Modeling extreme values with a GEV mixture probability distributions

Pascal Alain Dkengne Sielenou

2023-09-28

```
# library(xfun)
```

```
path <- ".."
```

```
xfun::in_dir(dir = path, expr = source("./src/generate_gev_sample.R"))
xfun::in_dir(dir = path, expr = source("./src/calculate_gev_inverse_cdf.R"))
xfun::in_dir(dir = path, expr = source("./src/estimate_gev_mixture_model_parameters.R"))
xfun::in_dir(dir = path, expr = source("./src/plot_gev_mixture_model_pdf.R"))
xfun::in_dir(dir = path, expr = source("./src/plot_several_standardized_block_maxima_mean.R"))
xfun::in_dir(dir = path, expr = source("./src/estimate_gev_mixture_model_quantile.R"))
```

```
library(readr)
```

```
Gnss_imar <- xfun::in_dir(dir = path, expr = read_csv("./applications/Gnss_imar.csv"))
```

```
## Rows: 20002 Columns: 25
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## dbl (25): version_major, version_minor, status, timestamp, latitude, longitu...
##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

```
Gnss_map_matching <- xfun::in_dir(dir = path, expr = read_csv("./applications/Gnss_map_matching.csv"))
```

```
## Rows: 20001 Columns: 25
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## dbl (25): version_major, version_minor, status, timestamp, latitude, longitu...
##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

#’ timestamp_position <- sapply(Gnss_map_matching$timestamp, function(ts) which.min(abs(ts - Gnss_imar$timestamp)))

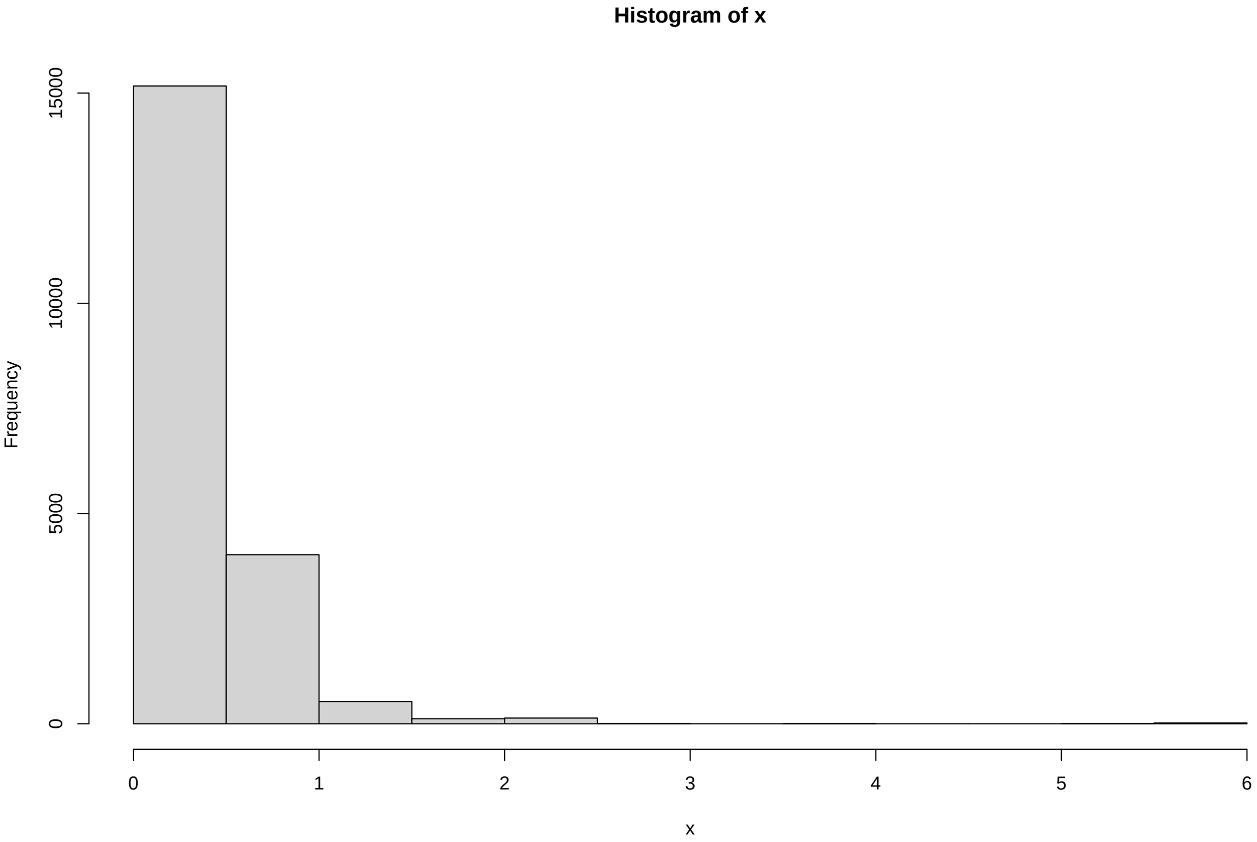
```
latitude_Gnss_map_matching_errors <- Gnss_imar$latitude[-1] - Gnss_map_matching$latitude

# timestamp_diff <- Gnss_imar$timestamp[-1] - Gnss_map_matching$timestamp
#
# head(timestamp_diff)
#
# tail(timestamp_diff)
```

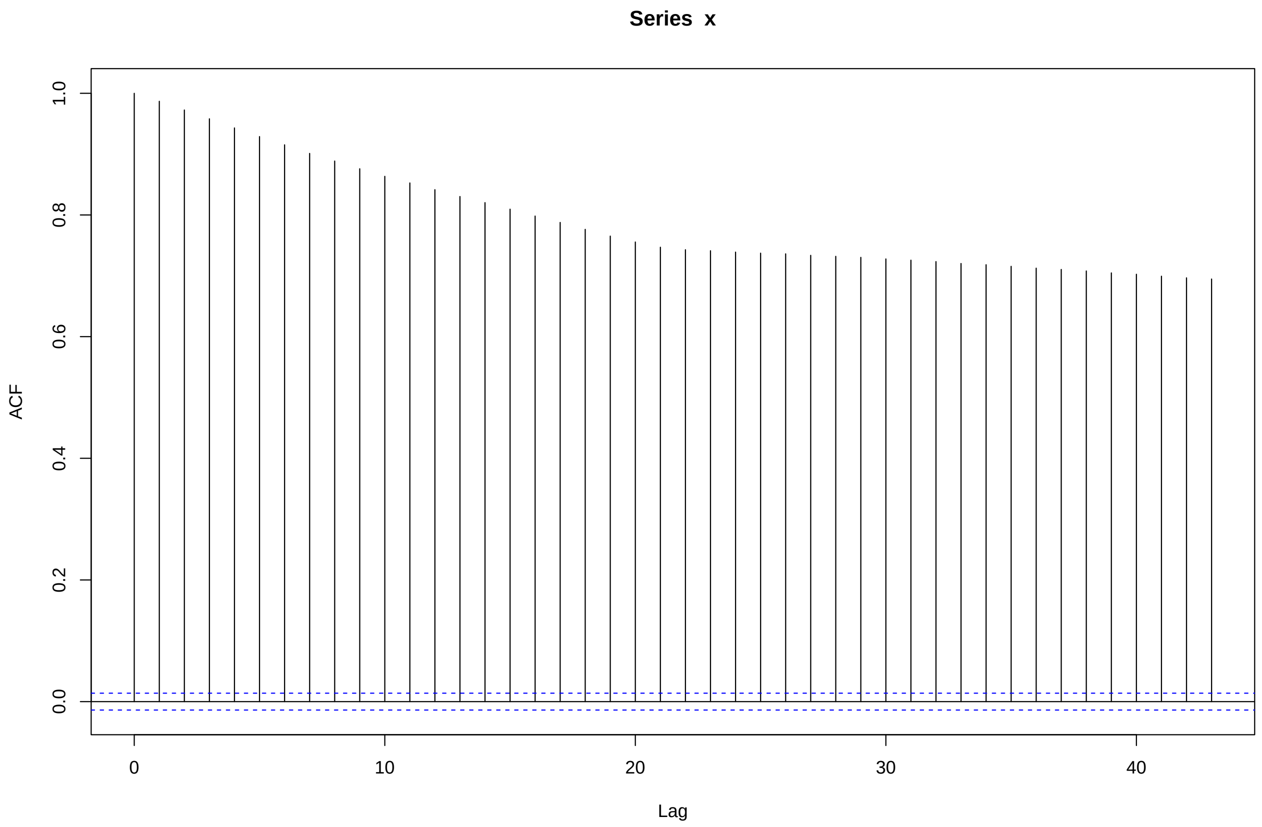
```
#
# range(timestamp_diff)
```

```
coefficient <- 10^(5)
x <- coefficient*abs(latitude_Gnss_map_matching_errors)
```

```
hist(x)
```



```
acf(x)
```



```
n <- length(x)
n
```

```
## [1] 20001
```

```
nlargest <- 2000

#
y <- extract_nlargest_sample(x, n = nlargest)
```

```
gev_mixture_model <- estimate_gev_mixture_model_parameters(x,
                                                            nsloc = NULL,
                                                            std.err = FALSE,
                                                            block_sizes = NULL,
                                                            minimum_nblocks = 50,
                                                            threshold = min(y),
                                                            nlargest = nlargest,
                                                            confidence_level = 0.95,
                                                            log_mv = TRUE,
                                                            log_pw = TRUE,
                                                            trace = FALSE)
```

```
##   Successful convergence.
##   Successful convergence.
```

```
names(gev_mixture_model)
```

```
##  [1] "data"
##  [2] "data_largest"
##  [3] "block_sizes"
##  [4] "equivalent_block_sizes"
##  [5] "rejected_block_sizes"
##  [6] "block_maxima_indexes_object"
##  [7] "gev_models_object"
##  [8] "extremal_indexes"
##  [9] "normalized_gev_parameters_object"
## [10] "weighted_normalized_gev_parameters_object"
## [11] "identic_weights_mw"
## [12] "pessimistic_weights_mw"
## [13] "pessimistic_weights_pw_shape"
## [14] "pessimistic_weights_pw_scale"
## [15] "pessimistic_weights_pw_loc"
## [16] "automatic_weights_mw"
## [17] "automatic_weights_mw_statistics"
## [18] "automatic_weights_pw_shape"
## [19] "automatic_weights_pw_scale"
## [20] "automatic_weights_pw_loc"
## [21] "automatic_weights_pw_statistics"
```

```
gev_mixture_model$block_sizes
```

```
##  [1]  2  3  4  5  6  7  8  9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26
## [26] 27 28 29 30 31 32 33 34 35 36 37 38 39 40
```

```
gev_mixture_model$normalized_gev_parameters_object
```

```
##             loc_star         scale_star        shape_star
## 2  0.805047838908850 0.0915708282410381 0.706330400471393
## 3  0.780500326355734 0.0725778684446514 0.689192757491651
## 4  0.764797925354116 0.0647494419217888 0.658381494613566
## 5  0.753084213092334 0.0563814657606010 0.668254799745940
## 6  0.745682872929703 0.0535027992671216 0.644372063934002
## 7  0.738314741742269 0.0501602923009216 0.646205529520483
## 8  0.733517318338475 0.0461054558676702 0.642437727014671
## 9  0.726117870708269 0.0468166844248344 0.617864058101191
## 10 0.720532953226680 0.0446934839667525 0.623252603291863
## 11 0.719910925286963 0.0399148174526985 0.648860335981693
## 12 0.704960369998868 0.0465001121157146 0.585209182597504
## 13 0.721172694429699 0.0357686001230337 0.662674478912404
## 14 0.703595686332608 0.0432065165668929 0.588667485677797
## 15 0.705206841650297 0.0380337514937296 0.621216871372568
## 16 0.699887167141014 0.0379541978487533 0.617504152506784
## 17 0.703468937167117 0.0361552379359654 0.617620067903273
## 18 0.692609686238964 0.0395351646809353 0.591624655621323
## 19 0.691491342736567 0.0387621016947383 0.592553535378669
## 20 0.695561773758290 0.0348545541831381 0.618563234399960
## 21 0.674620161723979 0.0422070730028164 0.562455139595575
## 22 0.676854014443636 0.0383184138896562 0.591057269440694
## 23 0.649881675133684 0.0465947891282851 0.534859636195468
## 24 0.684988751715025 0.0363618850876230 0.582051791214214
## 25 0.668487624812915 0.0388476761500032 0.591004317500299
## 26 0.685199943377586 0.0335233363718255 0.613380801960279
## 27 0.660025085255878 0.0381343454995609 0.576252649653111
## 28 0.673141077866813 0.0346638182229209 0.584703296970077
## 29 0.660666808270764 0.0361409237831552 0.592671535208803
## 30 0.650111910593117 0.0383678306804126 0.572520800313697
## 31 0.658462927439180 0.0379758629639002 0.563769848915829
## 32 0.681113551064684 0.0281429473220922 0.643235194437689
## 33 0.648338961689714 0.0352406540558787 0.611497819972599
## 34 0.653052432200312 0.0345653919848699 0.599784424711775
## 35 0.619848276867604 0.0451830036058980 0.533241890944053
## 36 0.655721353759653 0.0365173967482878 0.552722627780009
## 37 0.629151278256231 0.0403791916897468 0.535574610662510
## 38 0.623655439151820 0.0404549586836647 0.551468017209960
## 39 0.681258734302650 0.0249621259055186 0.676896276681649
## 40 0.660920138805951 0.0283366210174743 0.616243761195342
```

```
gev_mixture_model$weighted_normalized_gev_parameters_object
```

```
##                              loc_star         scale_star        shape_star
## identic_weights     0.692332349541744 0.0423631184637069 0.608363516541035
## pessimistic_weights 0.694113931667639 0.0425099670306353 0.610170904428398
## automatic_weights   0.805047838910980 0.0915708282409496 0.706330400471412
```

```
gev_mixture_model$automatic_weights_mw_statistics
```

```
## $function_value
## [1] 146.174987035113
##
## $gradient_value
## [1] 1.93267624126747e-12
##
## $function_reduction
## [1] 402.141624557218
##
## $number_iterations
## [1] 1
##
## $convergence
## [1] 0
##
## $message
## [1] "Successful convergence"
```

```
gev_mixture_model$automatic_weights_pw_statistics
```

```
## $function_value
## [1] 146.174987029032
##
## $gradient_value
## [1] 3.09228198602796e-11
##
## $function_reduction
## [1] 420.827064478688
##
## $number_iterations
## [1] 1
##
## $convergence
## [1] 0
##
```

```
## $message
## [1] "Successful convergence"
```

```
gev_mixture_model$automatic_weights_mw
```

```
##                    2                    3                    4
## 9.99999999998067e-01 0.00000000000000e+00 0.00000000000000e+00
##                    5                    6                    7
## 0.00000000000000e+00 0.00000000000000e+00 0.00000000000000e+00
##                    8                    9                   10
## 0.00000000000000e+00 0.00000000000000e+00 0.00000000000000e+00
##                   11                   12                   13
## 0.00000000000000e+00 0.00000000000000e+00 0.00000000000000e+00
##                   14                   15                   16
## 0.00000000000000e+00 0.00000000000000e+00 0.00000000000000e+00
##                   17                   18                   19
## 0.00000000000000e+00 0.00000000000000e+00 0.00000000000000e+00
##                   20                   21                   22
## 0.00000000000000e+00 0.00000000000000e+00 0.00000000000000e+00
##                   23                   24                   25
## 0.00000000000000e+00 0.00000000000000e+00 0.00000000000000e+00
##                   26                   27                   28
## 0.00000000000000e+00 0.00000000000000e+00 0.00000000000000e+00
##                   29                   30                   31
## 0.00000000000000e+00 0.00000000000000e+00 0.00000000000000e+00
##                   32                   33                   34
## 0.00000000000000e+00 0.00000000000000e+00 0.00000000000000e+00
##                   35                   36                   37
## 0.00000000000000e+00 0.00000000000000e+00 0.00000000000000e+00
##                   38                   39                   40
## 0.00000000000000e+00 0.00000000000000e+00 1.42108547152020e-14
```

```
gev_mixture_model$pessimistic_weights_pw_shape
```

```
##                  2                  3                  4                  5
## 0.0282546525351759 0.0277745599652025 0.0269318399975194 0.0271990632893522
##                  6                  7                  8                  9
## 0.0265571708306850 0.0266059071540105 0.0265058499665974 0.0258624418120546
##                 10                 11                 12                 13
## 0.0260021789001852 0.0266766345304368 0.0250315472078188 0.0270477064923402
##                 14                 15                 16                 17
## 0.0251182637442381 0.0259492992772571 0.0258531354493679 0.0258561323995070
##                 18                 19                 20                 21
## 0.0251926526549315 0.0252160644717146 0.0258805305412664 0.0244684094334872
##                 22                 23                 24                 25
## 0.0251783627462934 0.0238024227393362 0.0249526374563940 0.0251770295384283
##                 26                 27                 28                 29
## 0.0257467533851809 0.0248083523487031 0.0250188873103541 0.0252190401385995
##                 30                 31                 32                 33
## 0.0247159438499217 0.0245005994350321 0.0265269959489605 0.0256983183277967
##                 34                 35                 36                 37
## 0.0253990598556739 0.0237639476128451 0.0242314254449280 0.0238194469490610
##                 38                 39                 40
## 0.0242010435051740 0.0274351218457520 0.0258205709084172
```

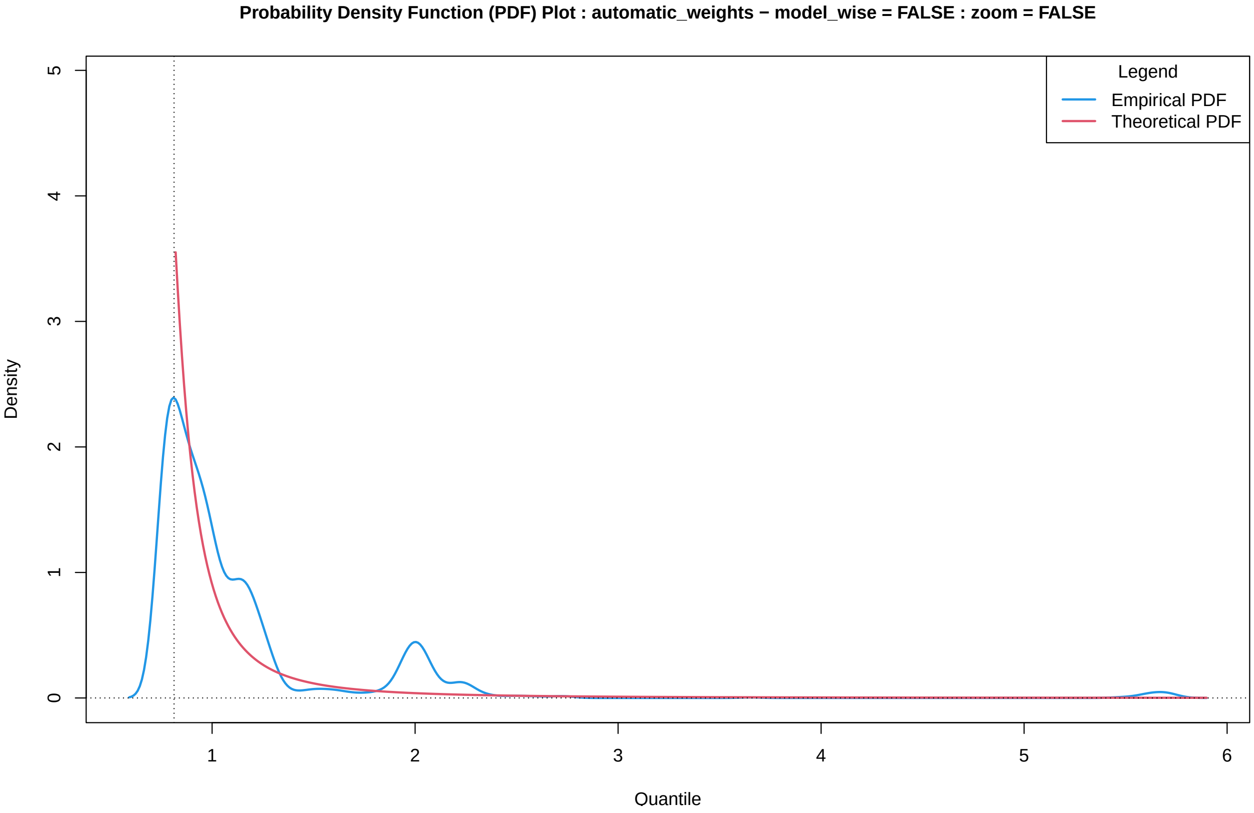
```r
gev_mixture_model$pessimistic_weights_pw_scale
```

```
##                  2                  3                  4                  5
## 0.0269323519651585 0.0264256539662027 0.0262195903052945 0.0260011008309597
##                  6                  7                  8                  9
## 0.0259263599618349 0.0258398455910452 0.0257352813823622 0.0257535915599968
##                 10                 11                 12                 13
## 0.0256989695298786 0.0255764556842864 0.0257454399763994 0.0254706296809070
##                 14                 15                 16                 17
## 0.0256607843970374 0.0255283899057705 0.0255263591100828 0.0254804794934931
##                 18                 19                 20                 21
## 0.0255667473550077 0.0255469902866631 0.0254473589921200 0.0256351507030716
##                 22                 23                 24                 25
## 0.0255356579128203 0.0257478775931350 0.0254857455060878 0.0255491765499834
##                 26                 27                 28                 29
## 0.0254135055527290 0.0255309580379408 0.0254425057285283 0.0254801147646263
##                 30                 31                 32                 33
## 0.0255369198342640 0.0255269121475830 0.0252771381894419 0.0254571861111911
##                 34                 35                 36                 37
## 0.0254400016416388 0.0257115527597871 0.0254897091448828 0.0255883354895055
##                 38                 39                 40
## 0.0255902743142134 0.0251968638637685 0.0252820341803008
```

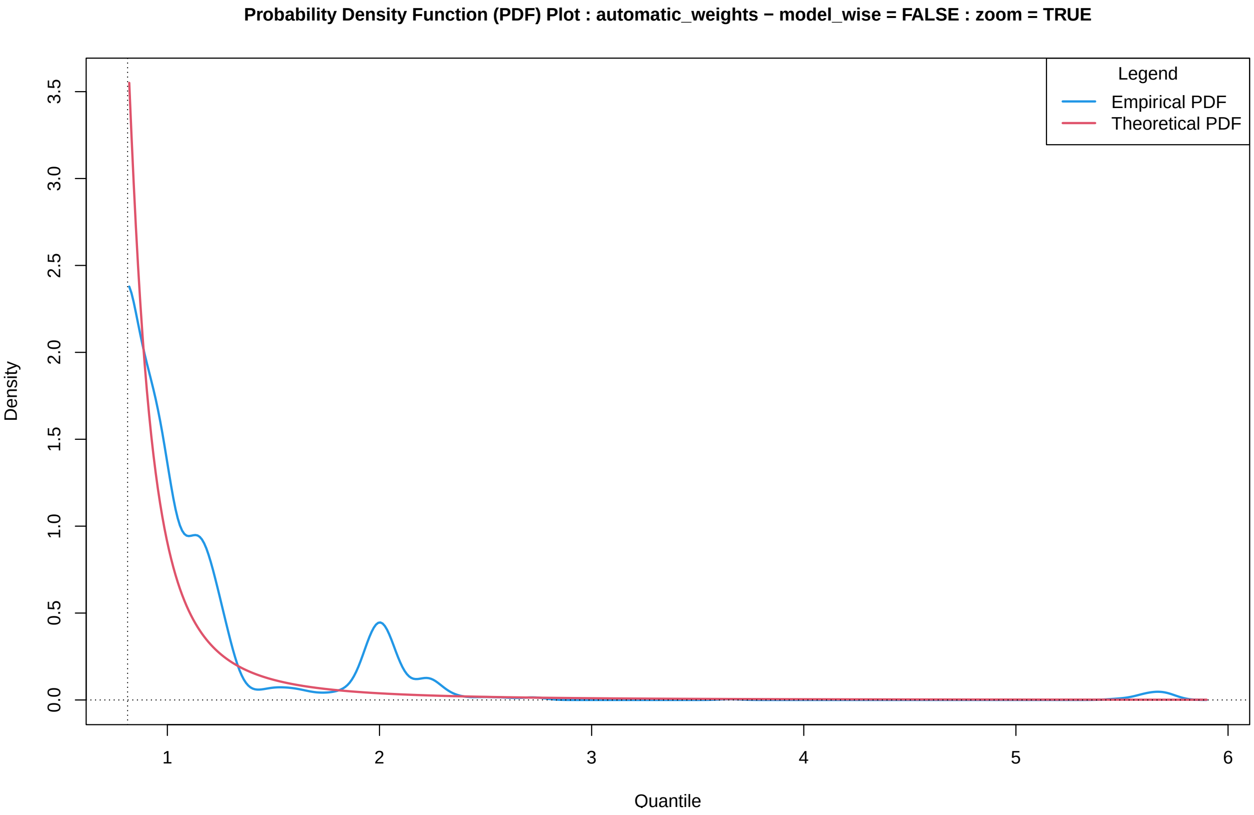
```r
gev_mixture_model$pessimistic_weights_pw_loc
```

```
##                  2                  3                  4                  5
## 0.0286748928960512 0.0279795648058090 0.0275436498716846 0.0272228937723737
##                  6                  7                  8                  9
## 0.0270221516720221 0.0268237806229449 0.0266954037752718 0.0264986015449073
##                 10                 11                 12                 13
## 0.0263510215365866 0.0263346355617462 0.0259438466712829 0.0263678847642808
##                 14                 15                 16                 17
## 0.0259084656749156 0.0259502418821032 0.0258125615737503 0.0259051820066177
##                 18                 19                 20                 21
## 0.0256253930336657 0.0255967510606375 0.0257011532069295 0.0251685261276696
##                 22                 23                 24                 25
## 0.0252248117517010 0.0245535332421329 0.0254308458484715 0.0250146515153075
##                 26                 27                 28                 29
## 0.0254362171982587 0.0248038572228666 0.0251313272838519 0.0248197795372158
##                 30                 31                 32                 33
## 0.0245591869869609 0.0247651399325541 0.0253324869214791 0.0245156833795560
##                 34                 35                 36                 37
## 0.0246315100886600 0.0238270708715208 0.0246973374619478 0.0240497684165845
##                 38                 39                 40
## 0.0239179572969925 0.0253361650409503 0.0248260679417380
```

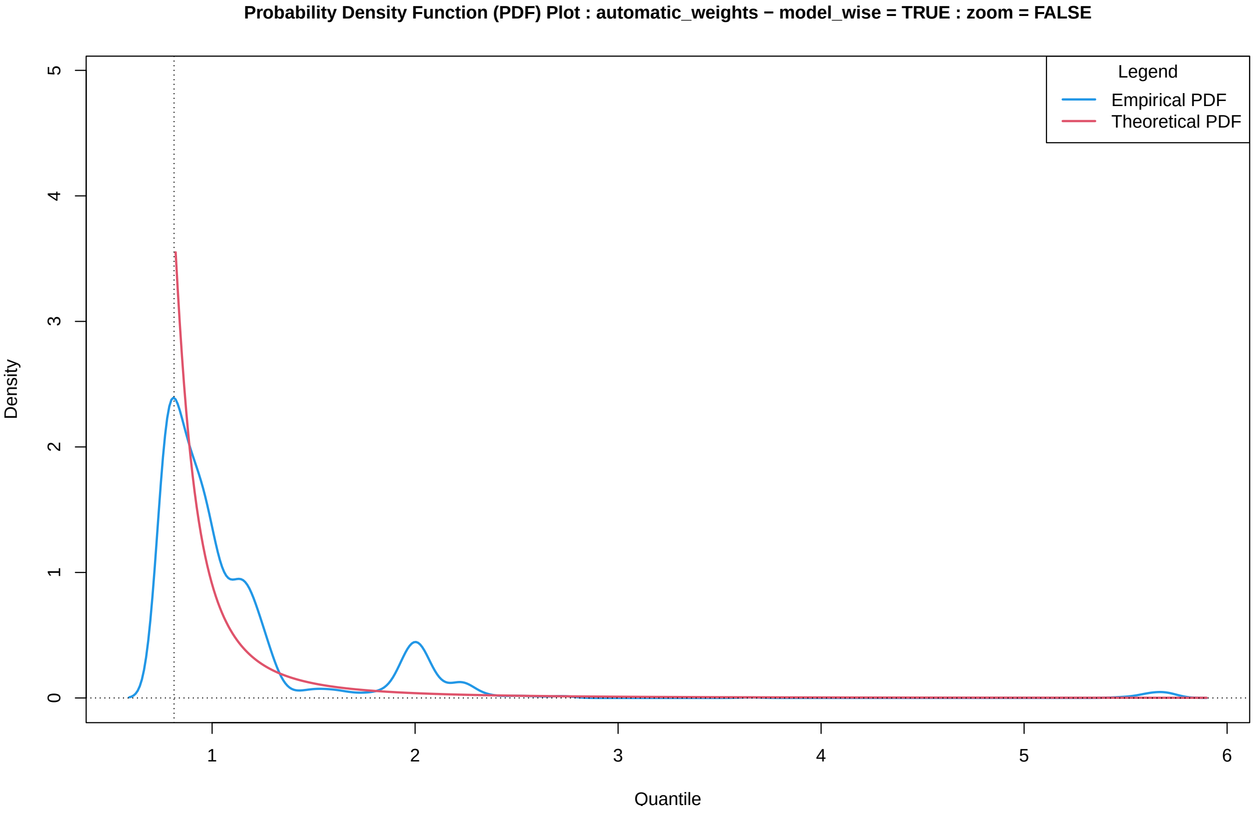
```r
plot_gev_mixture_model_pdf(gev_mixture_model,
                           type = "automatic_weights",
                           model_wise = FALSE,
                           zoom = FALSE,
                           xlab = "Quantile",
                           ylab = "Density",
                           main = "Probability Density Function (PDF) Plot")
```



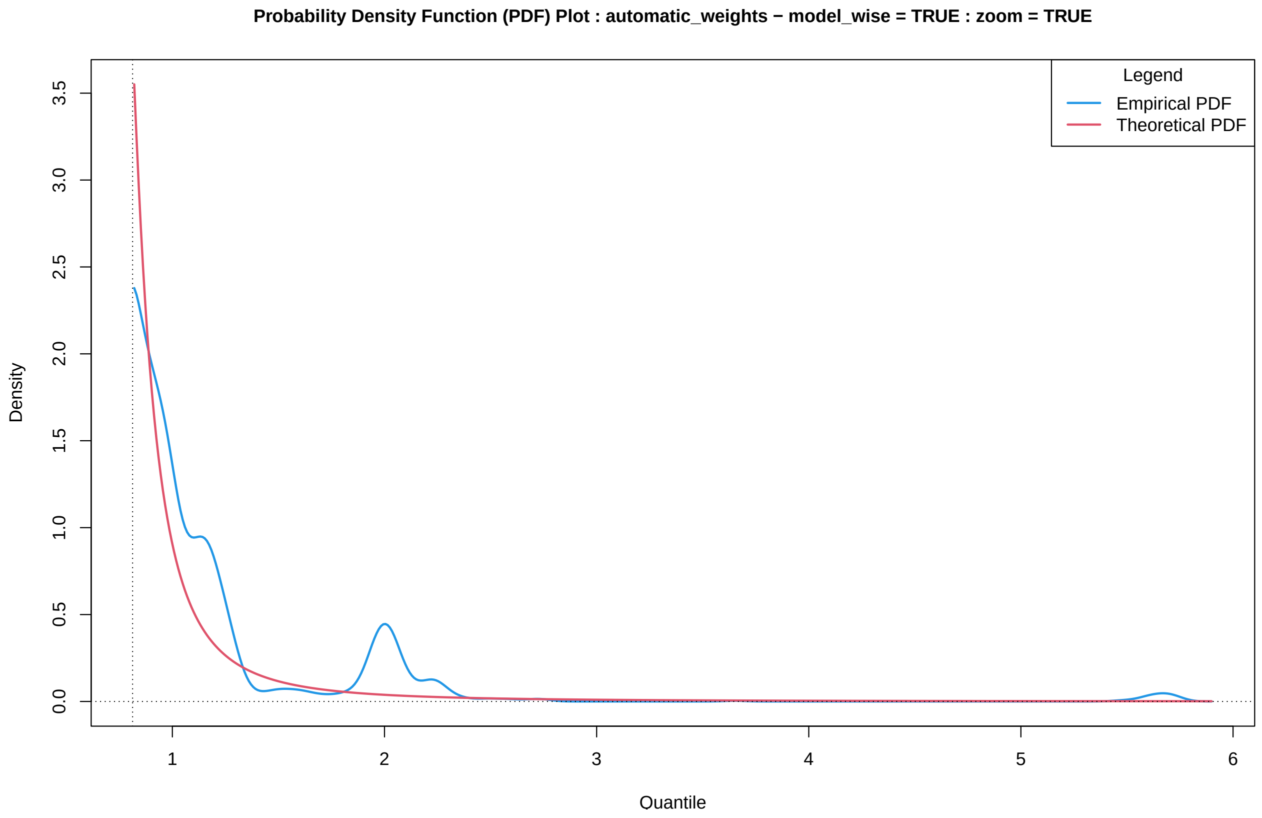
```
plot_gev_mixture_model_pdf(gev_mixture_model,
                           type = "automatic_weights",
                           model_wise = FALSE,
                           zoom = TRUE,
                           xlab = "Quantile",
                           ylab = "Density",
                           main = "Probability Density Function (PDF) Plot")
```



```
plot_gev_mixture_model_pdf(gev_mixture_model,
                           type = "automatic_weights",
                           model_wise = TRUE,
                           zoom = FALSE,
                           xlab = "Quantile",
                           ylab = "Density",
                           main = "Probability Density Function (PDF) Plot")
```



```
plot_gev_mixture_model_pdf(gev_mixture_model,
                           type = "automatic_weights",
                           model_wise = TRUE,
                           zoom = TRUE,
                           xlab = "Quantile",
                           ylab = "Density",
                           main = "Probability Density Function (PDF) Plot")
```



```
estimator_types <- c("automatic_weights_mw",
                     "pessimistic_weights_mw",
                     "identic_weights_mw",
                     "automatic_weights_pw",
                     "pessimistic_weights_pw",
                     "identic_weights_pw",
                     "empirical",
                     "confidence_interval_mw",
                     "confidence_interval_pw")
```

```
alpha <- 10^(-14)
```

```
rl_mw <- estimate_gev_mixture_model_quantile(gev_mixture_model,
                                              alpha = alpha,
                                              confidence_level = 0.95,
                                              do.ci = TRUE,
                                              estimator_type = estimator_types[1])

rl_mw
```

```
##   lower          estimate upper
## 1    NA 197141882.204619    NA
```

```
rl_pw <- estimate_gev_mixture_model_quantile(gev_mixture_model,
                                              alpha = alpha,
                                              confidence_level = 0.95,
                                              do.ci = TRUE,
                                              estimator_type = estimator_types[4])
```

```
rl_pw
```

```
##   lower         estimate upper
## 1    NA 197219149.663122    NA
```

```
rl_empirical <- estimate_gev_mixture_model_quantile(gev_mixture_model,
                                                    alpha = alpha,
                                                    confidence_level = 0.95,
                                                    do.ci = TRUE,
                                                    estimator_type = estimator_types[7])

rl_empirical
```

```
##   lower        estimate upper
## 1    NA 5.7475778000048    NA
```

```
est_rl_pw <- estimate_gev_mixture_model_quantile(gev_mixture_model,
                                                 alpha = alpha,
                                                 confidence_level = 0.95,
                                                 do.ci = TRUE,
                                                 estimator_type = estimator_types[9])

est_rl_pw
```

```
##                 lower        estimate            upper
## 2  -249876860.021693 196630730.323027 643138320.667748
## 3  -157133807.251733 96071227.7244053 349276262.700543
## 4  -68313708.5140478 35585011.1241869 139483730.762422
## 5  -94134171.2650508 41049083.5401527 176232338.345356
## 6    -48600926.01324 19741910.0921628 88084746.1975656
## 7  -52440945.4355982 19535558.4205393 91512062.2766768
## 8  -46606032.8783147 16113532.8126272 78833098.5035692
## 9  -24496616.3489815  8135150.0167915 40766916.3825645
## 10 -29753785.8171515 9066983.70931285 47887753.2357772
## 11 -58555447.6012769  16733915.923241  92023279.447759
## 12 -11325319.1032787  3208803.5693045 17742926.2418877
## 13 -84367083.1095887 22201922.5881909  128770928.28597
## 14  -12538765.249829 3295597.75001952 19129960.7498681
## 15 -30042620.2687485 7277116.04811729 44596852.3649831
## 16 -28093785.2083566 6544262.83300161 41182310.8743598
## 17 -27351996.4031822 6251500.54839346 39854997.4999691
## 18 -13782131.4714134 3278855.39184582  20339842.255105
## 19 -14445096.9301307 3300286.43646325 21045669.8030573
## 20 -29792040.3892429 6192362.39223582 42176765.1737146
## 21 -6906361.25558808 1532516.28099127 9971393.81757062
## 22 -14843996.0535165 3123727.93398785 21091451.9214922
## 23 -3783827.72427494 781749.625049398 5347326.97437373
## 24 -12356581.2728895 2299564.99163829  16955711.256166
## 25  -16974428.073936 3169982.19240537 23314392.4587467
## 26 -28477161.9616509 5148120.60270484 38773403.1670606
## 27 -10971015.3639959 2051957.18311189 15074929.7302197
## 28 -13832105.8952973 2368298.74600537  18568703.387308
## 29 -18350566.3365752 3085927.11354382 24522420.5636628
## 30 -11037983.1637816 1856081.78185332 14750146.7274883
## 31 -8134657.98416091 1436623.03493721 11007904.0540353
## 32 -63707571.5836918 10064331.3873086  83836234.358309
## 33 -33929388.3270195 5128869.00337148 44187126.3337625
## 34 -23424235.3940316 3608551.39811889 30641338.1902693
## 35 -4403521.23739552 724480.504576333 5852482.24654818
## 36 -7226772.32574309 1011404.60867848 9249581.54310004
## 37 -5145106.74984267 691418.799138906 6527944.34812048
## 38 -7036818.62001591 1082336.39829476 9201491.41660544
## 39 -175337184.396596 23233995.1270511 221805174.650698
## 40 -38341092.6433235 4702083.32042373  47745259.284171
```

```
est_rl_pw_range <- range(as.matrix(est_rl_pw))
est_rl_pw_range
```

```
## [1] -249876860.021693  643138320.667748
```

```
est_rl_mw <- estimate_gev_mixture_model_quantile(gev_mixture_model,
                                                 alpha = alpha,
                                                 confidence_level = 0.95,
                                                 do.ci = TRUE,
                                                 estimator_type = estimator_types[8])

est_rl_mw
```

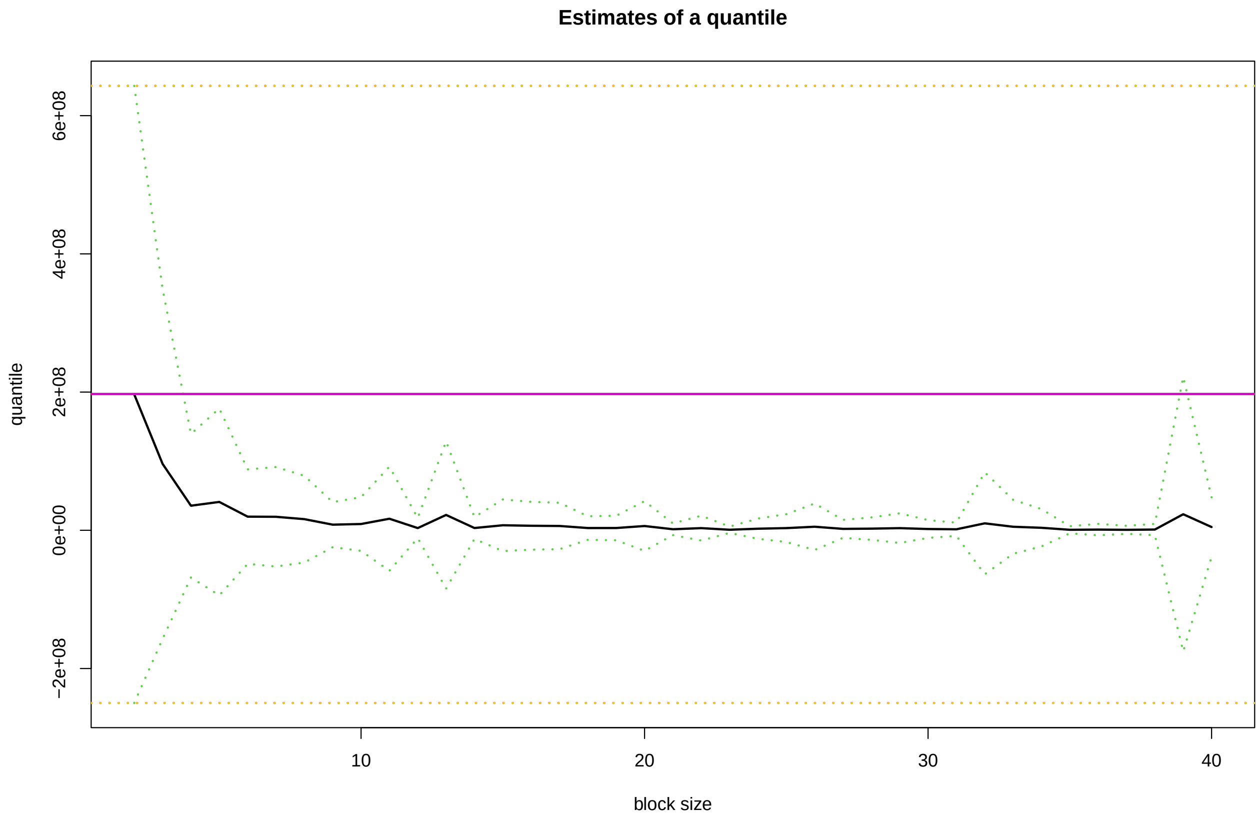
```
##                lower         estimate            upper
## 2  -249876860.021693 196630730.323027 643138320.667748
## 40 -38341092.6433235 4702083.32042373  47745259.284171
```

```
est_rl_mw_range <- range(as.matrix(est_rl_mw))
est_rl_mw_range
```

```
## [1] -249876860.021693  643138320.667748
```

```
matplot(x = rownames(est_rl_pw),
        y = est_rl_pw,
        xlab = "block size",
        ylab = "quantile",
        main = "Estimates of a quantile",
        cex = 1,
        cex.lab = 1,
        cex.axis = 1,
        type = "l",
        lty = c("dotted", "solid", "dotted"),
        lwd = c(2,2,2),
        col = c(3, 1, 3))

abline(h = rl_mw[2], col = 7, lwd = 2)
abline(h = rl_pw[2], col = 6, lwd = 2)
abline(h = est_rl_pw_range, col = 6, lty = "dotted", lwd = 2)
abline(h = est_rl_mw_range, col = 7, lty = "dotted", lwd = 2)
```


Estimates of a quantile
quantile
6e+08
4e+08
2e+08
0e+00
−2e+08
10
20
30
40
block size