Modeling extreme values with a GEV mixture probability distributions

Application to a wind speed data

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"))
```

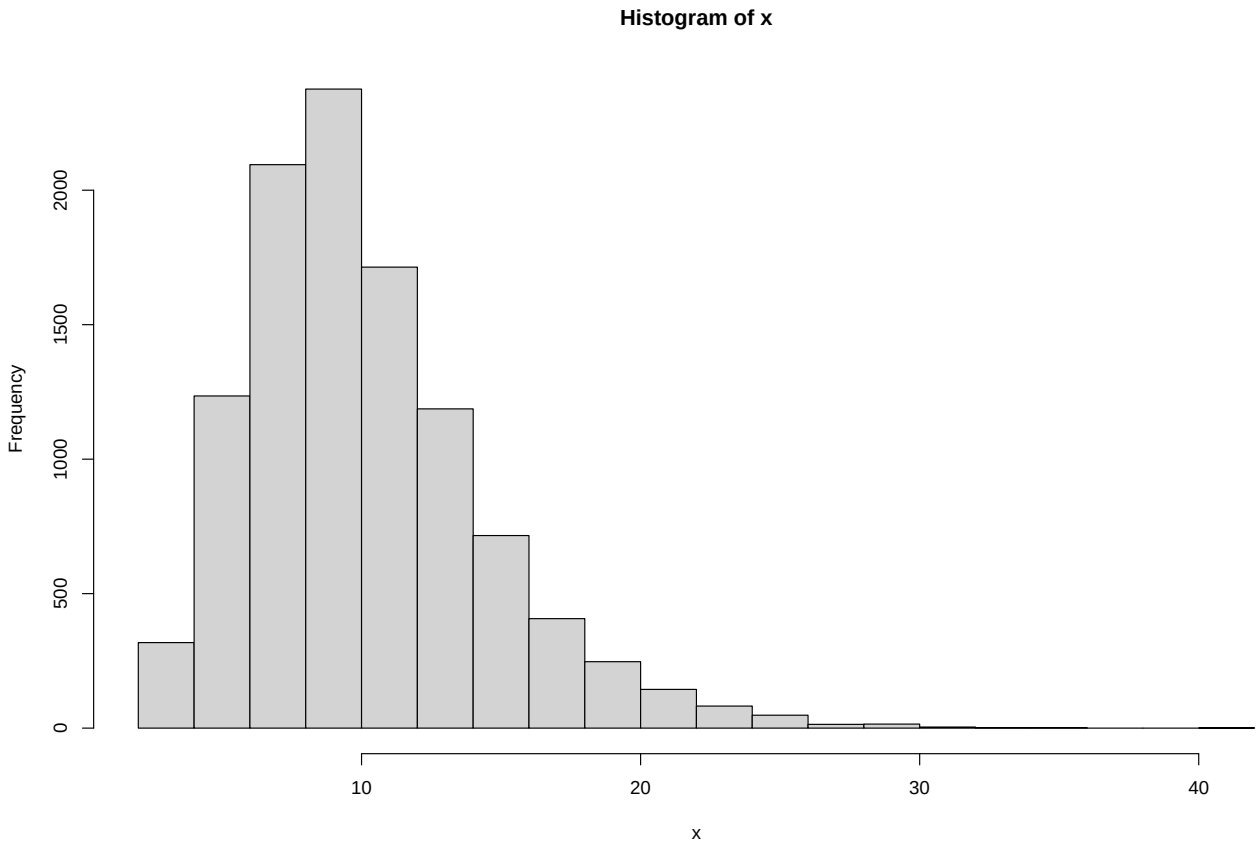
```
library(readr)
```

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

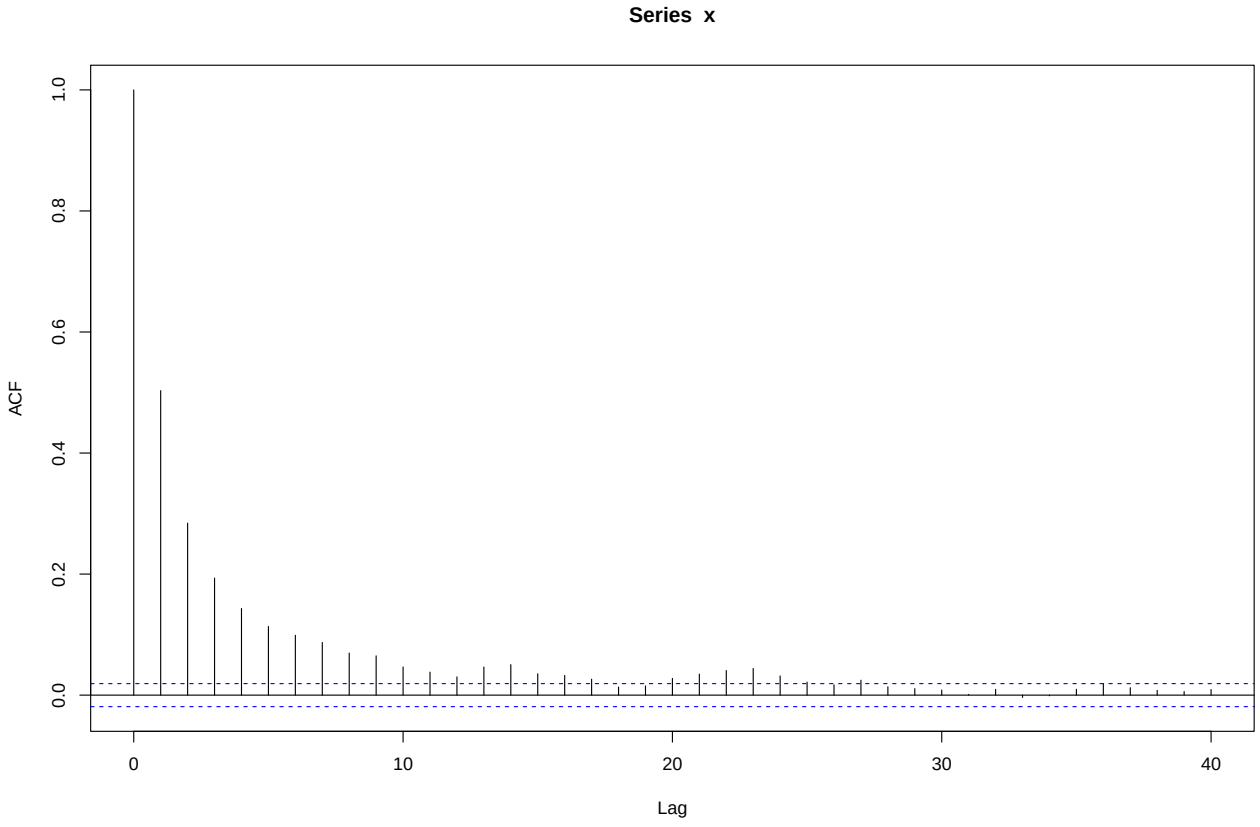
```
## Rows: 10627 Columns: 2
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## dbl  (1): Vent
## date (1): Date
##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

```
x <- vent$Vent
x <- x[!is.na(x)]
```

```
hist(x)
```


Histogram of x
Frequency
0
500
1000
1500
2000
10
20
30
40
x

```
acf(x)
```


Series x
ACF
0.0
0.2
0.4
0.6
0.8
1.0
0
10
20
30
40
Lag

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

```
## [1] 10607
```

```r
nlargest <- 1000

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

```r
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.
```

```r
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"
```

```r
gev_mixture_model$block_sizes
```

```
##  [1]  5  6  7  8  9 10 11 12 13 14 15 16 17 18 19 20
```

```r
gev_mixture_model$normalized_gev_parameters_object
```

```
##           loc_star       scale_star            shape_star
## 5  17.6205241237820 2.50470061406986  0.048443671023250333
```

```
## 6  17.7609100736487 2.44019943415645  0.047840767044563602
## 7  17.3707030848716 2.67269839188198  0.024805760894366619
## 8  17.3726806508740 2.71302038141482  0.016894559104128678
## 9  16.1270834085008 3.44863440129842 -0.047130198272530753
## 10 17.0616883773534 2.70817095719868  0.023388177884743604
## 11 18.2991917139322 2.20574482189218  0.064046844944789863
## 12 16.4330964161528 3.00050496503031  0.000389093032309497
## 13 16.8127671761745 2.82262103367131  0.010159128434603350
## 14 17.8222556674420 2.27217894417571  0.059525152489232798
## 15 19.3791292269775 1.54739563503278  0.165618933092516601
## 16 15.9078841226837 3.32180913031257 -0.028517578542985696
## 17 17.7361587211316 2.55425769669561  0.026915017674042148
## 18 18.5234333047301 1.73472549547027  0.132983274596403994
## 19 18.1213157299745 2.26374675834640  0.064391373571327071
## 20 18.1782068132748 2.28573561061218  0.057579180181332665
```

```
gev_mixture_model$weighted_normalized_gev_parameters_object
```

```
##                              loc_star       scale_star          shape_star
## identic_weights     17.5329392882190 2.53100901695372 0.04170832232200590
## pessimistic_weights 18.2757499577602 2.76096343747712 0.04440215996366131
## automatic_weights   17.4840313429108 2.66388375769000 0.00124929494078015
```

```
gev_mixture_model$automatic_weights_mw_statistics
```

```
## $function_value
## [1] 0.000702621261983087
##
## $gradient_value
## [1] 1.84290702009982e-06
##
## $function_reduction
## [1] 0.000304133463602511
##
## $number_iterations
## [1] 3872
##
## $convergence
## [1] 0
##
## $message
## [1] "Successful convergence"
```

```
gev_mixture_model$automatic_weights_pw_statistics
```

```
## $function_value
## [1] 0.000602530496063872
##
## $gradient_value
## [1] 0.0011359591549744
##
## $function_reduction
## [1] 0.000719589170664613
##
## $number_iterations
## [1] 3309
```

```
##
## $convergence
## [1] 0
##
## $message
## [1] "Successful convergence"
```

```
gev_mixture_model$automatic_weights_mw
```

```
##                  5                  6                  7                  8
## 0.0000000000000000 0.0000000000000000 0.0000000000000000 0.0000000000000000
##                  9                 10                 11                 12
## 0.0000000000000000 0.0000000000000000 0.0000000000000000 0.0000000000000000
##                 13                 14                 15                 16
## 0.0718468926136039 0.4304274863240161 0.0000000000000000 0.0000000000000000
##                 17                 18                 19                 20
## 0.4977256210623799 0.0000000000000000 0.0000000000000000 0.0000000000000000
```

```
gev_mixture_model$pessimistic_weights_pw_shape
```

```
##                  5                  6                  7                  8
## 0.0628381916881731 0.0628003177107239 0.0613702460940776 0.0608866491321286
##                  9                 10                 11                 12
## 0.0571105673652800 0.0612833103098220 0.0638263561247333 0.0598899348553282
##                 13                 14                 15                 16
## 0.0604779293253254 0.0635384044767897 0.0706500133858259 0.0581834986951883
##                 17                 18                 19                 20
## 0.0614998283147684 0.0683815218746935 0.0638483499200743 0.0634148807270679
```

```
gev_mixture_model$pessimistic_weights_pw_scale
```

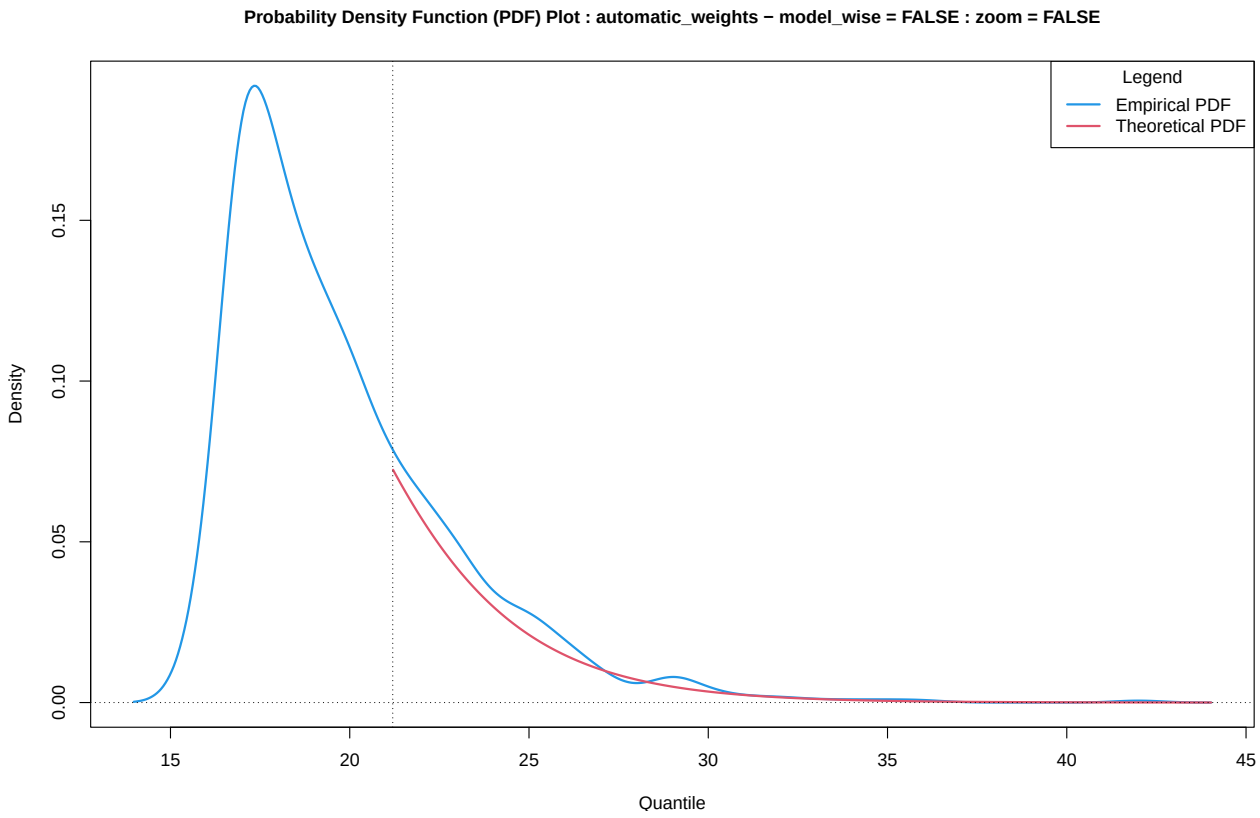
```
##                  5                  6                  7                  8
## 0.0542036718785900 0.0508178399511025 0.0641193654112906 0.0667576181484791
##                  9                 10                 11                 12
## 0.1393073217211691 0.0664346658376993 0.0401970198909310 0.0889925807545419
##                 13                 14                 15                 16
## 0.0744903116967280 0.0429581756518462 0.0208102087991340 0.1227140889178031
##                 17                 18                 19                 20
## 0.0569575205003194 0.0250976328867255 0.0425974672492494 0.0435445107043908
```

```
gev_mixture_model$pessimistic_weights_pw_loc
```

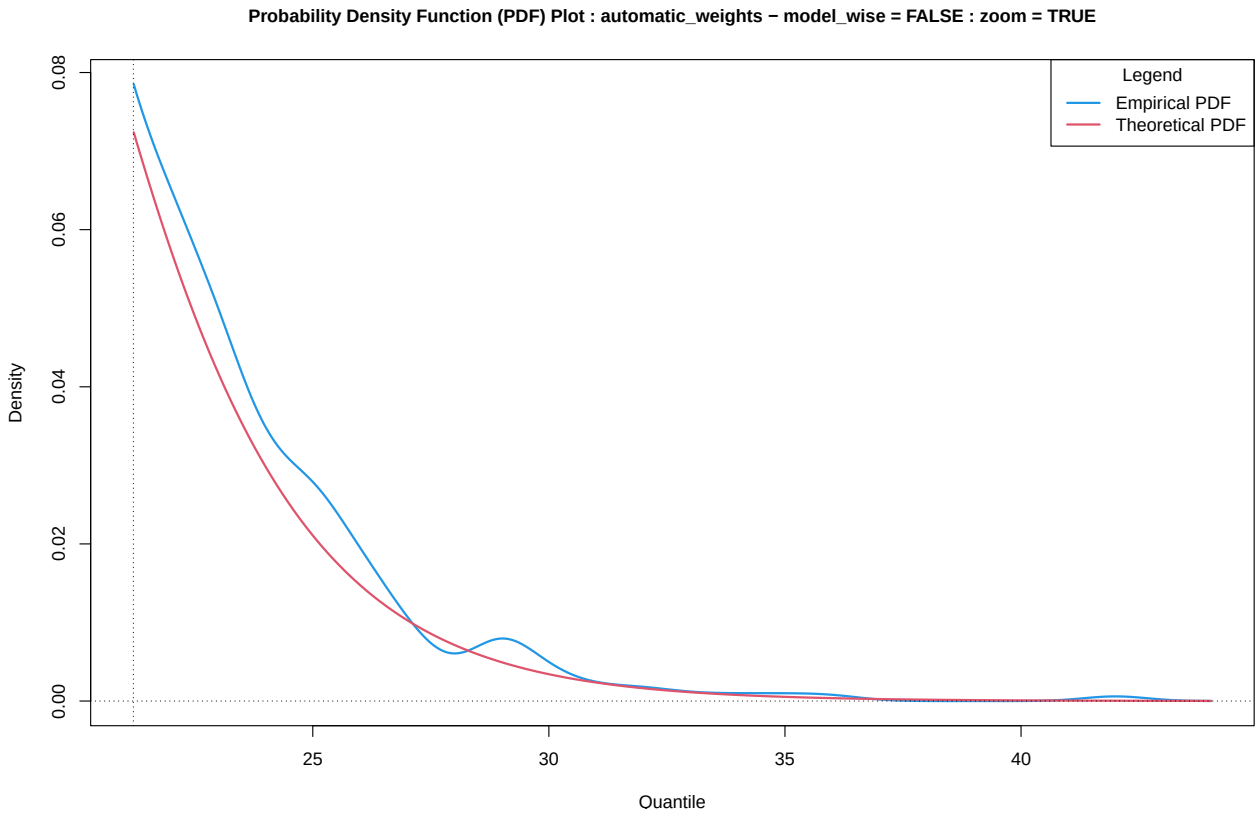
```
##                   5                   6                   7                   8
## 0.04667886849690563 0.05371420638304256 0.03636004578947129 0.03643202132460130
##                   9                  10                  11                  12
## 0.01048400594246828 0.02669445380385319 0.09201569475897257 0.01423727976399129
##                  13                  14                  15                  16
## 0.02081210149037658 0.05711250595831330 0.27093980400159878 0.00842035144300145
##                  17                  18                  19                  20
## 0.05240102566552134 0.11514598389559980 0.07702138986624887 0.08153026141603371
```

```
plot_gev_mixture_model_pdf(gev_mixture_model,
                           type = "automatic_weights",
                           model_wise = FALSE,
                           zoom = FALSE,
                           xlab = "Quantile",
                           ylab = "Density",
```

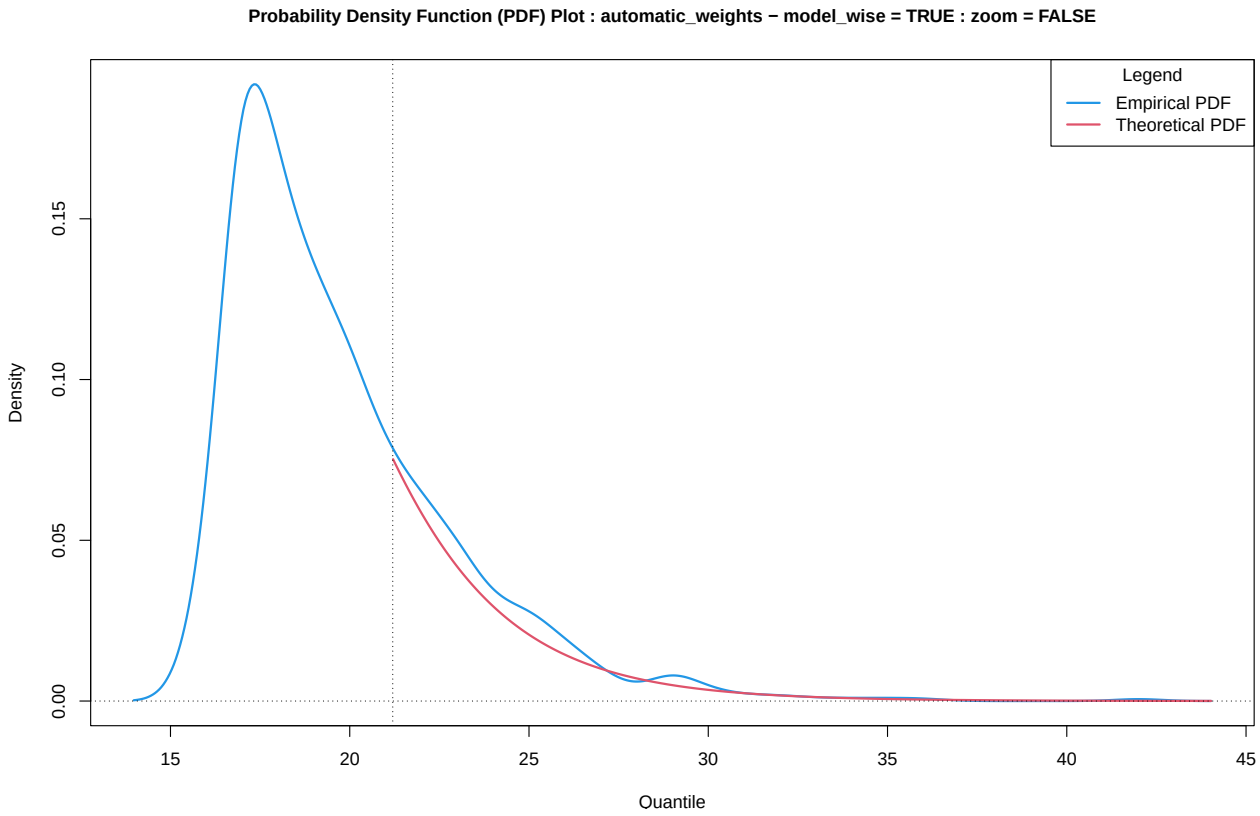
```
                        main = "Probability Density Function (PDF) Plot")
```



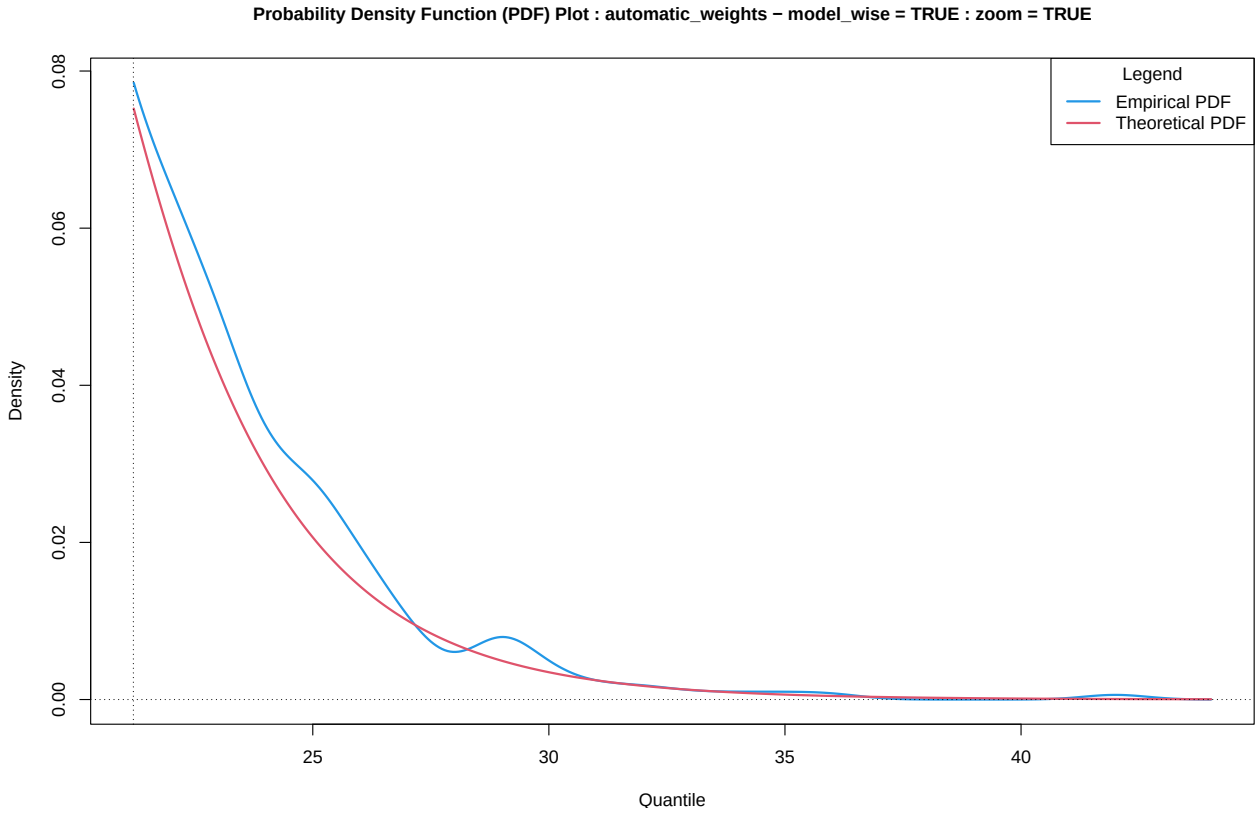

```
plot_gev_mixture_model_pdf(gev_mixture_model,
                           type = "automatic_weights",
                           model_wise = FALSE,
                           zoom = TRUE,
                           xlab = "Quantile",
                           ylab = "Density",
                           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^(-6)
```

```
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 53.4179549992172    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 48.2154823152573    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 41.936364000001    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
## 5  37.3607975255106  55.969842569361 74.5788876132114
## 6  36.2300132135648 54.9819985287164  73.733983843868
## 7  35.6271372782938 52.7754701568567 69.9238030354196
## 8  35.7410814419031 51.6577239923709 67.5743665428387
## 9  37.5355830862437 46.6514091423969   55.76723519855
## 10 34.7526697107768 52.6323512441578 70.5120327775388
## 11 33.2879128616886 55.5810750902905 77.8742373188925
## 12 34.9682650759231 50.8774534920081 66.7866419080931
## 13 34.0635036652888 51.0987339645002 68.1339642637116
## 14 32.9024180410563 55.1334636756482 77.3645093102402
## 15 14.9374069636966 72.3232465621667 129.709086160637
## 16 35.0624659206243 48.3663724445649 61.6702789685056
## 17 31.7296943126891 52.0029639301011 72.2762335475131
## 18 20.7186681866565 65.3127415515182  109.90681491638
## 19 27.8634590276772 56.4682591159479 85.0730592042187
## 20 29.0428103741655 55.2488294222344 81.4548484703033
```

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

```
## [1]  14.9374069636966 129.7090861606369
```

```
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
## 13 34.0635036652888 51.0987339645002 68.1339642637116
```

```
## 14 32.9024180410563 55.1334636756482 77.3645093102402
## 17 31.7296943126891 52.0029639301011 72.2762335475131
```

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

```
## [1] 31.7296943126891 77.3645093102402
```

```
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)
```