



HAL
open science

Feller-Pareto and Related Distributions: Numerical Implementation and Actuarial Applications

Christophe Dutang, Vincent Goulet, Nicholas Langevin

► **To cite this version:**

Christophe Dutang, Vincent Goulet, Nicholas Langevin. Feller-Pareto and Related Distributions: Numerical Implementation and Actuarial Applications. *Journal of Statistical Software*, 2022, 103 (6), 10.18637/jss.v103.i06 . hal-03725415

HAL Id: hal-03725415

<https://hal.science/hal-03725415v1>

Submitted on 10 Oct 2024

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



Feller-Pareto and Related Distributions: Numerical Implementation and Actuarial Applications

Christophe Dutang 
Université Paris-Dauphine

Vincent Goulet 
Université Laval

Nicholas Langevin
Université Laval

Abstract

Actuaries model insurance claim amounts using heavy tailed probability distributions. They routinely need to evaluate quantities related to these distributions such as quantiles in the far right tail, moments or limited moments. Furthermore, actuaries often resort to simulation to solve otherwise untractable risk evaluation problems. The paper discusses our implementation of support functions for the Feller-Pareto distribution for the R package **actuar**. The Feller-Pareto defines a large family of heavy tailed distributions encompassing the transformed beta family and many variants of the Pareto distribution.

Keywords: Feller-Pareto, distribution, transformed beta, beta of the second kind, **actuar**, R.

1. Introduction

Actuaries are experts in evaluating the likelihood and financial consequences of future events. A pivotal part of their work is the modeling of the size and frequency of insurance claims. With probability models for the claims process in hand, actuaries can compute insurance premiums, determine the amount a company has to set aside in its *reserve* to cope with future events, evaluate the risk the company will not be able to meet its obligations, or run simulations to compare business strategies or solve otherwise untractable problems.

Needless to say, the probability distribution is the cornerstone of actuarial and statistical modeling. In this data science golden age, one needs to be able to compute or simulate various quantities for a large array of probability distributions. Actuarial applications, for example, require accurate computation of probabilities or quantiles in the far right tail of distributions; easy computation of raw and limited moments; fast and reliable simulation of random variates (see, e.g., McNeil, Frey, and Embrechts 2015).

The R statistical system (R Core Team 2022a) provides in its base distribution a number of functions to compute the probability density function (PDF) or the probability mass function

(PMF), the cumulative distribution function (CDF) and the quantile function, as well as functions to generate variates for the most common discrete and continuous distributions of statistics.

The number of probability distributions implemented in base R has remained remarkably stable over time. The R Core Team has elected to delegate support for additional distributions to extension packages developed and maintained by the R community. The Comprehensive R Archive Network (CRAN) at <https://CRAN.R-project.org> is the central repository of R packages. In their extensive *CRAN Task View* on probability distributions, [Dutang and Kiener \(2022\)](#) list packages that support probability distributions not found in base R (263 packages as of March 07, 2022).

First released on CRAN in 2005, **actuar** ([Dutang, Goulet, and Pigeon 2008](#)), available at <https://CRAN.R-project.org/package=actuar>, provides specialized functions for actuarial applications, but also support functions for a large number of continuous size distributions useful for loss severity modeling; for phase-type distributions used in computation of ruin probabilities; for zero-truncated and zero-modified extensions of the discrete distributions commonly used in loss frequency modeling; for the heavy tailed Poisson-inverse Gaussian discrete distribution. The package also introduces support functions to compute raw moments, limited moments and the moment generating function (when it exists) of continuous distributions. These functions all prove useful in loss modeling and risk evaluation.

We introduce with this paper our latest addition to the range of heavy tailed probability models supported by **actuar**: the Feller-Pareto distribution and related Pareto distributions with a location parameter. To the best of our knowledge, the Feller-Pareto distribution has not yet been extensively used for insurance loss modeling. However, [Brazauskas \(2002\)](#) proposes its use to model the size of insurance claims and [Cummins, Dionne, McDonald, and Pritchett \(1990\)](#) relies on a special case to estimate the tail distribution and reinsurance premiums for fire claims.

The paper is organized as follows. For the reader not familiar with the Feller-Pareto family of distributions, we first summarize how it extends the transformed beta family in Section 2. Next, we discuss the theoretical and numerical challenges faced during the numerical implementation in Section 3. In particular in Section 3.2, we also compare various algorithms to generate random variates from the Feller-Pareto and related distributions. In Section 4, we provide a small case study based on real world data. The workhorses behind the R support functions are implemented in C for unrivaled speed and efficiency. We close the paper by briefly explaining how R package developers can now take advantage of these routines through **actuar**'s new package API in Section 5.

For the purpose of this paper, a heavy tailed distribution is a positive distribution without a finite moment generating function. Other definitions exist, such as the domain of attraction in extreme value theory ([Embrechts, Klüppelberg, and Mikosch 1997](#)).

2. Feller-Pareto distribution

The Pareto distribution is a well known probability distribution originally used by economists to model income in a society. Due to its heavy tail, the distribution also plays an important role in actuarial science to model loss cost, notably in reinsurance. In recent years, many authors proposed increasingly flexible variants of the Pareto that eventually led to the distri-

bution now generally known as the Feller-Pareto; see [Arnold \(2015\)](#); [Kleiber and Kotz \(2003\)](#) for exhaustive surveys.

2.1. Definition

[Arnold \(2015\)](#) proposes the Feller-Pareto distribution as the distribution of the random variable

$$X = \mu + \theta \left(\frac{U}{V} \right)^{1/\gamma}, \quad \gamma, \theta > 0, \mu \in \mathbb{R}, \quad (1)$$

where U and V are two independent gamma distributions with shape parameter $\tau > 0$ and $\alpha > 0$, respectively, and common scale parameter 1. In the sequel, we denote $U \sim \mathcal{G}(\tau, 1)$, $V \sim \mathcal{G}(\alpha, 1)$ and $X \sim \mathcal{FP}(\mu, \alpha, \gamma, \tau, \theta)$.

One will note the presence of the location parameter μ . In actuarial applications, this proves useful for the modeling of claim amounts above a certain threshold such as, for example, for reinsurance contracts stipulating that the reinsurance company is responsible for the total amount of losses above a certain limit.

It is well known that the distribution of the ratio U/V in (1) is a beta distribution of the second kind (see [Johnson, Kotz, and Balakrishnan 1994a](#)). Therefore, the CDF of X is given in terms of the (incomplete) beta function by

$$F(x) = I(\tau, \alpha; u) = \frac{\beta(\tau, \alpha; u)}{\beta(\alpha, \tau)}, \quad u = \frac{y}{1+y}, \quad y = \left(\frac{x-\mu}{\theta} \right)^\gamma, \quad x \geq \mu, \quad (2)$$

where $I(a, b; x)$ is the (regularized) incomplete beta function — or incomplete beta ratio — representing the CDF of the beta distribution (henceforth denoted $\mathcal{B}(a, b)$). We define

$$\beta(a, b; x) = \int_0^x t^{a-1} (1-t)^{b-1} dt, \quad a > 0, b > 0, 0 < x < 1,$$

as the incomplete beta function, and

$$\beta(a, b) = \int_0^1 t^{a-1} (1-t)^{b-1} dt = \frac{\Gamma(a)\Gamma(b)}{\Gamma(a+b)}.$$

as the beta function ([Abramowitz and Stegun 1972](#)).

The PDF corresponding to the CDF $F(x)$ above is

$$f(x) = \frac{\gamma((x-\mu)/\theta)^{\gamma\tau-1}}{\theta\beta(\alpha, \tau)[1 + ((x-\mu)/\theta)^\gamma]^{\alpha+\tau}}, \quad x \geq \mu. \quad (3)$$

We can rewrite this equation as follows:

$$f(x) = \frac{\gamma u^\tau (1-u)^\alpha}{(x-\mu)\beta(\alpha, \tau)}, \quad u = \frac{y}{1+y}, \quad y = \left(\frac{x-\mu}{\theta} \right)^\gamma, \quad x \geq \mu,$$

which highlights a link between the Feller-Pareto distribution and the beta (of the first kind). Indeed, it is also well known that if $U \sim \mathcal{G}(\tau, 1)$, $V \sim \mathcal{G}(\alpha, 1)$ and U and V are independent, then $B = V/(V+U) \sim \mathcal{B}(\alpha, \tau)$. We can rewrite (1) as

$$X = \mu + \theta(B^{-1} - 1)^{1/\gamma}, \quad (4)$$

thereby defining the Feller-Pareto as a transformation of a beta distribution. It is actually under this form that [Feller \(1968\)](#) and [Johnson, Kotz, and Balakrishnan \(1994b\)](#); [Johnson et al. \(1994a\)](#) introduce the Feller-Pareto.

One final remark that will prove useful in the sequel. Another well known result about beta distributions is that if B has a $\mathcal{B}(\alpha, \tau)$ distribution, then the reflected variable $\tilde{B} = 1 - B$ has a $\mathcal{B}(\tau, \alpha)$ distribution. Therefore, by a simple manipulation of (4), we can equivalently define the Feller-Pareto as the distribution of the random variable

$$X = \mu + \theta(\tilde{B}^{-1} - 1)^{-1/\gamma}, \quad (5)$$

where $\tilde{B} \sim \mathcal{B}(\tau, \alpha)$.

2.2. Moments

The computation of moments plays a central role in actuarial applications. The first raw moment — or expected value — is tightly linked to the concept of insurance premium, whereas the second central moment — or variance — measures the risk associated with this premium.

A general formula for the (raw) moment of order k

$$m_k = \mathbf{E} \left[X^k \right],$$

is easy to derive for the Feller-Pareto distribution with $\mu = 0$ using definition (1) as a ratio of independent gamma distributions. Indeed, in that case,

$$m_k = \mathbf{E} \left[\theta^k \left(\frac{U}{V} \right)^{k/\gamma} \right] = \theta^k \mathbf{E} \left[V^{-m/\gamma} \right] \mathbf{E} \left[U^{m/\gamma} \right].$$

As [Arnold \(2015\)](#) and [Klugman, Panjer, and Willmot \(2019\)](#) show, we readily obtain

$$m_k = \frac{\theta^k \Gamma(\tau + k/\gamma) \Gamma(\alpha - k/\gamma)}{\Gamma(\alpha) \Gamma(\tau)}, \quad -\tau\gamma < k < \alpha\gamma. \quad (6)$$

For the more general case $\mu \neq 0$, one derives the moment of order k using the binomial expansion

$$\mathbf{E} \left[\left(\mu + \theta \left(\frac{U}{V} \right)^{1/\gamma} \right)^k \right] = \sum_{j=0}^k \binom{k}{j} \mu^{k-j} \mathbf{E} \left[\theta^j \left(\frac{U}{V} \right)^{j/\gamma} \right].$$

The resulting general equation for the raw moment of order k of the Feller-Pareto is:

$$m_k = \sum_{j=0}^k \binom{k}{j} \frac{\mu^{k-j} \theta^j \Gamma(\tau + j/\gamma) \Gamma(\alpha - j/\gamma)}{\Gamma(\alpha) \Gamma(\tau)}. \quad (7)$$

Note, however, that due to constraints on the validity of the binomial expansion, the above result is only valid for nonnegative integral values of $k < \alpha\gamma$ ([Arnold 2015](#)).

2.3. Limited moments

Limited moments play an important role in actuarial science, especially the limited expected value as the (pure) premium for an insurance contract subject to a maximal insurable loss as is common in homeowners insurance ([Klugman et al. 2019](#), Chapter 3).

Let $a \wedge b = \min(a, b)$ denote the minimum of a and b . We define the limited moment of order k of a random variable X as

$$\mathbb{E}[(X \wedge x)^k] = \mathbb{E}[\min(X, x)^k].$$

The limited expected value is the special case with $k = 1$.

Like raw moments, the limited moments of the Feller-Pareto are simpler to derive in the case $\mu = 0$. Klugman *et al.* (2019, p. 463) already provide

$$\begin{aligned} \mathbb{E}[(X \wedge x)^k] &= \frac{\theta^k \Gamma(\tau + k/\gamma) \Gamma(\alpha - k/\gamma)}{\Gamma(\alpha) \Gamma(\tau)} I(\tau + j/\gamma, \alpha - j/\gamma; u) \\ &+ x^k (1 - I(\tau, \alpha; u)), \end{aligned} \quad (8)$$

for $x \geq 0$, $k > -\tau\gamma$, and with, as in Section 2.1, $u = y/(1 + y)$ and $y = ((x - \mu)/\theta)^\gamma$. (We will further discuss in Section 3.4 how $\Gamma(a)\Gamma(b)I(a, b; x)$ may also be valid for $b < 0$.)

Using again binomial expansion, we can similarly generalize the above formula for the limited moment of order k when $\mu \neq 0$:

$$\begin{aligned} \mathbb{E}[(X \wedge x)^k] &= \sum_{j=0}^k \binom{k}{j} \frac{\mu^{k-j} \theta^j \Gamma(\tau + j/\gamma) \Gamma(\alpha - j/\gamma)}{\Gamma(\alpha) \Gamma(\tau)} I(\tau + j/\gamma, \alpha - j/\gamma; u) \\ &+ x^k (1 - I(\tau, \alpha; u)), \end{aligned} \quad (9)$$

for $x \geq \mu$ and, this time, nonnegative integral values of k .

2.4. Generation of random variates

The definitions of Section 2.1 provide us with three straightforward algorithms to generate random variates from the Feller-Pareto distribution.

The first algorithm relies on definition (1) of the Feller-Pareto as a ratio of Gamma random variables.

Algorithm 1. Simulate variates from a $\mathcal{FP}(\mu, \alpha, \tau, \gamma, \theta)$ with PDF (3), $\mu \in \mathbb{R}$ and $\alpha, \tau, \gamma, \theta > 0$.

1. Generate u from a $\mathcal{G}(\tau, 1)$.
2. Generate v from a $\mathcal{G}(\alpha, 1)$.
3. Return $\mu + \theta(u/v)^{1/\gamma}$.

The second and third algorithms are very similar in that they both use transformations of a beta of the first kind, one using definition (4), the other using definition (5). We introduce the third algorithm here since it was actually the one used in **actuar** — with $\mu = 0$ — to simulate variates from the transformed beta distribution.

Algorithm 2. Simulate variates from a $\mathcal{FP}(\mu, \alpha, \tau, \gamma, \theta)$ with PDF (3), $\mu \in \mathbb{R}$ and $\alpha, \tau, \gamma, \theta > 0$.

1. Generate x from a $\mathcal{B}(\alpha, \tau)$.

2. Return $\mu + \theta(1/x - 1)^{1/\gamma}$.

Algorithm 3. Simulate variates from a $\mathcal{FP}(\mu, \alpha, \tau, \gamma, \theta)$ with PDF (3), $\mu \in \mathbb{R}$ and $\alpha, \tau, \gamma, \theta > 0$.

1. Generate x from a $\mathcal{B}(\tau, \alpha)$.
2. Return $\mu + \theta(1/x - 1)^{-1/\gamma}$.

2.5. Historical remarks

The later part of the twentieth century saw a lot of simultaneous independent work on probability distributions to model income and loss cost. On the one side, economists proposed distributions to model income — such as the Singh-Maddala or Dagum distributions — until McDonald (1984) introduced a four-parameter generalized beta of the second kind distribution with PDF

$$f(x) = \frac{ax^{ap-1}}{b^{ap}\beta(p, q)(1 + (x/b)^a)^{p+q}}, \quad a, b, p, q > 0, \quad x > 0. \quad (10)$$

This is exactly a Feller-Pareto with $\mu = 0$, $b = \theta$, $b = \gamma$, $p = \tau$ and $q = \alpha$. See Atkinson and Bourguignon (2015) for a thorough discussion of income distributions and Cummins *et al.* (1990, Table 2) for the interpretation as a mixture.

On the other side, as reported by Kleiber and Kotz (2003), actuaries also proposed various distributions to model insurance loss cost — such as the Fisk or Lomax distributions — until Venter (1983) proposed a distribution with a PDF identical to (10) and called it the transformed beta distribution.

2.6. Special cases

One will find many variants of the Pareto distribution in the literature, most of which can be considered special cases of the Feller-Pareto. We review some of them below. For the reader interested to know more, we highly recommend the extensive work of Arnold (2015) on the Pareto family of distributions.

As a first case of the Feller-Pareto distribution, let us consider the aptly named Pareto I, or Single Parameter Pareto (Klugman *et al.* 2019). We obtain this distribution by setting the location parameter μ equal to the scale parameter θ , and by setting the shape parameters γ and τ equal to 1. The resulting CDF is

$$F(x) = 1 - \left[\frac{x}{\theta} \right]^{-\alpha}, \quad x > \theta. \quad (11)$$

Even though this distribution appears to have two parameters, its name stems from the fact that only α is an actual parameter for estimation purposes, θ being a known threshold.

Next, the Pareto II distribution is obtained by setting $\gamma = \tau = 1$ in the Feller-Pareto. The CDF is then

$$F(x) = 1 - \left[1 + \frac{x - \mu}{\theta} \right]^{-\alpha}, \quad x > \mu. \quad (12)$$

When $\mu = 0$, we obtain what is generally simply called the Pareto distribution.

Then we have the Pareto III distribution, for which we can find many different definitions in the literature. The definition of [Arnold \(2015\)](#) is in line with the Pareto II, but with a different shape parameter. The CDF is

$$F(x) = 1 - \left[1 + \left(\frac{x - \mu}{\theta} \right)^\gamma \right]^{-1}, \quad x > \mu, \quad (13)$$

which is a Feller-Pareto with shape parameters $\alpha = \tau = 1$. [Johnson *et al.* \(1994b,a\)](#), on the other hand, use the original definition of Pareto himself

$$F(x) = 1 - \frac{Ce^{-\beta x}}{(x - \mu)^\alpha}.$$

However, as explained in [Kleiber and Kotz \(2003\)](#), parameter estimation is hard with this definition of the Pareto III. For this reason, we will keep the above [Arnold \(2015\)](#) definition.

Finally, the Pareto IV is a Feller-Pareto where only the shape parameter τ is set equal to 1. The CDF is

$$F(x) = 1 - \left[1 + \left(\frac{x - \mu}{\theta} \right)^\gamma \right]^{-\alpha}, \quad x > \mu. \quad (14)$$

The Feller-Pareto generalizes other common continuous extreme value distributions found in Appendix A of [Klugman *et al.* \(2019\)](#): the case $\mu = 0$ is the transformed beta distribution, as mentioned above; the case $\mu = 0, \tau = 1$ is the Burr distribution type 12; the case $\mu = 0, \tau = 1, \alpha = 1$ is the loglogistic distribution; the case $\mu = 0, \alpha = 1$ is the inverse Burr or Burr type 3 distribution. Figure 1 shows the relationships between the Feller-Pareto and the distributions of the transformed beta family of [Klugman *et al.* \(2019\)](#). One may connect the diagram of Figure 1 to the impressive diagram of relationships between univariate distributions of [Leemis and McQueston \(2008\)](#), starting with the Pareto or generalized Pareto distributions. [\(Cummins *et al.* 1990\)](#) also propose a diagram for the relationships between the members of generalized beta of the second kind family.

2.7. R implementation

As already mentioned in the introduction of this paper, R includes functions to compute the PDF, the CDF and the quantile function of standard probability laws, as well as functions to generate variates from these laws. For some root `foo` (e.g., `pois` for the Poisson distribution or `gamma` for the gamma distribution), the support functions are named `dfoo`, `pfoo`, `qfoo` and `rfoo`, respectively.

Since version 0.9-4, released in 2007, the `actuar` package provides `d`, `p`, `q` and `r` functions for all the probability laws useful for loss severity modeling found in Appendix A of [Klugman *et al.* \(2019\)](#) and not already present in base R, excluding the log- t but including the Loggamma distribution of [Hogg and Klugman \(1984\)](#). Version 2.0-0 of the package also added support for the Inverse Gaussian distribution. The package vignette “distributions” keeps the up-to-date list of supported distributions and root names of the R functions.

In addition to the `d`, `p`, `q` and `r` functions, `actuar` provides `m`, `lev` and `mgf` functions to compute, respectively, theoretical raw moments m_k , theoretical limited moments $\mathbb{E}[(X \wedge x)^k]$ and the moment generating function

$$M_X(t) = \mathbb{E} \left[e^{tX} \right],$$

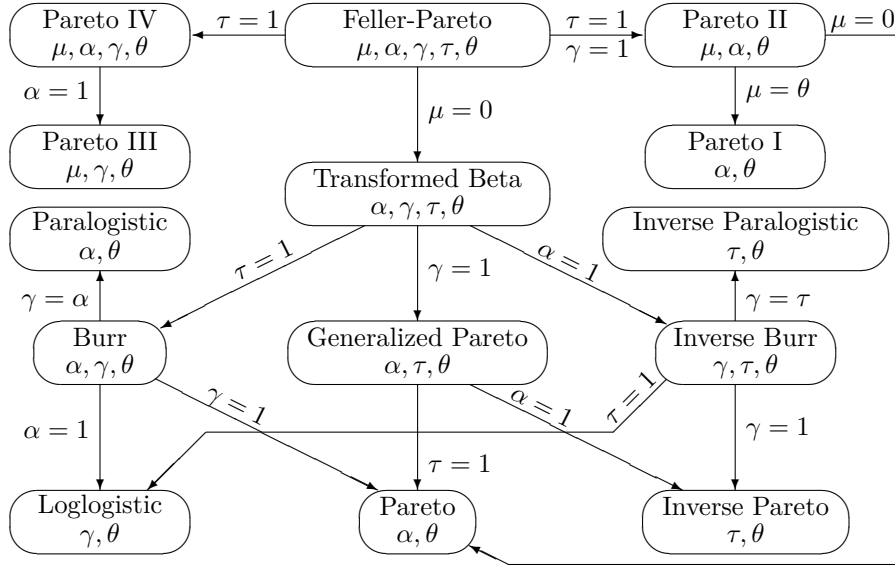


Figure 1: Interrelations between distributions of the Feller-Pareto and transformed beta families. This diagram is an extension of Figure 5.2 of Klugman *et al.* (2019).

Parameter	Argument
μ	min
α	shape1
γ	shape2
τ	shape3
$\lambda = 1/\theta$	rate
θ	scale

Table 1: Correspondence between the parameters of the Feller-Pareto distribution and the arguments of the `fpareto` family of R functions.

when the latter exists. The moment generating function of the Feller-Pareto distribution does not exist since the density (3) does not decrease at an exponential rate.

With version 3.0-0, released alongside this paper, **actuar** adds support for the Feller-Pareto, Pareto IV, Pareto III and Pareto II distributions. The Pareto I and the standard Pareto distribution without a location parameter were already supported by **actuar**. The root of the support functions are, respectively, `fpareto`, `pareto4`, `pareto3` and `pareto2`. Therefore, for the Feller-Pareto, the package now provides the functions `dfpareto`, `pfpareto`, `qfpareto`, `rfpareto`, `mfpareto` and `levfpareto`. Table 1 shows the correspondence between the parameters of the Feller-Pareto as defined in Section 2.1 and the arguments of the R functions.

In the sequel, we only discuss the implementation of the Feller-Pareto functions since the Pareto IV, III and II functions are merely special cases.

3. Computational challenges

This section discusses numerical and performance issues we encountered while implementing in R some of the functions of the `fpareto` family.

3.1. Cumulative distribution function

We first look at the implementation of `pfpareto` to compute the CDF (2) of the Feller-Pareto distributions. This requires to evaluate the incomplete beta ratio $I(a, b; x)$. Since this is just the CDF of the beta distribution, we have at our disposal the function `pbeta` in base R to carry the computation efficiently.

Actually, it proves not so simple. In our accuracy tests, we observed that when $y = [(x - \mu)/\theta]^\gamma$ gets “large”, then $u = y/(1 + y)$ becomes numerically equal to 1 and, accordingly, `pbeta` returns 1. This can represent a significant jump in the CDF of the Feller-Pareto; see the dashed line in Figure 2. In actuarial applications, events in the right tail of a heavy tailed probability distribution are the most critical ones for an insurance company: although very unlikely, they represent a huge financial exposure.

We needed a fix for the computation of the probabilities — or, equivalently, of the quantiles — in the far right tail of the distribution. It came from the symmetry property of the incomplete beta ratio. As stated in Abramowitz and Stegun (1972, Section 6.6.3):

$$I(a, b; x) = 1 - I(b, a; 1 - x). \quad (15)$$

Therefore, we can also write the CDF (2) of the Feller-Pareto distribution as

$$F(x) = 1 - I(\alpha, \tau; 1 - u), \quad u = \frac{y}{1 + y}, \quad y = \left(\frac{x - \mu}{\theta}\right)^\gamma, \quad x \geq \mu. \quad (16)$$

Now, the value $1 - u = 1/(1 + y)$ underflows to zero much later than $y/(1 + y)$ “overflows” to one for large values of y . Using the approach in (16) coupled with the standard argument `lower.tail = FALSE` of `pbeta` to compute $1 - I(a, b; x)$, we are able to compute probabilities accurately much farther in the right tail of the distribution; see the solid line in Figure 2.

The previous discussion also holds for very small values of y — although to a lesser extent since the set of representable floating point numbers is much more dense near zero (Goldberg 1991). Since any accuracy improvement in the right tail of the distribution should not be canceled by a deterioration in the left tail (near μ), our implementation of `pfpareto` actually computes the CDF with (2) when $u = y/(1 + y) \leq 0.5$, and with (16) otherwise.

3.2. Generation of random variates

The other main issue we faced in our R implementation was with the `rfpareto` function to generate random variates from the Feller-Pareto distribution. Section 2.4 outlined the three simulation algorithms at our disposal.

When comparing simulation algorithms, we are mostly interested in their relative speed and ability to reproduce the underlying distribution, especially for limit cases. We use functions `rgamma` and `rbeta` of base R to generate gamma and beta variates, respectively. Therefore, we leave aside considerations of randomness and efficiency since we are building on well

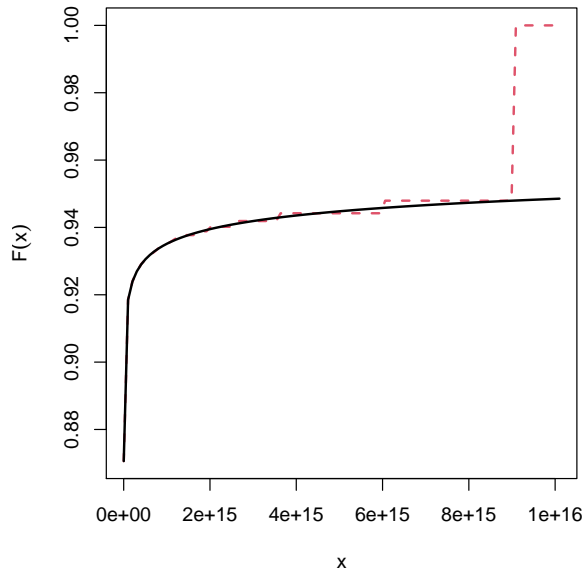


Figure 2: Cumulative distribution function of the Feller-Pareto distribution in the far right tail as computed with `pbeta` using (2) (red dashed line) and using (16) (black solid line).

Test	Replications	Relative time
<code>rbeta(n, shape1, shape2)</code>	100	1.000
<code>rgamma(2 * n, shape)</code>	100	1.312

Table 2: Relative time required to simulate $n = 10^6$ beta variates and $2n$ gamma variates (on average over the number of replications). Parameters `shape`, `shape1` and `shape2` uniformly distributed on $(0, 100)$.

established algorithms and routines to generate variates from these distributions (Cheng 1978; Ahrens and Dieter 1982).

One will first note that Algorithm 1 requires two gamma variates to generate a single Feller-Pareto variate, whereas Algorithms 2 and 3 require only one beta variate. This is an advantage for the beta-based algorithms, but not in a 2 : 1 ratio. Indeed, `rgamma` takes less time to generate a variate than `rbeta`. As Table 2 shows, generating *two* gamma variates is actually only about 30% slower than generating a *single* beta variate. These performance measurements carry over to the generation of Feller-Pareto variates, as shown in Table 3. (We obtained benchmarking times with the very convenient function `benchmark` of package `rbenchmark` (Kusnierczyk 2012).)

The other issue we investigated was performance of the algorithms in the main limit cases. First, when $\alpha \rightarrow 0$ and $\tau \rightarrow \infty$, variates from a $\mathcal{B}(\tau, \alpha)$ tend to 1. Due to the negative power in its second step, Algorithm 3 tends, in such a situation, to yield indeterminate values — “not a number”, or `NaN` in R terminology. Algorithm 2 does not exhibit this behavior. Figure 3 (left graphic) shows the effect on the empirical CDF of samples generated with Algorithms 2 and 3 for a small value of α and a large value of τ . Algorithm 2 is clearly preferable.

The indeterminate value issue of Algorithm 3 also arises in Algorithm 1 when $\alpha \rightarrow \tau \rightarrow 0$ and

Test	Replications	Relative time
Algorithm 3	100	1.000
Algorithm 2	100	1.011
Algorithm 1	100	1.307

Table 3: Relative time required to simulate 10^6 Feller-Pareto variates with three different algorithms (on average over the number of replications). Location parameter $\mu = 0$; scale parameter $\theta = 1$; shape parameters α and γ uniformly distributed on $(0, 100)$; shape parameter τ uniformly distributed on $(100, 1000)$.

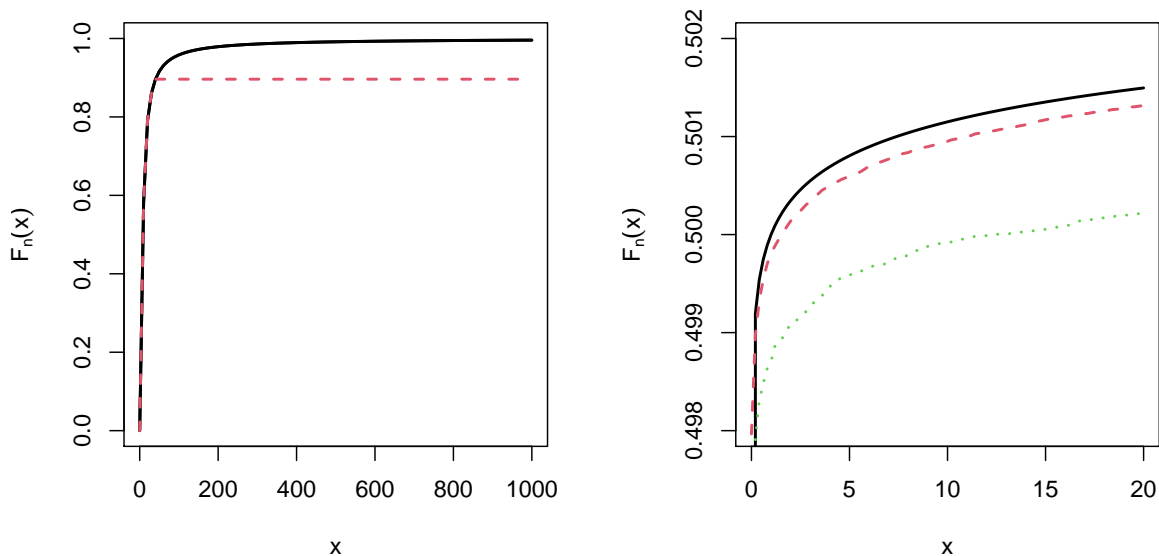


Figure 3: Comparison of the true CDF of the Feller-Pareto with empirical CDFs using three different simulation algorithms. Left: true CDF and Algorithm 2 (black solid line) *vs* Algorithm 3 (red dashed line) for $\alpha \rightarrow 0$ and $\tau \rightarrow \infty$. Right: true CDF (black solid line) *vs* Algorithm 2 (red dashed line) *vs* Algorithm 1 (green dotted line) for $\alpha \rightarrow \tau \rightarrow 0$. In both graphics $\mu = 0$, $\gamma = 10$ and $\theta = 1$.

both gamma variates may be numerically equal to 0. Figure 3 (right graphic) compares the theoretical CDF with typical empirical CDF of samples generated with Algorithms 1 and 2 (notice on the y -axis how this graphic is zoomed in). In our various tests, the empirical CDF with Algorithm 1 was almost consistently farther from the true CDF than the one with Algorithm 2.

As a result of the above observations, we recommend Algorithm 2 to generate Feller-Pareto random variates. This is the the algorithm we implemented in **actuar**. Accordingly, we also changed the algorithm used in the simulation of transformed beta variates from Algorithm 3 to Algorithm 2.

3.3. Moments

Numerical computation of raw moments does not pose significant difficulties. It is worth noting, though, that the function `mfpareto` computes the ratio of gamma functions in (6)

and (7) as a ratio of beta functions since

$$\begin{aligned} \frac{\Gamma(\tau + k/\gamma)\Gamma(\alpha - k/\gamma)}{\Gamma(\alpha)\Gamma(\tau)} &= \frac{\Gamma(\tau + k/\gamma)\Gamma(\alpha - k/\gamma)}{\Gamma(\alpha + \tau)} \frac{\Gamma(\alpha + \tau)}{\Gamma(\alpha)\Gamma(\tau)} \\ &= \frac{\beta(\tau + k/\gamma, \alpha - k/\gamma)}{\beta(\alpha, \tau)}. \end{aligned}$$

This way, we take advantage of the optimizations and careful handling of limit cases of routine `beta` of the C API of R (R Core Team 2022b).

Furthermore, in order to reduce the number of operations, we actually compute the moment of order k in the case $\mu \neq 0$ with the following expression that is algebraically equivalent to (7):

$$m_k = \mu^k + \frac{\mu^k}{\beta(\alpha, \tau)} \sum_{j=1}^k \binom{k}{j} \left(\frac{\theta}{\mu}\right)^j \beta(\tau + j/\gamma, \alpha - j/\gamma).$$

3.4. Limited moments

Before we turn to the computation of limited moments of the Feller-Pareto family of distributions, we must first study an alternative definition of the (regularized) incomplete beta function

$$I(a, b; x) = \frac{1}{\beta(a, b)} \int_0^x t^{a-1} (1-t)^{b-1} dt, \quad a > 0, b > 0, 0 < x < 1, \quad (17)$$

that is valid for negative values of the parameters. Klugman *et al.* (2019) introduce the alternative definition to extend the range of admissible values for limited expected value functions.

As seen in Abramowitz and Stegun (1972, Section 6.6), we have the following relation for the integral on the right hand side of (17):

$$\int_0^x t^{a-1} (1-t)^{b-1} dt = \frac{x^a}{a} F(a, 1-b; a+1; x),$$

where

$$F(a, b; c; z) = \frac{\Gamma(c)}{\Gamma(a)\Gamma(b)} \sum_{k=0}^{\infty} \frac{\Gamma(a+k)\Gamma(b+k)}{\Gamma(c+k)} \frac{z^k}{k!},$$

is the Gauss hypergeometric series. With the above definition, the incomplete beta function also admits negative, non integral values for parameters a and b .

Now, let

$$B(a, b; x) = \Gamma(a+b) \int_0^x t^{a-1} (1-t)^{b-1} dt, \quad (18)$$

for $a > 0$, $b \neq -1, -2, \dots$ and $0 < x < 1$. Clearly, when $b > 0$

$$B(a, b; x) = \Gamma(a)\Gamma(b)I(a, b; x).$$

Of more interest here is the case where $b < 0$, $b \neq -1, -2, \dots$ and $a > 1 + \lfloor -b \rfloor$. Integration by parts of (18) yields

$$\begin{aligned}
 B(a, b; x) = & -\Gamma(a+b) \left[\frac{x^{a-1}(1-x)^b}{b} + \frac{(a-1)x^{a-2}(1-x)^{b+1}}{b(b+1)} \right. \\
 & + \dots + \frac{(a-1)\cdots(a-r)x^{a-r-1}(1-x)^{b+r}}{b(b+1)\cdots(b+r)} \left. \right] \\
 & + \frac{(a-1)\cdots(a-r-1)}{b(b+1)\cdots(b+r)} \Gamma(a-r-1) \\
 & \times \Gamma(b+r+1) I(a-r-1, b+r+1; x),
 \end{aligned} \tag{19}$$

where $r = \lfloor -b \rfloor$. Following the terminology of **actuar**, we will call (18) the *beta integral*. (It is also known in the literature as the normalized incomplete beta function.) The package includes a C routine **betaint** — and a non exported R interface of the same name — that is a rather straightforward C implementation of (19). Its sole usage in the package is to evaluate the limited moments of distributions of the Feller-Pareto family.

Given the above, we rewrite the general expression (9) of the limited moment of order k as

$$\begin{aligned}
 \mathbb{E}[(X \wedge x)^k] = & \sum_{j=0}^k \binom{k}{j} \frac{\mu^{k-j} \theta^j B(\tau + j/\gamma, \alpha - j/\gamma; u)}{\Gamma(\alpha)\Gamma(\tau)} \\
 & + x^k (1 - I(\tau, \alpha; u))
 \end{aligned}$$

or, equivalently,

$$\begin{aligned}
 \mathbb{E}[(X \wedge x)^k] = & \frac{\mu^k B(\tau, \alpha; u)}{\Gamma(\alpha)\Gamma(\tau)} \\
 & + \frac{\mu^k}{\Gamma(\alpha)\Gamma(\tau)} \sum_{j=1}^k \binom{k}{j} \left(\frac{\theta}{\mu}\right)^j B(\tau + j/\gamma, \alpha - j/\gamma; u) \\
 & + x^k (1 - I(\tau, \alpha; u)).
 \end{aligned}$$

Note that for the expression above to be valid, $\alpha - j/\gamma$, $j = 0, 1, \dots, k$ must not be a negative integer. This is the expression that we implemented in the C routine **levfpareto**.

For the case $\mu = 0$ — distributions of the transformed beta family — we use (8) rewritten as

$$\mathbb{E}[(X \wedge x)^k] = \frac{\theta^k B(\tau + k/\gamma, \alpha - k/\gamma; u)}{\Gamma(\alpha)\Gamma(\tau)} + x^k (1 - I(\tau, \alpha; u)),$$

still with $\alpha - k/\gamma$ not a negative integer.

It is worth noting that the C routines behind limited moments functions carry a lot of computations in the log scale to delay numerical overflow and, therefore, to gain accuracy in the tails. For example, **levfpareto** and related routines compute

$$u = \frac{y}{1+y} = \frac{1}{1+y^{-1}}, \quad y = \left(\frac{x-\mu}{\theta}\right)^\gamma$$

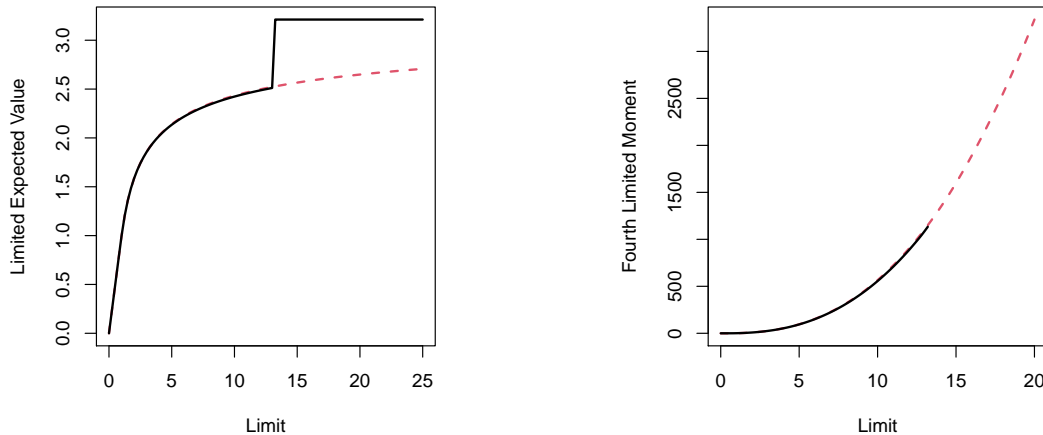


Figure 4: True limited moments as computed with `levtrbeta` in versions of **actuar** prior to 3.0-0 (black solid line) compared to their empirical counterparts (red dashed line). Left: first moment involving only positive values of b in the beta integral. Right: fourth moment involving negative values of b .

as $u = e^{\log u}$, with

$$\begin{aligned}\log u &= -\log(1 + e^{-\log y}) \\ \log y &= \gamma(\log(x - \mu) - \log \theta).\end{aligned}$$

The routines make use of function `log1pexp` of the C API of R to compute $\log(1 + e^x)$ accurately, notably for large x .

Upon analyzing the accuracy of the `levtrbeta` function in a previous version of **actuar**, we discovered discrepancies in the tail of the limited expected value function which in practice may lead to erroneous premium computation; see Section 4.2. Figure 4 shows the problem whether the beta integral (18) is called with a positive (left graphic) or negative (right graphic) value of its second parameter b . The cause is exactly the same as in Section 3.1: u gets numerically equal to 1 when y gets large. The cure is also the same: compute any instance of $I(a, b; u)$ as $1 - I(b, a; 1 - u)$ using $1 - u = (1 + y)^{-1}$ when $u > 0.5$. This also led us to add an argument to the `betaint` routine in order to pass the numerically accurate value of $1 - x$ to the incomplete beta function in (19) and, simultaneously, to all instances of this term in the sum.

The limited moments functions for all distributions of the Feller-Pareto family and the beta integral function incorporate the above improvements in version 3.0-0 of **actuar**. Figure 5 shows that their behavior in the tail is now consistent with the theoretical expressions.

4. Case study

We illustrate in this section how the Feller-Pareto distribution — and the functions we provide in **actuar** — can prove useful to model insurance loss costs, evaluate premiums and simulate data.

To this end, we use a dataset well known in actuarial circles as the *Danish dataset*. Collected by Copenhagen Reinsurance, it consists of aggregate claim amounts for 2,492 fire losses over

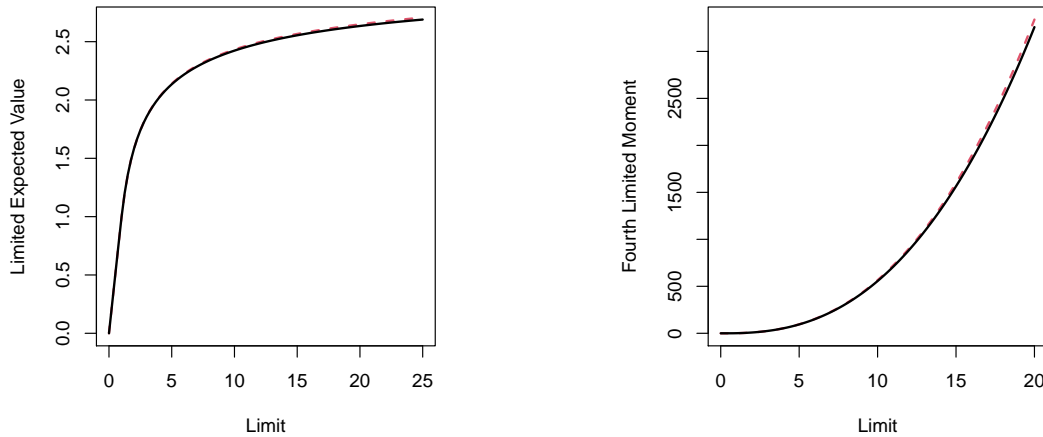


Figure 5: True limited moments as computed with `levtrbeta` in version 3.0-0 of **actuar** (black solid line) compared to their empirical counterparts (red dashed line).

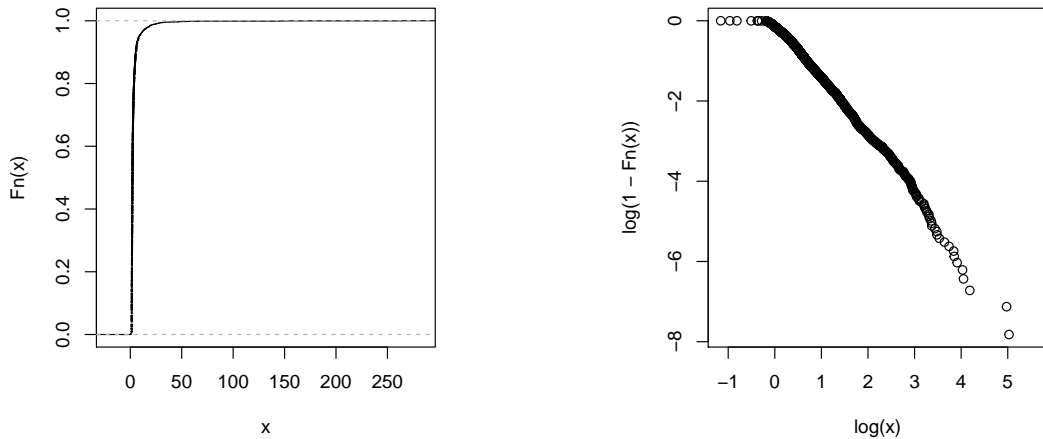


Figure 6: Empirical cumulative distribution function (left) and empirical survival function in log scale (right) for the Danish dataset.

the period 1980 to 1990, adjusted for inflation to reflect 1985 values. The values are expressed in millions of Danish Krone (DKK). The dataset has been studied by, e.g., [Embrechts *et al.* \(1997\)](#); [McNeil \(1997\)](#); [Resnick \(1997\)](#); [Davison \(2003\)](#). The R package **SMPracticals** ([Davison 2019](#)) provides the Danish dataset in the time series object `danish`. Since the dates of the events are irrelevant in our analysis, we use the data as a simple atomic vector.

```
R> data("danish", package = "SMPracticals")
R> danish.loss <- as.numeric(danish)
```

As Figure 6 shows, the Danish loss amounts exhibit a heavy, Pareto type, tail. [McNeil \(1997\)](#); [Resnick \(1997\)](#) propose alternative depictions of this behavior using a QQ-plot against the exponential distribution, a mean excess plot and a Hill plot. Summary statistics and right tail quantiles are also quite revealing.

```
R> summary(danish.loss)
```


Min.	1st Qu.	Median	Mean	3rd Qu.	Max.
0.3134	1.1572	1.6339	3.0627	2.6455	263.2504

```
R> quantile(danish.loss, probs = c(0.9, 0.95, 0.99, 1))
```

90%	95%	99%	100%
5.080440	8.406298	24.613784	263.250366

(McNeil 1997) fitted three distributions on the loss amounts: a Pareto I, a shifted lognormal and a generalized Pareto, the latter being a Pareto II distribution with a different parametrization widely used in extreme value theory (see, e.g., Embrechts *et al.* 1997). None of these distributions catch the steep curvature of the cumulative distribution function, although (McNeil 1997) shows the generalized Pareto successfully catches the tail behavior for losses over DKK 20,000,000.

4.1. Loss modeling

The first step in our case study will be to fit the Feller-Pareto distribution to the data using maximum likelihood estimation (MLE). Our main R tool will be function `fitdistr` from package `fitdistrplus` (Delignette-Muller and Dutang 2015), which wraps and extends the well known function `fitdistr` from package `MASS` (Venables and Ripley 2002).

Since the Danish dataset contains positive loss amounts, we have a natural value of $\mu = 0$ for the location parameter of the Feller-Pareto. We could try to estimate this parameter along with the shape parameters and the scale parameter, but this proves numerically cumbersome. Brazauskas (2002) provides explicit formulas for the Fisher information matrix when the location parameter is set to 0. However, for the current example, we simply rely on the numerical values provided by the R optimization routines.

We use the bounded BFGS optimization algorithm to restrict the parameters to positive values (Bonnans, Gilbert, Lemaréchal, and Sagastizábal 2006). For numerical reasons, we actually restrict the shape and scale parameters to $[\epsilon, \infty)$, where $\epsilon = 2^{-52}$ is the machine epsilon in double precision.

```
R> (fit.fp <- fitdistr(danish.loss, "fpareto", optim.method = "L-BFGS-B",
+   fix.arg = list(min = 0), start = list(shape1 = 1, shape2 = 1,
+   shape3 = 1, scale = 1), lower = .Machine$double.eps))
```

```
Fitting of the distribution ' fpareto ' by maximum likelihood
Parameters:
```

	estimate	Std. Error
shape1	0.07304845	0.01880457
shape2	17.75618267	4.31435803
shape3	0.80165273	0.22356892
scale	0.93134995	0.01474232

```
Fixed parameters:
```

	value
min	0

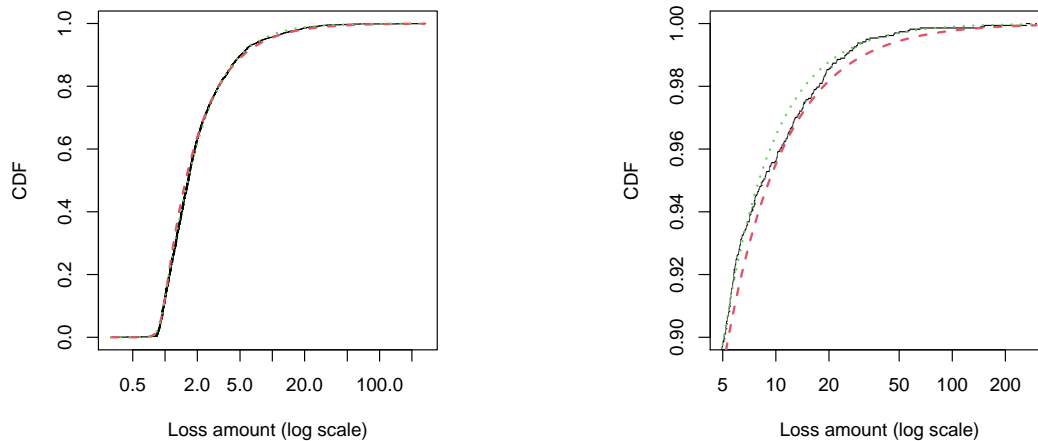


Figure 7: Comparison of CDF: empirical (solid black line), fitted Feller-Pareto (red dashed line), fitted Weibull – Inverse Weibull composite (green dotted line). Left: complete range of the distributions. Right: upper tails.

The graphics of Figure 7 compare the empirical and fitted CDFs. The fit of the Feller-Pareto is good through values of around DKK 20,000,000. Passed this point, the Feller-Pareto provides a heavier tail than the actual data.

In addition to the analysis of the goodness-of-fit plots, we can also check statistical measures or criteria (Klugman *et al.* 2019, Chapter 15) as provided by function `gofstat` of **fitdistrplus**.

```
R> gofstat(fit.fp)
```

```
Goodness-of-fit statistics
                                1-mle-fpareto
Kolmogorov-Smirnov statistic      0.04202179
Cramer-von Mises statistic        0.84124463
Anderson-Darling statistic       4.17626761

Goodness-of-fit criteria
                                1-mle-fpareto
Akaike's Information Criterion    7677.535
Bayesian Information Criterion    7700.818
```

Our intent here is merely to illustrate how the numerical tools we provide may prove useful to fit Feller-Pareto distributions to loss data. Still, the above statistics show that the model fares very well against the Weibull–Inverse Weibull mixture determined as best among 256 composite models by Grün and Miljkovic (2019) for the same dataset.

4.2. Expected losses

Our Feller-Pareto model has $\alpha\gamma = 1.297$. By (6), its expected value exists. We may therefore easily compute the expected losses (in million DKK) for the Danish dataset using function `mfpareto`.

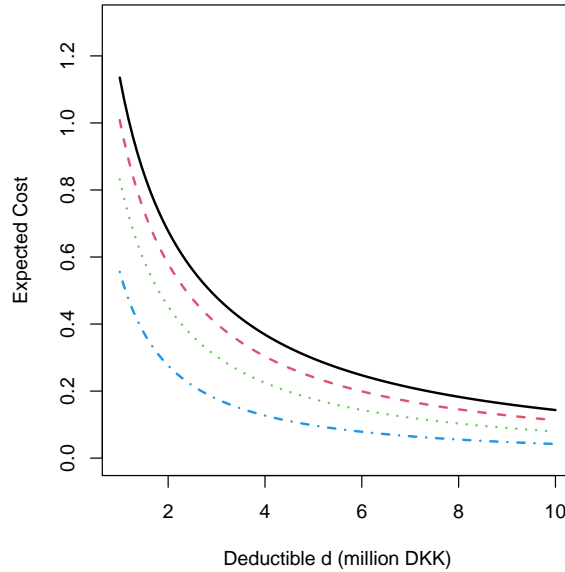


Figure 8: Expected loss cost of an excess-of-loss reinsurance contract as a function of the deductible for different cover values: $c = 4$ (solid black line), $c = 3$ (red dashed line), $c = 2$ (green dotted line), $c = 1$ (blue dot-dashed line).

```
R> par.fp <- fit.fp$estimate
R> mfpareto(1, min = 0, shape1 = par.fp["shape1"], shape2 = par.fp["shape2"],
+   shape3 = par.fp["shape3"], scale = par.fp["scale"])
```

```
[1] 3.977563
```

After [McNeil \(1997\)](#), we also compute the expected losses for an excess-of-loss reinsurance contract where the reinsurer is responsible for losses between d and $d + c$, where d is the deductible value and c , the cover. Let X be the original loss amount and X_{re} , the reinsurance cost. We have $X_{\text{re}} = \min(\max(X - d, 0), c) = \min(X, d + c) - \min(X, d)$.

The expected loss cost for the reinsurer is $\mathbf{E}[X_{\text{re}}] = \mathbf{E}[\min(X, d + c)] - \mathbf{E}[\min(X, d)]$. We used function `levfpareto` to compute these expected values and to draw [Figure 8](#), which displays the expected cost $\mathbf{E}[X_{\text{re}}]$ as a function of the deductible d for different values of covers c . One can see the drastic effect of the deductible and the cover on expected costs.

4.3. Simulation of aggregate claim amounts

For pricing and risk evaluation purposes, actuaries usually need to take into account not only the severity of losses, but also their frequency. Let random variable S represent the aggregate claim amount (or total amount of claims) of a portfolio of independent risks over a fixed period of time, random variable N represent the number of claims (or frequency) in the portfolio over that period, and random variable X_j represent the amount of claim j (or severity). Then, we have the random sum

$$S = X_1 + \cdots + X_N, \quad (20)$$

where we assume that X_1, X_2, \dots are mutually independent and identically distributed random variables each independent of N . When N has a Poisson distribution, we say that the distribution of S is a compound Poisson, see, e.g., [Klugman *et al.* \(2019\)](#).

But for a few simple cases, finding the distribution of the random variable S is a difficult task. However, one may turn to simulation to get good estimates of quantities related to this distribution.

Let us simulate a large sample from a compound Poisson distribution where the severity of losses is distributed according to our Feller-Pareto. This is straightforward using function `rcomppois` from package **actuar**. (See [Goulet and Pouliot \(2008\)](#) for an explanation of the syntax to specify simulation models.)

```
R> x <- rcomppois(1e6, lambda = 10, rfpareto(min = 0,
+   shape1 = par.fp["shape1"], shape2 = par.fp["shape2"],
+   shape3 = par.fp["shape3"], scale = par.fp["scale"]))
```

We may now, for example, get fairly good approximations of the upper tail quantiles of the distribution of S . Since the sample size of `x` is one million, we can reasonably assess quantiles up to probability level $p = 0.999999 = 1 - 10^{-6}$ based on the non-parametric estimator provided by base R function `quantile`. Otherwise, we will need either to have a higher number of simulations or to use the extreme value theory, e.g., [McNeil \(1997\)](#).

```
R> quantile(x, c(0.75, 0.9, 0.95, 0.99))
```

```
      75%      90%      95%      99%
37.52340 58.95447 84.15556 221.73017
```

5. Accessing the C routines

The actual workhorses behind the R functions of **actuar** presented in this paper are C routines. Package developers who need one or many of these routines in their C code can take advantage of them through the new package API of **actuar**. The header file `inst/include/actuarAPI.h` in the package installation directory contains the declarations of the more than 200 routines part of the API.

We direct the interested reader to the “distributions” package vignette for further instructions to access the C routines through the API. The companion package **expint** ([Goulet 2022](#)) ships with a complete test package implementing these instructions. See the vignette of the latter package for more information.

6. Conclusion

This paper introduced the implementation of the Feller-Pareto distribution and related Pareto distributions with a location parameter in version 3.0-0 of the R package **actuar**. The Feller-Pareto defines a large family of distributions encompassing the transformed beta family and many variants of the Pareto distribution.

A large proportion of the numerical challenges discussed in the paper are related to the fact that, for most distributions of the Feller-Pareto family, we need to evaluate the incomplete beta function for a value that can jump to 1 if not carefully computed to avoid numerical overflows. By applying a simple correction to many functions present in previous versions of **actuar**, we were able to fix long standing discrepancies in CDF and limited moment functions, as well as to provide robust new functions for the Feller-Pareto distribution. We also improved the simulation algorithm for all distributions of the Feller-Pareto family.

A case study based on a famous heavy tailed insurance dataset showed how one can use the various support functions for the Feller-Pareto distribution in modeling, graphics or simulation.

Acknowledgments

The authors wish to thank the editors of the journal as well as two anonymous referees for suggestions that led to substantive improvements to the paper.

References

- Abramowitz M, Stegun IA (1972). *Handbook of Mathematical Functions*. Dover.
- Ahrens JH, Dieter U (1982). “Generating Gamma Variates by a Modified Rejection Technique.” *Communications of the ACM*, **25**(1), 47–54. doi:10.1145/358315.358390.
- Arnold BC (2015). *Pareto Distributions*. 2nd edition. CRC Press.
- Atkinson AB, Bourguignon F (eds.) (2015). *Handbook of Income Distribution*, volume 2. Elsevier.
- Bonnans JF, Gilbert JC, Lemaréchal C, Sagastizábal CA (2006). *Numerical Optimization: Theoretical and Practical Aspects*. 2nd edition. Springer-Verlag. doi:10.1007/978-3-662-05078-1.
- Brazauskas V (2002). “Fisher Information Matrix for the Feller–Pareto Distribution.” *Statistics and Probability Letters*, **59**(2), 159–167. doi:10.1016/s0167-7152(02)00143-8.
- Cheng RCH (1978). “Generating Beta Variates With Nonintegral Shape Parameters.” *Communications of the ACM*, **21**(4), 317–322. doi:10.1145/359460.359482.
- Cummins JD, Dionne G, McDonald JB, Pritchett BM (1990). “Applications of the GB2 Family of Distributions in Modeling Insurance Loss Processes.” *Insurance: Mathematics and Economics*, **9**(4), 257–272. doi:10.1016/0167-6687(90)90003-v.
- Davison A (2019). **SMPracticals: Practicals for Use with Davison (2003) Statistical Models**. R package version 1.4-3, URL <https://CRAN.R-project.org/package=SMPracticals>.
- Davison AC (2003). *Statistical Models*, volume 11. Cambridge University Press.
- Delignette-Muller ML, Dutang C (2015). “**fitdistrplus**: An R Package for Fitting Distributions.” *Journal of Statistical Software*, **64**(4), 1–34. doi:10.18637/jss.v064.i04.

- Dutang C, Goulet V, Pigeon M (2008). “**actuar**: An R Package for Actuarial Science.” *Journal of Statistical Software*, **25**(7), 1–37. doi:10.18637/jss.v025.i07.
- Dutang C, Kiener P (2022). “CRAN Task View: Probability Distributions.” Version 2022-03-07, URL <https://CRAN.R-project.org/view=Distributions>.
- Embrechts P, Klüppelberg C, Mikosch T (1997). *Modelling Extremal Events for Insurance and Finance*. Springer-Verlag. doi:10.1007/978-3-642-33483-2.
- Feller W (1968). *An Introduction to Probability Theory and Its Applications*. 3rd edition. John Wiley & Sons.
- Goldberg D (1991). “What Every Computer Scientist Should Know About Floating-Point Arithmetic.” *ACM Computing Surveys*, **23**(1), 5–48. doi:10.1145/103162.103163.
- Goulet V (2022). **expint**: *Exponential Integral and Incomplete Gamma Function*. R package version 0.1-7, URL <https://CRAN.R-project.org/package=expint>.
- Goulet V, Pouliot LP (2008). “Simulation of Compound Hierarchical Models in R.” *North American Actuarial Journal*, **12**(4), 401–412. doi:10.1080/10920277.2008.10597532.
- Grün B, Miljkovic T (2019). “Extending Composite Loss Models Using a General Framework of Advanced Computational Tools.” *Scandinavian Actuarial Journal*, **8**, 642–660. doi:10.1080/03461238.2019.1596151.
- Hogg RV, Klugman SA (1984). *Loss Distributions*. John Wiley & Sons, New York.
- Johnson N, Kotz S, Balakrishnan N (1994a). *Continuous Univariate Distributions*, volume 2. 2nd edition. John Wiley & Sons.
- Johnson N, Kotz S, Balakrishnan N (1994b). *Continuous Univariate Distributions*, volume 1. 2nd edition. John Wiley & Sons.
- Kleiber C, Kotz S (2003). *Statistical Size Distributions in Economics and Actuarial Sciences*. John Wiley & Sons.
- Klugman SA, Panjer HH, Willmot GE (2019). *Loss Models: From Data to Decisions*. 5th edition. John Wiley & Sons.
- Kusnierczyk W (2012). **rbenchmark**: *Benchmarking Routine for R*. R package version 1.0.0, URL <https://CRAN.R-project.org/package=rbenchmark>.
- Leemis LM, McQueston JT (2008). “Univariate Distribution Relationships.” *The American Statistician*, **62**(1), 45–53. doi:10.1198/000313008x270448.
- McDonald JB (1984). “Some Generalized Functions for the Size Distribution of Income.” *Econometrica*, **52**(2), 647–664. doi:10.2307/1913469.
- McNeil AJ (1997). “Estimating the Tails of Loss Severity Distributions Using Extreme Value Theory.” *ASTIN Bulletin: The Journal of the IAA*, **27**(1), 117–137. doi:10.2143/ast.27.1.563210.

- McNeil AJ, Frey R, Embrechts P (2015). *Quantitative Risk Management: Concepts, Techniques and Tools*. Revised edition. Princeton University Press.
- R Core Team (2022a). *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria. URL <https://www.R-project.org/>.
- R Core Team (2022b). *Writing R Extensions*. R Foundation for Statistical Computing, Vienna, Austria. URL <https://CRAN.R-project.org/doc/manuals/R-exts.html>.
- Resnick SI (1997). “Discussion of the Danish Data on Large Fire Insurance Losses.” *ASTIN Bulletin: The Journal of the IAA*, **27**(1), 139–151. doi:10.2143/ast.27.1.563211.
- Venables WN, Ripley BD (2002). *Modern Applied Statistics with S*. 4th edition. Springer-Verlag. doi:10.1007/978-0-387-21706-2.
- Venter G (1983). “Transformed Beta and Gamma Distributions and Aggregate Losses.” *Proceedings of the Casualty Actuarial Society*, **70**, 156–193. doi:10.1080/03461238.1970.10405663.

Affiliation:

Christophe Dutang
CEREMADE, CNRS
Université Paris-Dauphine
Université Paris Sciences et Lettres (PSL)
Place du Maréchal de Lattre de Tassigny
75775 - PARIS CEDEX 16, France
E-mail: dutang@ceremade.dauphine.fr
URL: <http://dutangc.free.fr/>