We are IntechOpen,
the world's leading publisher of
Open Access books
Built by scientists, for scientists

4,800
Open access books available

122,000
International authors and editors

135M
Downloads

Our authors are among the

154
Countries delivered to

TOP 1%
most cited scientists

12.2%
Contributors from top 500 universities

BOOK CITATION INDEX INDEXED
CLARIVATE ANALYTICS

WEB OF SCIENCE™

Selection of our books indexed in the Book Citation Index
in Web of Science™ Core Collection (BKCI)

Interested in publishing with us?
Contact book.department@intechopen.com

Numbers displayed above are based on latest data collected.
For more information visit www.intechopen.com

Simulated Annealing of Constrained Statistical Functions

Barry Smith and Augustine Wong

Additional information is available at the end of the chapter

Abstract

In 1987, Corana et al. published a simulated annealing (SA) algorithm. Soon thereafter in 1993, Goffe et al. coded the algorithm in FORTRAN and showed that SA could uncover global optima missed by traditional optimization software when applied to statistical modeling and estimation in economics (econometrics). This chapter shows how and why SA can be used successfully to perform likelihood-based statistical inference on models where likelihood is constrained by often very complicated functions defined on a compact parameter space. These constraints arise because likelihood-based inference involves comparing the maxima of constrained versus unconstrained statistical optimization models. The chapter begins with a review of the relevant literature on SA and constrained optimization using penalty techniques. Next, a constrained optimization problem based in maximum likelihood stress-strength modeling is introduced, and its statistical and numerical properties are summarized. SA is then used to solve a sequence of penalty-constrained optimization problems, and the results are used to construct a confidence interval for the parameter of interest in the statistical model. The chapter concludes with a brief summary of the results and some ways we were able to enhance the performance of SA in this setting.

Keywords: SA, constrained optimization, penalty, likelihood

1. SA and penalty-constrained optimization

Several chapters in this book consider the foundations and development of the simulated annealing (SA) algorithm. In this chapter, we focus on just one version of the algorithm given in Ref. [1] and one FORTRAN implementation of the algorithm presented in Ref. [2]. The

reasons for this are efficiency and familiarity. The algorithm in Ref. [1] is tight and effective and the FORTRAN implementation in [2] is dependable, easily extensible and sufficiently fast that it can be applied to both complicated small-scale problems and to very large Monte Carlo studies such as the one in Ref. [3].

1.1. Background

The SA algorithm in Ref. [1] was developed as an approach to find the unconstrained global optimum of functions with a potential multiplicity of optima, some of which may lie on the boundaries of the function's domain. The function being optimized need not be differentiable or even continuous, but it must be bounded. As well, the domain of the function must be compact. The search algorithm in Ref. [1] depends upon function evaluations. Derivatives play no role. The reader is referred to the careful statement of the algorithm on pages 266–269 in Ref. [1]. Here we provide an overview of the algorithm and its implementation.

In a multivariate setting, the domain of the function is initially defined as a (perhaps very) large hypercube. Random paths of individual choice variable values are searched at each stage. The number of such searches is controlled by the user. The step sizes that partially define the random paths are part of the algorithm and will in general be different for each choice variable at each stage. The algorithm cycles through and individually changes (increases or decreases) all choice variables. Each time a variable change leads to a "better" value of the function being optimized, this point is accepted. Sometimes "worse" points are temporarily accepted. That is, sometimes, the algorithm deliberately allows a search path at a stage to contain "worse" points. By allowing the paths to meander through better and worse points in the domain, the algorithm allows the domain to be searched for better optima that can only be reached by first passing through the worse regions. In this way, the search process can escape from local optima that are dominated by one or more global optima. This distinguishes SA from traditional hill-climbing algorithms that use the local properties of the function to move always in better directions. Movements in worse directions are governed by a Metropolis decision. In a sense, the Metropolis decision can be thought of as the algorithm giving permission to the search process to move in a worse direction, depending upon the roll of a (weighted) die. As the algorithm progresses through subsequent ("cooling") stages, the chance that a worse point will be permitted/accepted decreases. In addition, as the algorithm progresses, the effective domain of the function (that part of the domain to which the search is effectively restricted) is contracted. In part, the search evolves in a fashion consistent with the overall (global versus local) topography of the surface of the function being optimized. Asymptotically, the algorithm converges to a domain that contains the best optimum encountered and which has a user-specified (small) volume. Convergence criteria and the number and length of searches at each stage are determined by parameters set by the user.

The FORTRAN implementation in Ref. [2] is faithful to the algorithm in Ref. [1], but it does include some features and suggestions that tend to help in deciding whether a global optimum has been reached and, in the initial stages of optimization, the features allow the researcher to search individual subsets of the domain of the function. The researcher can control the number

of searches, the initial "temperature" of the model and the rate at which temperature decreases. Uphill and rejected downhill moves are balanced by changes to the upper and lower bounds on parameter changes. In Ref. [2], the suggestion is made that starting the SA algorithm from a variety of randomly selected domain points may provide information about whether a global optimum has been found. This is balanced, to some extent, by the realization that the properties of SA that make it a global optimizer also tend to make it independent of initial values of the choice variables. Any indication of sensitivity to starting values is, therefore, a strong suggestion that the SA user-determined parameters should be changed to provide a more thorough search.

As with any mathematical tool, becoming adept at using SA to solve optimization problems requires practice. FORTRAN code for the SA implementation in Ref. [2] is widely available. This code, written in FORTRAN 77, is carefully documented and contains an example problem that illustrates many of the features of SA. These features, such as convergence criteria, cooling rate, lengths of search paths, and the like can be adjusted to study how the implementation works. It is straightforward to code and solve new problems. The advice in Ref. [2] and in the code is helpful in finding the set of search parameters that solves the problem at hand.

Finally, Goffe et al. [2] address the issue of the speed of SA and recommend the ways that the user can tune the implementation to run more quickly. The implementation was published in 1994, and since then multicore very fast processors with at least 64-bit single precision have become the norm. At the same time, though, there are optimization problems that are now being addressed at the limit of current technology.

1.2. Penalty-constrained optimization

The foregoing discussion has provided an overview of SA and how it can be applied to optimize a function. We now turn to a discussion of how SA can be used to find the global optimum of functions when there are constraints on the choice variables. Our principal concern is how to find the optimum of the statistical functions when the choice variables must satisfy an integral equality constraint. Our approach, however, is quite general and it can be adapted to solve optimization problems subject to several inequality as well as equality constraints that may or not involve integrals. At this point, it is helpful to introduce some notation.

Our choice variables are represented by the vector: θ. The objective function to be maximized is given by: $l(\theta)$. As well, there is a constraint given by: $R(\theta) = \psi_0$. In the statistical problem considered in the next section, we examine the unconstrained problem.

1.2.1. (Unconstrained problem) U: choose θ to maximize l(θ)

(Unconstrained Problem) U: Choose θ to maximize $l(\theta)$ is an important part of the analysis. It is almost always the case in statistical settings that the unconstrained problem can be solved in a straightforward manner using SA.

1.2.2. (Constrained problem) C: choose θ to maximize l(θ) subject to R(θ) = ψ₀

(Constrained Problem) C: Choose θ to maximize $l(\theta)$ subject to $R(\theta) = \psi_0$ can also be solved using simulated annealing. Indeed, as we will see, SA is a very natural approach to solving problem C.

Within a statistical setting, substitution and Lagrange multiplier techniques tend not to work well when attempting to solve C. Typically, the objective function, $l(\theta)$, and the constraint, $R(\theta) = \psi_0$ will not have properties that guarantee a straightforward solution to the optimization problem. For example, the constraint can reasonably be expected to be a highly nonlinear function of the choice variables, θ. In both the substitution and Lagrange approaches, it is necessary at each iteration to solve the constraint equation to express one choice variable in terms of all of the others. If one or more of the choice variables takes on an extreme value (very large or very small) during the iteration process, then this can lead in turn to an extreme constraint solution and such extreme values tend to perpetuate themselves through subsequent iterations. The traditional absence of some form of textbook concavity or convexity on the objective function and/or the constraint function typically results in a failed optimization attempt. Standard derivative-based optimizers often get lost when derivatives take extreme values or when the Hessian of the function is indefinite. In part, this is due to their strong dependence upon local properties of the function being optimized.

The penalty function approach provides an alternative way of dealing with the constraint, which does not require exact satisfaction of the constraint equation at each "iteration." In fact, the constraint equation only holds asymptotically. To clarify this, we begin by introducing the penalty function PL:

$$PL(\theta, \psi_0) = l(\theta) - k\big[R(\theta) - \psi_0\big]^2, \quad k > 0. \qquad (1)$$

This is the penalty function associated with the constrained optimization problem C introduced above. In this case, k is a positive parameter controlled by the researcher.

In a recent paper, Byrne [4] studied how the penalty functions like PL could be used to solve constrained optimization problems such as C. The following three conditions are introduced:

1. θ is chosen from a compact set.

2. The functions $l(\theta)$ and $R(\theta)$ are continuous.

3. Let θ_k^* be the vector that corresponds to the global maximum of PL in Eq. (1) when the parameter multiplying the squared term is k. We assume that each element in the sequence $\{\theta_k^*, k = 1, 2, \ldots\}$ exists.

Then, based on these conditions, it is proved in Ref. [4] that the sequence $\{\theta_k^*, k = 1, 2, \ldots\}$ converges to the θ^* that solves problem C.

This result is extremely important for applying simulated annealing to solve constrained optimization problems. First, SA chooses candidate optimizers from a compact set. Second, continuity of the penalty function PL is more than what is required for SA to reach a global

optimum. The complicating point is that the result in Ref. [4] is expressed in terms of the limit of a sequence of SA optimizations. But, in our experience, this is not a complication of considerable practical importance. In the first place, even for large values of k, SA is not helped by starting the iterations for the $(k + 1)$st solution at the optimal values from the kth solution. In practice, given that SA searches ("cools") sufficiently slowly and follows long enough paths in the domain, it tends to find the global solution of the $(k + 1)$ problem regardless of the starting values it is given. Second, there is a practical issue of how much accuracy can be expected. The SA algorithm terminates when successive improvements in the value of the objective function are all less that a user-specified threshold. This means that the contribution of the term $k[R(\theta) - \psi_0]^2$ must also be small in absolute value. In all of the problems we have considered, setting $k = 100, 000$ is certainly enough to get a high-quality estimate of θ^*. That is, it is reasonable to truncate the sequence at this value of k.

In concluding this section, we reconsider the question: "Why does a penalty approach work when the Lagrange approach fails?" In our experience, the Lagrange approach, which requires solutions of an equation to a given level of accuracy, is prone to problems of numerical accuracy and their propagation. Alternatively, the penalty approach never requires the constraint to be exactly satisfied. Instead, it increasingly discourages large squared deviations of the constraint function $R(\theta)$ from its constraint value ψ_0 as k increases. As a final point, when conditions are satisfied for the Lagrange multiplier formally to exist, it can be obtained as the limit of the partial derivative of the penalty function with respect to the parameter ψ_0 as k increases.

2. Modeling reliability using SA penalized likelihood

2.1. Background on reliability

We consider two variables X and Y. We refer to them respectively as Strength and Stress. For example, Strength could refer to the "time before failure" of a component such as a digital storage device. Alternatively, Stress might measure the total time that the device is used. From the standpoint of a manufacturer, X and Y are both random variables with distributions that can, in principle, be estimated from available breakdown and usage data. Reliability, R, is defined as the probability that the component will withstand the stress it faces in use. In particular,

$$R = P(Y < X). \tag{2}$$

A variety of distributions have been used for X and Y in the literature. The actual choice depends upon the process being studied. It is standard and reasonable to suppose that X and Y are independent.[1]

---

[1] That is, the probability distribution of Y does not depend upon any realized value of X and vice versa. If dependence is possible in a given setting, it is easily accommodated. Formally, (Eq. (2)) will continue to hold, but additional parameters, associated with the interdependence, may appear in both the objective function and the constraint. In a formal sense, the penalty approach is unchanged.

In this section, we suppose that both X and Y are distributed as exponentiated exponential distributions. Exponentiated exponential distributions, $EE(\alpha, \beta)$, have two parameters: $\alpha > 0$ controls shape and $\beta > 0$ controls scale. Adopting the notation introduced in Ref. [5], the cumulative distribution function is:

$$F(x; \alpha, \beta) = (1 - e^{-\beta x})^{\alpha}, \qquad \alpha > 0, \beta > 0, x > 0, \tag{3}$$

As in Ref. [6], we assume that X is distributed as $EE(\alpha_1, \beta_1)$ and Y is distributed as $EE(\alpha_2, \beta_2)$. We do not, however, constrain the scale parameters, β_1 and β_2, to be equal. As a result, the expression for reliability is:

$$R = P(Y < X) \quad = \int_0^\infty \alpha_1 \beta_1 \; (1 - e^{-\beta_1 x})^{\alpha_1 - 1} \; e^{-\beta_1 x} \; (1 - e^{-\beta_2 x})^{\alpha_2} \; dx. \tag{4}$$

There is no known closed-form solution for this integral. As noted in Ref. [7], introducing the change of variables: $z = \beta_1 x$ allows one to see that R is homogeneous of degree 0 in (β_1, β_2). The contours of R are, therefore, all constant along a line in a parameter space defined by $\beta_2 = \beta_1$.

2.2. The unconstrained EE likelihood equation and it properties

Following Ref. [7], we let $x = (x_1, \ldots, x_n)'$ and $y = (y_1, \ldots, y_m)'$ denote the realizations of random samples from $EE(\alpha_1, \beta_1)$ and $EE(\alpha_2, \beta_2)$, respectively. The log-likelihood function of the above model can be written:

$$
\begin{aligned}
l(\alpha_1, \beta_1, \alpha_2, \beta_2; x, y) \quad &= n\log\alpha_1 + n\log\beta_1 + (\alpha_1 - 1)\sum_{i=1}^{n}\log(1 - e^{-\beta_1 x_i}) - \beta_1\sum_{i=1}^{n} x_i \\
&+ m\log\alpha_2 + m\log\beta_2 + (\alpha_2 - 1)\sum_{j=1}^{m}\log(1 - e^{-\beta_2 y_j}) - \beta_2\sum_{j=1}^{m} y_j.
\end{aligned}
\tag{5}
$$

We denote the parameter vector as $\theta = (\alpha_1, \alpha_2, \beta_1, \beta_2)'$.

The Appendix in Ref. [7] contains a derivation of the properties of $l(\theta) = l(\alpha_1, \beta_1, \alpha_2, \beta_2; x, y)$. In particular, $l(\theta)$ is not a concave function of θ nor is it quasi- or pseudo-concave. There is a small region around the point where the gradient of $l(\theta)$ vanishes and in that region, the Hessian matrix is negative definite. Elsewhere in the parameter space, the determinant of the Hessian matrix changes sign frequently. Thus, extreme care must be taken in trying to maximize $l(\theta)$, using a derivative-based algorithm. We found that a variable-metric algorithm would work as long as an approximate Hessian matrix, constrained to be negative definite, is used over a restricted parameter space.

One example, which we consider in greater detail later in the paper, uses the following data with sample sizes of 11 and 9: $x = (2.1828, 0.5911, 1.0711, 0.9007, 1.7814, 1.3616, 0.8629, 0.2301, 1.5183, 0.8481, 1.0845)$ and $y = (0.8874, 1.1482, 0.8227, 0.4086, 0.5596, 1.1978, 1.1324, 0.5625, 1.0679)$. Our SA program quickly and easily solved the associated unconstrained maximum likelihood optimization problem.

2.3. Constrained likelihood maximization

As will be discussed in the next section, inference for the "parameter" $R(\theta)$ in our reliability model requires that the likelihood function $l(\theta)$ be maximized subject to the constraint $R(\theta) = \psi_0$ for a range of values of the constraint parameter ψ_0. These constrained optimization problems are all solved using the penalty function approach introduced in Section 1.2 and using the penalty function $PL(\theta, \psi_0)$ given in Eq. (1). The functions $l(\theta)$ and $R(\theta)$ can be thought of now as the unconstrained *EE* likelihood function and the reliability function, respectively.

3. Likelihood-based inference and penalty functions

In Eq. (5), we introduced the statistical log-likelihood function primarily as an example of a function that needs to be maximized (with and without constraint) in a statistical setting. In this section, we provide more details about likelihood functions and inference. Our brief discussion is not intended as a complete explanation of the underlying statistical notions. Rather, it is intended only to motivate some importance of constrained and unconstrained optimization within statistics. Our discussion is rooted in the example of Eq. (5).

3.1. Background on likelihood models in statistics

Likelihood is akin to probability. The difference in the notions for our purposes is that likelihood is measured in terms of the probability density governing the realizations of a continuous random variable. Technically, the probability of any one outcome, say x_0, of a continuous random variable is 0. The value of the density, say $h(x)$, evaluated at x_0 and multiplied by dx, that is, $h(x_0)\, dx$, can be thought of as approximately the probability that there will be a realization of the random variable X in a very small interval containing x_0. It is common to have situations where the realized (observed) values of a random variable X arise from a process of random sampling where the outcomes are independent of each other yet are governed by identical probability density functions, $h(x)$. The likelihood of a given sample of realized values is defined as the product of the densities corresponding to each of the outcomes in the sample; so the likelihood of a given sample is, up to a scaling factor, a notion similar to the probability of the sample. The likelihood of a sample of realizations of X and Y is, given our assumptions, the product of the likelihoods of the X and the Y samples. For a variety of reasons, it is often easier to work with a positive monotonic transformation of the sample likelihood. In particular, we work with the log-likelihood of the sample. In Eq. (5), we are given the log-likelihood of a sample where the densities come from possibly different exponentiated exponential distributions.

The derivation of the log-likelihood associated with a sample of realizations is just the beginning of the modeling process. Extensions include forecasting the next realization of a random variable or perhaps finding an interval where one can be 95% confident that the next realization of a function of the random variables will fall. For example, we could ask for a 95% confidence interval of the measure of reliability in Eq. (4), incorporating the information in the

sample given at the end of Section 2.2. These are questions of statistical inference. We answer these questions by solving the optimization problems.

The likelihood function given in Eq. (5) can be combined with the sample of 11 realizations of X and 9 realizations of Y given in Section 2.2. We can use the information in the data to estimate the unknown vector of parameters: $\theta = (\alpha_1, \alpha_2, \beta_1, \beta_2)$. One set of estimates of the parameters of the model is obtained by maximizing the likelihood function with the choice variables being the parameters. These maximum likelihood parameter estimates can be thought of as the parameter values that yield specific density functions that are most likely to have generated the data. Of course, 20 observations are not enough to achieve certainty, so there is a related theory about where the true (population) parameters lie in relation to their estimates. Indeed, there are probability distributions associated with the maximum likelihood parameter estimation process, and the parameter values that maximize the log-likelihood for a given sample of data realizations are themselves just realizations. These probability distributions or their approximations allow us to estimate how close the parameter realizations are to the true parameter values.

There is also a probability distribution for the maximized value of the log-likelihood function. This allows us to ask questions such as the following: do I induce a "significant" change in the maximized likelihood value when I constrain the parameter estimates (choice variables) to satisfy an additional condition or set of conditions. This leads back to the constrained optimization problem C in Section 1.2, and the penalty function in Eq. (1).

In the subsection that follows, we present the process of inference for our reliability model. The presentation is more technical.

3.2. Inference in the reliability model

Given $l(\theta)$ is the log likelihood function, we denote the unconstrained maximum likelihood estimator $\widehat{\theta}$ when $l(\theta)$ alone is maximized. As well, we define

$$j\left(\widehat{\theta}\right) = -\frac{\partial^2 l(\theta)}{\partial\theta\partial\theta'}\bigg|_{\widehat{\theta}} \tag{6}$$

as the observed information matrix evaluated at $\widehat{\theta}$. Finally, we let $\widehat{\theta}_\psi$ be the constrained maximum likelihood estimator of θ given by maximizing $l(\theta)$ subject to $R(\theta) = \psi$. Formally, $\widehat{\theta}_\psi$ can be obtained for any ψ in the range of $R(\theta)$ by maximizing $l(\theta)$ subject to the constraint $R(\theta) = \psi$ using the penalty function approach within SA.

The aim next is to obtain inference concerning $R = R(\theta)$, where $dim(R) = 1$. Two widely used likelihood-based methods for obtaining confidence interval for R are based on the asymptotic distribution of the maximum likelihood estimator $\widehat{\theta}$ and the (log) likelihood ratio statistic.

Taking θ as the true population parameter vector, $\left(\widehat{\theta}-\theta\right)'\left[var\left(\widehat{\theta}\right)\right]^{-1}\left(\widehat{\theta}-\theta\right)$ is asymptotically distributed as chi-square with degrees of freedom equal to $dim(\theta)$, and variance-covariance

matrix $\widehat{var}\left(\widehat{\theta}\right) \approx j^{-1}\left(\widehat{\theta}\right)$. Since $R = R(\theta)$ depends upon the entire vector of parameters, we approximate its variance by applying the Delta method to $\widehat{R} = R\left(\widehat{\theta}\right)$ and obtain:

$$\widehat{var}\left(\widehat{R}\right) \approx R'_\theta\left(\widehat{\theta}\right)\widehat{var}\left(\widehat{\theta}\right)R_\theta\left(\widehat{\theta}\right) \tag{7}$$

where

$$R_\theta\left(\widehat{\theta}\right) = \left.\frac{\partial R(\theta)}{\partial \theta}\right|_{\widehat{\theta}}. \tag{8}$$

Since $dim\left(\widehat{R}\right) = 1$, we have

$$\frac{\widehat{R}-R}{\sqrt{\widehat{var}\left(\widehat{R}\right)}} \tag{9}$$

asymptotically distributed as standard normal. An approximate $(1 - \alpha)100\ \%$ confidence interval for R based on $\widehat{\theta}$ is

$$\left(\widehat{R}-z_{\alpha/2}\sqrt{\widehat{var}\left(\widehat{R}\right)}, \widehat{R} + z_{\alpha/2}\sqrt{\widehat{var}\left(\widehat{R}\right)}\right) \tag{10}$$

where $z_{\alpha/2}$ is the $(1 - \alpha/2)100^{\text{th}}$ percentile of the standard normal distribution. This is our first confidence interval.

Alternatively, with regularity conditions stated in Refs. [8, 9], the log likelihood ratio statistic:

$$W(\psi)\ =\ 2\left[\ell\left(\widehat{\theta}\right)-\ell\left(\widehat{\theta}_\psi\right)\right] \tag{11}$$

is asymptotically distributed as chi square with 1 degree of freedom. Therefore, an approximate $(1 - \alpha)100\ \%$ confidence interval of R based on the likelihood ratio statistic is:

$$\left\{\psi : W(\psi) \leq \chi^2_{1,\alpha}\right\}. \tag{12}$$

The set of all constrained values ψ in the domain of R that cannot be rejected at the $(1 - \alpha)100\ \%$ as the true value $R(\theta)$ is defined in Eq. (10). These values form our second confidence interval.

It should be noted that both methods have rates of convergence $O(n^{-1/2})$. While the MLE-based interval is often preferred because of simplicity in calculation, the log-likelihood ratio method has the advantage that it is invariant to reparameterization and the MLE-based method is not. The results presented in Ref. [10] suggest that, in terms of coverage, the confidence interval based on the log-likelihood ratio statistic should be preferred to the MLE-based interval. In particular, when 95% confidence intervals for both statistics are compared, the interval from the log-likelihood ratio statistic is shorter and therefore more precise.

	95% Confidence interval
MLE	(0.306, 0.934)
χ^2	(0.378, 0.823)

Table 1. Interval estimates of R.

The results are summarized in **Table 1**. We note that, consistent with Ref. [10], the χ^2 interval is indeed shorter than the MLE interval.

4. Conclusion

This chapter has considered how SA can play an important role as a global optimizer of constrained likelihood-based statistical models. SA is naturally paired with the penalty function approach to constrained optimization. SA and the penalty approach both require compact domains and bounded functions. Penalty functions must be continuous and, within a statistical setting, this almost always holds. SA supplies the global optimization property that guarantees that the penalty function approach converges to the global constrained optimum. Even though our implementation of SA does not make use of derivatives, the converged penalty function will often be differentiable and Lagrange multipliers, gradients, and Hessian matrices can be calculated. The extension of these results to multiple constraints is computationally straightforward.

In this chapter, we have motivated the pairing of SA and penalty functions in a statistical setting. But the approach can be used to solve many other types of numerical constrained optimization problems. Over time, we have accumulated a considerable amount of experience solving constrained problems using SA and the penalty functions. One lesson stands out: SA is a global optimizer and, for the most part, it should be independent of initial conditions such as starting values of parameters (choice variables). If you find that you get a different optimum after changing the starting values, then it is likely that neither solution is the true global optimum. You can usually remedy this by increasing the initial temperature, slowing the rate of cooling, and/or increasing the length and number of search paths.

Author details

Barry Smith[1] and Augustine Wong[2]*

*Address all correspondence to: august@yorku.ca

1 Department of Economics, York University, Toronto, Canada

2 Department of Mathematics and Statistics, York University, Toronto, Canada

References

[1] Corana, A., Marchese, M., Martini, C. and Ridella, C. Minimizing multimodal functions of continuous variables with the simulated annealing algorithm. ACM Transactions on Mathematical Software. 1987; 13: 262–280.

[2] Goffe, W.L., Ferrier, G.D. and Rogers, J. Global optimization of statistical functions with simulated annealing. Journal of Econometrics. 1994; 60: 65–99.

[3] Li, X., and Smith, B. Diagnostic analysis and computational strategies for estimating discrete time duration models—a Monte Carlo study. Journal of Econometrics. 2015; 187 (1): 275–292.

[4] Byrne, C.L. Sequential unconstrained minimization: a survey. [Internet]. 2013. Available from: http://faculty.uml.edu/cbyrne/SUM.pdf.

[5] Gupta, R.D. and Kundu, D. Exponentiated exponential family: an alternative to gamma and Weibull distributions. Biometrical Journal. 2001; 43(1): 117–130.

[6] Kundu, D. and Gupta, R.D. Estimation of P[$X < Y$] for generalized exponential distributions. Metrika. 2005; 61: 291–308.

[7] Smith, B., Wang, S., Wong, A. and Zhou, X. A penalized likelihood approach to parameter estimation with integral reliability constraints. Entropy. 2015; 17: 4040–4063. DOI:10.3390/e17064040.

[8] Severeni, T. Likelihood Methods in Statistics. New York: Oxford University Press; 2000.

[9] Cox, D.R. and Hinkley, D.V. Thoeretical Statistics. London: Chapman and Hall; 1974.

[10] Doganaksoy, N. and Schmee, J. Comparisons of approximate confidence intervals for distributions used in life-data analysis. Technometrics. 1993; 35: 175–184.