Available online at www.sciencedirect.com
ScienceDirect
Applied Mathematics and Computation 183 (2006) 85–93
APPLIED MATHEMATICS AND COMPUTATION
www.elsevier.com/locate/amc

Estimating the parameters of Weibull distribution using simulated annealing algorithm

Babak Abbasi [a,*], Abdol Hamid Eshragh Jahromi [a], Jamal Arkat [b], Mehdi Hosseinkouchack [c]

[a] Department of Industrial Engineering, Sharif University of Technology, P.O. Box 11365-9414, Azadi Avenue, Tehran, Iran
[b] Department of Industrial Engineering, Iran University of Science and Technology, P.O. Box 11495-157, Farjam Avenue, Narmak, Tehran, Iran
[c] Department of Business and Economics, Johann Wolfgang Goethe University, P.O. Box 055, No. 17, Merton Street, Frankfurt am Main, Germany

Abstract

Weibull distribution plays an important role in failure distribution modeling in reliability studies. It is a hard work to estimate the parameters of Weibull distribution. This distribution has three parameters, but for simplicity, a parameter is omitted and as a result, the estimation of the others will be easily done. When the three-parameter distribution is of interest, the estimation procedure will be quite boring. Maximum likelihood estimation is a good method, which is usually used to elaborate on the parameter estimation. The likelihood function formed for the parameter estimation of a three-parameter Weibull distribution is very hard to maximize. Many researchers have studied this maximization problem. In this paper, we have briefly discussed this problem and proposed a new approach based on the simulated algorithm to solve that.

Keywords: Weibull probability distribution; Simulated annealing; Parameter estimation; Maximum likelihood estimation

1. Introduction

It is a known fact that three-parameter Weibull distribution family is extremely flexible and can fit very well an extremely wide range of empirical observations [1]. It exhibits a wide range of shapes for the density and hazard functions, which are suitable to model complex failure data sets. Also, it is especially useful as a failure model in analyzing the reliability of different types of systems.

* Corresponding author.
E-mail addresses: B_abbasi@Mehr.Sharif.edu, B.abbasi@gmail.com (B. Abbasi), Eshargh@Sharif.edu (A.H. Eshragh Jahromi), J.arkat@gmail.com (J. Arkat), M.kouchack@gmail.com (M. Hosseinkouchack).

0096-3003/$ - see front matter
doi:10.1016/j.amc.2006.05.063

The Weibull distribution probability density function is as follows [2]:

$$f(x) = \frac{\beta}{\eta}\left(\frac{x-\gamma}{\eta}\right)^{\beta-1} e^{-\left(\frac{x-\gamma}{\eta}\right)^{\beta}}; \quad \beta > 0, \ \eta > \gamma \geqslant 0. \tag{1}$$

Also the cumulative Weibull distribution function is given by

$$F(x) = 1 - e^{-\left(\frac{x-\gamma}{\eta}\right)^{\beta}}, \tag{2}$$

where β, η, and γ are shape, scale, and location parameters, respectively.

Successful application of Weibull distribution depends on having acceptable statistical estimates of the three parameters. Estimating the parameters of the three-parameter Weibull distribution family is intrinsically a very difficult task. Nosal and Nosal (2000) used Monte Carlo methods and APL (array processing language) to investigate the performance of the gradient random search minimization procedure for fitting a Weibull distribution to a given data set using minimum Kolmogorov–Smirnov distance approach [3].

Because of this difficulty, the three-parameter Weibull model is seldom used. Even the popular two-parameter model does not offer close estimates of the parameters and relies on numerical procedure.

The present study focuses on likelihood method, and uses simulated annealing to maximize the likelihood function. Simulated annealing is an algorithm that originates in material science engineering, originally introduced to find the equilibrium configuration of a collection of atoms at a given temperature [4].

The first one to use it as an algorithm to solve optimization problems was Kirkpatrick et al. [5]. This algorithm considers the optimization process as a Markovian chain that pushes the iterations towards a minimum or maximum. At each iteration or better to say each temperature value, the algorithm checks a number of finite different situations that the Markovian chain can be in to find a new point to move to. This procedure should be controlled with great care to assure the convergence towards a global maximum or minimum. But it should be taken into account that there is no need to have in depth knowledge on the Markovian chains. Altogether, it is a very good and easy to implement algorithm that helps engineers find their way through hard optimization problems.

The next section deals with the notion of maximum likelihood estimation. Section 3 gives some brief information about what the SA algorithm is, and the way it can be applied to the specific problem at hand is explained in Section 4. The following section illustrates the approach here presented focusing on three different problems. Section 6 concludes the current work followed by a section introducing some research areas to those interested in parameter estimation for a three-parameter Weibull distribution using metaheuristics.

2. Parameter estimation using MLE

Estimation theory is a cornerstone in statistical analyses, and there has been introduced several techniques to estimate parameters, of which *MLE* (maximum likelihood estimation), graphical procedure [6], moments method [7,8] and weighted least square method are some of the most interesting ones [9,10].

It is widely known that MLE estimators are asymptotically unbiased with the minimum variance, and maximum likelihood estimation is a commonly used technique for parameter estimation [11].

The MLE method has very desirable properties. Let $x_1, x_2 \ldots, x_n$ be a random sample of size n drawn, at random, from a probability density function, $f(x;\vec{\theta})$, of unknown parameters, $\vec{\theta}$. The likelihood function is as follows:

$$L = \prod_{i=1}^{n} f_{X_i}(x_i;\vec{\theta}), \tag{3}$$

where $\vec{\theta}$ is a vector of size m representing the unknown parameters, i.e.

$$\vec{\theta} = (\theta_1, \theta_2, \ldots, \theta_m). \tag{4}$$

The goal, here, is to find a vector, say $\vec{\theta}_0$, that maximizes the so-called likelihood function. To maximize L, we may equivalently use its logarithm, say $\mathrm{Ln}(L)$. Estimates are obtained through solving the following equation set:

$$\frac{\partial}{\partial\vec{\theta}}\mathrm{Ln}(L)=\vec{0}. \tag{5}$$

This equation set is hard to solve when applied to Weibull distribution, and many others. In this regard, we may be forced to reluctantly use numerical procedures.

The L function for Weibull distribution is as follows, where $\vec{\theta}=(\beta,\eta,\gamma)$:

$$L(x_1,\ldots,x_n,\beta,\eta,\gamma)=\prod_{i=1}^{n}\frac{\beta}{\eta}\left(\frac{x_i-\gamma}{\eta}\right)^{\beta-1}\mathrm{e}^{-\left(\frac{x_i-\gamma}{\eta}\right)^{\beta}};\quad \beta>0,\ \ \eta>\gamma\geqslant 0. \tag{6}$$

Its logarithm will be as follows:

$$\mathrm{Ln}(L(x_1,\ldots,x_n,\beta,\eta,\gamma))=n\,\mathrm{Ln}\left(\frac{\beta}{\eta}\right)+\sum_{\mathrm{i}=1}^{\mathrm{n}}\left(-\left(\frac{x_i-\gamma}{\eta}\right)^{\beta}+(\beta-1)\,\mathrm{Ln}\left(\frac{x_i-\gamma}{\eta}\right)\right). \tag{7}$$

It is very difficult to maximize L (or $\mathrm{Ln}(L)$) [12], using ordinary optimization techniques. Gradient method being dependent on the partial derivations of the objective function is not a good method to use here, because it is very hard to evaluate the gradient terms and the objective function itself at different points the algorithm needs. Also worth noting, that it is quite boring to derive the gradient of this complicated objective function.

3. General simulated annealing algorithm

As its name implies, the simulated annealing (SA) exploits an analogy between the way in which a metal cools and freezes into a minimum energy crystalline structure (the annealing process) and the search for a minimum in a more general system.

The algorithm is based upon that of Metropolis et al. [4], which was originally proposed as a means of finding the equilibrium configuration of a collection of atoms at a given temperature. The connection between this algorithm and mathematical minimization was first noted by Pincus [13], but it was Kirkpatrick et al. [5] who proposed that it form the basis of an optimization technique for combinatorial (and other) problems.

SAs major advantage over other methods is an ability to avoid becoming trapped at local minima. The algorithm employs a random search, which not only accepts changes that decrease objective function f, but also some changes that increase it. The latter are accepted with a probability

$$p=\mathrm{e}^{-\Delta/T}, \tag{8}$$

where Δ is the increase in f and T is a control parameter, which by analogy with the original application is known as the system 'temperature' irrespective of the objective function involved.

The implementation of the SA algorithm is remarkably easy. The following elements must be provided:

1. a representation of possible solutions,
2. a generator of random changes in solutions,
3. a means of evaluating the problem functions, and
4. an *annealing schedule* – an initial temperature and rules for lowering it as the search progresses.

Ease of use and provision of good solutions to real-world problems makes this algorithm be one of the most powerful and popular meta-heuristics to solve many optimization problems.

The basic structure of SA algorithm is presented in Table 1, where the following notation is used:

$S=$ the current solution,
$S^*=$ the best solution,

Table 1
Simulated annealing algorithm for minimization problem

```
Initialize the SA control parameter (T0, L)
Select an initial solution, S0
Set T = T0; Set S = S0; Set S* = S0; Calculate f(S0);
While the stop criterion is not reached do:
    Set n = 1;
    While n < L do:
        Generate solution Sn in the neighborhood of S0; Calculate Δ = f(Sn) − f(S);
        if Δ ⩽ 0
            S = Sn
        else
            generate a random number, r ∈ (0, 1)
            if (r ⩽ p = e^(−Δ/T));
                S = Sn; n = n + 1;
            end
        end
        if (f(S) < f(S*))
            S* = Sn;
        end
    end
    reduce the temperature T;
end
```

S_n = neighboring solution,
$f(S)$ = the value of objective function at solution S,
n = repetition counter,
T_0 = initial temperature,
I = number of repetition allowed at each temperature level,
p = probability of accepting S_n when it is not better than S.

It is obvious that this procedure just takes into account the minimization problems, hence while performing a maximization problem, the objective function is multiplied by (-1) to obtain a capable form.

The algorithm starts with an initial solution for the problem. As it is obvious from Table 1, SA has two cycles, inner and outer. In the inner cycle of the SA, repeated while $n < L$, a neighboring solution S_n of the current solution S is generated. If $\Delta \leqslant 0$ (S_n is better than S), then the generated solution replaces the current solution, otherwise the solution is accepted with a criterion probability, as noted before $p = e^{-\Delta/T}$. The value of the temperature, T, decreases in each iteration of the outer cycle of the algorithm.

As a meta-heuristic algorithm, the most important feature of this algorithm is the possibility of accepting a worse solution, hence allowing it to prevent falling into a local optimum trap. Obviously, the probability of accepting a worse solution decreases as the temperature decreases in each outer cycle. The performance of SA depends on the definition of the several control parameters:

1. The initial temperature (T_0) should be high enough that in the first iteration of the algorithm, the probability of accepting a worse solution is, at least, of 80% [5].
2. The most commonly used temperature reducing function is geometric; i.e. $T_i = CT_{i-1}$ in which $C < 1$ and constant. Typically, $0.75 \leqslant C \leqslant 0.95$.
3. The length of each temperature level (L) determines the number of solutions generated at a certain temperature, T.
4. The stopping criterion defines when the system has reached a desired energy level. Equivalently it defines:
   The total number of solutions generated.
   The temperature at which the desired energy level is reached (freezing temperature).
   The acceptance ratio (ratio between the number of solutions accepted and the number of solutions generated).

It is obvious that these control parameters are chosen with respect to the specific problem at hand. When adapting this general algorithm to a specific problem, the procedure to generate both initial and neighboring solutions is very important in addition to the control parameter. The details of proposed SA algorithm to parameter estimation problem are presented in the next section.

4. Applying SA algorithm to parameter estimation

To estimate the three parameters of Weibull distribution we are to maximize L (or $\mathrm{Ln}(L)$), using simulated annealing. In this regard, the steps of this algorithm are briefly looked into.

1. Obtain a random sample from the distribution the size of which should be big enough.
2. Determine control parameters of SA, i.e. T_0,T_f, C, I.
3. Generate random values a, b, c.
4. Compute the likelihood function, L, at this randomly generated solution.
5. While $T > T_0$
   $T = CT$.
6. For $i = 1$ to I
   6.1. Generate neighboring values, say a_1, b_1, c_1, for a, b, c.
   6.2. Compute the likelihood function at this new solution, L_0.
   6.3. Evaluate parameters.
      6.3.1. if $L_0 > L$ then $a = a_1$, $b = b_1$, $c = c_1$, and $L = L_0$
      6.3.2. else
         Generate a random value, u, of $Uni(0,1)$.
      6.3.2.1. if $u < e^{-\frac{(F_0 - F)}{T}}$ then $a = a_1$, $b = b_1$, $c = c_1$
7. Print a, b, c and L
   a, b, c are estimates of β, η, γ.

If the control parameters of SA are defined correctly, unbiased estimations with minimum variance will be straightforward.

5. Numerical examples

To illustrate the new approach three examples are considered. In addition, to know the effects of the sample size on the algorithm, samples of size 2500, 1000, 500, and 100 have been taken and used to estimate the parameters. It is straightforward from the primary estimation theory that the bigger the sample size the better the estimation. However, here as the sample size increases the more complicated will be the likelihood function to maximize. Therefore, the selection of the sample size is a matter of compromise. Also because good processors are at hand this problem is diminished. However, the time consumption has also been noted at the end of all tables of results.

For all these examples, the cooling rate has been considered 0.99, the initial and final temperatures 100 and 0.001 respectively. The Markovian chain considered finite with a length of five. To see how the algorithm approaches the maximum, and finally the estimations, its performance is illustrated for these three examples.

All the coding has been done using Matlab 7, and been run on a P4 processor with 256 MB of RAM.

Example 1. The first example here is to estimate the parameters of a Weibull distribution with $\vec{\theta} = (2,2,2)$. To get a better understanding of the algorithm there has been listed the likelihood function at real and estimated values for different sample sizes. Table 2 shows the results. It is quite clear that as the sample size increases the more will be better the estimation. But as noted before because of likelihood function will be more complicated with bigger sample sizes, the run time will be more for the algorithm to settle down to the estimates.

Table 2
Table of results for the first example

Weibull parameters		(2, 2, 2)		
Number of random numbers generated	2500	1000	500	100
Estimated parameters	(2.0298, 2.0338, 1.9995)	(1.9511, 1.9051, 2.0354)	(1.8329, 1.9306, 2.0718)	(1.8392, 1.6363, 2.25181)
Likelihood function at the real values	−3234.8	−1305.5	−618.3805	−137.2584
Likelihood function at the estimated value	−3235.4	−1301.8	−615.3697	−134.6563
Run time (s)	100.6719	96.2500	89.6094	72.2188

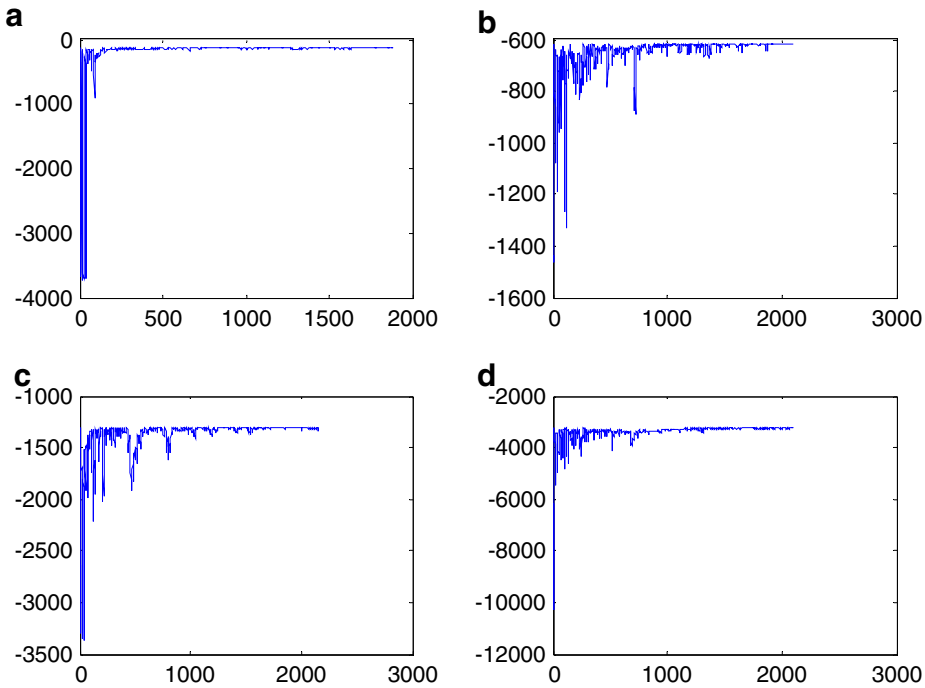

Fig. 1. SA approaches towards ML function maximum for Example 1: (a) sample size of 100, (b) sample size of 500, (c) sample size of 1000, (d) sample size of 2500.

Table 3
Table of results for the second example

Weibull parameters		(2, 3, 4)		
Number of random numbers generated	2500	1000	500	100
Estimated parameters	(2.0418, 3.0912, 3.9714)	(2.0940, 3.2164, 3.9295)	(1.8267, 2.9170, 4.1451)	(1.8974, 3.0067, 4.0707)
Likelihood function at the real values	−2436.6	−974.9306	−461.2994	−93.4527
Likelihood function at the estimated value	−2435.5	−974.2441	−458.2988	−93.1223
Run time (s)	104.1094	90.6250	87.2969	86.7500

As the simulated annealing is a randomized algorithm, the way it approaches its end can be of interest. Fig. 1 shows the likelihood function for all the values it has been evaluated during the process of maximization. It is quite clear that the algorithm converges, after a wide of feasible space search, to its maximum happening at the estimates of the parameters.

Example 2. This example considers a Weibull distribution with $\vec{\theta} = (2, 3, 4)$. The results contained in Table 3 and Fig. 2 are the same as that of first example.

Example 3. This example considers a Weibull distribution with $\vec{\theta} = (3, 2, 5)$. The results contained in Table 4 and Fig. 3 are the same as that of the first example.

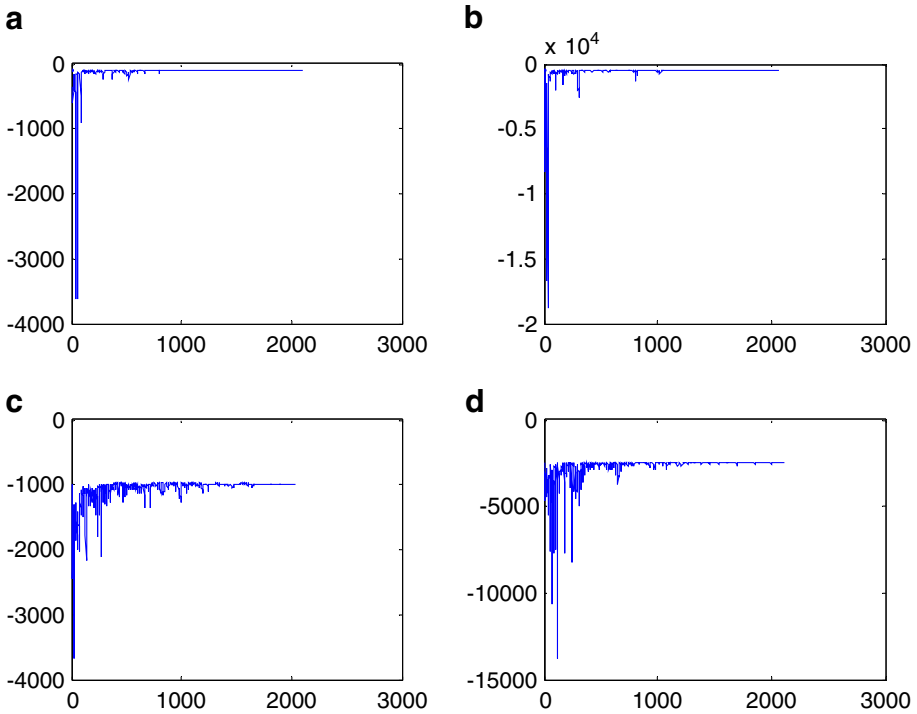

Fig. 2. SA approaches towards ML function maximum for Example 2: (a) sample size of 100, (b) sample size of 500, (c) sample size of 1000, (d) sample size of 2500.

Table 4
Table of results for the third example

Weibull parameters		(3,2,5)		
Number of random numbers generated	2500	1000	500	100
Estimated parameters	(2.9732, 1.9149, 5.0407)	(3.1251, 1.9814, 4.9547)	(2.9469, 1.9971, 5.0398)	(2.6096, 1.7596, 5.2287)
Likelihood function at the real values	−4282.2	−1746.6	−838.9136	−166.0052
Likelihood function at the estimated value	−4279.5	−1743.1	−838.5784	−165.1076
Run time (s)	102.9375	91.8125	86.7813	83.7031

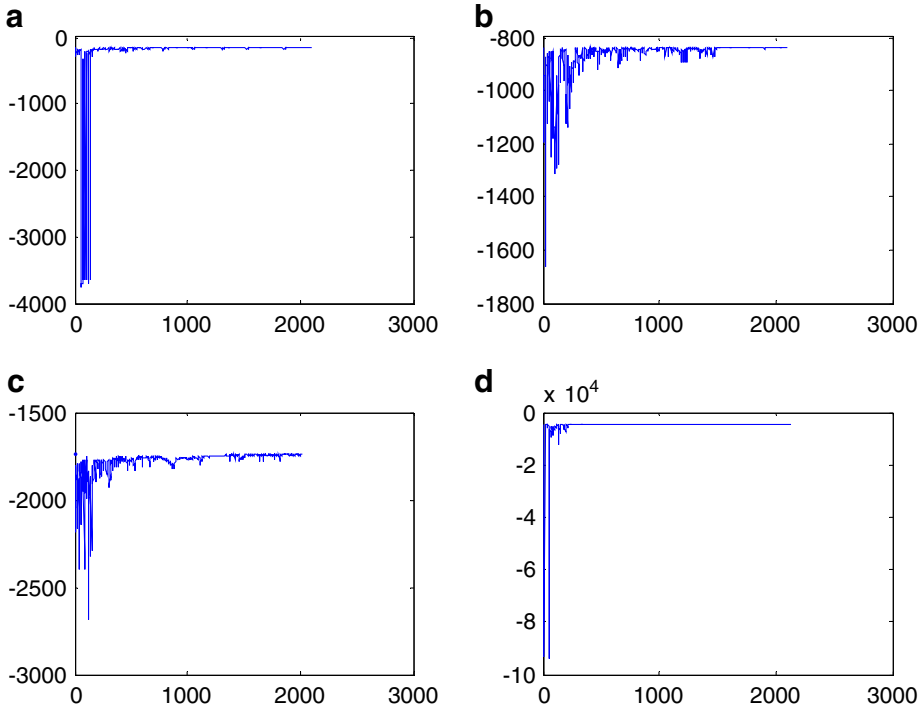

Fig. 3. SA approach towards ML function maximum for Example 3: (a) sample size of 100, (b) sample size of 500, (c) sample size of 1000, (d) sample size of 2500.

6. Conclusion

Based on the results shown the proposed approach yields good results for a very important estimation problem. Due to the power and accuracy of this new approach, it will be possible to use three-parameter Weibull distribution to better model the real problems in which a two-parameter Weibull distribution is not well performing.

Also, the examples have been chosen with a diligence to cover a somehow wide range of problems.

7. Future research

The widespread use of parameter estimation has turned it into a field that is of great interest to many practitioners and researchers. Something very interesting to research is a nice analysis on the SA algorithm itself with regard to the estimation problem for Weibull distribution with different sample sizes to establish a method of how to choose that. This problem is a matter of controlling statistical errors introduced as first and second type errors.

At the other hand as there are other well-established meta-heuristics, such as genetic algorithms, Bee and Ant Colony, etc., a discussion of how good the estimation can be using them, may lead to a better estimation procedure.

References

[1] M. Nosal, A.H. Legge, S.V. Krupa, Application of a stochastic, Weibull probability generator for replacing missing data on ambient concentrations of gaseous pollutants, Environmental Pollution 108 (2000) 439–446.

[2] N.L. Johnson, S. Kotz, Continuous Univariate Distributions, John Wiley & Sons, New York, 1970.

[3] M. Nosal, E.-M.Nosal, Three-parameter Weibull generator for replacing missing observations, in: Proceedings of the 2003 Hawaii International Conference on Statistics and Related Fields in Honolulu, Hawaii, June 5–8, 2003.

[4] N. Metropolis, A.W. Rosenbluth, M.N. Rosenbluth, A.H. Teller, E. Teller, Equations of state calculations by fast computing machines, Journal of Chemical Physics 21 (1958) 1087–1092.
[5] S. Kirkpatrick, C.D. Gerlatt Jr., M.P. Vecchi, Optimization by simulated annealing, Science 220 (1983) 671–680.
[6] R. Ross, Graphical methods for plotting and evaluating Weibull distributed data, in: Proc. Int. Conf. Prop. Appl. Diel. Mater, vol. 1, Brisbane, Australia, 1994, pp. 250–253.
[7] E.H. Lehman, Shapes, moments and estimators of the Weibull distribution, Transactions of IEEE on Reliability 11 (1962) 32–38.
[8] J.S. White, The moments of log-Weibull order statistics, Technometrics 11 (2) (1969) 374–386.
[9] J.S. Usher, Weibull component reliability-prediction in the presence of masked data, IEEE Transactions on Reliability 45 (1996) 229–232.
[10] L.J. Bain, C.E. Antle, Estimation of parameters in the Weibull distribution, Technometrics 9 (1967) 621–627.
[11] S.D. Dubey, Asymptotic properties of several estimators of Weibull parameters, Technometrics 7 (1965) 423–434.
[12] D.R. Thoman, L.J. Bain, C.E. Antle, Inferences on the parameters of the Weibull distribution, Technometrics 11 (1969) 445–460.
[13] M. Pincus, A Monte Carlo method for the approximate solution of certain types of constrained optimization problems, Operations Research 18 (1970) 1225–1228.