 0_{(49)})$	0.8180	0.7961	0.9735	0.1774	0.6135	0.9505	0.3370	0.7063	0.9696	0.2633
100	60	$(0_{(59)}, 40)$	0.8180	0.6906	0.9281	0.2375	0.6531	0.9581	0.3050	0.7599	0.8958	0.1359
		$(40, 0_{(59)})$	0.8180	0.7974	0.9605	0.1631	0.1135	0.9295	0.8160	0.6962	0.9741	0.2779
100	80	$(0_{(79)}, 20)$	0.8180	0.8008	0.9623	0.1615	0.6036	0.8926	0.2890	0.7572	0.9099	0.1527
		$(20, 0_{(79)})$	0.8180	0.7284	0.8922	0.1637	0.7428	0.8929	0.1501	0.7436	0.8892	0.1456

Table 9: True value of C_{py} and its 95% CRIs when $\alpha = 7.0, \theta = 4.0$, and $\gamma = 0.50$.

n	m	$(R_1, \cdots, R_m)$	C_{py}	Prior-0			Prior-I		
				Lower	Upper	Length	Lower	Upper	Length
20	10	$(0_{(9)}, 10)$	0.2043	0.1501	0.4475	0.2974	0.0747	0.2501	0.1754
		$(10, 0_{(9)})$	0.2043	0.0597	0.2022	0.1525	0.0952	0.2850	0.1899
20	15	$(0_{(14)}, 5)$	0.2043	0.1663	0.4001	0.2337	0.1061	0.2681	0.1620
		$(5, 0_{(14)})$	0.2043	0.0715	0.2098	0.1383	0.1236	0.3243	0.2006
40	15	$(0_{(14)}, 25)$	0.2043	0.1153	0.2933	0.1780	0.1401	0.3474	0.2073
		$(25, 0_{(14)})$	0.2043	0.1665	0.4062	0.2396	0.0982	0.2536	0.1554
40	30	$(0_{(29)}, 10)$	0.2043	0.1434	0.2753	0.1318	0.2080	0.3878	0.1798
		$(10, 0_{(29)})$	0.2043	0.2106	0.3906	0.1801	0.1518	0.2902	0.1384
60	30	$(0_{(29)}, 30)$	0.2043	0.1246	0.2382	0.1136	0.1309	0.2603	0.1294
		$(30, 0_{(29)})$	0.2043	0.1136	0.2173	0.1036	0.1611	0.2938	0.1326
60	40	$(0_{(39)}, 20)$	0.2043	0.1748	0.3033	0.1285	0.1335	0.2348	0.1013
		$(20, 0_{(39)})$	0.2043	0.1648	0.2886	0.1239	0.1828	0.3164	0.1336
80	40	$(0_{(39)}, 40)$	0.2043	0.1211	0.2127	0.0916	0.1290	0.2289	0.0999
		$(40, 0_{(39)})$	0.2043	0.1478	0.2562	0.1084	0.2026	0.3600	0.1574
80	50	$(0_{(49)}, 30)$	0.2043	0.1790	0.2882	0.1092	0.2010	0.3229	0.1220
		$(30, 0_{(49)})$	0.2043	0.1469	0.2463	0.0994	0.1704	0.2868	0.1164

Table 9: Continued.

n	m	$(R_1, \cdots, R_m)$	C_{py}	Prior-0			Prior-I		
				Lower	Upper	Length	Lower	Upper	Length
100	60	$(0_{(59)}, 40)$	0.2043	0.1813	0.2848	0.1035	0.1373	0.2196	0.0823
		$(40, 0_{(59)})$	0.2043	0.2097	0.3212	0.1115	0.1868	0.2949	0.1081
100	80	$(0_{(79)}, 20)$	0.2043	0.1378	0.2041	0.0664	0.1528	0.2288	0.076
		$(20, 0_{(79)})$	0.2043	0.1750	0.2554	0.0804	0.2040	0.3008	0.0967

Table 10: True value of C_{py} and its 95% CRIs when $\alpha = 4.40, \theta = 5.79$, and $\gamma = 0.77$.

n	m	$(R_1, \cdots, R_m)$	C_{py}	Prior-0			Prior-I		
				Lower	Upper	Length	Lower	Upper	Length
20	10	$(0_{(9)}, 10)$	0.8180	0.7691	1.0438	0.2747	0.5618	0.9786	0.4168
		$(10, 0_{(9)})$	0.8180	0.6363	1.0184	0.3821	0.5888	0.9965	0.4077
20	15	$(0_{(14)}, 5)$	0.8180	0.8125	1.0336	0.2211	0.5742	0.9386	0.3645
		$(5, 0_{(14)})$	0.8180	0.7516	1.0189	0.2673	0.7455	1.0266	0.2812
40	15	$(0_{(14)}, 25)$	0.8180	0.6105	0.9495	0.3390	0.6226	0.9758	0.3531
		$(25, 0_{(14)})$	0.8180	0.5923	0.9498	0.3575	0.5964	0.9509	0.3545
40	30	$(0_{(29)}, 10)$	0.8180	0.8602	1.0218	0.1615	0.6672	0.9304	0.2632
		$(10, 0_{(29)})$	0.8180	0.7540	0.9739	0.2199	0.8814	1.0291	0.1476
60	30	$(0_{(29)}, 30)$	0.8180	0.7504	0.9770	0.2266	0.6036	0.8657	0.2621
		$(30, 0_{(29)})$	0.8180	0.7519	0.9742	0.2223	0.6978	0.9387	0.2409
60	40	$(0_{(39)}, 20)$	0.8180	0.6979	0.9042	0.2063	0.7053	0.9238	0.2184
		$(20, 0_{(39)})$	0.8180	0.6086	0.8415	0.2329	0.7245	0.9306	0.2060
80	40	$(0_{(39)}, 40)$	0.8180	0.8670	1.0067	0.1397	0.7292	0.9301	0.2009
		$(40, 0_{(39)})$	0.8180	0.658	0.8842	0.2263	0.6186	0.8507	0.2320
80	50	$(0_{(49)}, 30)$	0.8180	0.7423	0.9275	0.1852	0.7084	0.9039	0.1955
		$(30, 0_{(49)})$	0.8180	0.7865	0.9557	0.1692	0.7469	0.9232	0.1763
100	60	$(0_{(59)}, 40)$	0.8180	0.7055	0.8875	0.1820	0.6287	0.8110	0.1823
		$(40, 0_{(59)})$	0.8180	0.7905	0.9412	0.1507	0.7062	0.8906	0.1844
100	80	$(0_{(79)}, 20)$	0.8180	0.8057	0.9423	0.1366	0.6786	0.8407	0.1621
		$(20, 0_{(79)})$	0.8180	0.7243	0.8818	0.1575	0.7245	0.8803	0.1558

Also, we can generate the progressive type-II censored sample of size $r = 5$ taken from sample size $n = 20$ with censoring scheme $R = (0_{(4)}, 15)$ based on the dataset II defined by Zimmer and Hubele [33]. A progressive type-II censored sample produced from the real-dataset II is obtained as follows:

0.9 1.5 3.2 7.5 11.1.

The descriptive statistics for the considered datasets are reported in Table 1. For the previous datasets considered, based on a progressive type-II we have computed the point estimates of C_{py} using ML and Bootstrap method, the results are shown in Table 2, and we also determined the 95% CIs based on MLEs and the 95% bootstrap (Boot-p and Boot-t) CIs of C_{py}, and the results are displayed in Table 3. Now, we want to calculate the Bayes estimates of C_{py} against SE loss functions. Since we do not have prior information about the unknown parameters, we assume the noninformative gamma priors for α, θ, and γ. This prior distribution is the case in which hyperparameters are identified as $a_i = b_i = 0, i = 1, 2, 3$. We perform the MCMC algorithm described in Section 5 to generate a sequence of 10,000 random vectors iteratively with different starting points for the parameters α, θ, and γ, and discard the first 1000 values as "burn-in." The results of Bayes estimates are reported in Table 2 and also calculated the 95% CRIs, the results are shown in Table 3. The MCMC results are shown in Table 4 for the posterior mean, median, mode, standard deviation (SD), and skewness (Sk) of C_{py}.

7. Simulations

In this section, the Monte Carlo simulation study has been implemented to compare the performances of the classical estimation methods and the Bayesian estimation approach for prior-0 and prior-I distributions under SE loss function of the index C_{py} for TPBXIID. This simulation was carried out considering different values of n and m and by choosing $(\alpha, \theta, \gamma) = (7.0, 4.0, 0.50)$ and $(4.40, 5.79, 0.77)$ with $L = 0.6$, $U = 6.0$, and $p_0 = 0.95$, respectively. Two different priors are used for Bayesian computation in order to compare the Bayes estimates: (a) noninformative gamma prior (prior-0), the hyperparameter values as $a_i = b_i = 0$, and (b) informative gamma prior (prior-I), for this prior, we arbitrarily selected the hyperparameter values as $a_i = 1.7$ and $b_i = 2.2$ for different parameter sets. We applied the MCMC method with using 10000 MCMC samples and discard the first 1000 values as "burn-in" under SE loss function. We compare the performances of MLEs and Bayes estimates in terms of the MSE, which is calculated as follows:

$$\text{MSE} = \sum_{i=1}^{M} \frac{\left(\widehat{C}_{\text{py}}(i) - C_{\text{pyExact}}\right)^2}{M}. \quad (25)$$

We have used two different sampling schemes as follows:

(i) Scheme I: $R_1 = n - m, R_i = 0$ for $i \neq 1$.

(ii) Scheme II: $R_m = n - m, R_i = 0$ for $i \neq m$. Point (classical in addition to the Bayesian) estimates of C_{py} for TPBXIID are displayed in Tables 5 and 6. Also, the 95% CIs based on MLEs and the 95% bootstrap (Boot-p and Boot-t) CIs of C_{py} were determined, and the results are summarized in Tables 7 and 8. Also, the results of 95% CRIs are given in Tables 9 and 10.

8. Conclusions

In this paper, we considered classical and Bayesian point estimation methods of the index C_{py} and used two practical examples to illustrate the methods proposed. Generally, we considered the MLEs and bootstrap (Boot-p and Boot-t) for classical estimation methods in order to get the estimates of the unknown parameters and the C_{py} index. Since theoretical comparison of these methods is not feasible, we have carried out comprehensive simulation study to compare these methods with different sample sizes and different combinations of the unknown parameters. Therefore, we considered Bayesian inference of the unknown parameters of the TPBXIID and the index C_{py} using MCMC approach. In addition, we have considered the 95% CIs based on MLEs and the 95% bootstrap (Boot-p and Boot-t) CIs of the index C_{py}. We note the following from the previous results:

(1) It can be seen that from all tables, for increasing values of n and m, the MSEs decreasing.

(2) It is observed from Tables 5 and 6 that the Bayes estimators perform better under prior-I than under prior-0 in addition to it performs better than classical methods of estimation in terms of MSEs.

(3) It is observed from Tables 7 and 8 that the Boot-t CIs give more accurate results than the Boot-p and ACIs since the lengths of the Boot-t CIs are less than the lengths of Boot-p and ACIs, for different sample sizes.

(4) It is evident that, from Table 3, the Bayesian estimation method gives smallest average widths from the other estimation methods.

Data Availability

No data were used to support the findings of this study.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

References

[1] J. M. Juran, *Juran's Quality Control Handbook*, McGraw-Hill, New York, NY, USA, 3rd. edition, 1974.

[2] V. E. Kane, "Process capability indices," *Journal of Quality Technology*, vol. 18, no. 1, pp. 41–52, 1986.

[3] L. K. Chan, F. Spiring, and H. Xiao, *An OC Curve Approach for Analyzing the Process Capability Index Cpk*, Technical Report, Department of Statistics, University of Manitoba, Winnipeg, Canada, 1988.

[4] W. L. Pearn, S. Kotz, and N. L. Johnson, "Distributional and inferential properties of process capability indices," *Journal of Quality Technology*, vol. 24, no. 4, pp. 216–231, 1992.

[5] B. H. Gunter, "The use and abuse of C_{pk}," *Qual Prog*, vol. 22, no. 3, pp. 108-109, 1989.

[6] J. A. Clements, "Process capability calculations for non-normal distributions," *Qual Prog*, vol. 95, p. 100, 1989.

[7] R. N. Rodriguer, "Recent developments in process capability analysis," *Journal of Quality Technology*, vol. 24, no. 4, pp. 176–187, 1992.

[8] A. M. Polansky, "A smooth nonparametric approach to process capability," *Quality and Reliability Engineering International*, vol. 14, no. 1, pp. 43–48, 1998.

[9] A. B. Yeh and S. Bhattcharya, "A robust process capability index," *Communications in Statistics–Simulation and Computation*, vol. 27, no. 2, pp. 565–589, 1998.

[10] M. Perakis and E. Xekalaki, "A process capability index that is based on the proportion of conformance," *Journal of Statistical Computation and Simulation*, vol. 72, no. 9, pp. 707–718, 2002.

[11] S. S. Maiti, M. Saha, and A. K. Nanda, "On generalizing process capability indices," *Quality Technology & Quantitative Management*, vol. 7, no. 3, pp. 279–300, 2010.

[12] Z. Y. Huiming, Y. Jun, and H. Liya, "Bayesian evaluation approach for process capability based on sub samples," in *Proceedings of the 2007 IEEE International Conference on Industrial Engineering and Engineering Management*, pp. 1200–1203, Singapore, December 2007.

[13] R. Miao, X. Zhang, D. Yang, Y. Zhao, and Z. Jiang, "A conjugate Bayesian approach for calculating process capability indices," *Expert Systems with Applications*, vol. 38, no. 7, pp. 8099–8104, 2011.

[14] C.-W. Wu and T.-Y. Lin, "A Bayesian procedure for assessing process performance based on the third-generation capability index," *Journal of Applied Statistics*, vol. 36, no. 11, pp. 1205–1223, 2009.

[15] M. Kargar, M. Mashinchi, and A. Parchami, "A Bayesian approach to capability testing based on Cpk with multiple samples," *Quality and Reliability Engineering International*, vol. 30, no. 5, pp. 615–621, 2014.

[16] S. S. Maiti and M. Saha, "Bayesian estimation of generalized process capability indices," *Journal of Probability and Statistics*, vol. 2012, Article ID 819730, 15 pages, 2012.

[17] M. A. W. Mahmoud, R. M. EL-Sagheer, A. A. Soliman, and A. H. Abd Ellah, "Inferences of the lifetime performance index with Lomax distribution based on progressive type-ii censored data," *Economic Control of Quality*, vol. 29, no. 1, pp. 39–51, 2014.

[18] S. Ali and M. Riaz, "On the generalized process capability under simple and mixture models," *Journal of Applied Statistics*, vol. 41, no. 4, pp. 832–852, 2014.

[19] M. Saha, S. Dey, A. S. Yadav, and S. Kumar, "Classical and Bayesian inference of Cpy for generalized Lindley distributed quality characteristic," *Quality and Reliability Engineering International*, vol. 35, no. 8, pp. 2593–2611, 2019.

[20] I. W. Burr, "Cumulative frequency functions," *The Annals of Mathematical Statistics*, vol. 13, no. 2, pp. 215–232, 1942.

[21] D. R. Wingo, "Maximum likelihood methods for fitting the Burr type XII distribution to life test data," *Biometrical Journal*, vol. 25, no. 1, pp. 77–84, 1983.

[22] D. R. Wingo, "Maximum likelihood estimation of Burr XII distribution parameters under Type II censoring," *Microelectronics Reliability*, vol. 33, no. 9, pp. 1251–1257, 1993.

[23] W. J. Zimmer, J. B. Keats, and F. K. Wang, "The Burr XII distribution in reliability analysis," *Journal of Quality Technology*, vol. 30, no. 4, pp. 386–394, 1998.

[24] Q. Shao, "Estimation for hazardous concentrations based on NOEC toxicity data: an alternative approach," *Environmetrics*, vol. 11, no. 5, pp. 583–595, 2000.

[25] R. M. EL-Sagheer, "Estimation of parameters of Weibull-Gamma distribution based on progressively censored data," *Statistical Papers*, vol. 59, no. 2, pp. 725–757, 2018.

[26] W. H. Greene, *Econometric Analysis*, Prentice-Hall, New York, NY, USA, 4th edition, 2000.

[27] B. Efron, "The jackknife, the bootstrap and other resampling plans," in *Proceedings of the CBMS/NSF Regional Conference Series in Applied Mathematics*, SIAM, Philadelphia, PA, USA, 1982.

[28] P. Hall, "Theoretical comparison of bootstrap confidence intervals," *The Annals of Statistics*, vol. 16, no. 3, pp. 927–953, 1988.

[29] N. Balakrishnan and R. A. Sandhu, "A simple simulational algorithm for generating progressive type-II censored samples," *The American Statistician*, vol. 49, no. 2, pp. 229-230, 1995.

[30] M.-H. Chen and Q.-M. Shao, "Monte Carlo estimation of Bayesian Credible and HPD intervals," *Journal of Computational and Graphical Statistics*, vol. 8, no. 1, pp. 69–92, 1999.

[31] N. Metropolis, A. W. Rosenbluth, M. N. Rosenbluth, A. H. Teller, and E. Teller, "Equation of state calculations by fast computing machines," *The Journal of Chemical Physics*, vol. 21, no. 6, pp. 1087–1092, 1953.

[32] V. Leiva, C. Marchant, H. Saulo, M. Aslam, and F. Rojas, "Capability indices for Birnbaum-Saunders processes applied to electronic and food industries," *Journal of Applied Statistics*, vol. 41, no. 9, pp. 1881–1902, 2014.

[33] L. S. Zimmer and N. F. Hubele, "Quantiles of the sampling distribution of Cpm," *Quality Engineering*, vol. 10, pp. 309–329, 1998.