# Modeling extreme values with a GEV mixture probability distributions

## Application to a wind speed data

Pascal Alain Dkengne Sielenou

2023-10-12

```r
# library(xfun)
```

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

```r
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_gev_mixture_model_cdf.R"))
xfun::in_dir(dir = path, expr = source("./src/estimate_gev_mixture_model_quantile.R"))
```

```r
library(readr)
```

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

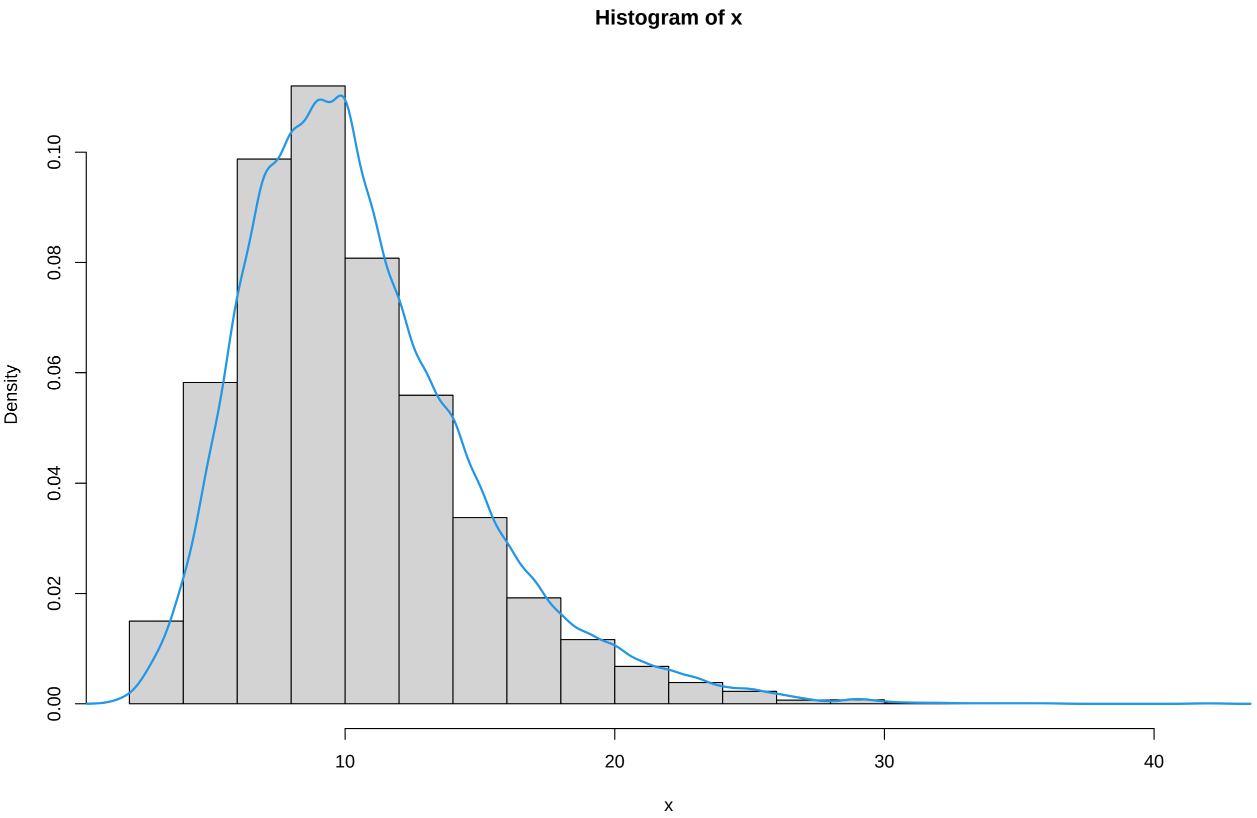
```
## Rows: 10627 Columns: 2
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## dbl  (1): Vent
## date (1): Date
##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types =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

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

n
```

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

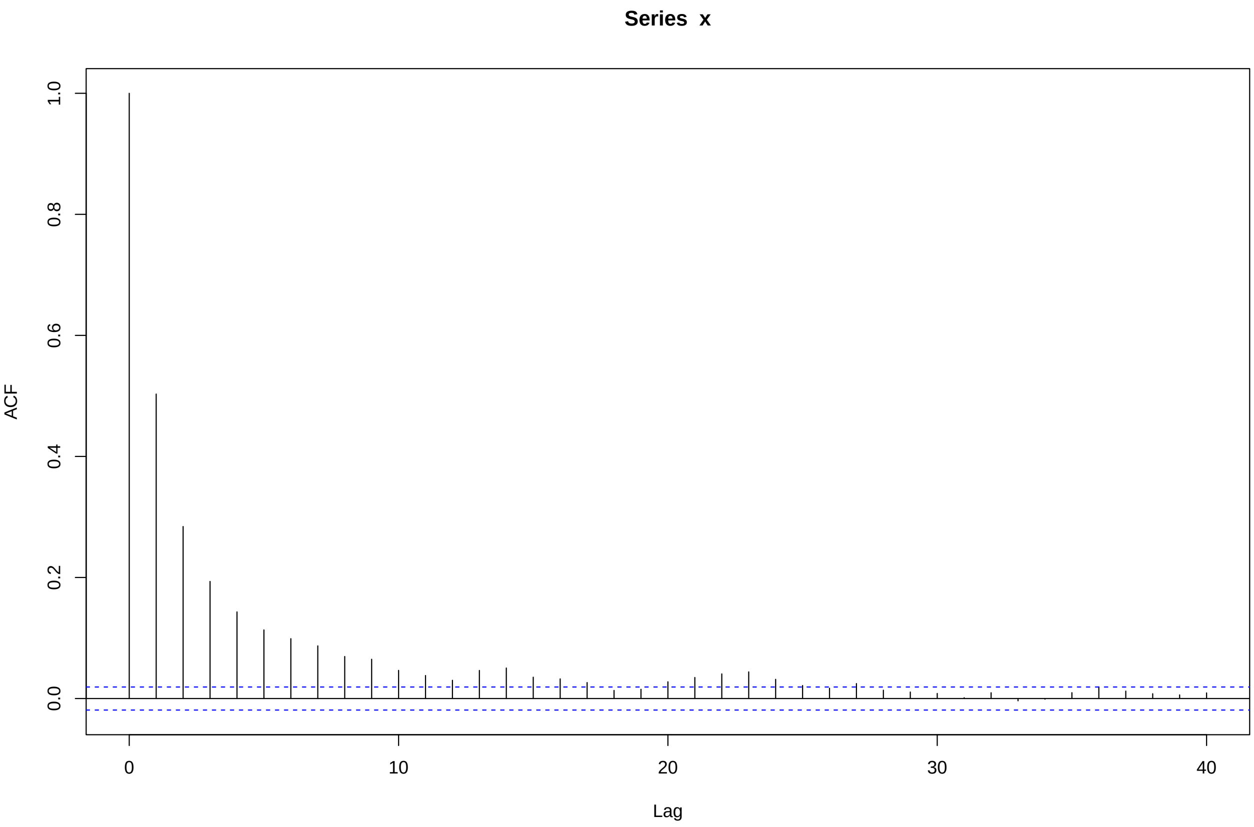
```r
# Histogram of all data
```

```r
hist(x, prob = TRUE)
lines(density(x),
      lwd = 2,
      col = 4)
```



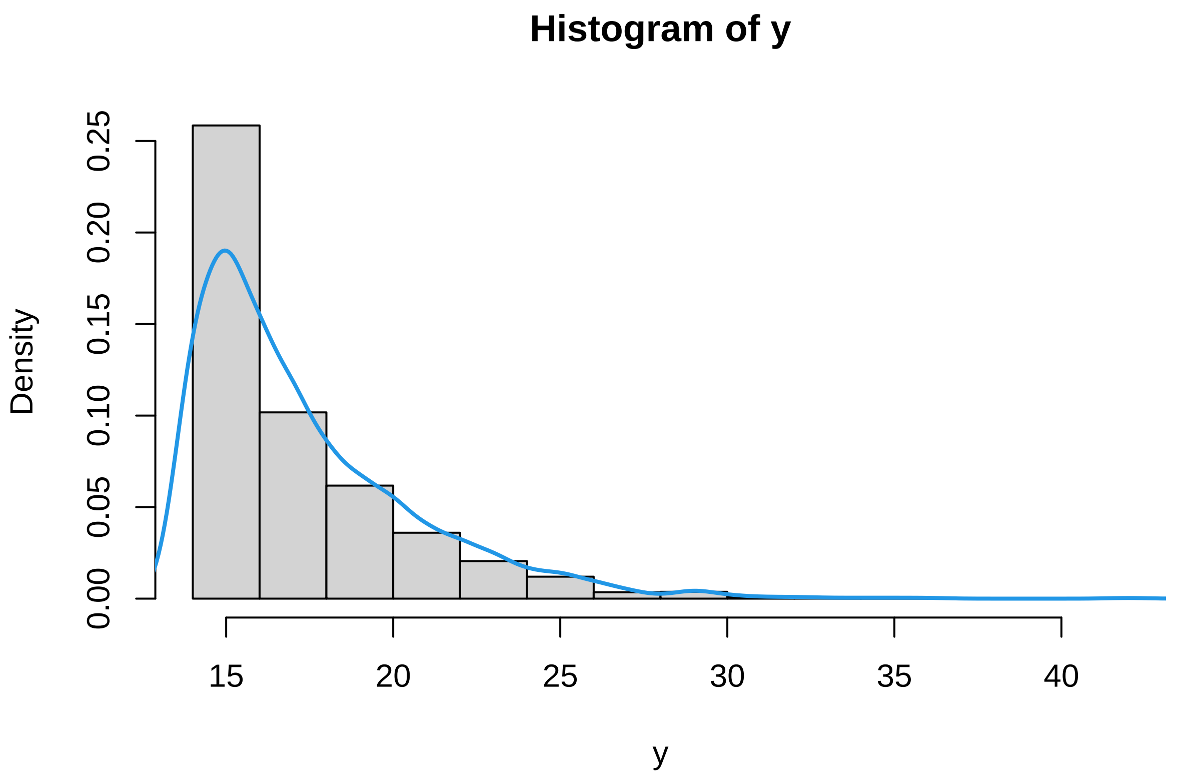
```
# Autocorrelation function of all data
```

```
acf(x)
```



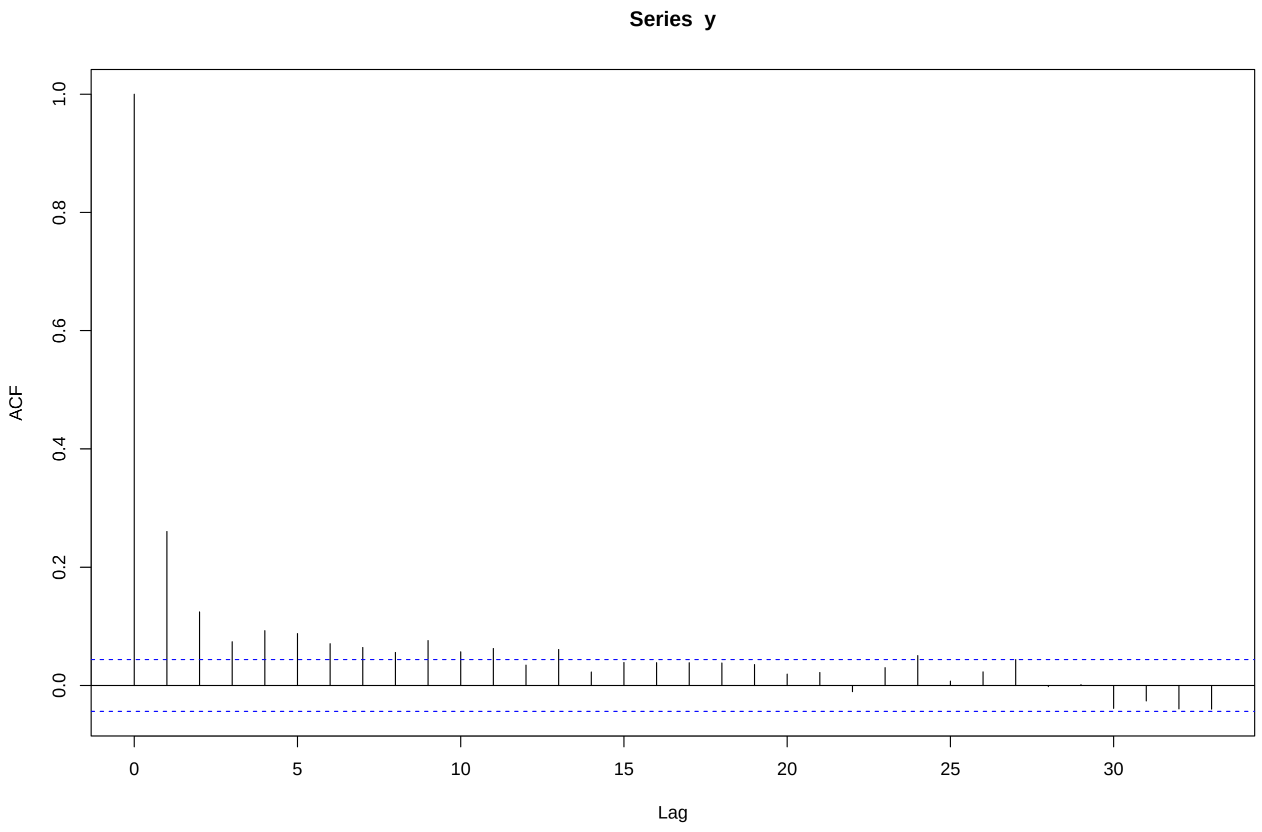
```
# Histogram of the largest data
```

```
nlargest <- 2000
y <- extract_nlargest_sample(x, n = nlargest)
hist(y, prob = TRUE)
lines(density(y),
      lwd = 2,
      col = 4)
```



```
# Autocorrelation function of the largest data
```

```
acf(y)
```



```
# Estimation of gev mixture models
```

```
gev_mixture_model <- estimate_gev_mixture_model_parameters(x = x,
                                                           block_sizes = NULL,
                                                           minimum_nblocks = 50,
                                                           threshold = NULL,
                                                           nlargest = nlargest,
                                                           confidence_level = 0.95,
                                                           use_extremal_index = TRUE,
                                                           use_lower_threshold = FALSE,
                                                           maximum_iterations = 1500,
                                                           log_mv = TRUE,
                                                           log_pw = TRUE,
                                                           trace = FALSE,
                                                           method = "MLE")
```

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

```
gev_mixture_model$extremal_indexes
```

```
##           19           20           21           22           23           24 
## 0.7083622670 0.7235605139 0.7083622670 0.7235605139 0.6882408990 0.7246083284 
##           25           26           27           28           29           30 
## 0.7235605139 0.6379524564 0.6691287041 0.6882408990 0.7235605139 0.6882408990 
##           31           32           33           34           35           36 
## 0.6379524564 0.6882408990 0.6379524564 0.6882408990 0.7083622670 0.6379524564 
##           37           38           39           40 
## 0.6691287041 0.6379524564 0.6879635631 0.6379524564
```

```
gev_mixture_model$normalized_gev_parameters_object
```

```
##         loc_star  scale_star        shape_star
## 19 12.207454222 3.916602815 -0.0500647997067
## 20 13.339404221 3.386243511 -0.0222016275348
## 21 12.120283966 3.906377152 -0.0533039821013
## 22 12.528390898 3.658864734 -0.0349411538984
## 23 13.584349251 3.118116705  0.0001124426024
## 24 13.554242389 3.240016830 -0.0182563959382
## 25 12.795352420 3.669494229 -0.0379260347583
## 26 14.978565180 2.628178877  0.0304036700219
## 27 10.228845186 4.593014880 -0.0769598337587
## 28 12.414671325 3.512081028 -0.0144312981038
## 29 12.492112381 3.837265211 -0.0474754050521
## 30 13.017878862 3.465678541 -0.0271683273755
## 31 14.581985396 2.526069010  0.0524553323527
## 32 12.440046281 3.655038452 -0.0329641740013
## 33 13.900613980 2.897107860  0.0175588295455
## 34 10.571721606 4.405852497 -0.0597603934781
## 35  7.956573550 5.758776233 -0.1175170546092
## 36 14.591645767 2.681754521  0.0306909008983
## 37  7.925418866 5.304655126 -0.0924866381669
## 38 14.338779818 2.813414984  0.0174012536989
## 39 10.898293258 4.099321612 -0.0442725396631
## 40 12.046054094 3.729905045 -0.0315836341529
```

```
gev_mixture_model$full_normalized_gev_parameters_object
```

```
##        loc_star  scale_star       shape_star
## 19 10.845287729 3.984799407 -0.0500647997067
## 20 12.239768642 3.410657211 -0.0222016275348
## 21 10.760912723 3.978837052 -0.0533039821013
## 22 11.337770161 3.700466397 -0.0349411538984
## 23 12.419394313 3.117985715  0.0001124426024
## 24 12.507480291 3.259126933 -0.0182563959382
## 25 11.600694862 3.714802853 -0.0379260347583
## 26 13.805256657 2.592505992  0.0304036700219
## 27  8.354642295 4.737253223 -0.0769598337587
## 28 11.098956573 3.531068500 -0.0144312981038
## 29 11.240898510 3.896667096 -0.0474754050521
## 30 11.716450747 3.501036166 -0.0271683273755
## 31 13.459820122 2.467205457  0.0524553323527
## 32 11.066020272 3.700332084 -0.0329641740013
## 33 12.603514014 2.874332303  0.0175588295455
## 34  8.907108905 4.505330407 -0.0597603934781
## 35  5.930172098 5.996912963 -0.1175170546092
## 36 13.394496360 2.645012927  0.0306909008983
## 37  5.754026888 5.505479871 -0.0924866381669
## 38 13.079106471 2.791495089  0.0174012536989
## 39  9.352302896 4.167766532 -0.0442725396631
## 40 10.357536185 3.783234577 -0.0315836341529
```

```
gev_mixture_model$automatic_weights_pw_shape
```

```
##               19               20               21               22
##  0.000000000e+00 -8.470329473e-22  0.000000000e+00  1.694065895e-21
##               23               24               25               26
## -2.613890736e-22  4.235164736e-22  1.694065895e-21 -3.388131789e-21
##               27               28               29               30
##  0.000000000e+00  2.541098842e-21  0.000000000e+00  1.694065895e-21
##               31               32               33               34
##  1.694065895e-21  2.541098842e-21 -2.964615315e-21  0.000000000e+00
##               35               36               37               38
##  8.573504150e-01 -8.470329473e-22  1.426495850e-01  8.470329473e-22
##               39               40
##  0.000000000e+00 -8.470329473e-22
```

```
gev_mixture_model$automatic_weights_pw_scale
```

```
##             19             20             21             22             23
## 0.008604648234 0.015274183782 0.008411807552 0.002350393953 0.039669027732
##             24             25             26             27             28
## 0.023460073684 0.002177092531 0.105720327576 0.050752094523 0.008860131893
##             29             30             31             32             33
## 0.005808750156 0.010460251249 0.121519363084 0.002351980994 0.070212505432
##             34             35             36             37             38
## 0.034697821705 0.179762244080 0.099100787202 0.112489099832 0.080629114683
##             39             40
## 0.014729864597 0.002958435527
```

```
gev_mixture_model$automatic_weights_pw_loc
```

```
##                19                20                21                22                23 
## 0.000000000e+00 0.000000000e+00 0.000000000e+00 0.000000000e+00 0.000000000e+00 
##                24                25                26                27                28 
## 0.000000000e+00 0.000000000e+00 9.852168793e-01 0.000000000e+00 0.000000000e+00 
##                29                30                31                32                33 
## 0.000000000e+00 0.000000000e+00 1.475123633e-02 0.000000000e+00 0.000000000e+00 
##                34                35                36                37                38 
## 0.000000000e+00 6.776263578e-21 2.708832333e-05 1.355252716e-20 4.796019839e-06 
##                39                40 
## 0.000000000e+00 0.000000000e+00
```

```
gev_mixture_model$weighted_normalized_gev_parameters_object[3, ]
```

```
##                      loc_star  scale_star    shape_star
## automatic_weights 14.97270159 3.802991485 -0.1139464761
```

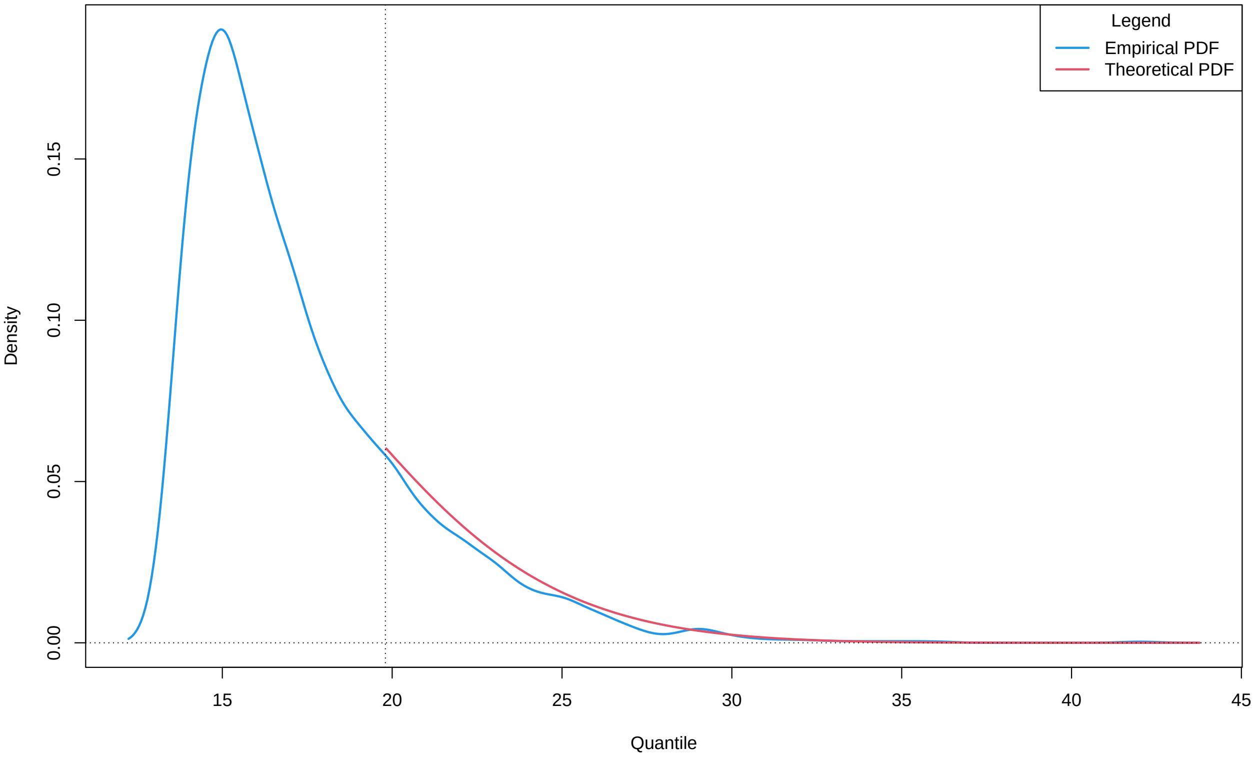
```
gev_mixture_model$automatic_weights_mw
```

```
##                19                20                21                22                23 
## 0.000000000e+00 2.103725991e-01 0.000000000e+00 0.000000000e+00 0.000000000e+00 
##                24                25                26                27                28 
## 0.000000000e+00 0.000000000e+00 7.896274009e-01 0.000000000e+00 0.000000000e+00 
##                29                30                31                32                33 
## 0.000000000e+00 0.000000000e+00 0.000000000e+00 0.000000000e+00 0.000000000e+00 
##                34                35                36                37                38 
## 0.000000000e+00 0.000000000e+00 0.000000000e+00 0.000000000e+00 0.000000000e+00 
##                39                40 
## 3.552713679e-15 0.000000000e+00
```

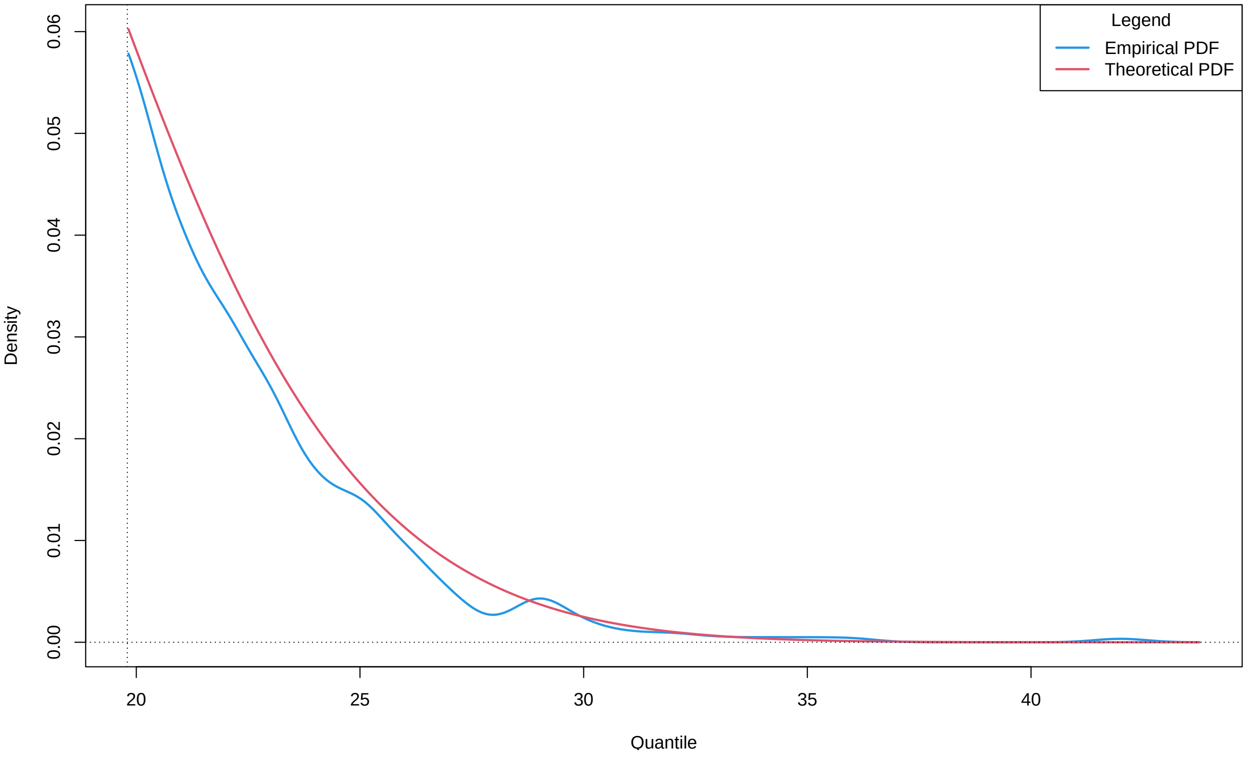
```
# Model diagnostics

## GEV mixture model with respect to parameters
```

```
par(mfrow = c(2, 1))
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")
```

**Probability Density Function (PDF) Plot : automatic_weights − model_wise = FALSE : zoom = FALSE**

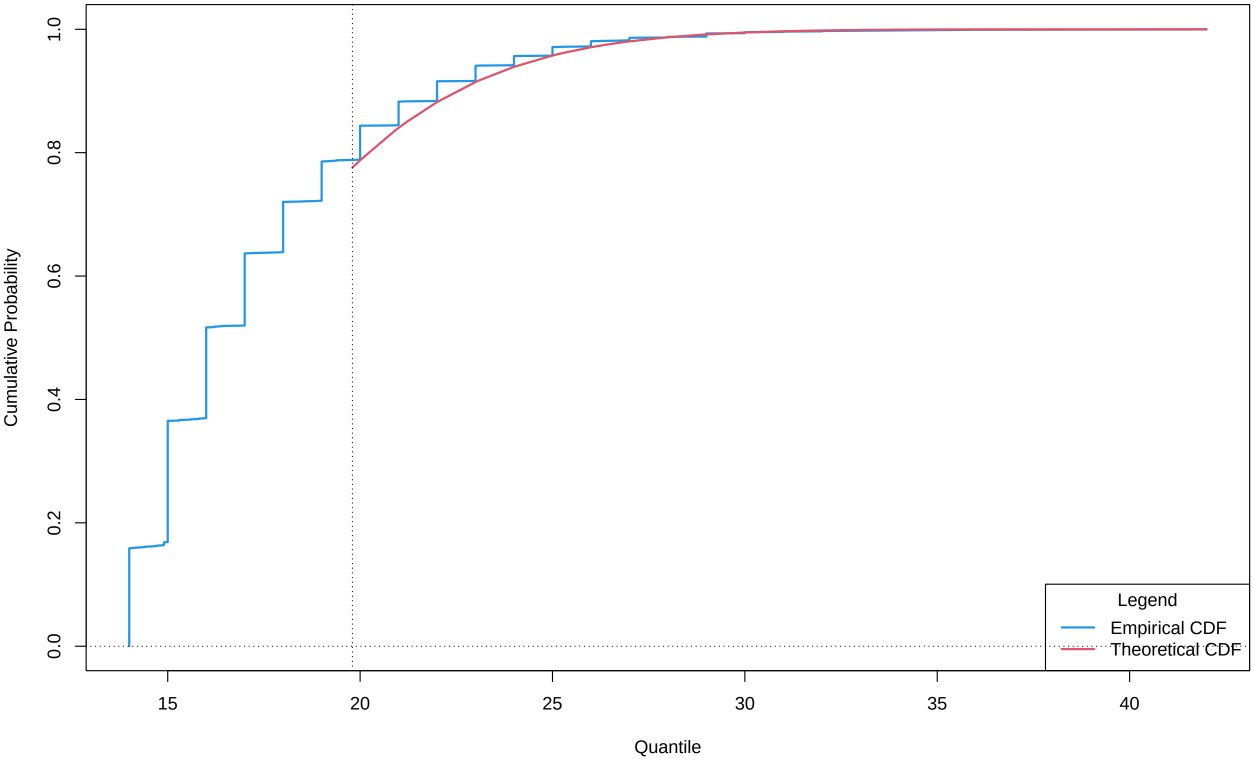


**Probability Density Function (PDF) Plot : automatic_weights − model_wise = FALSE : zoom = TRUE**

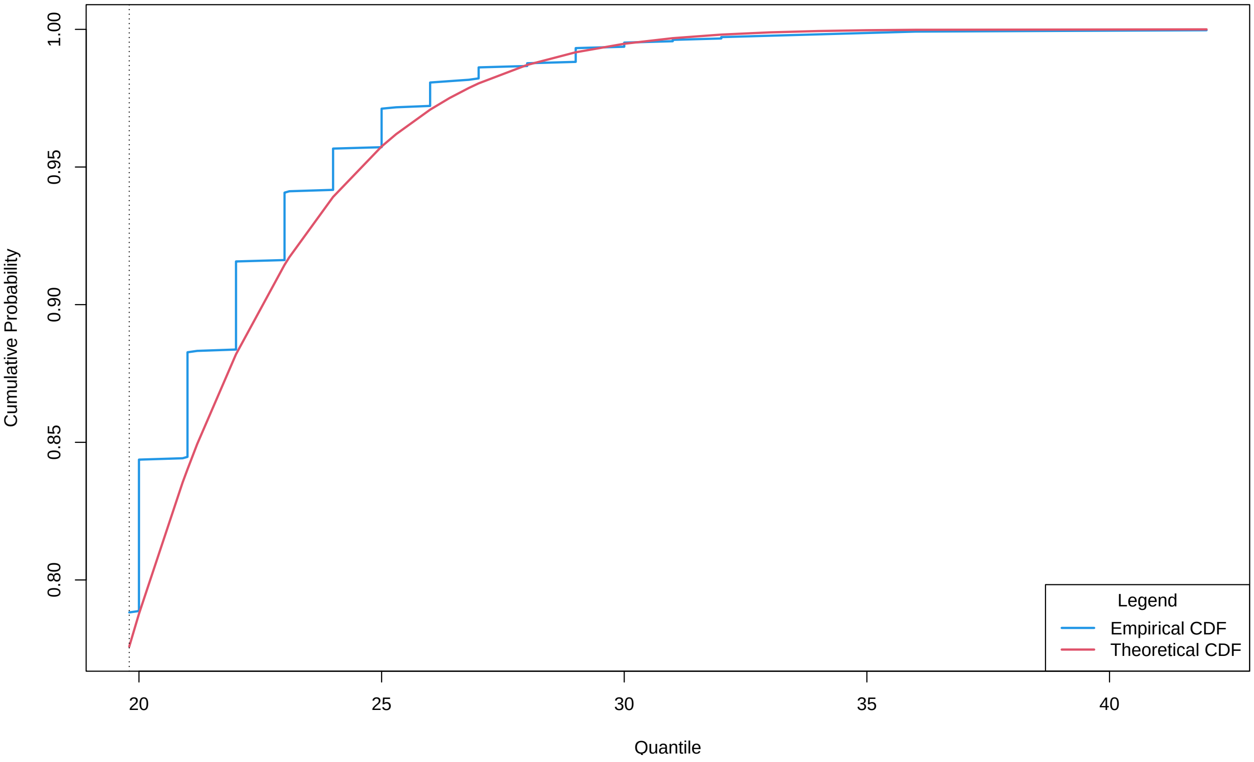


```r
par(mfrow = c(2, 1))
plot_gev_mixture_model_cdf(gev_mixture_model,
                           type = "automatic_weights",
                           model_wise = FALSE,
                           zoom = FALSE,
                           xlab = "Quantile",
                           ylab = "Cumulative Probability",
                           main = "Cumulative Distribution Function (CDF) Plot")

plot_gev_mixture_model_cdf(gev_mixture_model,
                           type = "automatic_weights",
                           model_wise = FALSE,
                           zoom = TRUE,
                           xlab = "Quantile",
                           ylab = "Cumulative Probability",
                           main = "Cumulative Distribution Function (CDF) Plot")
```

Cumulative Distribution Function (CDF) Plot : automatic_weights − model_wise = FALSE : zoom = FALSE

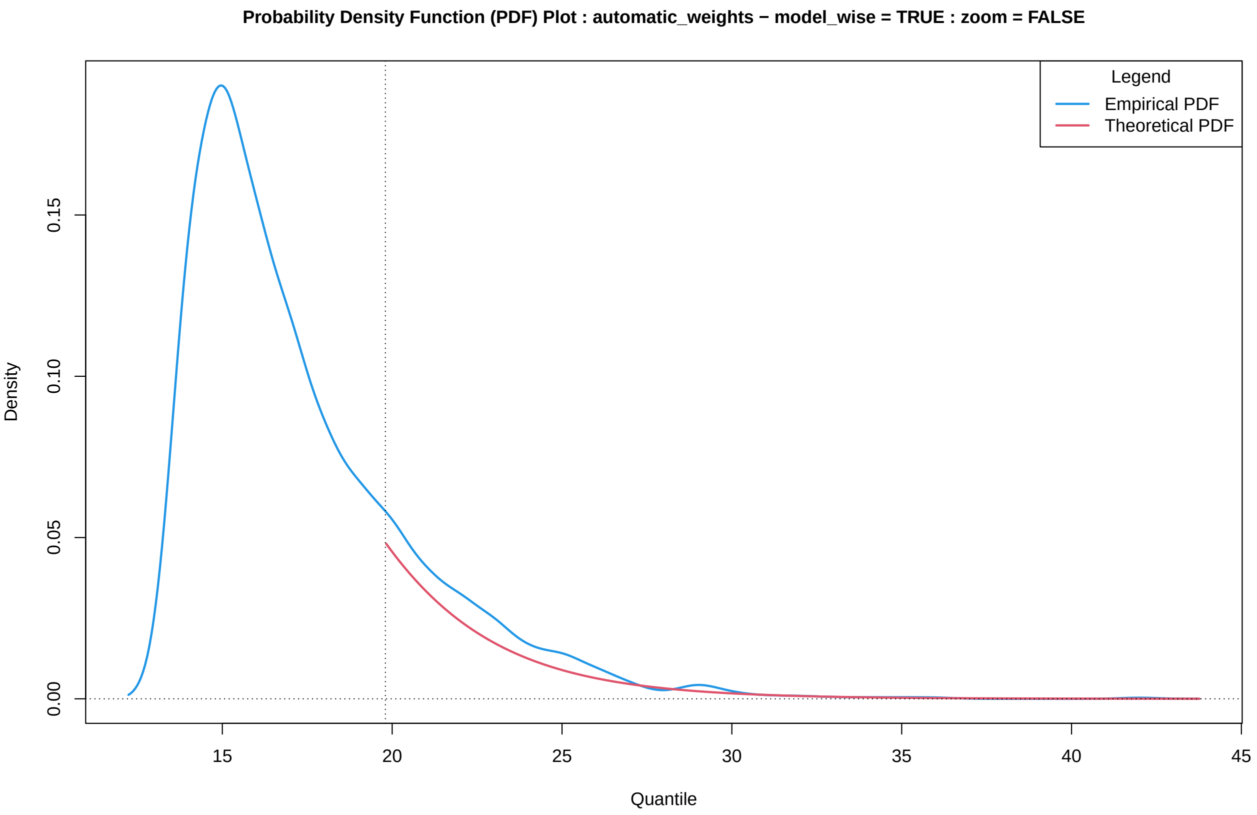


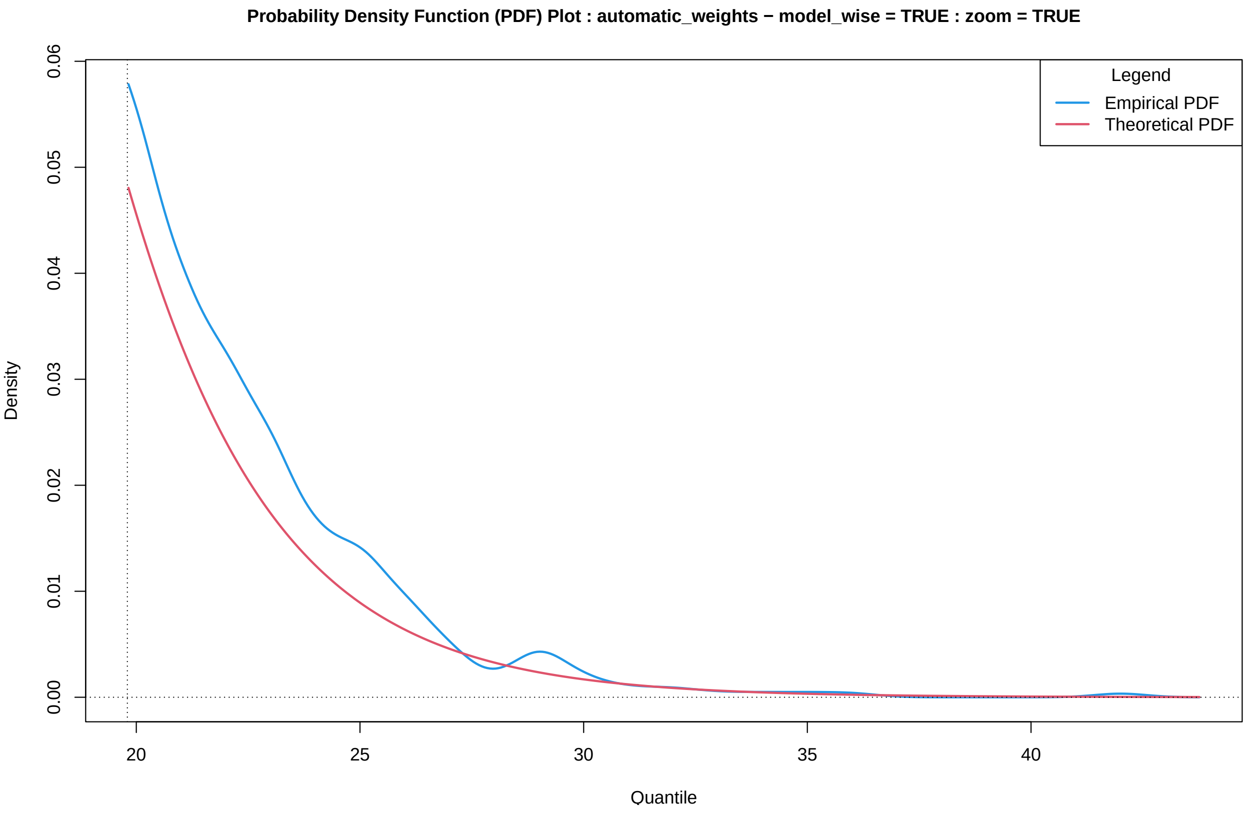
Cumulative Distribution Function (CDF) Plot : automatic_weights − model_wise = FALSE : zoom = TRUE



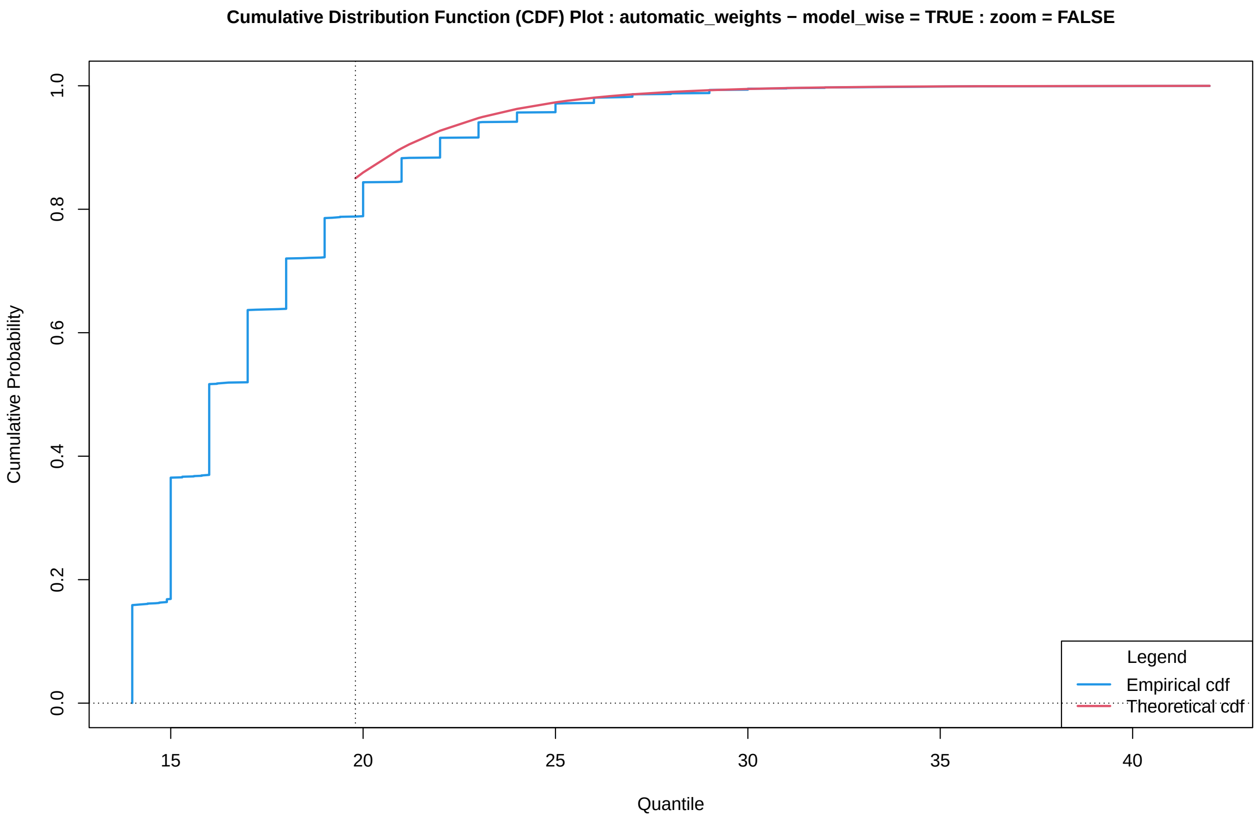
```r
## GEV mixture model with respect to distribution functions
```

```r
par(mfrow = c(2, 1))
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")
```

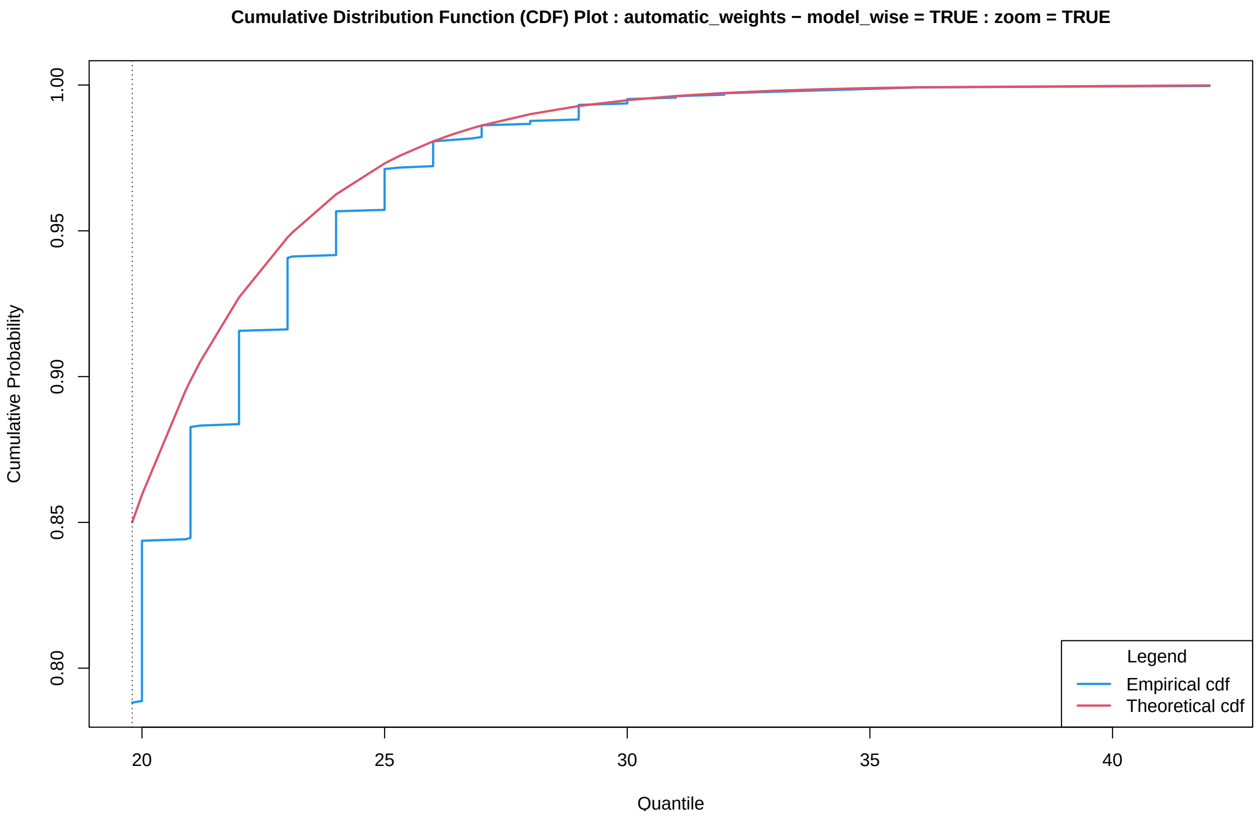

Probability Density Function (PDF) Plot : automatic_weights − model_wise = TRUE : zoom = FALSE
Legend
Empirical PDF
Theoretical PDF
Density
0.00
0.05
0.10
0.15
Quantile
15
20
25
30
35
40
45


Probability Density Function (PDF) Plot : automatic_weights − model_wise = TRUE : zoom = TRUE
Legend
Empirical PDF
Theoretical PDF
Density
0.00
0.01
0.02
0.03
0.04
0.05
0.06
Quantile
20
25
30
35
40

```r
par(mfrow = c(2, 1))
plot_gev_mixture_model_cdf(gev_mixture_model,
                           type = "automatic_weights",
                           model_wise = TRUE,
                           zoom = FALSE,
                           xlab = "Quantile",
                           ylab = "Cumulative Probability",
                           main = "Cumulative Distribution Function (CDF) Plot")

plot_gev_mixture_model_cdf(gev_mixture_model,
                           type = "automatic_weights",
                           model_wise = TRUE,
                           zoom = TRUE,
                           xlab = "Quantile",
                           ylab = "Cumulative Probability",
                           main = "Cumulative Distribution Function (CDF) Plot")
```



Cumulative Distribution Function (CDF) Plot : automatic_weights − model_wise = TRUE : zoom = FALSE



Cumulative Distribution Function (CDF) Plot : automatic_weights − model_wise = TRUE : zoom = TRUE

```
# Estimation of an extreme quantile
```

```
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)

## Quantile from GEV mixture model with respect to parameters
```

```
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[2]
```

```
##      estimate
## 1 39.98600531
```

```
## Quantile from GEV mixture model with respect to distribution functions
```

```
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[2]
```

```
##      estimate
## 1 52.90721502
```

```
## Quantiles from equivalent estimated GEV models
```

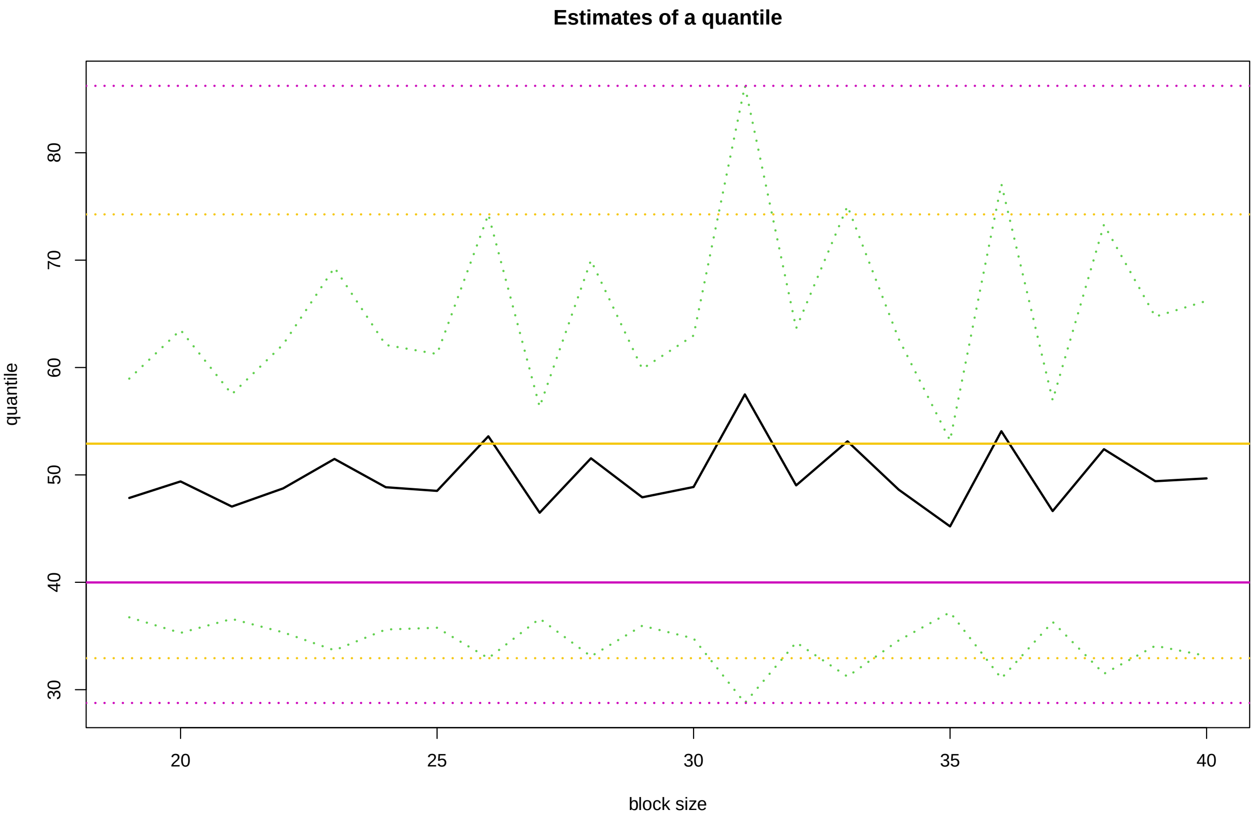
```
est_rl_pw <- estimate_gev_mixture_model_quantile(gev_mixture_model,
                                                 alpha = alpha,
                                                 confidence_level = 0.95,
                                                 do.ci = TRUE,
                           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

est_rl_pw
```

```
##          lower    estimate       upper
## 19 36.73894742 47.85236169 58.96577596
## 20 35.28553705 49.39264923 63.49976141
## 21 36.58822201 47.05089367 57.51356534
## 22 35.33878071 48.74509012 62.15139954
## 23 33.66726168 51.48624863 69.30523558
## 24 35.60339584 48.85253654 62.10167724
## 25 35.76710077 48.51228069 61.25746061
## 26 32.93288798 53.59639174 74.25989549
## 27 36.61501877 46.47590335 56.33678794
## 28 33.10367716 51.54617073 69.98866429
## 29 35.95465175 47.91375395 59.87285614
## 30 34.77118115 48.87427781 62.97737447
## 31  28.7623101 57.49608026 86.22985042
## 32 34.39767525 49.02571103 63.65374681
## 33 31.21680629 53.12656753 75.03632878
## 34 34.58978906 48.62261825 62.65544744
## 35 37.20402629 45.20376486 53.20350344
## 36 31.04940328 54.06964824 77.08989321
## 37 36.30953608 46.63150769 56.95347929
## 38 31.48043736 52.39378453  73.3071317
## 39 34.08702608 49.41292762 64.73882916
## 40 33.14324463 49.67484671 66.20644879
```

```
## Comparison of estimated quantiles
```

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

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

```
matplot(x = rownames(est_rl_pw),
        y = est_rl_pw,
        xlab = "block size",
        ylab = "quantile",
        main = "Estimates of a quantile",
        ylim = range(c(est_rl_pw_range)),
        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)
```



```
# Legend:
# yellow: Quantile from GEV mixture model with respect to distribution functions
# pink: Quantile from GEV mixture model with respect to parameters
```