

Modeling extreme values with a GEV mixture probability distributions

Standard exponential distribution

Pascal Alain Dkengne Sielenou

2023-10-28

```
# Load useful functions
```

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

```
# Generate a random sample
```

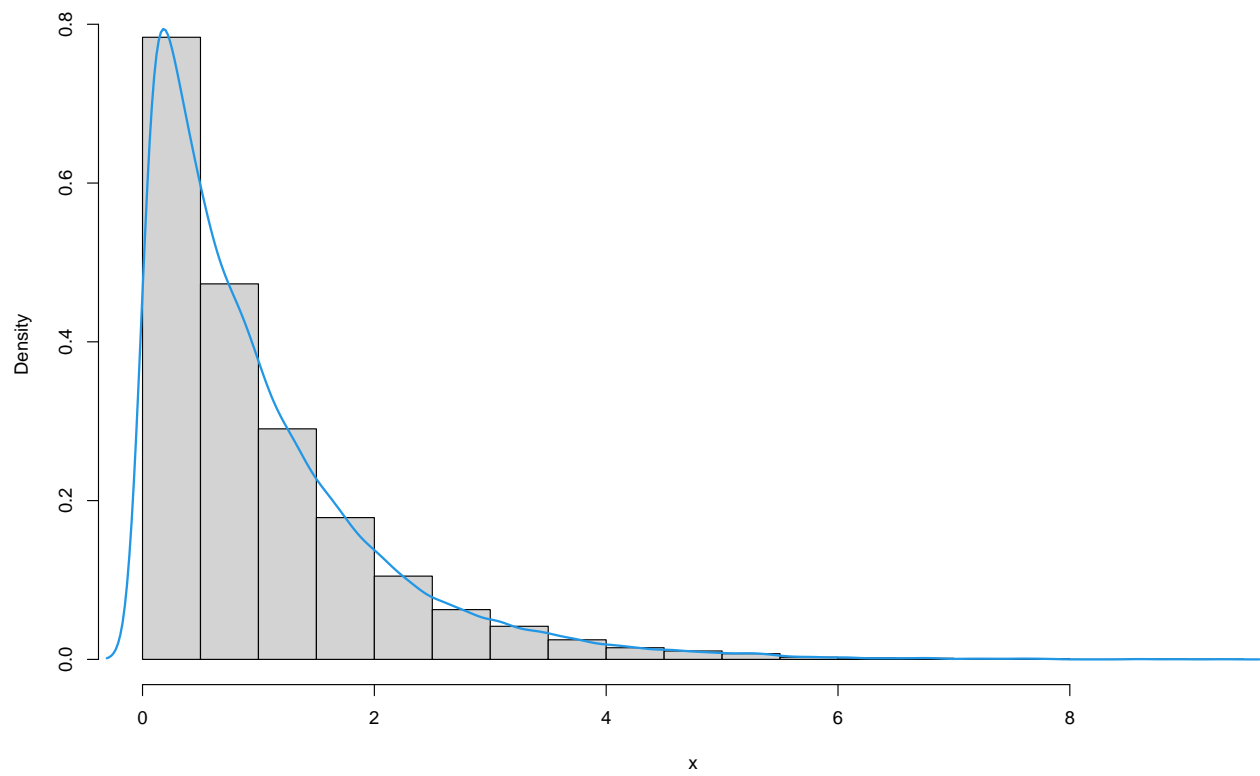
```
n <- 20000
```

```
set.seed(1122)
x <- rexp(n = n)
```

```
# Histogram of all data
```

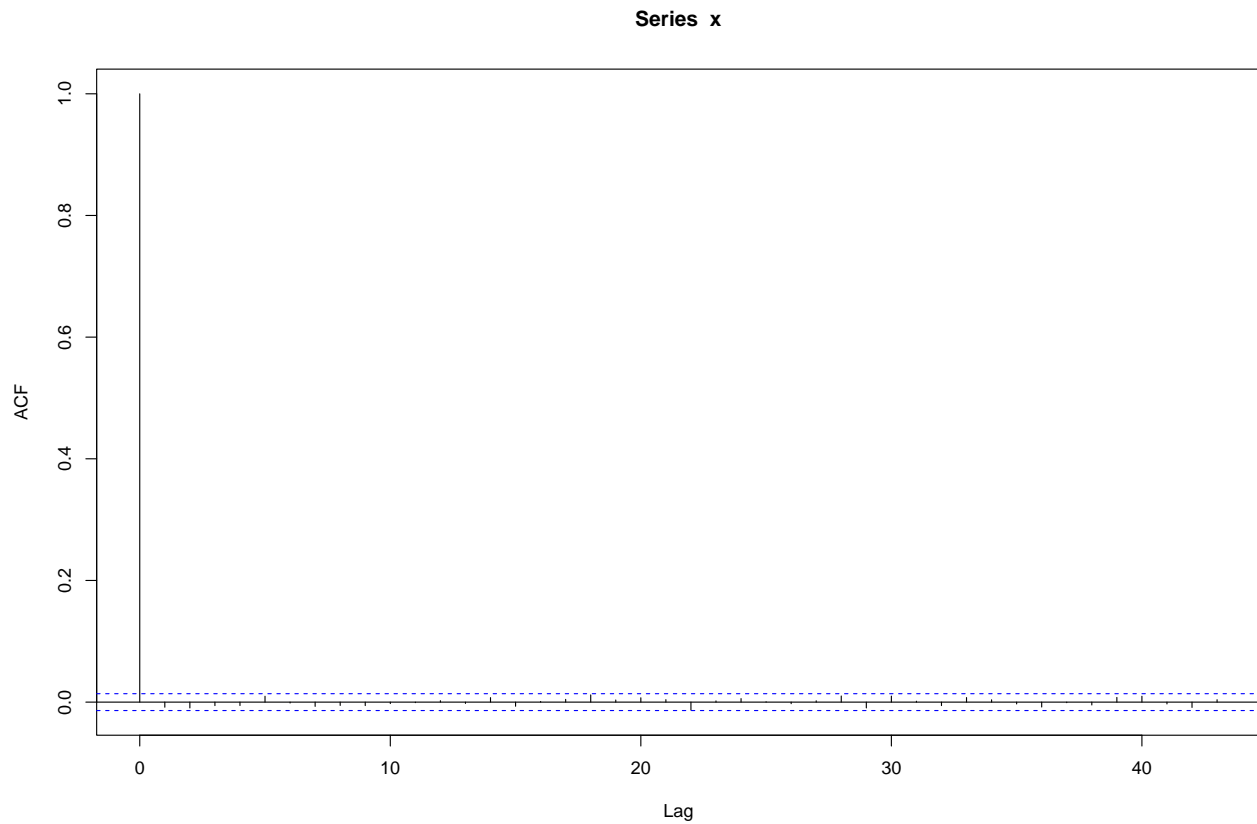
```
dens_x <- density(x)
hist(x, prob = TRUE, ylim = range(dens_x$y))
lines(dens_x, lwd = 2, col = 4)
```

Histogram of x



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

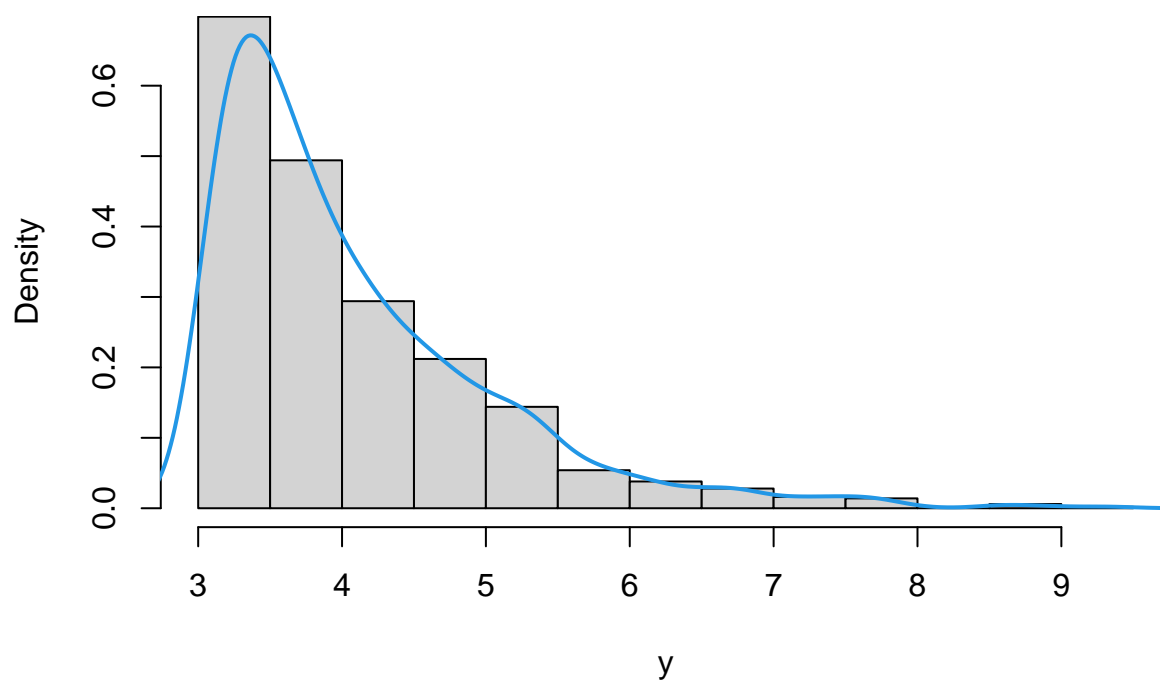
```
acf(x)
```



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

```
nlargest <- 1000  
y <- extract_nlargest_sample(x, n = nlargest)  
dens_y <- density(y)  
hist(y, prob = TRUE, ylim = range(dens_y$y))  
lines(density(y), lwd = 2, col = 4)
```

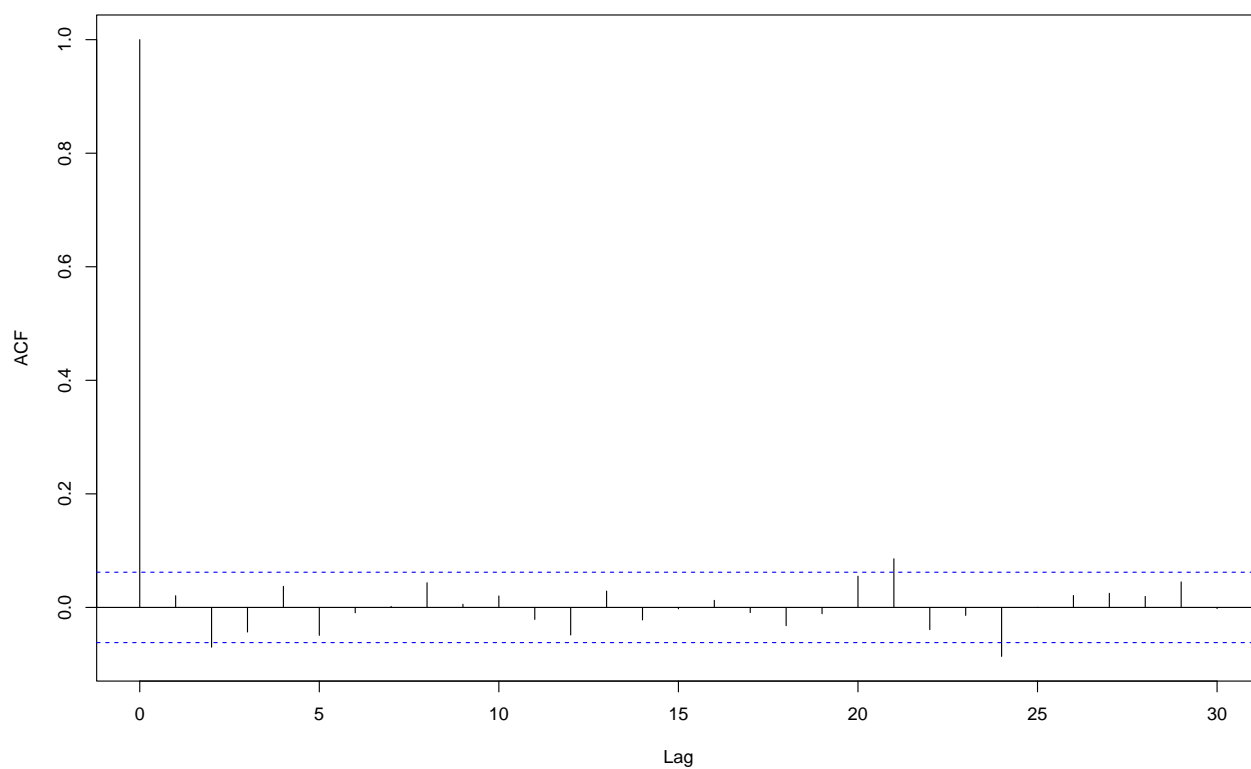
Histogram of y



Autocorrelation function of the largest data

`acf(y)`

Series 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
```

```
##          11          12          13          14          15          16
## 0.9900009242 1.0000000000 0.9609670108 1.0000000000 0.9609670108 0.9609670108
##          17          18          19          20
## 0.9609670108 1.0000000000 1.0000000000 1.0000000000
```

```
gev_mixture_model$normalized_gev_parameters_object
```

```
##      loc_star  scale_star  shape_star
## 11 3.295748201 0.9764477537 -0.036678131753
## 12 3.522773537 0.8042194762 0.023268099931
## 13 3.347347009 0.9754901572 -0.039148696475
## 14 3.377601536 0.9349268788 -0.023112178277
## 15 3.122292373 1.1696169467 -0.094710563158
## 16 2.931105537 1.1775333422 -0.080313159733
## 17 2.657091744 1.3690784947 -0.119321409565
## 18 3.752558379 0.7760365143 0.008517987581
## 19 3.878534390 0.6894152458 0.045461735709
## 20 3.578059712 0.9702756582 -0.063034629214
```

```
gev_mixture_model$full_normalized_gev_parameters_object
```

```
##      loc_star  scale_star  shape_star
## 11 3.285933676 0.9768077321 -0.036678131753
## 12 3.522773537 0.8042194762 0.023268099931
## 13 3.308477389 0.9770118521 -0.039148696475
## 14 3.377601536 0.9349268788 -0.023112178277
## 15 3.075635928 1.1740358048 -0.094710563158
## 16 2.884146774 1.1813047488 -0.080313159733
## 17 2.602451923 1.3755981952 -0.119321409565
## 18 3.752558379 0.7760365143 0.008517987581
## 19 3.878534390 0.6894152458 0.045461735709
## 20 3.578059712 0.9702756582 -0.063034629214
```

```
gev_mixture_model$automatic_weights_pw_shape
```

```
##          11          12          13          14
```

```

## 0.000000000e+00 5.003931839e-17 0.000000000e+00 -3.388131789e-21
##          15          16          17          18
## -1.355252716e-20 0.000000000e+00 0.000000000e+00 0.000000000e+00
##          19          20
## 1.000000000e+00 -6.776263578e-21

gev_mixture_model$automatic_weights_pw_scale

##          11          12          13          14          15
## 1.979257069e-05 2.106932985e-01 1.937756174e-05 1.055065246e-04 2.831812967e-04
##          16          17          18          19          20
## 2.955207653e-04 6.243595086e-04 2.829522561e-01 5.049735388e-01 3.316840595e-05

gev_mixture_model$automatic_weights_pw_loc

##          11          12          13          14          15
## 0.03820822164 0.11818422716 0.04523113754 0.06725519263 0.01222334111
##          16          17          18          19          20
## 0.02136803196 0.03469324583 0.22884552886 0.29019149249 0.14379958077

gev_mixture_model$weighted_normalized_gev_parameters_object[3, ]

##          loc_star  scale_star  shape_star
## automatic_weights 3.612603816 0.7388644418 0.04546173571

gev_mixture_model$automatic_weights_mw

##          11          12          13          14          15          16
## 0.00000000000 0.8210939617 0.00000000000 0.00000000000 0.00000000000 0.1789060383
##          17          18          19          20
## 0.00000000000 0.00000000000 0.00000000000 0.00000000000

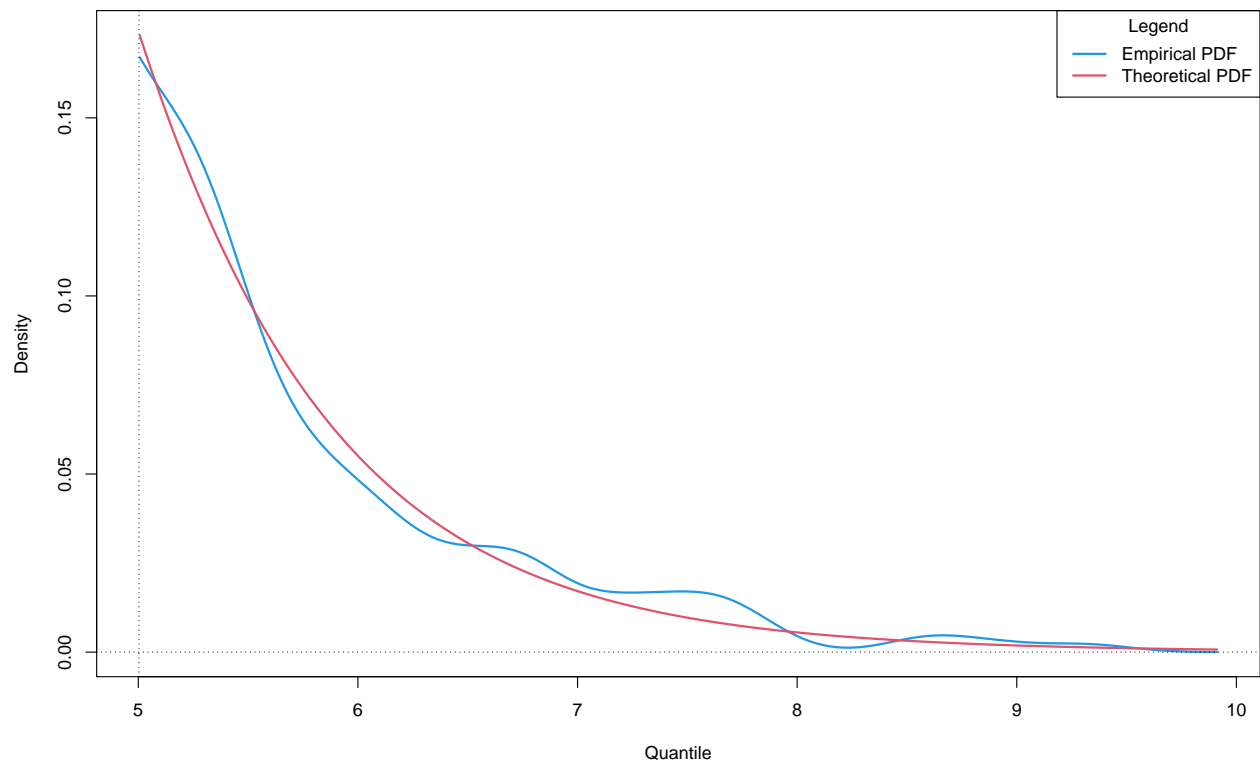
# Model diagnostics

## GEV mixture model with respect to parameters

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")

```

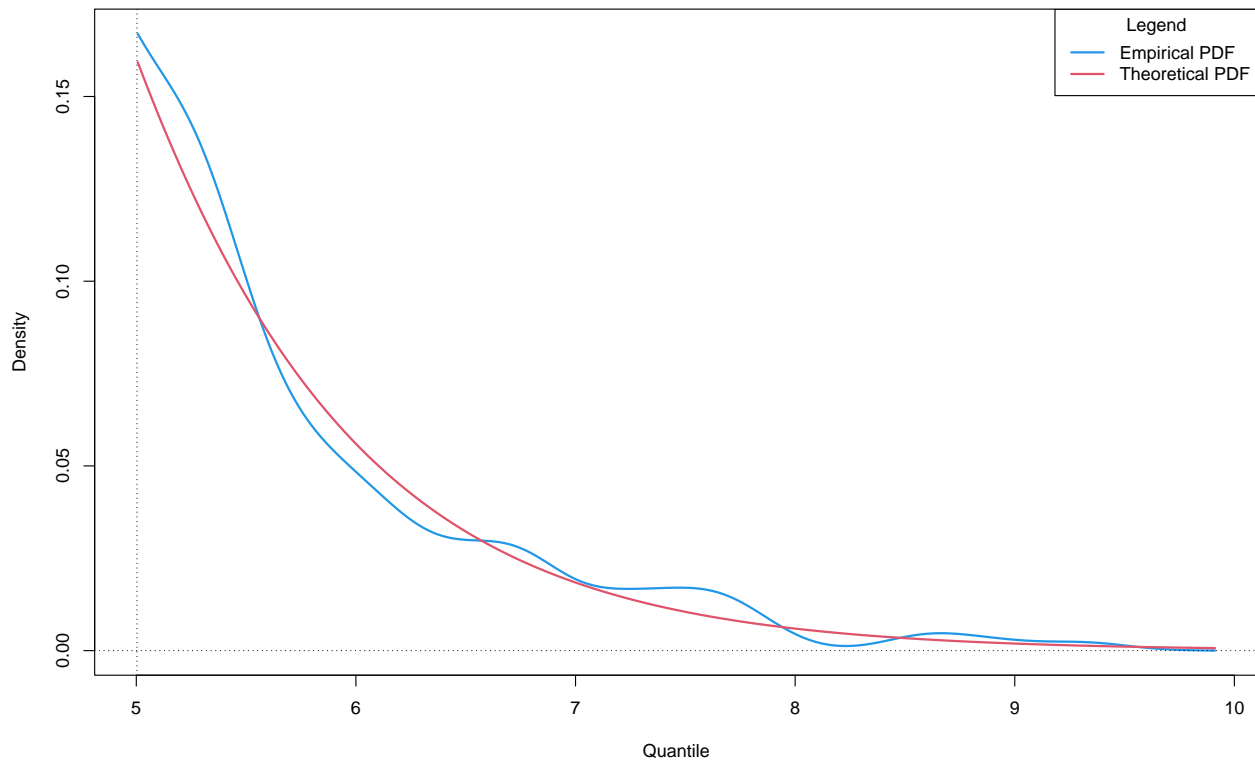
Probability Density Function (PDF) Plot : automatic_weights – model_wise = FALSE : zoom = TRUE



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

```
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 = 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",
                     "model_wise",
                     "parameter_wise",
                     "empirical")
```

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

```
## Quantile from the true distribution
```

```
true_rl <- qexp(p = 1 - alpha)
true_rl
```

```
## [1] 32.2369909
```

```
## 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
```



```
## [1] 43.69524681
```

```
## 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
```

```
## [1] 33.96224551
```

```
## Quantiles from equivalent estimated distributions in GEV mixture model with respect to parameters
```

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

```
##           lower    quantile      upper  
## 11  -9.1708649954 20.14888062 49.46862624  
## 12  -57.2416652813 34.26013771 125.76194070  
## 13  -11.8445433300 19.70857462 51.26169257  
## 14  -26.6436413027 22.32791317 71.29946764  
## 15   -0.6234629226 14.54148175 29.70642642  
## 16   -6.2762240410 15.95651806 38.18926017  
## 17   -1.0972552539 13.68961610 28.47648745  
## 18  -62.3546965548 27.64126723 117.63723101  
## 19 -123.9406858239 41.27750351 206.49569285  
## 20  -11.0333129468 16.22433668 43.48198630
```

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

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

```
## Quantiles from equivalent estimated GEV distributions in GEV mixture model respect to distribution f
```

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

```
##           lower    quantile      upper  
## 12  -57.241665281 34.26013771 125.76194070  
## 16   -6.276224041 15.95651806 38.18926017
```

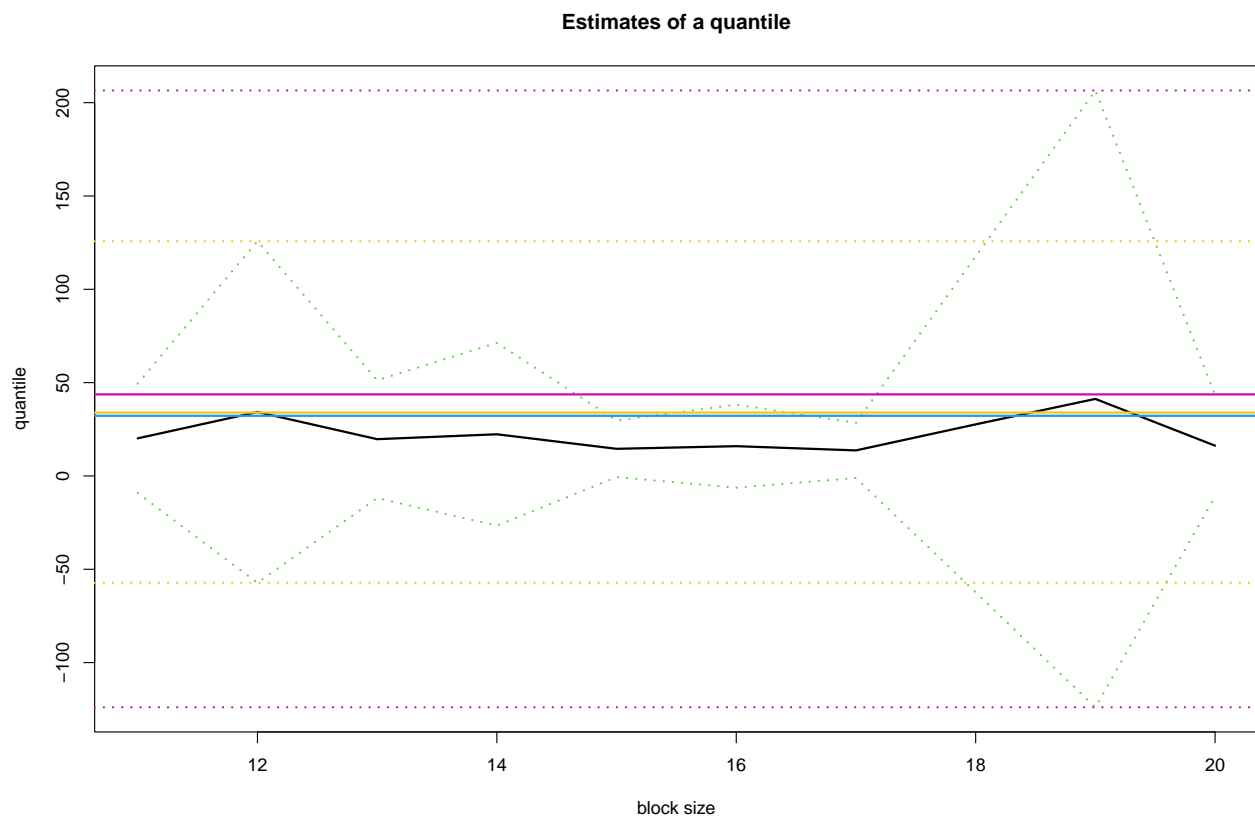
```
est_rl_mw_range <- range(as.matrix(est_rl_mw))
```

```
est_rl_mw_range
```

```
## [1] -57.24166528 125.76194070
```

```
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, true_rl, rl_pw)),  
        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 = true_rl, col = 4, lwd = 2)  
abline(h = rl_mw, col = 7, lwd = 2)  
abline(h = rl_pw, 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:  
# blue: Quantile from the true distribution  
# yellow: Quantile from GEV mixture model with respect to distribution functions  
# pink: Quantile from GEV mixture model with respect to parameters
```