A Unified Approach for Probability Distribution Fitting with **fitdistrplus**

M-L. Delignette-Muller[1], C. Dutang[2,3]

[1] VetAgro Sud Campus Vétérinaire - Lyon [2] ISFA - Lyon, [3] AXA GRM - Paris,

OUTLINE

MAXIMUM LIKELIHOOD ESTIMATION - BRIEF REMINDER

Assuming a sample $(X_i)_{1\leq i\leq n} \overset{i.i.d.}{\sim} X$, the likelihood

$$\mathcal{L}(\theta, x_1, \ldots, x_n) = \prod_{i=1}^{n} f_X(x_i, \theta),$$

where f_X is the generic mass probability/density function.
The MLE estimator θ_{MLE} maximizes the likelihood

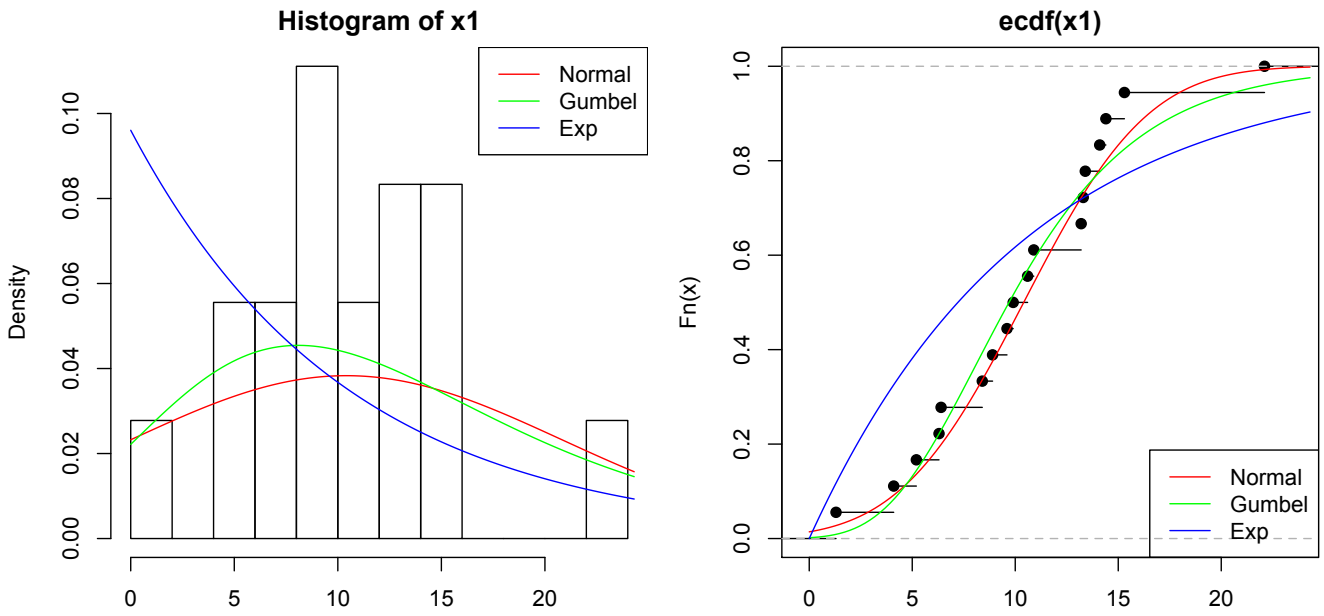
$$\theta_{\text{MLE}} = \arg\max_{\theta\in\Theta} \mathcal{L}(\theta, x_1, \ldots, x_n).$$

Example

```
> (f1 <- mledist(x1,"norm"))
$estimate
     mean        sd
10.411111  4.747033
$convergence
[1] 0
$loglik
[1] -53.57625
$hessian
          mean       sd
mean 0.7987816 0.000000
sd   0.0000000 1.597564
```

FUNCTION MLEDIST WITH NON R-BASE DISTRIBUTIONS

```
> dgumbel<-function(x,a,b) 1/b*exp((a-x)/b)*exp(-exp((a-x)/b))
> (f2 <- mledist(x1,"gumbel",start=list(a=10,b=5)))
$estimate
       a        b
8.094333 4.375401
$convergence
[1] 0
$loglik
[1] -54.09525
$hessian
           a          b
a  0.9400408 -0.4418806
b -0.4418806  1.9124424
```



OPTIONAL ARGUMENTS OF MLEDIST

Fixed arguments

```
> (f4 <- mledist(x1, "gumbel",
start=list(b=5), fix.arg=list(a=7) ))
$estimate
       b
4.248811

$convergence
[1] 0

$loglik
[1] -54.60187

$hessian
         b
b 1.710569

$optim.function
[1] "optim"

> f2
$estimate
       a        b
8.094333 4.375401
```

Custom optimization

```
> fit1 <- mledist(x1, "gamma")
>
> fit1bis <- mledist(x1, "gamma", optim.method="BFGS")
>
> #wrap genoud function
> mygenoud <- function(fn, par, ...)
+ {
+  require(rgenoud)
+ res <- genoud(fn, starting.values=par, ...)
+  standardres <- c(res, convergence=0)
+  return(standardres)
+ }
>
> #custom optimization call
> fit2 <- mledist(x1, "gamma", custom.optim=mygenoud,
+  nvars=2, Domains=cbind(c(0,0), c(10, 10)),
+  boundary.enforcement=1, print.level=0, hessian=TRUE)
>
> cbind(NelderMead=fit1$estimate, BFGS=fit1bis$estimate,
+ Genoud=fit2$estimate)
      NelderMead      BFGS    Genoud
shape  3.5747819 3.5768812 3.5742223
rate   0.3433516 0.3435683 0.3433094
```

MOMENT MATCHING ESTIMATION

It consists in equating the theoretical moments and the empirical moments

$$E\left[X^k;\theta\right] = \frac{1}{n}\sum_{i=1}^{n} X_i^k, \text{ for } k = 1,\ldots,p.$$

with $\theta \in \mathbb{R}^p$.

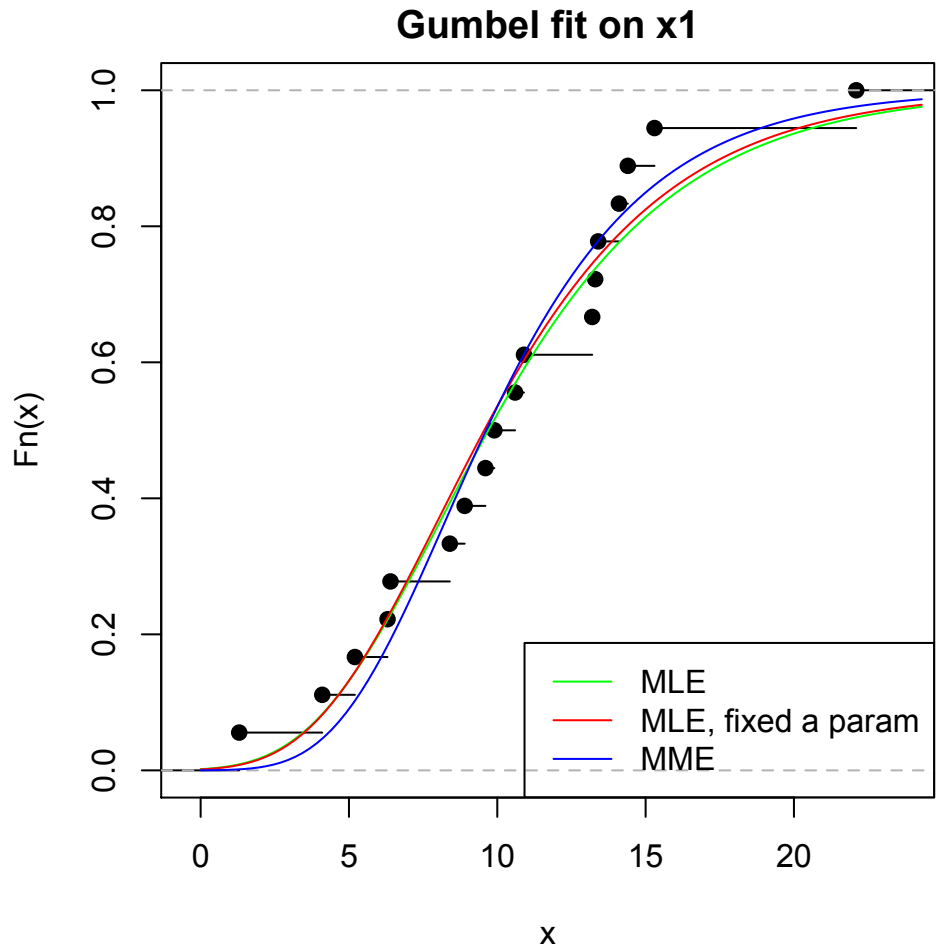
θ_{MME} can be computed in two ways, either by closed formulas (e.g. exponential distribution) or by square residual numeric minimization.

Example with closed formulas

```
> (g1 <- mmedist(x1, "norm"))
$estimate
     mean        sd
10.411111  4.747033

$convergence
[1] 0

$order
[1] 1 2

$memp
NULL

$loglik
[1] -53.57625

$method
[1] "closed formula"

> cbind(MLE=f1$estimate, MME=g1$estimate)
           MLE       MME
mean 10.411111 10.411111
sd    4.747033  4.747033
```

MMEDIST - EXAMPLE WITH NUMERICAL OPTIMIZATION

```
> #empirical raw moment
> memp <- function(x, order)
+  ifelse(order == 1, mean(x),
+ sum(x^order)/length(x))
>
> #euler constant
> euler <- 0.5772156649
>
> #theoretical raw moment
> mgumbel <- function(order, a, b)
+ {
+  mean <- a + b*euler
+  if(order == 1)
+  return(mean)
+  else
+  return(mean^2 + pi^2*b^2/6)
+ }
>
> g2 <- mmedist(x1, "gumbel", order=c(1, 2),
+ memp="memp", start=c(10, 5))
>
> cbind(MLE=f2$estimate, MLEfix=c(8,
+ f4$estimate[1]), MME=g2$estimate)
       MLE   MLEfix      MME
a 8.094333 8.000000 8.260669
b 4.375401 4.248811 3.713298
```



QUANTILE MATCHING ESTIMATION

It consists in equating the theoretical quantiles and the empirical quantiles

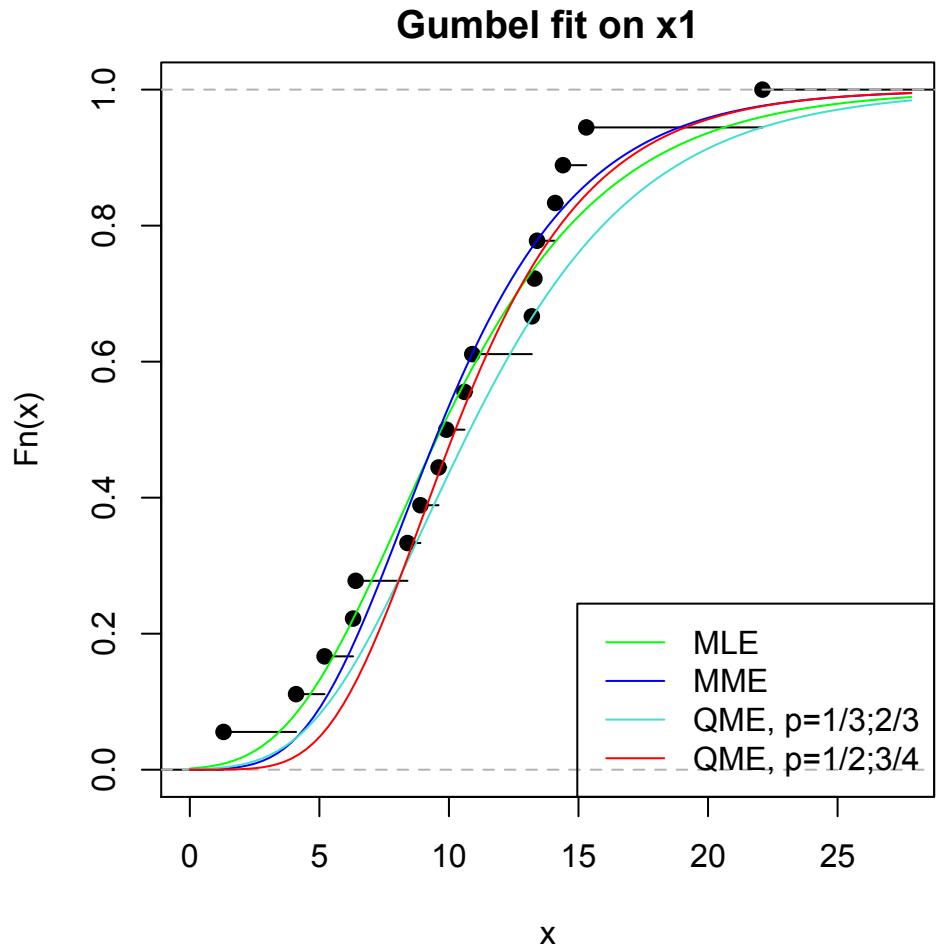
$$q_{n,p_k} = F_X^{-1}(p_k), \text{ for } k = 1, \ldots, p$$

where q_{n,p_k} is the empirical quantile and $F_X^{-1}(p_k)$ the theoretical one. p_k are given probabilities on which θ_{QME} is computed numerically.
Example with normal distribution

```
> (h1 <- qmedist(x1, "norm", prob=c(1/2, 2/3)))
$estimate
     mean        sd
10.250030  6.926297

$convergence
[1] 0

$value
[1] 2.722893e-09

$hessian
          mean        sd
mean 4.0000000 0.8614546
sd   0.8614546 0.3710520

$probs
[1] 0.5000000 0.6666667

$optim.function
[1] "optim"

$loglik
[1] -55.60913

> h1bis <- qmedist(x1, "norm", prob=c(1/3, 2/3))
>
> cbind(MLE=f1$estimate, MME=g1$estimate,
+ QME1=h1$estimate, QME2=h1bis$estimate)
           MLE       MME      QME1      QME2
mean 10.411111 10.411111 10.250030 10.983327
sd    4.747033  4.747033  6.926297  5.223794
>
```

QMEDIST EXAMPLE

```
> #empirical quantiles computed with
> #the quantile() function
>
> #theoretical quantiles
> qgumbel <- function(p, a, b)
+  a - b*log(-log(p))
>
> h2 <- qmedist(x1, "gumbel",
+ prob=c(1/3, 2/3), start=list(a=10, b=5))
> h2bis <- qmedist(x1, "gumbel",
+ prob=c(1/2, 3/4), start=list(a=10, b=5))
>
>
> cbind(MLE=f2$estimate, MME=g2$estimate,
+ QME1=h2$estimate, QME2=h2bis$estimate)
       MLE      MME      QME1     QME2
a 8.094333 8.260669 9.157968 8.947923
b 4.375401 3.713298 4.514493 3.553285
```



Maximum goodness-of-fit estimation

It consists in maximizing a goodness of fit statistics, or equivalently minimizing a distance. Generally, we use the following statistics

- Cramér-von Mises:

$$\Delta^2_{\text{CvM}} = \int_{\mathbb{R}} \left(F_n(x) - F_X(x)\right)^2 dx,$$

- Kolmogorov Smirnov:

$$\Delta^2_{\text{KS}} = \sup_x \left|F_n(x) - F_X(x)\right|,$$

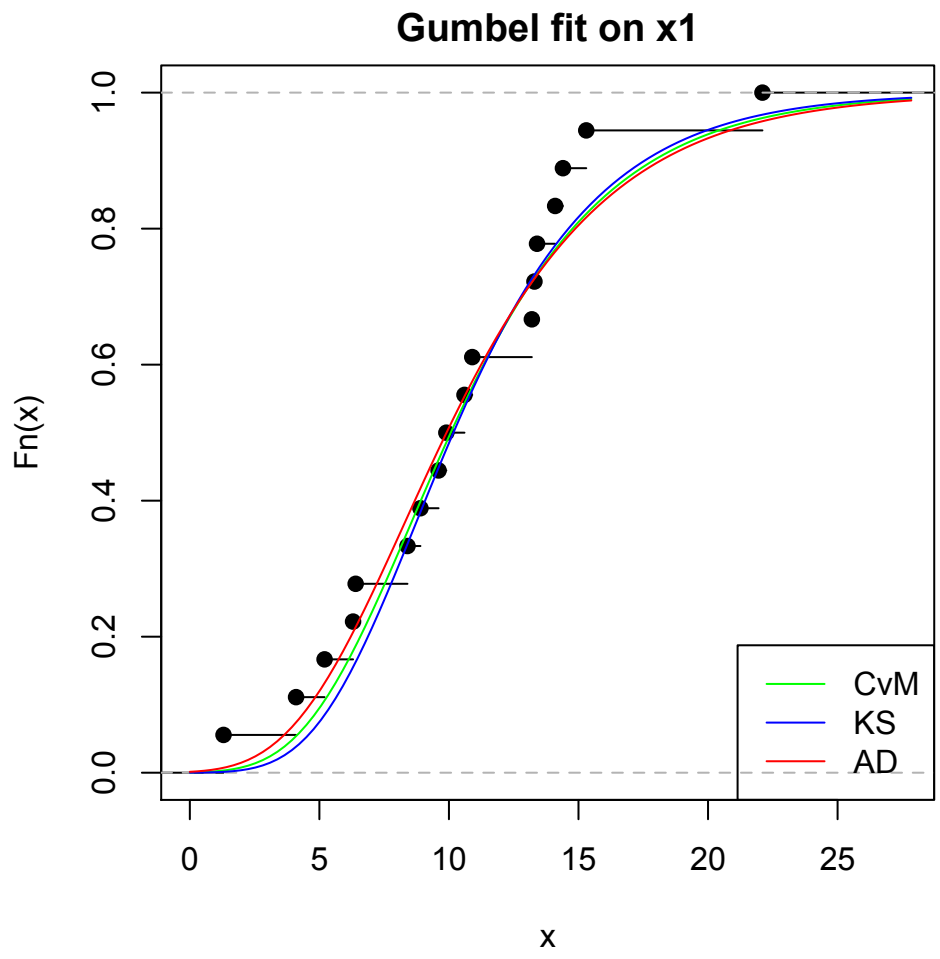
- Anderson Darling:

$$\Delta^2_{\text{AD}} = n \int_{\mathbb{R}} \frac{\left(F_n(x) - F_X(x)\right)^2}{F_X(x)(1 - F_X(x))} dx,$$

with F_n the empirical cdf and F_X the theoretical ones.

MGEDIST EXAMPLES

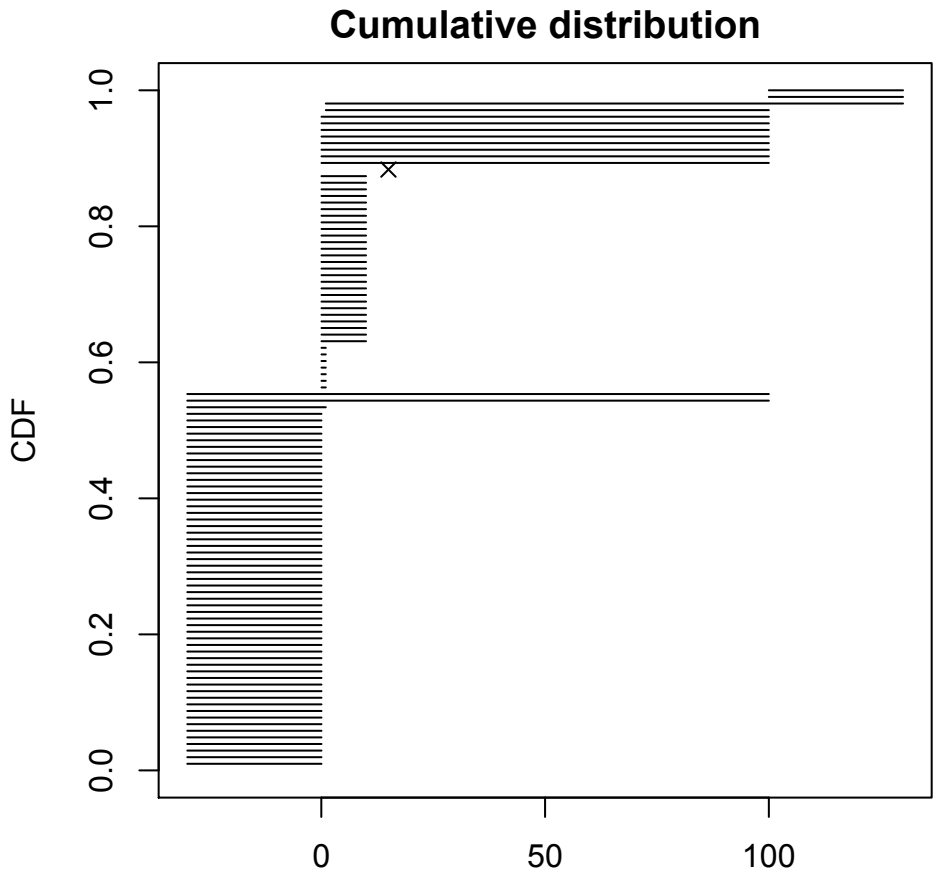
```
> i1_1 <- mgedist(x1, "norm", "CvM")
> i1_2 <- mgedist(x1, "norm", "KS")
> i1_3 <- mgedist(x1, "norm", "AD")
>
> cbind(MLE=f1$estimate, CvM= i1_1$estimate,
+ KS= i1_2$estimate, AD= i1_3$estimate)
           MLE      CvM        KS        AD
mean 10.411111 10.34687 10.643541 10.336154
sd    4.747033  4.64827  4.595217  4.763116
>
>
> i2_1 <- mgedist(x1, "gumbel", "CvM",
+ start=list(a=10, b=5))
> i2_2 <- mgedist(x1, "gumbel", "KS",
+ start=list(a=10, b=5))
> i2_3 <- mgedist(x1, "gumbel", "AD",
+ start=list(a=10, b=5))
>
> cbind(MLE=f2$estimate, CvM= i2_1$estimate,
+ KS= i2_2$estimate, AD= i2_3$estimate)
       MLE      CvM       KS       AD
a 8.094333 8.550061 8.736596 8.298506
b 4.375401 4.146123 3.916195 4.385616
>
```



CENSORED DATA

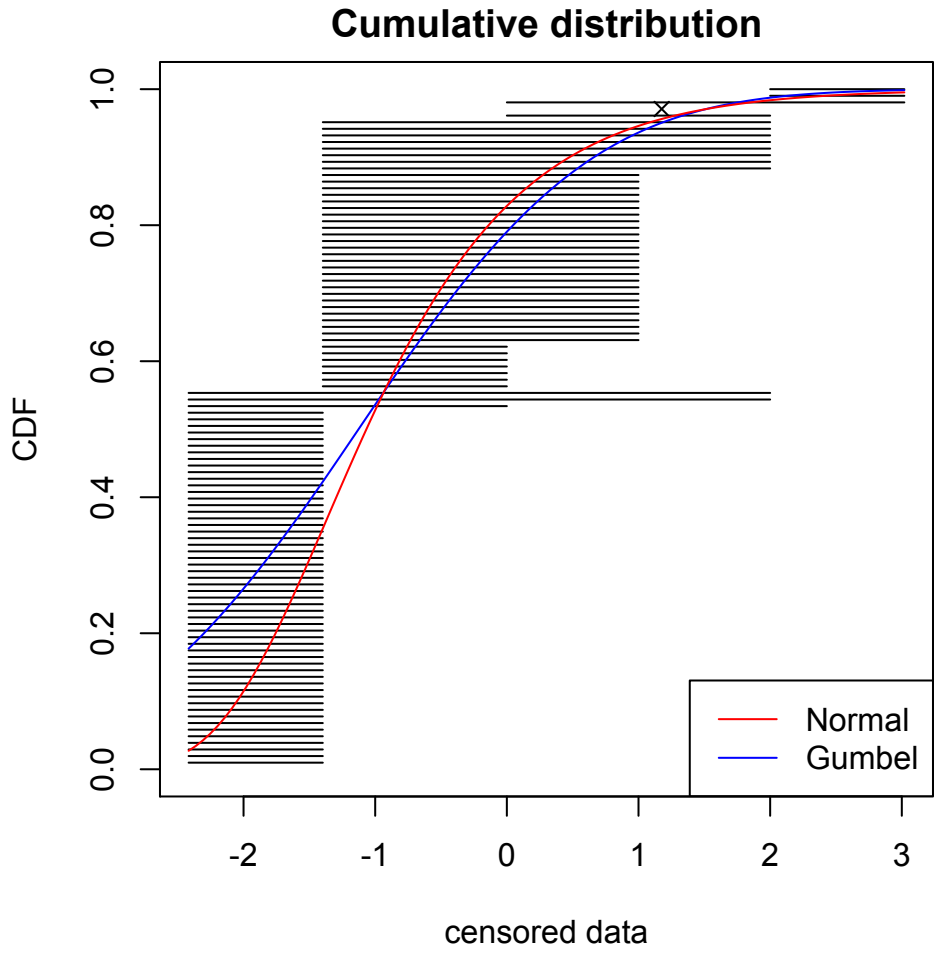
The ith obersvation x_i is not known exactly, but rather somewhere on an interval $x_i \in]l_i, u_i[$ with possible infinite bound. Non censored case is $l_i = u_i = x_i$.

```
> head(smokedfish, 10)
    left right
1     NA  0.04
2     NA  0.04
3   0.04 10.00
4   0.04 10.00
5     NA  1.00
6     NA  0.04
7   0.04 10.00
8   0.04 10.00
9     NA  0.04
10 15.00 15.00
>
> plotdistcens(smokedfish)
>
```



On the use of mledistcens

Taking into account left and/or right censoring in the (log-)likelihood, maximum likelihood estimation can be carried out.



Conclusion (1/2)

Functionalities of the **fitdistrplus** package

- MLE: Extends the **MASS** `fitdistr` function with fixed arguments, custom optimization algorithms, possible censoring,
- MME: Provides a generic function to perform moment matching estimation with the raw or centered moments,
- QME: Based on the **stats** `quantile` function, provides the quantile matching estimation,
- MGE: Maximum goodness-of-fit is now available with the usual statistical distance and their variants.

So we can fit any probability distributions.

For specific probability distributions, please look at the task view
`http://cran.r-project.org/web/views/Distributions.html`

CONCLUSION (2/2) - UNIFIED APPROACH WITH FITDIST

```
> f0 <- fitdist(x1, "gamma", method="mle")
>
> summary(f0)
Fitting of the distribution ' gamma '
by maximum likelihood
Parameters :
       estimate Std. Error
shape 3.5747819  1.1403248
rate  0.3433516  0.1175915

Loglikelihood:  -54.44954
AIC:  112.8991
BIC:  114.6798
Correlation matrix:
          shape      rate
shape 1.0000000 0.9313999
rate  0.9313999 1.0000000
>
> plot(f0, col="turquoise")
>
> descdist(x1, boot=10, boot.col="turquoise")
summary statistics
------
min:  1.3   max:  22.1
median:  10.25
mean:  10.41111
estimated sd:  4.884657
estimated skewness:  0.3433588
estimated kurtosis:  3.755991
>
```