

Article

# Estimation of Beta-Pareto Distribution Based on Several Optimization Methods

Badreddine Boumaraf <sup>1,2,3,†</sup> , Nacira Seddik-Ameur <sup>2,†</sup> and Vlad Stefan Barbu <sup>3,\*,†</sup> 

<sup>1</sup> Department of Mathematics and Informatics, University of Souk-Ahras, Souk Ahras 41000, Algeria; b.boumaaraf@univ-soukahras.dz

<sup>2</sup> Laboratory of Probability and Statistics LaPS, University of Badji Mokhtar of Annaba, Annaba 23000, Algeria; naciraseddik@yahoo.fr

<sup>3</sup> Laboratory of Mathematics Raphaël Salem, University of Rouen-Normandy, 76801 Saint Étienne du Rouvray, France

\* Correspondence: barbu@univ-rouen.fr

† The authors contributed equally to this work.

Received: 28 February 2020; Accepted: 10 June 2020; Published: 1 July 2020



**Abstract:** This paper is concerned with the maximum likelihood estimators of the Beta-Pareto distribution introduced in Akinsete et al. (2008), which comes from the mixing of two probability distributions, Beta and Pareto. Since these estimators cannot be obtained explicitly, we use nonlinear optimization methods that numerically provide these estimators. The methods we investigate are the method of Newton-Raphson, the gradient method and the conjugate gradient method. Note that for the conjugate gradient method we use the model of Fletcher-Reeves. The corresponding algorithms are developed and the performances of the methods used are confirmed by an important simulation study. In order to compare between several concurrent models, namely generalized Beta-Pareto, Beta, Pareto, Gamma and Beta-Pareto, model criteria selection are used. We firstly consider completely observed data and, secondly, the observations are assumed to be right censored and we derive the same type of results.

**Keywords:** maximum likelihood estimators; nonlinear optimization methods; Beta-Pareto distribution; Beta distribution; Pareto distribution; model selection; right-censored data

## 1. Introduction

In this work we are interested in the four-parameter distribution called Beta-Pareto (BP) distribution, introduced recently by Akinsete et al. (2008) [1]. This distribution is a generalization of several models like Pareto, logbeta, exponential, arcsine distributions, in the sense that these distributions can be considered as special cases of the BP distribution, by making transformation of the variable or by setting special values of the parameters (cf. Reference [1]).

It is well known that the family of Pareto distribution and the corresponding generalizations have been extensively used in various fields of applications, like income data [2], environmental studies [3] or economical-social studies [4]. Concerning the generalizations of the Pareto distribution, we can cite the Burr distribution [5], the power function [6] and the logistic distribution. It is important also to stress that heavy tailed phenomena can be successfully modelled by means of (generalized) Pareto distributions [7]. Note that this BP model is based on Pareto distribution which is known to be heavy tailed, as it was shown in Reference [1] using some theoretical and application arguments. We would like to stress that this is important in practice because it can be used to describe skewed data better than other distributions proposed in the statistical literature. A classical example of this kind is given by the exceedances of Wheaton River flood data which are highly skewed to the right.

All these reasons show the great importance of statistically investigating any generalization of Pareto distributions. Thus the purpose of our work is to provide algorithmic methods for practically computing the maximum likelihood estimators (MLEs) of the parameters of the BP distribution, to carry out intensive simulation studies in order to investigate the quality of the proposed estimators, to consider model selection criteria in order to choose among candidate models and also to take into account right-censored data, not only completely observed data. Note that right-censored data are often observed in reliability studies and survival analysis, where experiments are generally conducted over a fixed time period. At the end of the study, generally we cannot observe the failure times of all the products or death (remission) of all patients due to loss to follow-up, competing risks (death from other causes) or any combination of these. In such cases, we do not always have the opportunity of observing these survival times, observing only the censoring times. Consequently, it is of crucial importance for this type of data and applications to develop a methodology that allows the estimation in the presence of right censoring.

Generally, numerical methods are required when the MLEs of the unknown parameters of any model cannot be obtained. For the BP distribution, we propose the use of several nonlinear optimization methods: Newton-Raphson’s method, the gradient and the conjugate gradient methods. For the Newton-Raphson’s method, the evaluation of the approximation of the Hessian matrix is widely studied in the literature. In this work, we are interested in the *BFGS* method (from Broyden-Fletcher-Goldfarb-Shanno), the *DFP* method (from Davidon-Fletcher-Powell) and the *SR1* method (Symmetric Rank 1). For the conjugate gradient method we use the model of Fletcher-Reeves.

It is well known that the gradient and conjugate gradient methods are better than the Newton-Raphson’s method. Nonetheless, most statistical studies use the Newton-Raphson’s method instead of the gradient or conjugate gradient methods, maybe because it is much more easier to put in practice. Our interest is to put in practice the gradient method and the conjugate gradient method in our framework of BP estimation and also present the Newton-Raphson method for comparison purposes only.

The structure of the article is as follows. In the next section we introduce the BP distribution and we give some elements of MLE for such a model. Section 3 is devoted to the numerical optimization methods used for obtaining the MLEs for the parameters of interest. Firstly, we briefly recall the numerical methods that we use (Newton-Raphson, gradient and conjugate gradient). Secondly, we present the corresponding algorithm for the method of conjugate gradient, which is the most complex one. We end this section by investigating through simulations the accuracy of these three methods. In Section 4 we use model selection criteria (AIC, BIC, AICc) in order to chose between several concurrent models, namely between the Generalized Beta-Pareto, Beta, Pareto, Gamma, BP models respectively. In Section 5 we assume that we have at our disposal right-censored observations and we derive the same type of results as we did for complete observations.

## 2. BP Distribution and MLEs of the Parameters

A random variable  $X$  has a BP distribution with parameters  $\alpha, \beta, \theta, k > 0$ , if its probability density function is

$$f(x; \alpha, \beta, k, \theta) := \frac{k}{\theta B(\alpha, \beta)} \left[ 1 - \left(\frac{x}{\theta}\right)^{-k} \right]^{\alpha-1} \left(\frac{x}{\theta}\right)^{-k\beta-1}, x \geq \theta,$$

with  $B(\alpha, \beta) = \frac{\Gamma(\alpha)\Gamma(\beta)}{\Gamma(\alpha+\beta)}$ ,  $\Gamma(x), x > 0$ , being the Gamma function. Consequently, the support of  $X$  is  $[\theta, +\infty)$ . The corresponding cumulative distribution function can be written as

$$F(x; \alpha, \beta, k, \theta) = 1 - \frac{\left(\frac{x}{\theta}\right)^{-k\beta}}{\beta B(\alpha, \beta)} - \frac{\left(\frac{x}{\theta}\right)^{-k\beta}}{B(\alpha, \beta)} \sum_{n=1}^{\infty} \frac{\prod_{i=1}^n (i - \alpha)}{n!(\beta + n)} \left(\frac{x}{\theta}\right)^{-kn}, x \geq \theta, \alpha, \beta, \theta, k > 0.$$

In Figures 1–5, the pdf, survival, cdf, hazard and cumulative hazard functions are presented for several values of the parameters.

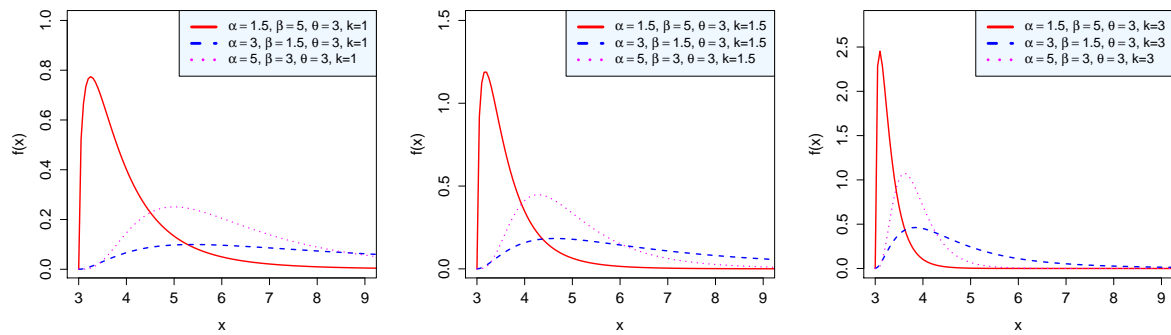


Figure 1. Density of the BP distribution.

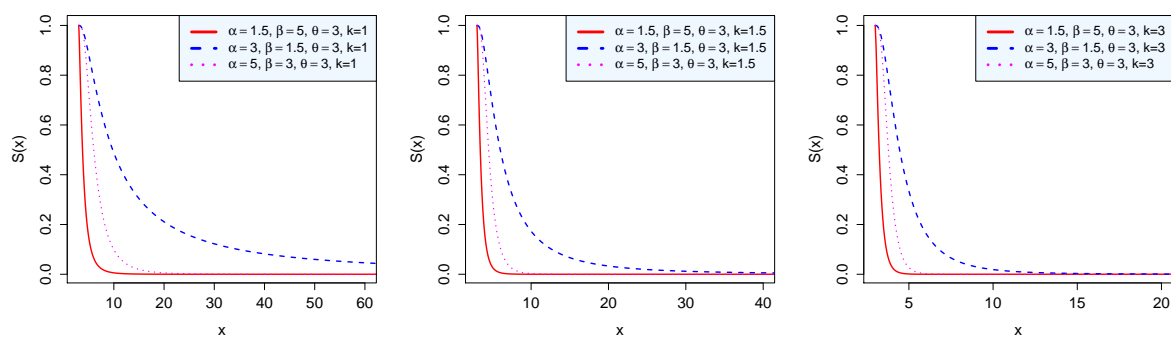


Figure 2. Survival function of the BP distribution.

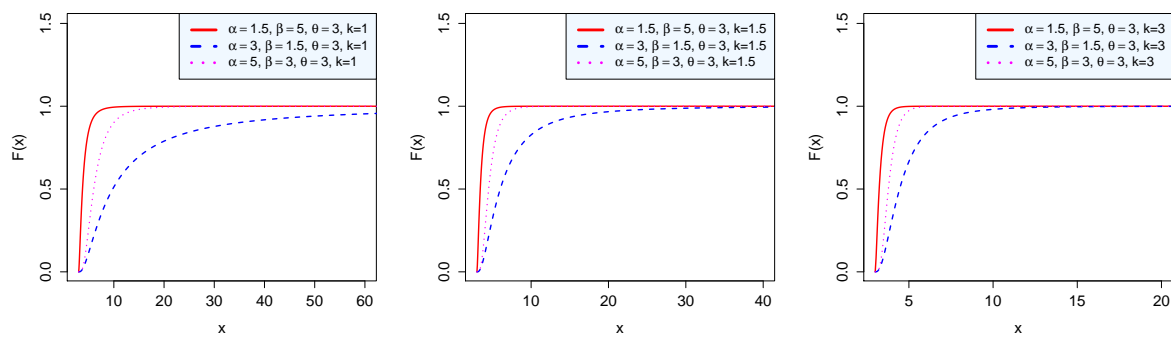


Figure 3. Cumulative distribution function of the BP distribution.

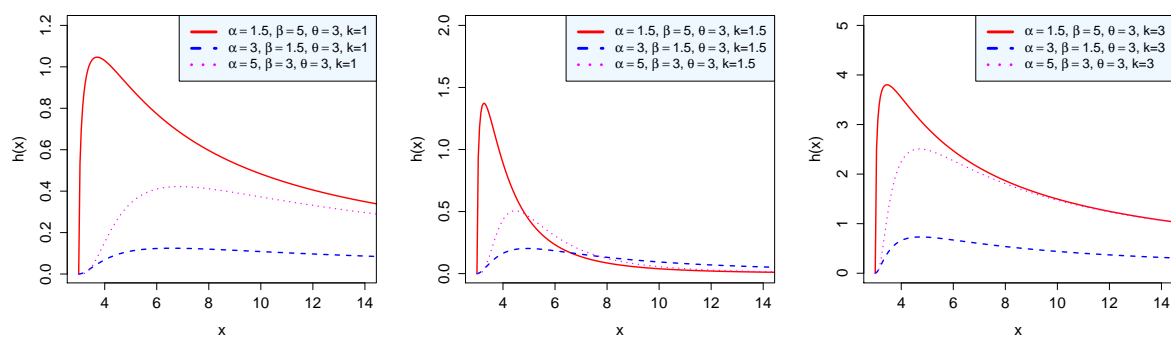


Figure 4. Hazard rate of the BP distribution.

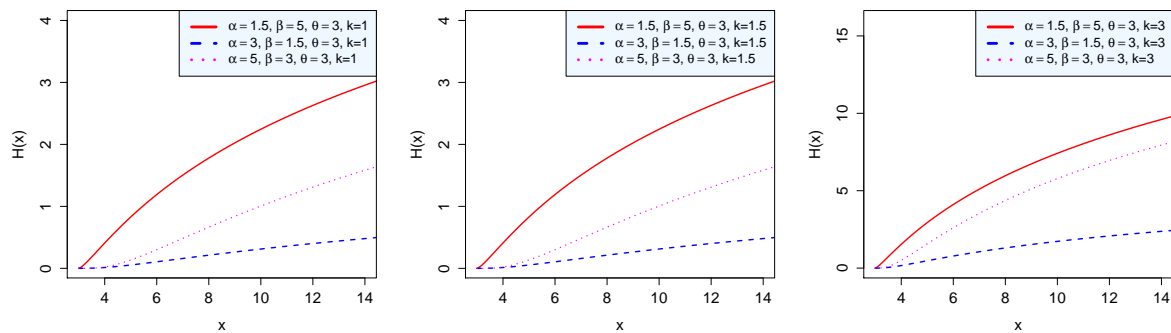


Figure 5. Cumulative hazard rate of the BP distribution.

Let us now consider an i.i.d. (independent and identically distributed) sample  $(x_1, x_2, x_3, \dots, x_n)$  of a random variable  $X$  following a BP distribution with density  $f(x; \alpha, \beta, k, \theta)$ . The corresponding log-likelihood function can be written as

$$l(\alpha, \beta, k, \theta \mid x_1, x_2, x_3, \dots, x_n) = n \ln k - n \ln \theta + n(\ln \Gamma(\alpha + \beta) - \ln \Gamma(\alpha) - \ln \Gamma(\beta)) + (\alpha - 1) \sum_{i=1}^n \ln \left[ 1 - \left( \frac{x_i}{\theta} \right)^{-k} \right] - (k\beta + 1) \sum_{i=1}^n \ln \left( \frac{x_i}{\theta} \right). \tag{1}$$

As usual, when we need to see the log-likelihood function as a function of the parameters  $\alpha, \beta, k, \theta$  only, we shall write  $l(\alpha, \beta, k, \theta)$  instead of  $l(\alpha, \beta, k, \theta \mid x_1, x_2, \dots, x_n)$ . The score equations are thus given by

$$\frac{\partial l(\alpha, \beta, k, \theta)}{\partial \alpha} = n [\Psi(\alpha + \beta) - \Psi(\alpha)] + \sum_{i=1}^n \ln \left( 1 - \left( \frac{x_i}{\theta} \right)^{-k} \right) = 0, \tag{2}$$

$$\frac{\partial l(\alpha, \beta, k, \theta)}{\partial \beta} = n[\Psi(\alpha + \beta) - \Psi(\beta)] - k \sum_{i=1}^n \ln \left( \frac{x_i}{\theta} \right) = 0, \tag{3}$$

$$\frac{\partial l(\alpha, \beta, k, \theta)}{\partial k} = \frac{n}{k} - \sum_{i=1}^n \left[ \beta + (\alpha - 1) \left( 1 - \left( \frac{x_i}{\theta} \right)^k \right)^{-1} \right] \ln \left( \frac{x_i}{\theta} \right) = 0, \tag{4}$$

where the function  $\Psi$  is defined by  $\Psi(x) := \frac{\Gamma'(x)}{\Gamma(x)}, x > 0$ .

As  $x \geq \theta$ , the MLE of the parameter  $\theta$  is the first order statistic  $x_{(1)}$  and the MLEs of  $\alpha, \beta$  and  $k$  are obtained by solving the system (2)–(4).

Using Equations (2)–(4) we compute the elements of the Fisher information matrix as follows [1]:

$$\begin{aligned} \frac{\partial^2 l(\alpha, \beta, k, \theta)}{\partial \alpha^2} &= n[\Psi'(\alpha + \beta) - \Psi'(\alpha)], \\ \frac{\partial^2 l(\alpha, \beta, k, \theta)}{\partial \alpha \partial \beta} &= n\Psi'(\alpha + \beta), \\ \frac{\partial^2 l(\alpha, \beta, k, \theta)}{\partial k \partial \beta} &= - \sum_{i=1}^n \ln \left( \frac{x_i}{\theta} \right), \\ \frac{\partial^2 l(\alpha, \beta, k, \theta)}{\partial k \partial \alpha} &= \sum_{i=1}^n \ln \left( \frac{x_i}{\theta} \right) \left[ \left( \frac{x_i}{\theta} \right)^k - 1 \right]^{-1}, \\ \frac{\partial^2 l(\alpha, \beta, k, \theta)}{\partial k^2} &= - \frac{n}{k^2} - (\alpha - 1) \sum_{i=1}^n \left( \frac{x_i}{\theta} \right)^k \left[ \frac{\ln \left( \frac{x_i}{\theta} \right)}{1 - \left( \frac{x_i}{\theta} \right)^k} \right]^2, \\ \frac{\partial^2 l(\alpha, \beta, k, \theta)}{\partial \beta^2} &= n[\Psi'(\alpha + \beta) - \Psi'(\beta)]. \end{aligned}$$

There are no closed formulas for solving these equations, so numerical methods are required.

### 3. Numerical Optimization for Solving the Score Equations

In this section we are interested in proposing numerical methods for solving the score equations of the MLEs of the BP parameters. We will consider three methods: the Newton-Raphson’s method, the gradient method and the conjugate gradient method. Firstly, we will recall the general lines of these optimization methods. Secondly, we will give a detailed description of the application of the most complex method, namely of the conjugate gradient one, for computing the MLEs of the BP distribution. We close the section by presenting numerical results of the implementation of these methods in our BP framework.

#### 3.1. General Description of Optimization Methods

Let us first recall the main lines of the three numerical methods that we have considered. The objectives of these methods will be to minimize/maximize a function  $l : \mathbb{R}^p \rightarrow \mathbb{R}$ ; in our case,  $p = 3$  (the three parameters of the BP distribution to be estimated,  $\alpha, \beta, k$ ). As it was mentioned in the previous section, although the number of parameters of the BP distribution is 4 (the parameters are  $\alpha, \beta, k$  and  $\theta$ ), the MLE of the fourth parameter  $\theta$  can be immediately obtained as the first order statistics,  $\hat{\theta} = x_{(1)}$ . For this reason, although the function  $l$  has 4 arguments, in the optimization methods only 3 will be in fact involved.

##### 3.1.1. Newton-Raphson’s Method

*Newton’s method:* Let us denote by  $V := (\alpha, \beta, k, \theta)$  a current point of the function  $l$ , let  $\nabla l(V_k)$  be the gradient and  $H_k$  be the Hessian function at an iteration  $V_k$  of the current point  $V$ . Newton’s method consists in taking the descent direction

$$d_k = -H_k^{-1} \nabla l(V_k).$$

*Near-Newton method:* Often in practice, the inverse of the Hessian,  $H_k^{-1}$ , is very difficult to evaluate when the function  $l$  is not analytic. The gradient is always more or less accessible (by inverse methods). As the Hessian cannot be computed exactly, we try to evaluate an approximation. Among the methods that approximate the Hessian, three are retained here: the method BFGS (for Broyden-Fletcher-Goldfarb-Shanno), the method DFP (for Davidon-Fletcher-Powell) and the method SR1 (for Symmetric Rank 1 method) [8].

We introduce the notation  $s_k = V_{k+1} - V_k$  and  $y_k = \nabla l(V_{k+1}) - \nabla l(V_k)$  and we choose  $H_0$  a definite positive matrix; for convexity reasons the identity matrix is usually chosen.

*Update of the Hessian by the method BFGS:* In this approach, the approximation of the Hessian is given by

$$H_{k+1} = H_k + \frac{y_k y_k^T}{y_k^T s_k} - \frac{H_k s_k s_k^T H_k}{s_k^T H_k s_k}.$$

##### 3.1.2. Gradient Method

This algorithm is based on the fact that, in the vicinity of a point  $V$ , the function  $l$  decreases most strongly in the opposed direction of the gradient of  $l$ ,

$$d = -\nabla l(V).$$

Let us fix an arbitrary  $\epsilon > 0$ . The algorithm can be described as follows:

1. **Step 0 (Initialization):**  $V_0 = (\alpha_0, \beta_0, k_0, \theta_0)$  that satisfies the conditions for the parameters of the BP distribution; set  $k = 0$  and go to **Step 1**
2. **Step 1:** Computation of  $d_k = -\nabla l(V_k)$ ; if  $\| \nabla l(V_k) \| \leq \epsilon$ , then STOP; If not, go to Step 2.

3. **Step 2:** Compute  $\lambda_k$  the solution of the problem

$$\min_{\lambda \in \mathbb{R}^+} l(V_k + \lambda d_k)$$

and  $V_{k+1}$ ,

$$V_{k+1} = V_k - \lambda_k \nabla l(V_k).$$

Set  $k = k + 1$  and go to **Step 1**.

The algorithm that uses this direction of descent is called the gradient algorithm or the deepest descent algorithm. The algorithm generally requires a finite number of iterations, depending on the regularity of the function  $l$ . In practice, we often observe that  $\nabla l(V)$  is a good direction of descent, but the convergence to the solution is generally slow. Despite its poor numerical performance, this algorithm is worth studying.

### 3.1.3. Conjugate Gradient Method

The conjugate gradient method is one of the most famous and widely used methods for solving minimization problems. It is mostly used for big size problems. This method was proposed in 1952 by Hestenes and Steifel for the minimization of strictly convex quadratic functions [9].

Many mathematicians have used this method for the nonlinear (non-quadratic) case. This was done for the first time in 1964 by Fletcher and Reeves [10], then in 1969 by Polak and Ribiere [11]; another variant was studied in 1987 by Fletcher [12]. The strategy adopted was to use a recursive sequence

$$V_{k+1} = V_k + \lambda_k d_k,$$

where  $\lambda_k$  is a properly chosen positive real constant called “step” and  $d_k$  is a non-zero real vector called “direction”. As the conjugate gradient algorithm is used to solve nonlinear functions, we should note that nonlinear conjugate gradient methods are not unique.

Algorithm for the conjugate gradient method:

The Algorithm is initialized by the step 0 of the simple gradient.

0. We have  $d_0 = -\nabla l(V_0)$  as long as a convergence criterion is not verified.

1. Determination of a step  $\lambda_k$  by some linear search method. Computation of a new iteration

$$V_{k+1} = V_k + \lambda_k d_k.$$

2. Evaluation of a new gradient

$$\nabla l(V_{k+1}).$$

3. Computation of the real  $\omega_{k+1}$  by some methods, for example Fletcher and Reeves (see below),

$$\omega_{k+1} = \frac{\nabla l(V_{k+1})^T \nabla l(V_{k+1})}{\nabla l(V_k)^T \nabla l(V_k)}. \tag{5}$$

4. Construction of a new descent direction:

$$d_{k+1} = -\nabla l(V_{k+1}) + \omega_{k+1} d_k.$$

5. Increment

$$k = k + 1.$$

Several methods exist for computing the term  $\omega_{k+1}$ ; we will be concerned in the sequel by the methods of Fletcher-Reeves, of Polack-Ribiere, and of Hestenes-Stiefel (a variant of the method of Fletcher-Reeves). It should be noted that this last method is particularly effective in the case when the

function is quadratic and when the linear search is carried out exactly. The corresponding  $\omega_{k+1}$  for these methods is given by:

*Fletcher-Reeves*

$$\omega_{k+1} = \frac{\nabla l(V_{k+1})^T \nabla l(V_{k+1})}{\nabla l(V_k)^T \nabla l(V_k)};$$

*Polack-Ribiere*

$$\omega_{k+1} = \frac{\nabla l(V_{k+1})^T (\nabla l(V_{k+1}) - \nabla l(V_k))}{\nabla l(V_k)^T \nabla l(V_k)};$$

*Hestenes-Stiefel*

$$\omega_{k+1} = \frac{\nabla l(V_{k+1})^T (\nabla l(V_{k+1}) - \nabla l(V_k))}{(\nabla l(V_{k+1}) - \nabla l(V_k))^T \nabla l(V_k)}.$$

Noting that

$$l(V_{k+1}) = l(V_k) + (V_{k+1} - V_k) \frac{dl}{dV_k}(V_k) + Rest(V_{k+1} - V_k), \tag{6}$$

we have the recurrence

$$V_{k+1} = V_k + \lambda_k d_k,$$

where the step of descent  $\lambda_k$  is obtained with an exact or inaccurate linear search. The descent vector  $d_k$  is obtained by using a conjugate gradient algorithm based on the recurrence formula

$$d_k = -\nabla l(V_k) - \omega_k d_{k-1},$$

where

$$\omega_k = \frac{\|\nabla l(V_k)\|^2}{\|\nabla l(V_{k-1})\|^2}.$$

### 3.2. Optimization Methods for Computing the MLEs of the BP Distribution

At this stage, we are interested in describing the application of the three optimization methods for computing the MLEs of the BP distribution. In the sequel we will present in details only the most complex one, namely the conjugate gradient method. Following the same line, the other two methods can also be adapted for computing the MLEs of the BP distribution. Note also that we will give in the next section numerical results for all three methods.

So, our purpose is to numerically determine the MLEs of the BP parameters by calculating the maximum of the log-likelihood function  $l(\alpha, \beta, k, \theta)$  given in (1).

As we have previously mentioned, the MLE of the parameter  $\theta$  is the first order statistic, so, in the sequel, the iterates  $\theta_0 = \theta_1 = \dots = x_{(1)}$  are fixed.

To compute the MLEs of the parameters, we proceed as follows:

1. **Step 0:** Initialization  $V_0 = (\alpha_0, \beta_0, k_0, \theta_0)$ .
2. **Step 1:** Computation of the gradient

$$\begin{aligned} \nabla l(\alpha, \beta, k, \theta) &= \begin{pmatrix} \frac{\partial l(\alpha, \beta, k, \theta)}{\partial \alpha} \\ \frac{\partial l(\alpha, \beta, k, \theta)}{\partial \beta} \\ \frac{\partial l(\alpha, \beta, k, \theta)}{\partial k} \end{pmatrix} \\ &= \begin{pmatrix} n [\Psi(\alpha + \beta) - \Psi(\alpha)] + \sum_{i=1}^n \ln \left( 1 - \left( \frac{x_i}{\theta} \right)^{-k} \right) \\ n [\Psi(\alpha + \beta) - \Psi(\beta)] - k \sum_{i=1}^n \ln \left( \frac{x_i}{\theta} \right) \\ \frac{n}{k} - \sum_{i=1}^n \left[ \beta + (\alpha - 1) \left( 1 - \left( \frac{x_i}{\theta} \right)^k \right)^{-1} \right] \ln \left( \frac{x_i}{\theta} \right) \end{pmatrix}. \end{aligned}$$

If  $\nabla l(\alpha, \beta, k, \theta) \leq \epsilon$ , then stop,  $V_0 = (\alpha_0, \beta_0, k_0, \theta_0)$  is the optimal vector.

**Else:** go to the next step.

**3. Step 2:** Computation of the direction of descent

If  $i = 0$  :

$$d_0 = -\nabla l(\alpha_0, \beta_0, k_0, \theta_0) = - \begin{pmatrix} n [\Psi(\alpha_0 + \beta_0) - \Psi(\alpha_0)] + \sum_{i=1}^n \ln \left( 1 - \left( \frac{x_i}{\theta_0} \right)^{-k_0} \right) \\ n [\Psi(\alpha_0 + \beta_0) - \Psi(\beta_0)] - k_0 \sum_{i=1}^n \ln \left( \frac{x_i}{\theta_0} \right) \\ \frac{n}{k_0} - \sum_{i=1}^n \left[ \beta_0 + (\alpha_0 - 1) \left( 1 - \left( \frac{x_i}{\theta_0} \right)^{k_0} \right)^{-1} \right] \ln \left( \frac{x_i}{\theta_0} \right) \end{pmatrix}.$$

If  $i > 0$  :  $d_{i+1} = -\nabla l(\alpha_i, \beta_i, k_i, \theta_i) + \omega_i d_i$ ; when we use the method of Fletcher-Reeves (5), we will have the result

$$\omega_i^{FR} = \frac{\|\nabla l(\alpha_{i+1}, \beta_{i+1}, k_{i+1}, \theta_{i+1})\|^2}{\|\nabla l(\alpha_i, \beta_i, k_i, \theta_i)\|^2}.$$

**4. Step 3:** Computation of  $V_1 = (\alpha_1, \beta_1, k_1, \theta_1)$

$V_1 = V_0 + \lambda_0 d_0$  such as  $d_0 = -\nabla l(\alpha_0, \beta_0, k_0, \theta_0)$ .

**Determination of a step  $\lambda_0$**

We can find  $\lambda_0$  with exact and inaccurate linear search methods. In our case, the use of the exact linear search helps us to have a fast convergence. The exact linear search method is to solve the problem

$$\min_{\lambda_0 \in \mathbb{R}^+} l(V_0 + \lambda_0 d_0). \tag{7}$$

For this, we will look for the value of  $\lambda_0$  which cancels the first derivative of the function  $l(V_0 + \lambda_0 d_0)$ ,



$$\begin{aligned} \alpha_0 - \lambda_0 \left\{ n [\Psi(\alpha_0 + \beta_0) - \Psi(\alpha_0)] + \sum_{i=1}^n \ln \left( 1 - \left( \frac{x_i}{\theta_0} \right)^{-k_0} \right) \right\} &= 0, \\ \beta_0 - \lambda_0 \left\{ n [\Psi(\alpha_0 + \beta_0) - \Psi(\beta_0)] - k_0 \sum_{i=1}^n \ln \left( \frac{x_i}{\theta_0} \right) \right\} &= 0, \\ k_0 - \lambda_0 \left\{ \frac{n}{k_0} - \sum_{i=1}^n \left[ \beta_0 + (\alpha_0 - 1) \left( 1 - \left( \frac{x_i}{\theta_0} \right)^{k_0} \right)^{-1} \right] \ln \left( \frac{x_i}{\theta_0} \right) \right\} &= 0. \end{aligned}$$

Thus we can deduce the exact value of  $\lambda_0$

$$\begin{aligned} \lambda_0 &= (\alpha_0 + \beta_0 + k_0) / \left[ 2n\Psi(\alpha_0 + \beta_0) - n\Psi(\alpha_0) - n\Psi(\beta_0) - k_0 \sum_{i=1}^n \ln \left( \frac{x_i}{\theta_0} \right) \right. \\ &\quad \left. + \sum_{i=1}^n \ln \left( 1 - \left( \frac{x_i}{\theta_0} \right)^{-k_0} \right) + \frac{n}{k_0} - \sum_{i=1}^n \left[ \beta_0 + (\alpha_0 - 1) \left( 1 - \left( \frac{x_i}{\theta_0} \right)^{k_0} \right)^{-1} \right] \ln \left( \frac{x_i}{\theta_0} \right) \right]. \end{aligned}$$

If  $\lambda_0$  is positive, we accept it and we go to next step. If not, we use inexact methods of linear search to have an approximation of the optimal value  $\lambda_0$  and we go to next step. The main inexact methods are the so-called inaccurate linear search methods of Armijo [13], of Goldstein [14], of Wolfe [15] and of Wolfe strong [16].

**Construction of the vector  $V_1$**

$$V_1 = V_0 + \lambda_0 d_0 = \begin{pmatrix} \alpha_0 + \lambda_0 \left[ -n\{\Psi(\alpha_0 + \beta_0) - \Psi(\alpha_0)\} - \sum_{i=1}^n \ln(1 - (\frac{x_i}{\theta_0})^{-k_0}) \right] \\ \beta_0 + \lambda_0 \left[ -n\{\Psi(\alpha_0 + \beta_0) - \Psi(\beta_0)\} + k_0 \sum_{i=1}^n \ln(\frac{x_i}{\theta_0}) \right] \\ k_0 + \lambda_0 \left[ -\frac{n}{k_0} + \sum_{i=1}^n \left[ \beta_0 + (\alpha_0 - 1) \left( 1 - (\frac{x_i}{\theta_0})^{k_0} \right)^{-1} \right] \ln(\frac{x_i}{\theta_0}) \right] \end{pmatrix},$$

where  $\lambda_0$  was previously obtained.

Set  $i = i + 1$  and go to Step 1.

Based on Zoutendijk’s theorem [17] and globally convergent Riemannian conjugate gradient method [18], the convergence of the algorithm is ensured.

**3.3. Numerical Results**

We carried out an important simulation study using the programming software R. In the sequel we present the results obtained by means of the three optimization methods.

In Tables 1–3, the MLEs of the parameters  $\alpha$ ,  $\beta$  and  $k$  of the BP distribution are presented, together with the standard deviations (SDs), considering the three optimization methods. Note the convergence of the estimators obtained with the gradient and conjugate gradient methods and note also that the Newton’s method does not converge with a non-quadratic or nonlinear function.

The Tables 4–6 present the MLEs of the parameters for the conjugate gradient method and gradient methods, as well as the bias and the mean square errors (MSEs) of the estimators. Note that the MSEs of the estimators have very small values, smaller than  $10^{-4}$ .

**Table 1.** Results of simulation with conjugate gradient method for  $m = 10,000$  iterations, with  $\alpha = 23$ ,  $\beta = 34, k = 19, \theta = 0.15$ .

$n$	$\hat{\alpha}$	$\hat{\beta}$	$\hat{k}$	$SD(\hat{\alpha})$	$SD(\hat{\beta})$	$SD(\hat{k})$
10	22.7873	33.7865	19.9800	$1.06 \times 10^{-4}$	$1.06 \times 10^{-4}$	$9.44 \times 10^{-6}$
15	22.7910	33.7902	19.9802	$1.04 \times 10^{-4}$	$1.04 \times 10^{-4}$	$9.47 \times 10^{-6}$
20	22.7947	33.7939	19.9804	$1.02 \times 10^{-4}$	$1.04 \times 10^{-4}$	$1.11 \times 10^{-5}$
30	22.8020	33.8012	19.9808	$9.89 \times 10^{-5}$	$9.97 \times 10^{-5}$	$1.01 \times 10^{-5}$
40	22.8094	33.8086	19.9812	$9.52 \times 10^{-5}$	$9.62 \times 10^{-5}$	$1.02 \times 10^{-5}$
50	22.8168	33.8160	19.9906	$9.15 \times 10^{-5}$	$9.17 \times 10^{-5}$	$4.26 \times 10^{-6}$
100	22.8536	33.8528	19.9926	$7.31 \times 10^{-5}$	$7.35 \times 10^{-5}$	$3.72 \times 10^{-6}$
150	22.8904	33.8896	19.9946	$5.47 \times 10^{-5}$	$5.52 \times 10^{-5}$	$2.83 \times 10^{-6}$
200	22.9273	33.9265	19.9966	$3.63 \times 10^{-5}$	$3.65 \times 10^{-5}$	$1.45 \times 10^{-6}$
300	23.0009	34.0001	20.0006	$4.78 \times 10^{-7}$	$4.01 \times 10^{-7}$	$8.09 \times 10^{-7}$

**Table 2.** Results of simulation with gradient method for  $m = 10,000$  iterations, with  $\alpha = 23, \beta = 34, k = 19, \theta = 0.15$ .

$n$	$\hat{\alpha}$	$\hat{\beta}$	$\hat{k}$	$SD(\hat{\alpha})$	$SD(\hat{\beta})$	$SD(\hat{k})$
10	22.7873	33.7865	19.7790	$2.12 \times 10^{-5}$	$2.12 \times 10^{-5}$	$2.20 \times 10^{-5}$
15	22.7910	33.7902	19.7992	$2.08 \times 10^{-5}$	$2.09 \times 10^{-5}$	$1.99 \times 10^{-5}$
20	22.7947	33.7939	19.7994	$2.05 \times 10^{-5}$	$2.06 \times 10^{-5}$	$2.00 \times 10^{-5}$
30	22.8020	33.8012	19.7998	$1.97 \times 10^{-5}$	$1.97 \times 10^{-5}$	$1.99 \times 10^{-5}$
40	22.8094	33.8086	19.8002	$1.90 \times 10^{-5}$	$1.90 \times 10^{-5}$	$1.98 \times 10^{-5}$
50	22.8168	33.8160	19.8126	$1.83 \times 10^{-5}$	$1.83 \times 10^{-5}$	$1.86 \times 10^{-5}$
100	22.8536	33.8528	19.9026	$1.46 \times 10^{-5}$	$1.47 \times 10^{-5}$	$9.76 \times 10^{-6}$
150	22.8904	33.8896	19.9046	$1.09 \times 10^{-5}$	$1.11 \times 10^{-5}$	$9.69 \times 10^{-6}$
200	22.9273	33.9265	19.9066	$7.26 \times 10^{-6}$	$7.32 \times 10^{-6}$	$9.30 \times 10^{-6}$
300	23.0009	34.0001	19.9106	$9.56 \times 10^{-8}$	$8.21 \times 10^{-8}$	$8.83 \times 10^{-6}$

**Table 3.** Results of simulation with Newton’s method for  $m = 10,000$  iterations, with  $\alpha = 23, \beta = 34, k = 19, \theta = 0.15$ .

$n$	$\hat{\alpha}$	$\hat{\beta}$	$\hat{k}$	$SD(\hat{\alpha})$	$SD(\hat{\beta})$	$SD(\hat{k})$
10	22.5508	33.4207	19.6611	$2.01 \times 10^{-2}$	$3.35 \times 10^{-2}$	$1.14 \times 10^{-2}$
15	22.5508	33.4207	19.6611	$1.34 \times 10^{-2}$	$2.23 \times 10^{-2}$	$7.65 \times 10^{-3}$
20	22.5508	33.420	19.6684	$1.01 \times 10^{-2}$	$1.67 \times 10^{-2}$	$5.49 \times 10^{-3}$
30	22.7707	33.8507	19.8198	$6.72 \times 10^{-3}$	$7.42 \times 10^{-4}$	$1.08 \times 10^{-3}$
40	22.7707	33.8507	19.8243	$5.04 \times 10^{-3}$	$5.56 \times 10^{-4}$	$7.71 \times 10^{-4}$
50	22.7707	33.8507	19.8239	$4.03 \times 10^{-3}$	$4.45 \times 10^{-4}$	$1.03 \times 10^{-4}$
100	22.7707	33.8507	19.8198	$2.01 \times 10^{-3}$	$2.22 \times 10^{-4}$	$1.03 \times 10^{-4}$
150	22.7707	33.8507	19.8197	$1.34 \times 10^{-3}$	$1.48 \times 10^{-4}$	$2.16 \times 10^{-4}$
200	22.7707	33.8507	19.8197	$2.62 \times 10^{-4}$	$1.11 \times 10^{-4}$	$1.62 \times 10^{-4}$
300	22.7707	33.8507	19.8197	$1.75 \times 10^{-4}$	$7.42 \times 10^{-5}$	$1.08 \times 10^{-4}$

**Table 4.** MLEs of  $\alpha$  with the conjugate gradient method and gradient method.

$\alpha$				
		$\hat{\alpha}$ (bias)	MSE	
n	$\hat{\alpha}_{GC}$ (bias)	$\hat{\alpha}_G$ (bias)	$MSE_{GC}$	$MSE_G$
10	22.787 (0.220)	23.009786028 (−0.0097860)	$1.36 \times 10^{-9}$	$3.58 \times 10^{-7}$
20	22.794 (0.205)	23.009786227 (−0.0097862)	$5.44 \times 10^{-9}$	$3.58 \times 10^{-9}$
30	22.802 (0.197)	23.009786057 (−0.0097860)	$1.22 \times 10^{-8}$	$3.58 \times 10^{-7}$
40	22.802 (0.197)	23.009786022 (−0.0097860)	$1.22 \times 10^{-8}$	$3.58 \times 10^{-7}$
50	22.809 (0.190)	23.009786225 (−0.0097862)	$9.89 \times 10^{-11}$	$3.58 \times 10^{-7}$
100	22.809 ( $3.4 \times 10^{-8}$ )	23.009786237 (−0.009786)	$9.89 \times 10^{-11}$	$3.58 \times 10^{-7}$
150	22.816 (0.109)	23.009786240 (−0.009786)	$3.06 \times 10^{-7}$	$3.58 \times 10^{-7}$
200	22.890 (0.072)	23.009786240 (−0.009786)	$5.44 \times 10^{-7}$	$3.58 \times 10^{-7}$
300	23.001 (−0.001)	23.009786241 (−0.009786)	$1.22 \times 10^{-6}$	$3.58 \times 10^{-7}$
1000	23.002 (−0.002)	23.009786241 (−0.009786)	$1.37 \times 10^{-5}$	$3.58 \times 10^{-7}$

**Table 5.** MLEs of  $\beta$  with the conjugate gradient method and gradient method.

$\beta$				
		$\hat{\beta}$ (bias)	MSE	
n	$\hat{\beta}_{GC}$ (bias)	$\hat{\beta}_G$ (bias)	$MSE_{GC}$	$MSE_G$
10	33.786 (0.006)	33.9995 (0.000409)	$1.27 \times 10^{-9}$	$3.57 \times 10^{-7}$
20	33.793 (0.206)	33.9997 (0.000225)	$5.04 \times 10^{-9}$	$3.58 \times 10^{-9}$
30	33.801 (0.198)	33.9996 (0.000348)	$8.76 \times 10^{-9}$	$3.57 \times 10^{-7}$
40	33.801 (0.198)	33.9996 (0.000364)	$1.22 \times 10^{-8}$	$3.57 \times 10^{-7}$
50	33.808 (0.191)	33.9997 (0.000229)	$2.13 \times 10^{-8}$	$3.58 \times 10^{-7}$
100	33.816 (0.183)	33.9997 (0.00022)	$1.65 \times 10^{-10}$	$3.58 \times 10^{-7}$
150	33.889 (0.110)	33.9997 (0.000217)	$3.06 \times 10^{-7}$	$3.58 \times 10^{-7}$
200	33.926 (0.073)	33.9997 (0.000218)	$5.41 \times 10^{-7}$	$3.58 \times 10^{-7}$
300	34.000 (−0.0006)	33.9997 (0.000217)	$1.21 \times 10^{-6}$	$3.58 \times 10^{-7}$
1000	34.001 (0.001)	33.9997 (0.000217)	$1.37 \times 10^{-5}$	$3.58 \times 10^{-7}$

**Table 6.** MLEs of  $k$  with the conjugate gradient method and gradient method.

$k$				
		$\hat{k}$ (bias)	MSE	
n	$\hat{k}_{GC}$ (bias)	$\hat{k}_G$ (bias)	$MSE_{GC}$	$MSE_G$
10	19.989 (0.0001)	20.046 (−0.046)	$1.98 \times 10^{-11}$	$4.40 \times 10^{-8}$
20	19.989 (0.0105)	20.037 (−0.037)	$1.48 \times 10^{-11}$	$3.50 \times 10^{-9}$
30	19.989 (0.0101)	20.042 (−0.042)	$1.74 \times 10^{-10}$	$3.82 \times 10^{-7}$
40	19.989 (0.0102)	20.041 (−0.041)	$1.43 \times 10^{-10}$	$3.77 \times 10^{-7}$
50	19.990 ( 0.0096)	20.037 (0.037)	$1.00 \times 10^{-10}$	$3.52 \times 10^{-7}$
100	19.990 (0.0094)	20.037 (0.037)	$9.89 \times 10^{-11}$	$3.53 \times 10^{-7}$
150	19.994 (0.0054)	20.037 (0.037)	$1.65 \times 10^{-10}$	$3.53 \times 10^{-7}$
200	19.996 (0.0032)	20.037 (0.037)	$1.16 \times 10^{-9}$	$3.53 \times 10^{-7}$
300	23.001 (−0.0014)	20.037 (−0.037)	$1.67 \times 10^{-9}$	$3.58 \times 10^{-7}$
1000	20.000 (−0.0006)	20.037 (−0.037)	$3.10 \times 10^{-9}$	$3.53 \times 10^{-7}$

The interest of the method of conjugated gradient comes from the fact that it converges quickly towards the minimum; we can show that in  $N$  dimensions it only needs maximum  $N$  calculation steps, if the function is exactly quadratic. The inconvenient of the Newton’s method is that it requires to know the Hessian of the function in order to determine the descent step. We have seen that the conjugate gradient algorithm chooses optimally the descent directions through  $V(\alpha, \beta, k, \theta)$ .

In Figures 6–8 we also present the evolution of the MSEs of the computed estimators. The three numerical methods are carried out for  $m = 10,000$  iterations.

As we see in these figures, the MLEs of the parameters  $\alpha, \beta$ , and  $k$  obtained by Newton's, gradient and conjugate gradient methods are  $\sqrt{n}$ -consistent. And we can say that the conjugate gradient method gives the best results. In this way, we have numerically checked the well known properties of MLEs, namely the consistency and asymptotic normality, by verifying that the values of the mean square errors obtained are  $\sqrt{n}$ -consistent, which confirm the theory of the method used.

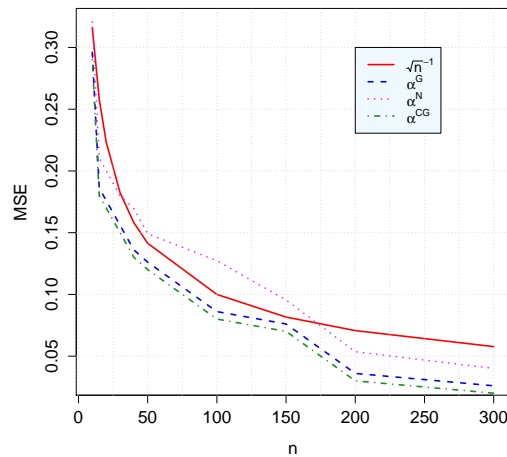


Figure 6. The MSE of  $\hat{\alpha}$  computed with Newton's, gradient and conjugate gradient methods.

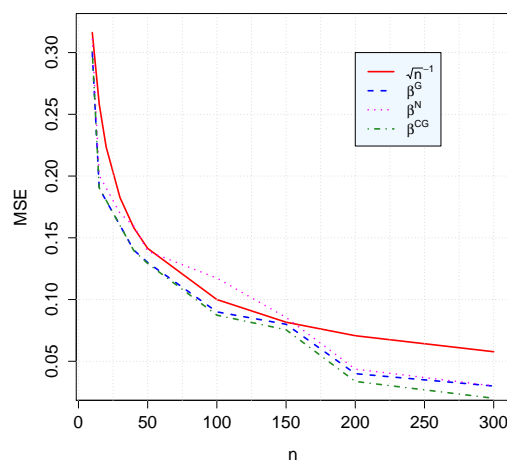


Figure 7. The MSE of  $\hat{\beta}$  computed with Newton's, gradient and conjugate gradient methods.

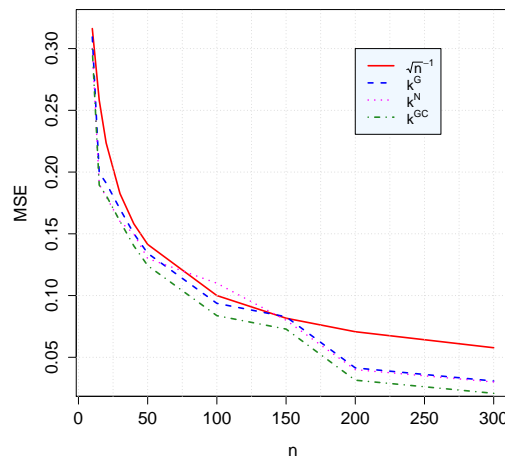


Figure 8. The MSE of  $\hat{k}$  computed with Newton’s, gradient and conjugate gradient methods.

#### 4. Model Selection

We want also to take into account model selection criteria in order to choose among several candidate models, BP, Beta, Pareto, Gamma and Generalized Beta-Pareto distributions. The Generalized BP (GBP) distribution is a 5-parameter distribution introduced in Reference [19], with a different form given in Reference [20]. The density of this distribution is defined by

$$g(x; \alpha, \beta, k, \theta, c) := \frac{c}{B(\alpha, \beta)} \left[ 1 - \left( \frac{\theta}{x} \right)^k \right]^{c\theta-1} \left[ 1 - \left( 1 - \left( \frac{\theta}{x} \right)^k \right)^c \right]^{\beta-1} \frac{k}{\theta} \left( \frac{\theta}{x} \right)^{k+1}, \quad x \geq \theta > 0, \quad (8)$$

with the parameters  $\alpha, \beta, \theta, c, k > 0$ .

For the model selection problem, we considered the general information criterion (GIC)

$$GIC(K) = -2 \ln L + I \times K,$$

where:  $L$  is the likelihood,  $K$  is the number of parameters of the model,  $I$  is an index for the penalty of the model. The following well-known criteria are obtained for different values of  $I$ :

$$AIC(K) = -2 \ln L + 2K$$

$$BIC(K) = -2 \ln L + \ln(n)K$$

and

$$AICc(K) = AIC(K) + \frac{2K(K+1)}{n-K-1},$$

where  $n$  is the sample size.

We have simulated data according to BP distribution, Pareto (P) distribution, Gamma (G) distribution and GBP distribution. We have computed the three criteria AIC, BIC and AICc and recorded in Tables 7–10 the number of times in 1000 iterations that each of the models is selected (minimum value of the corresponding criteria).

Several remarks need to be done. First, since the support of a BP distribution is  $[\theta, +\infty)$ , it makes sense to compare data from BP distribution with data from Gamma, Pareto and GBP distributions. For example, when  $\theta$  is close to 0, the support of the BP distribution is close to the support of Gamma distribution. BP distribution reduces to Pareto distribution in the case where  $\alpha = \beta = 1$ , and the support is the same.

Second, note that we have added the Beta (B) distribution as one of the candidate distributions to be chosen by the information criteria. Although this is not a “real” candidate (since the support is  $(0, 1)$ ), we wanted to check also the model selection criteria in the presence of an “unusual” model. Clearly, it does not make sense to compare data from BP, GBP, Gamma or Pareto distribution, on the one side, with data from Beta distribution, on the other side. Nonetheless, it is important to have an idea about how the model criteria behave in this case.

We can notice that all criteria choose the correct model in all the 4 situations, for moderate or even small values of the sample size.

We can also remark that, when the underlying model is the GBP, for small values of the sample size  $n$  (less than 50), the criteria fail in choosing the correct model; nonetheless, starting from reasonable values of the sample size  $n$  (around 50 or 100), the correct model is chosen by all criteria in most of the cases. This phenomenon could be generated by the fact that the number of parameters in the GBP model is the highest one, 5, which has an influence on the information criteria.

We have also noticed a strange phenomenon: also when the underlying model is the GBP, for small values of the sample size  $n$ , the model preferred by the criteria is the Beta model instead of the GBP, that is to say the “unusual” model, as previously explained. The fact is more accentuated for the BIC criteria, so surely it is related to the penalization due to the number of parameters.

Another remark related to the difference between BP and GBP models needs to be done. When comparing Tables 7 and 10, we can notice an asymmetry between the model selection criteria when the underlying true models are the BP, respectively the GBP model. On the one side, when the underlying model is the GBP (Table 7), the BP model is chosen by all criteria in 15–20% of cases for small values of the sample size. On the other side, when the underlying model is the BP (Table 10), the GBP model is never chosen. Although we think that this phenomenon could be related to the number of parameters of each model, to the presence of the other models investigated by the criteria as candidate models (Pareto, Beta, Gamma), or to the particular case that we considered in simulations, we do not have a complete explanation for this phenomenon.

**Table 7.** Smaller AIC/BIC/AICc scores in 1000 simulations from a GBP distribution  $GBP(\alpha, \beta, k, \theta, c)$  with  $\alpha = 2.7, \beta = 1.2, k = 5.5, \theta = 0.15, c = 0.825$ .

n	AIC					BIC					AICc				
	BP	B	P	G	GBP	BP	B	P	G	GBP	BP	B	P	G	GBP
10	222	712	21	19	26	229	719	13	17	22	227	753	5	3	12
20	156	784	13	17	30	179	710	7	13	91	176	784	2	3	30
30	145	567	10	15	263	155	603	4	11	227	151	784	1	7	57
50	96	278	8	12	606	106	307	3	7	577	111	604	0	5	244
100	43	48	0	0	909	43	60	0	0	897	43	230	0	0	727
200	38	0	0	0	962	38	0	0	0	962	38	0	0	0	959
300	20	0	0	0	980	20	0	0	0	980	20	0	0	0	980
500	5	0	0	0	995	5	0	0	0	995	5	0	0	0	995
1000	0	0	0	0	1000	0	0	0	0	1000	0	0	0	0	1000

**Table 8.** Smaller AIC/BIC/AICc scores in 1000 simulations from a Pareto distribution  $P(k, \theta)$  with  $k = 0.075, \theta = 0.15$ .

n	AIC					BIC					AICc				
	BP	B	P	G	GBP	BP	B	P	G	GBP	BP	B	P	G	GBP
70	26	7	783	3	181	21	11	915	5	48	23	10	964	2	1
80	12	3	786	0	199	19	2	909	1	69	13	4	976	1	6
90	4	0	780	0	216	7	0	912	0	81	3	0	997	0	10
100	0	0	970	0	30	0	0	995	0	5	0	0	992	0	8
150	0	0	974	0	24	0	0	996	0	4	0	0	997	0	3
200	0	0	980	0	20	0	0	996	0	4	0	0	998	0	2
300	0	0	984	0	16	0	0	997	0	3	0	0	998	0	2
500	0	0	992	0	8	0	0	995	0	5	0	0	998	0	2
1000	0	0	1000	0	0	0	0	1000	0	0	0	0	1000	0	0

**Table 9.** Smaller AIC/BIC/AICc scores in 1000 simulations from a Gamma distribution  $G(\alpha, \beta)$  with  $\alpha = 0.5, \beta = 0.9$ .

n	AIC					BIC					AICc				
	BP	B	P	G	GBP	BP	B	P	G	GBP	BP	B	P	G	GBP
10	0	9	2	988	1	0	9	2	988	1	0	9	2	989	0
20	3	13	9	947	28	4	21	12	972	3	2	7	6	600	385
30	2	12	5	893	85	4	13	9	957	17	1	6	5	793	195
50	1	11	5	823	160	1	8	4	962	25	0	4	2	950	44
100	0	9	3	779	209	0	5	1	955	72	0	1	0	980	19
150	0	0	0	807	193	0	0	0	914	86	0	0	0	977	23
200	0	0	0	776	224	0	0	0	898	102	0	0	0	988	12
500	0	0	0	843	157	0	0	0	906	94	0	0	0	998	2
1000	0	0	0	915	85	0	0	0	949	51	0	0	0	998	2
2000	0	0	0	975	25	0	0	0	982	18	0	0	0	998	2

**Table 10.** Smaller AIC/BIC/AICc scores in 1000 simulations from a BP distribution  $BP(\alpha, \beta, k, \theta)$  with  $\alpha = 0.5, \beta = 0.9, k = 0.8, \theta = 0.15$ .

n	AIC					BIC					AICc				
	BP	B	P	G	GBP	BP	B	P	G	GBP	BP	B	P	G	GBP
10	560	19	0	421	0	558	20	0	422	0	525	29	0	446	0
20	648	1	0	351	0	645	1	0	354	0	620	3	0	377	0
30	48	1	0	351	0	645	1	0	354	0	620	3	0	377	0
50	648	1	0	351	0	645	1	0	354	0	620	3	0	377	0
100	873	0	0	127	0	871	0	0	129	0	820	0	0	180	0
150	890	0	0	110	0	888	0	0	112	0	844	0	0	156	0
200	931	0	0	69	0	931	0	0	69	0	892	0	0	108	0
500	962	0	0	38	0	962	0	0	38	0	928	0	0	72	0
1000	999	0	0	1	0	999	0	0	1	0	998	0	0	2	0
2000	1000	0	0	0	0	1000	0	0	0	0	1000	0	0	0	0

### 5. MLEs for Right-Censored Data

Let us consider that the random right-censorship  $C$  is non-informative. Roughly speaking, a censorship is non-informative in the sense that the censoring distribution provides no information about the lifetime distribution. If the censoring mechanism is independent of the survival time, then we will have non-informative censoring. In fact, in practice we almost always mean independent censoring by non-informative censoring. It is well known that if the censoring variable  $C$  depends on the lifetime  $X$ , we run into the problem of non-identifiability: starting from observations, we cannot make inference about  $X$  or  $C$ , because several different distributions for  $(X, C)$  could provide the same distribution for the observation. For these reasons, researchers almost always consider independence between censoring and lifetime. In most practical examples this makes sense; for instance, if the lifetime represents the remission time of patients and the censoring mechanism comes from loss to follow-up (patients change town, for instance), it is natural to consider that we have independence between censoring and lifetime.

In our case, suppose that the variables  $X$  and  $C$  have respectively the probability density functions  $f$  and  $g$  and survival functions  $S$  and  $G$ ; all the information is contained in the couple  $(T_j, \delta_j)$  where  $T_j = \min(X_j, C_j)$  is the observed time and  $\delta_j = 1_{(X_j \leq C_j)}$  is the censorship indicator. So, the contribution to the likelihood for the individual  $j$  is

$$L_j = [f(t_j|V)G(t_j)]^{\delta_j} [g(t_j)S(t_j|V)]^{1-\delta_j}.$$

Note that the term  $[f(t_j|V)G(t_j)]^{\delta_j}$  corresponds to the case  $\delta_j = 1$ , that is to say to the case when the lifetime is observed; in this situation, the contribution to the likelihood of the lifetime  $X$  is  $f(t_j|V)$ , while the contribution to the likelihood of the censorship is  $G(t_j)$ . The second term has an analogous interpretation in the case when  $\delta_j = 0$ , that is to say in the case when the censorship is observed.

We also assume that there are no common parameters between the censoring and the lifetime; consequently, the parameter  $V$  does not appear in the distribution of the censorship. The useful part of the likelihood (for obtaining the MLEs of interest, that is, the MLEs of the BP distribution) is then reduced to

$$L = \prod_{j=1}^m f(t_j|V)^{\delta_j} S(t_j|V)^{1-\delta_j}$$

and the log-likelihood is

$$\begin{aligned} l &:= \ln L = \sum_{j=1}^m \delta_j \ln f(t_j|V) + \sum_{j=1}^m (1 - \delta_j) \ln S(t_j|V) \\ &= \sum_{j=1}^m \delta_j \left[ \ln \left( \frac{k}{\theta B(\alpha, \beta)} \right) + (\alpha - 1) \ln \left( 1 - \left( \frac{t_j}{\theta} \right)^{-k} \right) + (-k\beta - 1) \ln \left( \frac{t_j}{\theta} \right) \right] \\ &\quad + \sum_{j=1}^m (1 - \delta_j) \ln \left[ \frac{\left( \frac{t_j}{\theta} \right)^{-k\beta}}{\beta B(\alpha, \beta)} + \frac{\left( \frac{t_j}{\theta} \right)^\beta}{\beta B(\alpha + \beta)} \sum_{n=1}^{\infty} \frac{\prod_{i=1}^n (i - \alpha)}{n!(\beta + n)} \left( \frac{t_j}{\theta} \right)^{-nk} \right], \end{aligned}$$

where  $x \geq \theta$  and  $\alpha, \beta, k, \theta > 0$ . Consequently we get

$$\begin{aligned} l &= \ln \left( \frac{k}{\theta} \right) \sum_{j=1}^m \delta_j - m \ln(B(\alpha, \beta)) + \sum_{j=1}^m \delta_j (\alpha - 1) \ln \left( 1 - \left( \frac{t_j}{\theta} \right)^{-k} \right) - \sum_{j=1}^m \delta_j \ln \left( \frac{t_j}{\theta} \right) \\ &\quad - k\beta \sum_{j=1}^m \ln \left( \frac{t_j}{\theta} \right) + \sum_{j=1}^m (1 - \delta_j) \ln \left[ \beta^{-1} + \sum_{n=1}^{\infty} \frac{\prod_{i=1}^n (i - \alpha)}{n!(\beta + n)} \left( \frac{t_j}{\theta} \right)^{-nk} \right]. \end{aligned} \tag{9}$$



Setting

$$z(t_j, \alpha, \beta, k, \theta) := \beta^{-1} + \sum_{n=1}^{\infty} \frac{\prod_{i=1}^n (i - \alpha)}{n!(\beta + n)} \left(\frac{t_j}{\theta}\right)^{-nk}$$

and

$$\begin{aligned} z_{\alpha}(t_j, \alpha, \beta, k, \theta) &:= \frac{\partial z}{\partial \alpha}(t_j, \alpha, \beta, k, \theta), \\ z_{\beta}(t_j, \alpha, \beta, k, \theta) &:= \frac{\partial z}{\partial \beta}(t_j, \alpha, \beta, k, \theta), \\ z_k(t_j, \alpha, \beta, k, \theta) &:= \frac{\partial z}{\partial k}(t_j, \alpha, \beta, k, \theta), \end{aligned}$$

we obtain the score functions

$$\begin{aligned} \frac{\partial l(\alpha, \beta, k, \theta)}{\partial \alpha} &= m [\Psi(\alpha + \beta) - \Psi(\alpha)] + \sum_{j=1}^m \delta_j \ln \left[ 1 - \left(\frac{t_j}{\theta}\right)^{-k} \right] + \sum_{j=1}^m (1 - \delta_j) \frac{z_{\alpha}(t_j, \alpha, \beta, k, \theta)}{z(t_j, \alpha, \beta, k, \theta)} \\ \frac{\partial l(\alpha, \beta, k, \theta)}{\partial \beta} &= m [\Psi(\beta) - \Psi(\beta + \alpha)] - k \sum_{j=1}^m \ln \left(\frac{t_j}{\theta}\right) + \sum_{j=1}^m (1 - \delta_j) \frac{z_{\beta}(t_j, \alpha, \beta, k, \theta)}{z(t_j, \alpha, \beta, k, \theta)} \\ \frac{\partial l(\alpha, \beta, k, \theta)}{\partial k} &= k^{-1} \sum_{j=1}^m \delta_j - \sum_{j=1}^m \left[ \beta + \delta_j \ln \left(\frac{t_j}{\theta}\right) \left( 1 - \left(\frac{t_j}{\theta}\right)^k \right)^{-1} \right] \ln \left(\frac{t_j}{\theta}\right) + \sum_{j=1}^m (1 - \delta_j) \frac{z_k(t_j, \alpha, \beta, k, \theta)}{z(t_j, \alpha, \beta, k, \theta)}. \end{aligned}$$

### 5.1. Conjugate Gradient Method for Parameter MLEs of the BP Distribution with Right-Censored Data

As we have already mentioned, the MLE of the parameter  $\theta$  is the first order statistic. Let us fix an arbitrary  $\epsilon > 0$ . The algorithm based on the conjugate gradient that we propose is as follows.

1. **Step 0:** Initialization  $V_0 = (\alpha_0, \beta_0, k_0, \theta_0)$ .
2. **Step 1:** Computation of the gradient

$$\begin{aligned} \nabla l(\alpha, \beta, k, \theta) &= \begin{pmatrix} \frac{\partial l(\alpha, \beta, k, \theta)}{\partial \alpha} \\ \frac{\partial l(\alpha, \beta, k, \theta)}{\partial \beta} \\ \frac{\partial l(\alpha, \beta, k, \theta)}{\partial k} \end{pmatrix} \\ &= \begin{pmatrix} m [\Psi(\alpha + \beta) - \Psi(\alpha)] + \sum_{j=1}^m \delta_j \ln \left[ 1 - \left(\frac{t_j}{\theta}\right)^{-k} \right] + \sum_{j=1}^m (1 - \delta_j) \frac{z_{\alpha}(t_j, \alpha, \beta, k, \theta)}{z(t_j, \alpha, \beta, k, \theta)} \\ m [\Psi(\beta) - \Psi(\beta + \alpha)] - k \sum_{j=1}^m \ln \left(\frac{t_j}{\theta}\right) + \sum_{j=1}^m (1 - \delta_j) \frac{z_{\beta}(t_j, \alpha, \beta, k, \theta)}{z(t_j, \alpha, \beta, k, \theta)} \\ k^{-1} \sum_{j=1}^m \delta_j - \sum_{j=1}^m \left[ \beta + \delta_j \ln \left(\frac{t_j}{\theta}\right) \left[ 1 - \left(\frac{t_j}{\theta}\right)^k \right]^{-1} \right] \ln \left(\frac{t_j}{\theta}\right) + \sum_{j=1}^m (1 - \delta_j) \frac{z_k(t_j, \alpha, \beta, k, \theta)}{z(t_j, \alpha, \beta, k, \theta)} \end{pmatrix}. \end{aligned}$$

If  $\nabla l(\alpha, \beta, \theta, k) \leq \epsilon$ , then stop,  $V_0 = (\alpha_0, \beta_0, k_0, \theta_0)$  is the optimal vector. If not, go to the next step.

3. **Step 2:** Computation of the direction of descent

If  $i = 0$ , then

$$d_0 = -\nabla l(\alpha_0, \beta_0, k_0, \theta_0)$$

$$= \begin{pmatrix} -m [\Psi(\alpha_0 + \beta_0) + \Psi(\alpha_0)] - \sum_{j=1}^m \delta_j \ln \left[ 1 - \left( \frac{t_j}{\theta_0} \right)^{-k_0} \right] - \sum_{j=1}^m (1 - \delta_j) \frac{z_\alpha(t_j, \alpha_0, \beta_0, k_0, \theta_0)}{z(t_j, \alpha_0, \beta_0, k_0, \theta_0)} \\ -m [\Psi(\beta_0) + \Psi(\beta_0 + \alpha_0)] - k_0 \sum_{j=1}^m \ln \left( \frac{t_j}{\theta_0} \right) - \sum_{j=1}^m (1 - \delta_j) \frac{z_\beta(t_j, \alpha_0, \beta_0, k_0, \theta_0)}{z(t_j, \alpha_0, \beta_0, k_0, \theta_0)} \\ -k_0^{-1} \sum_{j=1}^m \delta_j + \sum_{j=1}^m \left[ \beta_0 - \delta_j \ln \left( \frac{t_j}{\theta_0} \right) \left[ 1 - \left( \frac{t_j}{\theta_0} \right)^{k_0} \right]^{-1} \right] \ln \left( \frac{t_j}{\theta_0} \right) - \sum_{j=1}^m (1 - \delta_j) \frac{z_k(t_j, \alpha_0, \beta_0, k_0, \theta_0)}{z(t_j, \alpha_0, \beta_0, k_0, \theta_0)} \end{pmatrix}.$$

If  $i > 0$ , then  $d_{i+1} = -\nabla l(\alpha_i, \beta_i, k_i, \theta_i) + \omega_i d_i$ ; using the Fletcher-Reeves' method we get

$$\omega_i^{FR} = \frac{\|\nabla l(\alpha_{i+1}, \beta_{i+1}, k_{i+1}, \theta_{i+1})\|^2}{\|\nabla l(\alpha_i, \beta_i, k_i, \theta_i)\|^2}.$$

**4. Step 3:** Computation of  $V_1 = (\alpha_1, \beta_1, k_1, \theta_1)$

$$V_1 = V_0 + \lambda_0 d_0, \text{ such that } d_0 = -\nabla l(\alpha_0, \beta_0, k_0, \theta_0).$$

**Computation of a step  $\lambda_0$ :** We can find  $\lambda_0$  with exact and inaccurate linear search methods. In this case, the use of the exact linear search helps us to have a fast convergence. The exact linear search method is to solve the problem

$$\min_{\lambda_0 \in \mathbb{R}^+} l(V_0 + \lambda_0 d_0).$$

To solve this problem, we must find the value of  $\lambda_0$  such that  $\nabla l(V_0 + \lambda_0 d_0) = 0$  :

$$\begin{aligned} \alpha_0 - \lambda_0 \times & \left[ -m [\Psi(\alpha_0 + \beta_0) + \Psi(\alpha_0)] - \sum_{j=1}^m \delta_j \ln \left[ 1 - \left( \frac{t_j}{\theta_0} \right)^{-k_0} \right] - \sum_{j=1}^m (1 - \delta_j) \frac{z_\alpha(t_j, \alpha_0, \beta_0, k_0, \theta_0)}{z(t_j, \alpha_0, \beta_0, k_0, \theta_0)} \right] = 0, \\ \beta_0 - \lambda_0 & \left[ -m [\Psi(\beta_0) + \Psi(\beta_0 + \alpha_0)] - k_0 \sum_{j=1}^m \ln \left( \frac{t_j}{\theta_0} \right) - \sum_{j=1}^m (1 - \delta_j) \frac{z_\beta(t_j, \alpha_0, \beta_0, k_0, \theta_0)}{z(t_j, \alpha_0, \beta_0, k_0, \theta_0)} \right] = 0, \\ k_0 - \lambda_0 \times & \left[ -k_0^{-1} \sum_{j=1}^m \delta_j + \sum_{j=1}^m \left[ \beta_0 - \delta_j \ln \left( \frac{t_j}{\theta_0} \right) \left[ 1 - \left( \frac{t_j}{\theta_0} \right)^{k_0} \right]^{-1} \right] \ln \left( \frac{t_j}{\theta_0} \right) - \sum_{j=1}^m (1 - \delta_j) \frac{z_k(t_j, \alpha_0, \beta_0, k_0, \theta_0)}{z(t_j, \alpha_0, \beta_0, k_0, \theta_0)} \right] = 0. \end{aligned}$$

Consequently

$$\alpha_0 + \beta_0 + k_0 + \lambda_0 \times \begin{pmatrix} \left[ -m [\Psi(\alpha_0 + \beta_0) + \Psi(\alpha_0)] - \sum_{j=1}^m \delta_j \ln \left[ 1 - \left( \frac{t_j}{\theta_0} \right)^{-k_0} \right] - \sum_{j=1}^m (1 - \delta_j) \frac{z_\alpha(t_j, \alpha_0, \beta_0, k_0, \theta_0)}{z(t_j, \alpha_0, \beta_0, k_0, \theta_0)} \right] \\ + \left[ -m [\Psi(\beta_0) + \Psi(\beta_0 + \alpha_0)] - k_0 \sum_{j=1}^m \ln \left( \frac{t_j}{\theta_0} \right) - \sum_{j=1}^m (1 - \delta_j) \frac{z_\beta(t_j, \alpha_0, \beta_0, k_0, \theta_0)}{z(t_j, \alpha_0, \beta_0, k_0, \theta_0)} \right] \\ + \left[ -k_0^{-1} \sum_{j=1}^m \delta_j + \sum_{j=1}^m \left[ \beta_0 - \delta_j \ln \left( \frac{t_j}{\theta_0} \right) \left[ 1 - \left( \frac{t_j}{\theta_0} \right)^{k_0} \right]^{-1} \right] \ln \left( \frac{t_j}{\theta_0} \right) - \sum_{j=1}^m (1 - \delta_j) \frac{z_k(t_j, \alpha_0, \beta_0, k_0, \theta_0)}{z(t_j, \alpha_0, \beta_0, k_0, \theta_0)} \right] \end{pmatrix} = 0$$

and we obtain

$$\begin{aligned} \lambda_0 = & -(\alpha_0 + \beta_0 + k_0) / \left[ -m [\Psi(\alpha_0 + \beta_0) + \Psi(\alpha_0)] - \sum_{j=1}^m \delta_j \ln \left[ 1 - \left( \frac{t_j}{\theta_0} \right)^{-k_0} \right] \right. \\ & - \sum_{j=1}^m (1 - \delta_j) \frac{z_\alpha(t_j, \alpha_0, \beta_0, k_0, \theta_0)}{z(t_j, \alpha_0, \beta_0, k_0, \theta_0)} - m [\Psi(\beta_0) + \Psi(\beta_0 + \alpha_0)] - k_0 \sum_{j=1}^m \ln \left( \frac{t_j}{\theta_0} \right) \\ & - \sum_{j=1}^m (1 - \delta_j) \frac{z_\beta(t_j, \alpha_0, \beta_0, k_0, \theta_0)}{z(t_j, \alpha_0, \beta_0, k_0, \theta_0)} - k_0^{-1} \sum_{j=1}^m \delta_j + \sum_{j=1}^m \left[ \beta_0 - \delta_j \ln \left( \frac{t_j}{\theta_0} \right) \left[ 1 - \left( \frac{t_j}{\theta_0} \right)^{k_0} \right]^{-1} \right] \ln \left( \frac{t_j}{\theta_0} \right) \\ & \left. - \sum_{j=1}^m (1 - \delta_j) \frac{z_k(t_j, \alpha_0, \beta_0, k_0, \theta_0)}{z(t_j, \alpha_0, \beta_0, k_0, \theta_0)} \right]. \end{aligned}$$

5.2. Numerical Results for BP Parameter Estimation with Censored Data

To show the efficiency of the MLEs obtained by the gradient and conjugate gradient method when data are right censored, an important simulation study was realized. The results of the MLEs and their SDs are summarized in the Tables 11 and 12.

**Table 11.** Results of simulation with samples of censored data using the conjugate gradient method with  $m = 10,000, \alpha = 13, \beta = 23, k = 18, \theta = 0.15$ .

n	$\hat{\alpha}$	$\hat{\beta}$	$\hat{k}$	$SD(\hat{\alpha})$	$SD(\hat{\beta})$	$SD(\hat{k})$
10	12.0111	23.5087	17.7891	1.701444	0.766839	0.036204
15	12.0541	23.8580	17.7891	0.733933	0.659689	0.036204
20	12.0967	23.9454	17.7891	0.183483	0.476273	0.036204
30	12.0106	23.4682	18.0907	0.117429	0.067367	0.036204
40	12.2676	23.5793	18.0907	0.007339	0.137393	0.036204
50	12.3533	23.1487	18.0907	$8.39 \times 10^{-5}$	$8.67 \times 10^{-5}$	$4.71 \times 10^{-6}$
100	12.7817	23.0266	18.0907	$8.47 \times 10^{-5}$	$8.83 \times 10^{-5}$	$4.71 \times 10^{-6}$
150	12.8293	22.9632	18.0907	$8.55 \times 10^{-5}$	$9.50 \times 10^{-5}$	$4.71 \times 10^{-6}$
200	12.8940	22.8416	17.8626	$4.78 \times 10^{-7}$	$4.01 \times 10^{-7}$	$8.09 \times 10^{-7}$
300	12.9777	22.7847	17.8277	$1.58 \times 10^{-8}$	$4.66 \times 10^{-8}$	$3.33 \times 10^{-10}$
500	12.9785	22.8175	17.8626	$2.85 \times 10^{-8}$	$9.44 \times 10^{-9}$	$3.35 \times 10^{-10}$
1000	13.0002	23.0004	17.0073	$7.00 \times 10^{-9}$	$1.19 \times 10^{-10}$	$5.69 \times 10^{-9}$

**Table 12.** Results of simulation with samples of censored data using the gradient method with  $m = 10,000, \alpha = 13, \beta = 23, k = 18, \theta = 0.15$ .

n	$\hat{\alpha}$	$\hat{\beta}$	$\hat{k}$	$SD(\hat{\alpha})$	$SD(\hat{\beta})$	$SD(\hat{k})$
10	12.8685	22.9473	15.9886	0.000319	0.000527	0.001948
15	12.9168	22.9444	16.8106	$7.03 \times 10^{-6}$	$9.83 \times 10^{-5}$	$2.82 \times 10^{-2}$
20	12.9336	22.9669	16.2010	$2.11 \times 10^{-5}$	$1.00 \times 10^{-4}$	$9.67 \times 10^{-3}$
30	12.9336	23.0119	16.8935	$2.11 \times 10^{-5}$	$5.79 \times 10^{-6}$	$3.35 \times 10^{-2}$
40	12.9336	23.0095	16.8626	$2.12 \times 10^{-5}$	$3.88 \times 10^{-6}$	$3.15 \times 10^{-2}$
50	12.9420	23.0256	17.4303	$2.82 \times 10^{-5}$	$1.80 \times 10^{-5}$	$5.93 \times 10^{-2}$
100	12.9504	22.9386	17.4201	$3.52 \times 10^{-5}$	$7.21 \times 10^{-5}$	$4.68 \times 10^{-2}$
150	12.9588	22.9327	17.7439	$4.22 \times 10^{-5}$	$7.24 \times 10^{-5}$	$5.72 \times 10^{-2}$
200	12.9672	22.9290	18.0424	$4.93 \times 10^{-5}$	$6.92 \times 10^{-5}$	$6.61 \times 10^{-2}$
300	12.9756	22.9404	18.3535	$5.63 \times 10^{-5}$	$4.24 \times 10^{-5}$	$7.58 \times 10^{-2}$
500	12.9756	22.9381	19.7080	$5.63 \times 10^{-5}$	$4.08 \times 10^{-5}$	$8.82 \times 10^{-2}$
1000	12.9840	23.0030	19.7933	$6.35 \times 10^{-5}$	$2.29 \times 10^{-7}$	$1.05 \times 10^{-1}$

To evaluate and compare the performance of the estimators obtained by the proposed methods, we present in the Tables 13 and 14 the bias and MSEs of the estimators.

5.3. Model Selection for Censored Data

In this section, the censored data are used with the AIC, BIC and AICc criteria to choose the best statistical model for our statistical population. These results are presented in Tables 15–18. As for the uncensored data, we can conclude that the information criteria choose the correct model even for small

or medium values of the sample size  $n$ . Note that similar remarks as the ones done in Section 4 for model selection for complete (uncensored) data hold true also here.

**Table 13.** Gradient method.

	$\hat{\alpha}$	$\hat{\beta}$	$\hat{k}$
$E(\hat{V})$	12.9450	22.9656	17.6040
bias	-0.0242	-0.0349	0.3027
MSE	$5.66 \times 10^{-5}$	$1.41 \times 10^{-5}$	$7.27 \times 10^{-2}$

**Table 14.** Conjugate gradient method.

	$\hat{\alpha}$	$\hat{\beta}$	$\hat{k}$
$E(\hat{V})$	13.0029	23.002452	17.9714
bias	-0.0790	0.0261	0.0328
MSE	$1.82 \times 10^{-7}$	$3.90 \times 10^{-9}$	$3.42 \times 10^{-10}$

**Table 15.** Smaller AIC/BIC/AICc scores in 1000 simulations from a censored Pareto distribution  $P(k, \theta)$  with  $k = 7.5, \theta = 0.15$ .

n	AIC				BIC				AICc						
	BP	B	P	G	GBP	BP	B	P	G	GBP	BP	B	P	G	GBP
10	7	10	233	745	5	6	9	233	748	4	10	4	233	738	15
20	5	12	417	563	3	3	6	417	570	4	7	6	417	559	11
30	1	7	561	429	2	5	6	561	425	3	2	6	561	428	3
50	0	5	734	261	0	0	1	734	265	0	0	2	734	264	0
100	0	2	912	86	0	0	0	912	88	0	0	1	912	87	0
200	0	0	939	61	0	0	0	939	61	0	0	0	939	61	0
500	0	0	973	27	0	0	0	973	27	0	0	0	973	27	0
1000	0	0	996	4	0	0	0	996	4	0	0	0	996	4	0

**Table 16.** Smaller AIC/BIC/AICc scores in 1000 simulations from a censored Gamma distribution  $G(\alpha, \beta)$  with  $\alpha = 2, \beta = 1$ .

n	AIC				BIC				AICc						
	BP	B	P	G	GBP	BP	B	P	G	GBP	BP	B	P	G	GBP
10	0	1	52	947	0	0	1	52	947	0	0	1	52	947	0
20	0	0	64	936	0	0	0	64	936	0	0	0	64	936	0
30	0	1	52	947	0	0	1	52	947	0	0	1	52	947	0
50	0	0	63	937	0	0	0	63	937	0	0	0	63	937	0
100	0	1	56	943	0	0	1	56	943	0	0	1	56	943	0
200	0	2	52	946	0	0	2	52	946	0	0	2	52	946	0
500	0	1	53	946	0	0	0	53	947	0	0	1	53	946	0
1000	0	1	61	938	0	0	0	61	939	0	0	1	61	938	0

**Table 17.** Smaller AIC/BIC/AICc scores in 1000 simulations from a censored BP distribution  $BP(\alpha, \beta, k, \theta)$  with  $\alpha = 2.7, \beta = 1.9, k = 0.48; \theta = 0.15$ .

n	AIC					BIC					AICc				
	BP	B	P	G	GBP	BP	B	P	G	GBP	BP	B	P	G	GBP
10	453	11	25	502	9	434	7	24	524	11	421	12	22	579	7
20	600	8	15	370	7	532	6	15	438	9	531	15	10	439	5
30	636	7	9	344	4	598	5	8	382	7	611	12	6	369	2
50	704	3	6	286	1	665	2	7	323	3	688	7	2	302	1
100	711	0	0	289	0	680	0	0	320	0	699	0	0	301	0
200	714	0	0	286	0	682	0	0	318	0	708	0	0	298	0
500	741	0	0	259	0	736	0	0	264	0	741	0	0	259	0
1000	752	0	0	248	0	748	0	0	252	0	752	0	0	248	0
2000	755	0	0	245	0	753	0	0	247	0	755	0	0	245	0

**Table 18.** Smaller AIC/BIC/AICc scores in 1000 simulations from a censored GBP distribution  $GBP(\alpha, \beta, k, \theta, c)$  with  $\alpha = 1.5, \beta = 0.25, k = 0.075, \theta = 0.15, c = 0.001125$ .

n	AIC					BIC					AICc				
	BP	B	P	G	GBP	BP	B	P	G	GBP	BP	B	P	G	GBP
10	5	0	1	0	994	5	0	1	0	994	5	0	1	0	994
20	4	0	7	0	989	3	0	7	0	990	5	0	6	0	989
30	11	0	1	0	988	9	0	1	0	990	11	0	1	0	988
50	19	0	1	0	980	19	0	1	0	980	19	0	1	0	980
100	24	0	1	0	975	24	0	1	0	975	24	0	1	0	975
200	33	0	2	0	965	33	0	2	0	965	34	0	2	0	964
300	53	0	4	0	943	51	0	4	0	945	52	0	4	0	944
500	76	0	1	0	923	75	0	1	0	924	76	0	1	0	923
1000	122	0	2	0	876	118	0	2	0	880	123	0	0	0	877
2000	238	0	2	0	760	237	0	2	0	761	237	0	2	0	761

### 6. Conclusions

In this paper, we have developed different optimization methods to determine the maximum likelihood estimators of the four parameters of the Beta-Pareto distribution. The results obtained showed that all the used methods give  $\sqrt{n}$ -consistent estimators in both complete and right-censored data samples and particularly the conjugate gradient method gives the best results. Using classical model selection criteria, our study proves that this new model can be used instead of several alternative models such as Beta, Pareto, Gamma, and Generalized Beta-Pareto. Another important contribution of our work is the use of right-censored data as an input for the estimation procedure and model selection techniques.

**Author Contributions:** All authors have equally contributed to different stages of the article. All authors have read and agreed to the published version of the manuscript.

**Funding:** This research received no external funding.

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

## References

1. Akinsete, A.; Famoye, F.; Lee, C. The Beta-Pareto distribution. *Statistics* **2008**, *42*, 547–563. [[CrossRef](#)]
2. Asmussen, S. *Applied Probability and Queues*; Springer: New York, NY, USA, 2003.
3. Zhang, J. Likelihood moment estimation for the generalized Pareto distribution. *Aust. N. Z. J. Stat.* **2007**, *49*, 69–77. [[CrossRef](#)]
4. Levy, M.; Levy, H. Investment talent and the Pareto wealth distribution: Theoretical and experimental analysis. *Rev. Econ. Stat.* **2003**, *85*, 709–725. [[CrossRef](#)]
5. Burroughs, S.M.; Tebbens, S.F. Upper-truncated power law distributions. *Fractals* **2001**, *9*, 209–222. [[CrossRef](#)]
6. Newman, M.E. Power laws, Pareto distributions and Zipf's law. *Contemp. Phys.* **2005**, *46*, 323–351. [[CrossRef](#)]
7. Choulakian, V.; Stephens, M.A. Goodness-of-fit tests for the generalized Pareto distribution. *Technometrics* **2001**, *43*, 478–484. [[CrossRef](#)]
8. Conn, A.R.; Gould, N.I.M.; Toint, P.L. Convergence of quasi-Newton matrices generated by the symmetric rank one update. *Math. Program.* **1991**, *50*, 177–195. [[CrossRef](#)]
9. Hestenes, M.R.; Stiefel, E. Methods of conjugate gradients for solving linear systems. *J. Res. Natl. Bureau Stand.* **1952**, *49*, 409–436. [[CrossRef](#)]
10. Fletcher, R.; Reeves, C.M. Function minimization by conjugate gradients. *Computer J.* **1964**, *7*, 149–154. [[CrossRef](#)]
11. Polak, E.; Ribiere, G. Note sur la convergence de méthodes de directions conjuguées. *ESAIM Math. Model. Numer. Anal.* **1969**, *3*, 35–43. (In French) [[CrossRef](#)]
12. Fletcher, R. *Practical Methods of Optimization*; 2nd ed.; Wiley: Hoboken, NJ, USA, 2000.
13. Armijo, L. Minimization of functions having Lipschitz continuous first partial derivatives. *Pac. J. Math.* **1966**, *16*, 1–3. [[CrossRef](#)]
14. Goldstein, A.A. On steepest descent. *J. Soc. Ind. Appl. Math. Ser. A Control* **1965**, *3*, 147–151. [[CrossRef](#)]
15. Wolfe, P. Convergence conditions for ascent methods. *SIAM Rev.* **1969**, *11*, 226–235. [[CrossRef](#)]
16. Fletcher, R.; Powell, M.J.D. A rapidly convergent descent method for minimization. *Comp. J.* **1963**, *6*, 163–168. [[CrossRef](#)]
17. Sato, H.; Iwai, T. A new, globally convergent Riemannian conjugate gradient method. *Optimization* **2013**, *64*, 1011–1031. [[CrossRef](#)]
18. Dai, Y.H.; Yuan, Y. A nonlinear conjugate gradient method with a strong global convergence property. *SIAM J. Optim.* **1999**, *10*, 177–182. [[CrossRef](#)]
19. Nassar, M.M.; Nada, N.K. The beta generalized Pareto distribution. *J. Stat. Adv. Theory Appl.* **2011**, *6*, 1–17.
20. Mahmoudi, E. The beta generalized Pareto distribution with application to lifetime data. *Math. Comp. Simul.* **2011**, *81*, 2414–2430. [[CrossRef](#)]



© 2020 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<http://creativecommons.org/licenses/by/4.0/>).