Parameter Estimation of a Mixture Distribution

Let $X_1, \dots, X_n$ be a stationary time series with extremal index $\theta \in (0,1]$ (see Beirlant et al, 2004) and $W_1, \dots, W_n$ iid, heavy-tailed waiting times. We can show that the limiting distribution of the excess waiting times with threshold u_n ($u_n \to x_R$) is a mixture distribution out of the dirac measure at point zero and the Mittag-Leffler distribution $ML(\beta, \theta^{-1/\beta})$, $\beta \in (0,1]$, with weights $(1-\theta, \theta)$:

$$\mathbb{P}_{\beta,\theta} := (1-\theta)\delta_0 + \theta \cdot ML(\beta, \theta^{-1/\beta}). \tag{1}$$

We want to estimate the two unknown parameters β and θ with the method of moments. But since the ML-distribution does not have finite moments we have a look at the fractional moments. Let $q \in (0,1)$. The q-th fractional moment of a $ML(\beta, \gamma)$-distributed random variable Y is:

$$\mathbb{E}(Y^q) = \frac{q \cdot \pi \cdot \gamma^q}{\beta \cdot \Gamma(1-q)\sin(\frac{q\cdot\pi}{\beta})}, \qquad \text{for } 0 < q < \beta \tag{2}$$

(Cahoy, 2013). Then the q-th fractional moment of $T \sim \mathbb{P}_{\beta,\theta}$ is

$$\begin{aligned} m_q := m_q(T) :&= \mathbb{E}(T^q) = (1-\theta)\mathbb{E}(Y^q) + \theta\mathbb{E}(X^q) = \theta\mathbb{E}(X^q) \\ &= \theta^{\frac{\beta-q}{\beta}} \cdot \frac{q \cdot \pi}{\beta \cdot \Gamma(1-q)\sin(\frac{q\cdot\pi}{\beta})} \end{aligned} \tag{3}$$

with $Y \sim \delta_0$ and $X \sim ML(\beta, \theta^{-1/\beta})$, $q < \beta$. The empirical q-th fractional moment $\hat{m}_q = \frac{1}{n}\sum_{i=1}^n T_i^q$, $T_1, \dots, T_n \overset{iid}{\sim} \mathbb{P}_{\beta,\theta}$, is an unbiased estimator of m_q and for $2q < \beta$ the estimator $\hat{m}_q$ is consistent in mean square:

$$\begin{aligned} \mathrm{Var}(\hat{m}_q) &= \mathbb{E}(\hat{m}_q^2) - \mathbb{E}(\hat{m}_q)^2 \\ &= \frac{1}{n^2}\mathbb{E}\left(\sum_{i\neq j} T_i^q T_j^q + \sum_{i=1}^n T_i^{2q}\right) - m_q^2 \\ &= \frac{1}{n^2}\left(n(n-1)\mathbb{E}(T_1^q)^2 + n\mathbb{E}(T_1^{2q})\right) - m_q^2 \\ &= \frac{n-1}{n}m_q^2 + \frac{1}{n}m_{2q} - m_q^2 \overset{n\to\infty}{\longrightarrow} 0. \end{aligned} \tag{4}$$

Since $\lim\limits_{x\to 0} \frac{\sin(x\pi)}{x\pi} = 1$ (easily shown by using L'Hopital's rule) and $\Gamma(1) = 1$ it follows

$$\lim_{q\to 0} m_q = \theta. \tag{5}$$

By solving (3) for θ and using the empirical fractional moment $\hat{m}_q$, we get an estimator $\hat{\theta}_q(\beta)$ depending on β:

$$\hat{\theta}_q(\beta) = \left[\frac{\beta \cdot \Gamma(1-q)\sin(\frac{q\cdot\pi}{\beta})}{q \cdot \pi} \cdot \widehat{m}_q\right]^{\frac{\beta}{\beta-q}} \tag{6}$$

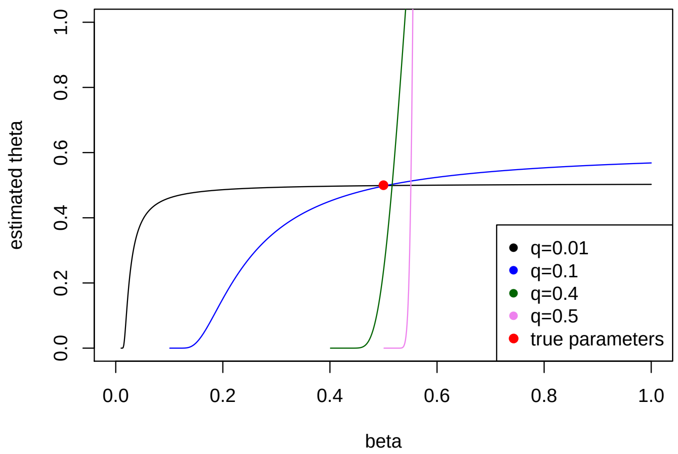

Figure 1: Estimation $\hat{\theta}_q(\beta)$ depending on β for various fractions q. It is based on $n = 10000$ $\mathbb{P}_{0.5,0.5}$-distributed data.

Therefore, we have to estimate β first by calculating the root of $\hat{\theta}_{q_1}(\beta) - \hat{\theta}_{q_2}(\beta)$ on $(\max(q_1, q_2), 1]$ (we choose $\max(q_1, q_2)$ as the lower interval limit because $\beta > q_1, q_2$ has to be fulfilled, otherwise the fractional moments don't exist). It is important to choose the fractions q_1 and q_2 small enough (smaller than the (unknown) β) otherwise we can't find β by calculating the root of $\hat{\theta}_{q_1}(\beta) - \hat{\theta}_{q_2}(\beta)$ because $\beta \notin (\max(q_1, q_2), 1]$ for $\beta \leqslant q_1, q_2$. $\hat{\theta}_q(\beta)$ is not unbiased but for smaller q the bias gets smaller because of (5).

Figure 1 shows $\hat{\theta}_q(\beta)$ depending on β for various fractions q. We can see that the curves of lower fractions q intersect the true parameter while the curve of $p = 0.5 = \beta$ doesn't. As well we can see that for small fraction q the estimator $\hat{\theta}_q(\beta)$ is less dependend of β.
Now we look how it works with `R`:

```
set.seed(2345)
## chosen parameters:
# 1) true parameters
beta <- 0.5
theta <- 0.5
# 2) random sample
n <- 10000                                             # sample size
# 3) fractions
q1 <- 0.01
q2 <- 0.05
## sample:
prob <- sample( c(0,1) , size = n , replace = T , prob = c(1-theta , theta))
```

```r
# we need the R-package 'MittagLeffleR' to sample ML-distributed values:
# install.packages("MittagLeffleR")
daten <- sapply(prob ,                          # sample
                function(x){ ifelse( x == 0 , return(0) ,
                        return( MittagLeffleR::rml(1 , tail = beta ,
                        scale = theta^(-1/beta))
                        ) ) } )
## functions:
fct_empfracmom <- function( q , data ){       # emp. frac. moment
                r <- 1/(length(data))*sum(data^q)
                return(r)       }
fct_theta <- function( beta , q , data ){     # estimator for theta
                r <- ( ( beta*gamma(1-q)*sin(pi*q/beta) )/( q*pi )*
                fct_empfracmom(q , data) )^(beta/(beta-q))
                return(r)               }
fct_root <- function( beta , q1 , q2 , data){
                r <- fct_theta( beta , q1 , data) - fct_theta( beta , q2 , data)
                return(r)                  }
fct_beta <- function(q1 , q2 , data){         # estimator for beta
                r <- uniroot(fct_root , c(max(q1,q2)+0.001,1) , q1 = q1 ,
                q2 = q2 , data = data)
                return(r$root)      }
## estimating beta and theta
# estimating beta by calculating the root:
beta_hat1 <- fct_beta(q1=q1 , q2=q2 , data = daten); beta_hat1

## [1] 0.4992572

# estimating theta by using beta_hat1
theta_hat11 <- fct_theta(beta = beta_hat1 , q=q1 , data = daten); theta_hat11

## [1] 0.5020829

theta_hat12 <- fct_theta(beta = beta_hat1 , q=q2 , data = daten); theta_hat12

## [1] 0.5020837
```

```
# error:
abs(beta-beta_hat1); abs(theta-theta_hat11); abs(theta-theta_hat12)

## [1] 0.0007427727
## [1] 0.00208289
## [1] 0.002083718

## -> it works

# estimating theta by calculating the q-th emp. frac. moment with q very small:
theta_hat2 <- fct_empfracmom(10^{-6} , data=daten); theta_hat2

## [1] 0.5021004

beta_hat2 <- uniroot(function(x){fct_theta(x , q=q1 , data=daten)-theta_hat2} ,
interval = c(q1,1))$root; beta_hat2

## [1] 0.5003904

abs(beta-beta_hat2); abs(theta-theta_hat2)

## [1] 0.0003903841
## [1] 0.002100401

## -> works as well;
## -> unsolved problem: which fraction q is small enough
##    to estimate theta reliably?
```