Estimating Parameters of Pareto Distribution under Interval and Fuzzy Uncertainty

Nitaya Buntao
Department of Applied Statistics
King Mongkut's University of Technology North Bangkok
1518 Piboonsongkhram Road, Bangsue
Bangkok 10800 Thailand
Email: taltanot@hotmail.com

***Abstract*—In many application areas, we encounter heavy-tail distributions – for example, such distributions are ubiquitous in financial applications. These distributions are often described by Pareto law. There exist techniques for estimating the parameters of such corresponding Pareto distributions based on the sample $x_1, \ldots, x_n$. In practice, we often only know the values x_i with interval (or, more generally, fuzzy) uncertainty. In this paper, we show how to estimate the parameters of the Pareto distribution under such uncertainty.**

I. Formulation of the Problem

Need for Pareto distributions. In most applications of statistical methods to science and engineering, researchers use either the normal distribution or distributions related to normal – such as lognormal; see, e.g., [6], [8]. In these distributions, the probability of a value decreases exponentially with this value. As a result, large deviations are practically impossible.

For example, for a normal distribution with mean a and standard deviation σ, the probability of a value x to be outside the "three sigma" interval $[a - 3\sigma, a + 3\sigma]$ is approximately 0.1%, and the probability to be outside the "six sigma" interval $[a - 6\sigma, a + 6\sigma]$ is approximately 10^{-8}.

In practice, however, we often encounter random processes in which large deviations are possible. An example of such distributions are returns in financial markets. In financial markets, large deviations from the average are possible. In many cases, such situations are well described by Pareto distributions, in which the probability density is proportional to $\rho(x) \sim x^{-\alpha}$ for some $\alpha > 0$ (and for all $x \geq x_0$). For example, for financial markets, the possibility to use Pareto distributions is described in [7].

Estimating parameters of the Pareto distribution. Large deviations describe crises, so their analysis is very important. To get accurate predictions of the possible large deviations, we must get accurate estimates of the parameters x_0 and α based on the observed data values $x_1, \ldots, x_n$.

In [2], it was shown that by applying the Maximum Likelihood techniques to the Pareto distribution, we get the following estimates:

$$\hat{x}_0 = \min(x_1, \ldots, x_n); \tag{1}$$

and

$$\hat{\alpha} = n \cdot \left(\sum_{i=1}^{n} \ln\left(\frac{x_i}{\min(x_1, \ldots, x_n)} \right) \right)^{-1}. \tag{2}$$

Need to take into account interval and fuzzy uncertainty. In practice, we rarely know the exact values of x_i. For example, in financial situations, we can take, as x_i, the price of the financial instrument at the i-th moment of time – e.g., on the i-th day. However, the price does not remain stable during the day – it fluctuates. Of course, we can always arbitrarily select a value, but it is more reasonable to consider the whole range $[\underline{x}_i, \overline{x}_i]$ of the daily prices instead of a single value x_i.

Different values x_i from the corresponding intervals lead, in general, to different estimates for x_0 and α. It is therefore desirable to find the range of all resulting values of x_0 and α. Estimating this range under interval uncertainty is a particular case of a general problem of *interval computations*; see, e.g., [1], [4].

Not all the values within the interval $[\underline{x}_i, \overline{x}_i]$ may be equally reasonable to consider. Some of these values may be flukes caused by accidental errors. While it is difficult to decide for sure, financial experts can usually tell to what extent the corresponding values are possible. This extent is usually formulated not in precise terms, but by using words from a natural language. For example, an expert may say that some values are most probably flukes, while some other values are most probably reasonable.

To describe these natural-language statements, it is reasonable to use *fuzzy logic*, a formalism specifically designed to formalize such statements; see, e.g., [3], [5]. Based on the information about the possibility of different values $x_i \in [\underline{x}_i, \overline{x}_i]$, it is desirable to conclude what is the degree of possibility of different values x_0 and α from the corresponding intervals.

From the computational viewpoint, fuzzy data processing can be reduced to interval data processing. An alternative way to describe a membership function $\mu_i(x_i)$ is to describe, for each possible value $\alpha \in [0, 1]$, the set of all values x_i for which the degree of possibility is at least α. This set

$$\{x_i : \mu_i(x_i) \geq \alpha\}$$

is called an *alpha-cut* and is denoted by $X_i(\alpha)$.

It is known (see, e.g., [3], [5]), that for alpha-cuts, Zadeh's extension principle takes the following form: for every α, we have

$$R(\alpha) = \{R(x_1, \ldots, x_n) : x_i \in X_i(\alpha)\}.$$

Thus, for every α, finding the alpha-cut of the resulting membership function $\mu(R)$ is equivalent to applying interval computations to the corresponding intervals $X_1(\alpha)$, $\ldots$, $X_n(\alpha)$.

Because of this reduction, in the following text, we will only consider the case of interval uncertainty.

II. First Result: Estimating x_0 under Interval Uncertainty

Problem: reminder. Let us first estimate the range of the estimate $x_0 = \min(x_1, \ldots, x_n)$ when $x_1 \in [\underline{x}_1, \overline{x}_1]$, $\ldots$, $x_n \in [\underline{x}_n, \overline{x}_n]$.

How we can solve this problem. The function $x_0 = \min(x_1, \ldots, x_n)$ is a (non-strictly) increasing function of each of its variables.

Thus, the largest possible value of this function is attained when each of the variables x_i attains its largest possible value $x_i = \overline{x}_i$. So, the largest possible value of x_0 is equal to $\min(\overline{x}_1, \ldots, \overline{x}_n)$.

Similarly, the smallest possible value of this function is attained when each of the variables x_i attains its smallest possible value $x_i = \underline{x}_i$. Thus, the smallest possible value of x_0 is equal to $\min(\underline{x}_1, \ldots, \underline{x}_n)$.

So, we arrive at the following result.

The interval of possible values of x_0: result. The interval $[\underline{x}_0, \overline{x}_0]$ of possible values of the parameter x_0 can be computed as follows:

$$\underline{x}_0 = \min(\underline{x}_1, \ldots, \underline{x}_n); \tag{3}$$

$$\overline{x}_0 = \min(\overline{x}_1, \ldots, \overline{x}_n). \tag{4}$$

III. Estimating α under Interval Uncertainty: Analysis of the Problem

Reducing the problem to simpler ones: idea. We want to find the range $[\underline{\alpha}, \overline{\alpha}]$ of the estimate α, as described by the formula (2). To simplify our analysis, let us reduce this problem to several simpler ones.

Reducing the problem to simpler ones: first step. First, according to the description of the estimate α (formula (2)), this estimate has the form

$$\alpha = \frac{n}{r}, \tag{5}$$

where we denote

$$r = \sum_{i=1}^{n} \ln\left(\frac{x_i}{\min(x_1, \ldots, x_n)}\right). \tag{6}$$

Since the function $\dfrac{n}{r}$ is decreasing,

- the largest possible value $\overline{\alpha}$ of $\alpha = \dfrac{n}{r}$ is attained when r takes the smallest possible value, and
- the smallest possible value $\underline{\alpha}$ of $\alpha = \dfrac{n}{r}$ is attained when r takes the largest possible value.

So, if we can find the range $[\underline{r}, \overline{r}]$ of possible values of r, we can then find the range $[\underline{\alpha}, \overline{\alpha}]$ for α as follows:

$$\underline{\alpha} = \frac{n}{\overline{r}}; \quad \overline{\alpha} = \frac{n}{\underline{r}}. \tag{7}$$

Thus, the original problem of computing the range of a complex expression (2) can be reduced to the auxiliary problem of computing the range of a somewhat simpler expression (6).

Reducing the problem to simpler ones: second step. To reduce the problem further, let us further simplify the expression (6). For this simplification, we can use the fact that r is the sum of several logarithms, and the sum of the logarithms is equal to the logarithm of the product:

$$r = \ln(S), \tag{8}$$

where we denote

$$S \stackrel{\text{def}}{=} \prod_{i=1}^{n} \frac{x_i}{\min(x_1, \ldots, x_n)} = \frac{\prod\limits_{i=1}^{n} x_i}{\min(x_1, \ldots, x_n))^n}. \tag{9}$$

Since the function $\ln(S)$ is increasing,

- the largest possible value $\overline{r}$ of $r = \ln(S)$ is attained when S takes the largest possible value, and
- the smallest possible value $\underline{r}$ of $r = \ln(S)$ is attained when S takes the smallest possible value.

So, if we can find the range $[\underline{S}, \overline{S}]$ of possible values of S, we can then find the range $[\underline{r}, \overline{r}]$ for r as follows:

$$\underline{r} = \ln(\underline{S}); \quad \overline{r} = \ln(\overline{S}). \tag{10}$$

Thus, the problem of computing the range of a complex expression (6) can be reduced to the auxiliary problem of computing the range of a somewhat simpler expression (9).

Further reduction. When we know that x_j is the smallest of n values $x_1, \ldots, x_n$, then the expression (9) can be simplified even further:

$$S = \frac{\prod\limits_{i=1}^{n} x_i}{x_j^n}. \tag{11}$$

By canceling the terms x_j in the numerator and in the denominator, we cam further simplify this expression into

$$S = \frac{\prod\limits_{i \neq j} x_i}{x_j^{n-1}}. \tag{12}$$

Let us show how after this reduction, we can explicitly compute both bounds $\underline{S}$ and $\overline{S}$.

Computing $\overline{S}$: analysis. The expression (12) is increasing as a function of all the variables x_i with $i \neq j$ and decreasing as a function of the remaining variable x_j. Thus, its largest possible value is attained when:

- all the variables x_i with $i \neq j$ attains their largest possible values $\overline{x}_i$, while
- the variable x_i attains its smallest possible value $\underline{x}_j$.

The corresponding expression is equal to

$$S_j = \frac{\prod\limits_{i \neq j} \overline{x}_i}{\underline{x}_j^{n-1}}. \tag{13}$$

Multiplying both numerator and denominator by $\overline{x}_j$, we conclude that

$$S_j = \frac{\prod\limits_{i=1}^{n} \overline{x}_i}{\overline{x}_j \cdot \underline{x}_j^{n-1}}. \tag{14}$$

This expression is only possible when $x_j \leq x_i$ for all $i \neq j$, i.e., when $\underline{x}_j \leq \overline{x}_i$ for all i. A number is smaller than several numbers if it is smaller than the smallest of them, i.e., if

$$\underline{x}_j \leq \min(\overline{x}_1, \ldots, \overline{x}_n). \tag{15}$$

It should be mentioned that the right-hand side of this inequality has already appeared in this text – as the upper endpoint $\overline{x}_0$ for the parameter x_0.

Among the values S_j corresponding to all such j, we need to choose the largest one. According to (14), each of the values S_j is the result of dividing the same product $\prod\limits_{i=1}^{n} \overline{x}_i$ by the value $\overline{x}_j \cdot \underline{x}_j^{n-1}$. Thus, the largest possible value S_j corresponds to the smallest possible value of the product $\overline{x}_j \cdot \underline{x}_j^{n-1}$.

The largest value $\overline{S}$ of S corresponds to the largest value $\overline{r}$ of r and thus, to the smallest value $\underline{\alpha}$ of α. Thus, we arrive at the algorithm for computing $\underline{\alpha}$ that is described in the next section.

Computing $\underline{S}$: analysis. Let $x_1, \ldots, x_n$ be the values at which the function S attains its minimum, and let x_j be the smallest of these values.

If all the values x_i are equal to each other, then we get $S = 1$. In this case, we can increase all the values until we reach the upper endpoint $\overline{x}_i$ of one of the intervals. Then, we get $x_i = \overline{x}_i$, and for every other k, we get $\overline{x}_i = x_i = x_k \leq \overline{x}_k$, hence $\overline{x}_i \leq \overline{x}_k$ for all k, and $\overline{x}_i = \min(\overline{x}_1, \ldots, \overline{x}_n)$ $(= \overline{x}_0)$.

For this i, we have $x_i = \overline{x}_i$, and for all other $k \neq i$, we get $x_k = \max(\overline{x}_i, \underline{x}_k)$. Let us show that a similar formula holds when not all the coordinates of the optimizing vector $(x_1, \ldots, x_n)$ are equal to each other.

Indeed, the expression (12) is increasing as a function of all the variables x_i with $i \neq j$ and decreasing as a function of the remaining variable x_j.

Thus, if we could increase x_j without changing all other values x_i – and still preserve the conditions $x_j \leq \overline{x}_j$ and the inequalities $x_j \leq x_i$ – we would be able to further decrease the value (12). Since we started with the values for which S attains its minimum, such a increase in x_j is impossible. The fact that we cannot increase x_j without violating the constraints $x_j \leq \overline{x}_j$ and $x_j \leq x_i$ means that at least in one of the constraints, we have equality. Thus:

- we either have $x_j = \overline{x}_j$,
- or we have $x_j < \overline{x}_j$ and $x_j = x_i$ for some $i \neq j$.

Let us consider the second case, when we have several values x_i for which $x_j = x_i$. Let n_j be the total number of such values x_j. Then, the product forming the numerator of the formula (11) for S contains n_j terms equal to x_j, i.e., it contains the factor $x_j^{n_j}$. Dividing both the numerator and the denominator of the formula (11) by this factor, we conclude that

$$S = \frac{\prod\limits_{i: x_i > x_j} x_i}{x_j^{n-n_j}}. \tag{16}$$

If for all the indices i for which $x_i = x_j$, we have $x_i < \overline{x}_i$, then we can increase this common value $x_j = x_i = \ldots$ without changing any other value x_k and thus, further decrease S. So, the fact that we have selected the minimizing vector implies that at least for one i, we have $x_j = x_i = \overline{x}_i$.

Thus, for the minimizing vector, the smallest value $\min(x_1, \ldots, x_n)$ is attained at one of the upper endpoints $\overline{x}_i$. Since this value $\overline{x}_i$ is the smallest, we get $\overline{x}_i \leq x_k$ for all $k \neq i$, and since $x_k \leq \overline{x}_k$, we conclude that $\overline{x}_i \leq \overline{x}_k$ for all k. Thus, the minimal value $\overline{x}_i = \min(x_1, \ldots, x_n)$ is the smallest of n upper endpoints:

$$x_j = \min(\overline{x}_1, \ldots, \overline{x}_n) = \overline{x}_0. \tag{17}$$

For every $k \neq i$, we select the smallest possible value $x_k \in [\underline{x}_k, \overline{x}_k]$ for which $x_k \geq \overline{x}_k$, i.e., the value $x_k = \max(\overline{x}_i, \underline{x}_k)$.

The smallest value $\underline{S}$ of S corresponds to the smallest value $\underline{r}$ of r and thus, to the largest value $\overline{\alpha}$ of α. Thus, we arrive at the algorithm for computing $\overline{\alpha}$ that is described in the corresponding section.

IV. Algorithm for Computing $\underline{\alpha}$

First stage. To find $\underline{\alpha}$, first, we compute the value

$$\overline{x}_0 = \min(\overline{x}_1, \ldots, \overline{x}_n). \tag{18}$$

Comment. If we have already computed the range $[\underline{x}_0, \overline{x}_0]$, for x_0, then we do not need to compute anything: we just borrow the corresponding value $\overline{x}_0$.

Second stage. We test $j = 1, \ldots, n$, and among all the indices j for which $\underline{x}_j \leq \overline{x}_0$, we select the one with the smallest possible value of the product $\overline{x}_j \cdot \underline{x}_j^{n-1}$.

Final formula. The smallest possible value of α is attained when x_j takes the value $\underline{x}_j$ while all other variables take the values $\overline{x}_i$. For these values, $\min(x_1, \ldots, x_n) = \underline{x}_j$, hence the j-th term in the sum (2) disappears, and the expression (2) takes the form

$$\underline{\alpha} = n \cdot \left(\sum_{i \neq j} \ln\left(\frac{\overline{x}_i}{\underline{x}_j} \right) \right)^{-1}. \tag{19}$$

Computation time. At each stage, this algorithm takes the linear number of steps, i.e., the number of steps bounded by

the number of variables n. Thus, overall, we have a linear-time algorithm.

This computation time is asymptotically optimal. Indeed, we need to take into account each of the intervals $[\underline{x}_i, \overline{x}_i]$. We need at least one computation step to read each of these values. Thus, the overall number of computation steps cannot be smaller than n. So, our algorithm that takes times $\leq \text{const} \cdot n$ is asymptotically optimal.

V. Algorithm for Computing $\overline{\alpha}$

First stage. To find $\overline{\alpha}$, first, we compute the value

$$\overline{x}_0 = \min(\overline{x}_1, \ldots, \overline{x}_n). \tag{20}$$

Comment. If we have already computed the range $[\underline{x}_0, \overline{x}_0]$, for x_0, then we do not need to compute anything: we just borrow the corresponding value $\overline{x}_0$.

Second stage. For each $k = 1, \ldots, n$, we take $x_k = \max(\overline{x}_0, \underline{x}_k)$, and then compute the corresponding value α as

$$\overline{\alpha} = n \cdot \left(\sum_{k=1}^{n} \ln\left(\frac{\max(\overline{x}_0, \underline{x}_k)}{\overline{x}_0} \right) \right)^{-1}. \tag{21}$$

Computation time. This algorithm also takes linear time and is, thus, also asymptotically optimal.

Acknowledgments

The author would like to thank Sa-aat Niwitpong, Hung T. Nguyen, Tony Wang, and Vladik Kreinovich for their encouragement and advise, and anonymous referees for their useful suggestions.

References

[1] L. Jaulin, M. Kieffer, O. Didrit, and E. Walter, *Applied Interval Analysis, with Examples in Parameter and State Estimation, Robust Control and Robotics*, Springer-Verlag, London, 2001.

[2] S. Kinsella and F. O'Brien, "Maximum likelihood estimation of stable Paretian distribution applied to index and option data", *Proceedings of the INFINITI Conference on International Finance*, Dublin, Ireland, June 8–9, 2009.

[3] G. Klir and B. Yuan, *Fuzzy Sets and Fuzzy Logic: Theory and Applications*, Upper Saddle River, New Jersey: Prentice Hall, 1995.

[4] R. E. Moore, R. B. Kearfott, and M. J. Cloud, *Introduction to Interval Analysis*, SIAM Press, Philadelphia, Pennsylvania, 2009.

[5] H. T. Nguyen and E. A. Walker, *First Course on Fuzzy Logic*, CRC Press, Boca Raton, Florida, 2006.

[6] S. Rabinovich, *Measurement Errors and Uncertainties: Theory and Practice*, American Institute of Physics, New York, 2005.

[7] S. T. Rachev and S. Mittnik, *Stable Paretian Models in Finance*, Wiley Publishers, New York, 2000.

[8] D. J. Sheskin, *Handbook of Parametric and Nonparametric Statistical Procedures*, Chapman & Hall/CRC, Boca Raton, Florida, 2004.