The Empirical Cumulative Distribution Function, its Inaccuracy and Probability Plotting

Version 1.0.0, 2004-05-03

K. Uusitalo

This article intends to show, how the theory of empirical cumulative distribution function (ecdf) and order statistics can be used to draw ecdf and probability plots showing not only a point for each observation but also the inaccuracy related to the probability (or transformed probability) axis. In addition a program code written in R (R Development Core Team, 2003) for drawing these plots is provided in the appendix.

The empirical cumulative distribution function F of n i.i.d. observations can be estimated the more accurately the larger n is. Let's examine closer the rth order statistic x_r of the n observations i.e. the rth observation of an ordered sample $x_1 \le x_2 \le ... \le x_n$ and the ecdf value of that observation $F(x_r)$.

Even though $F(x_r)$ cannot be known exactly, its distribution can be derived. The following derivation is somewhat modified from Mischke (1979):
- The probability that $r-1$ sample observations are less than x is $[F(x)]^{r-1}$.
- The probability that one sample observation is x (= x_r) is $f(x)dx = dF(x)$.
- The probability that $n-r$ sample observations are greater than x is $[1-F(x)]^{n-r}$.
- The number of ways that n observations can consist of $r-1$ cases less than x, one case equal to x and $n-r$ cases greater than x is $\frac{n!}{(r-1)!1!(n-r)!}$.

=> The probability that the rth order statistic x_r lies between x and $x+dx$ is

$$g_r(x)dx = \frac{n!}{(r-1)!(n-r)!}[F(x)]^{r-1}[1-F(x)]^{n-r}\,dF = \beta_{r,n-r+1}(F)dF \quad \forall r,n \in Z_+$$

where β is the density function of the beta distribution and $\beta_{r,n-r+1}(F)$ can be used to estimate the ecdf F of the rth observation x_r. Therefore the expectation and median of the beta distribution can be used as estimates of $F(x_r)$:

$$\hat{F}(x_r) = \mathrm{E}(\beta_{r,n-r+1}(F)) = \int_0^1 F\beta_{r,n-r+1}(F)dF = \frac{r}{n+1} \text{ (expected ecdf)}$$

$$\tilde{F}(x_r) = \mathrm{md}(\beta_{r,n-r+1}(F)) \text{ (median ecdf)}$$

The latter of the two is more difficult to calculate exactly, Jacquelin(1993) shows an exact method and approximate formulas for $\tilde{F}(x_r)$. Even though some of the simple to use approximate formulas are very good, there usually isn't any need to use approximations as modern statistical software such as R is capable of calculating the exact median as well as other quantiles of the beta distribution.

Fig. 1. shows an example of ecdf and its inaccuracy as represented by the beta density function for 30 random variates generated from the N(0,1) distribution. The information can be plotted in a

similar way also in other kinds of plots which use untransformed empirical probability axis. An example of this kind of plot is the probability-probability plot where one axis shows the theoretical cdf with estimated parameters and the other the empirical cdf.

There are, however other kinds of probability plots having an axis with transformed F. If the transformed axis is $u = M(F)$, then the density $\beta_{r,n-r+1}(F)$ showing the distribution of $F(x_r)$ must also be transformed to show it correctly distributed on the transformed axis and it becomes

$$\beta_{r,n-r+1}(F)\left[\frac{dM(F)}{dF}\right]^{-1}.$$

In some kinds of probability plots the transformation is the quantile function (also called inverse cdf) of some theoretical distribution like the standard normal distribution in the normal quantile-quantile plot or the Gumbel distribution in the return level plot which is used in extreme value analysis. In the following the theoretical cdf will be denoted by $H(u)$ and its density function by $h(u)$.

If M is the inverse cdf or quantile function H^{-1} of some theoretical distribution having density function $h(u)$, then $\left[\frac{dM(F)}{dF}\right]^{-1} = \left[\frac{dH^{-1}(F)}{dF}\right]^{-1} = \frac{dH(u)}{du} = h(u)$. In this case the transformed distribution

$$\beta_{r,n-r+1}(F)h(u) = \beta_{r,n-r+1}(H(u))h(u)$$

is the distribution of the rth theoretical distribution order statistic u_r.

The expectation of the rth order statistic u_r becomes

$$\hat{u}_r = E[\beta_{r,n-r+1}(H(u))h(u)] = \int_{-\infty}^{\infty} u\beta_{r,n-r+1}(H(u))h(u)du$$

and the median is

$$\tilde{u}_r = H^{-1}[\mathrm{md}(\beta_{r,n-r+1}(F))] = H^{-1}[\tilde{F}(x_r)].$$

Sometimes also a third alternative (among many others) is used as plotting positions in probability plots having the above mentioned $u = H^{-1}(F)$ transformation: $H^{-1}[\hat{F}(x_r)] = H^{-1}(\frac{r}{n+1})$ but this isn't theoretically well grounded. However using either $\hat{u}_r$ or $\tilde{u}_r$ as plotting positions is theoretically sound. Royston (1982) gives algorithms that can be used for both approximate and exact computation of $\hat{u}_r$ in the special case that $H(u)$ an $h(u)$ are the cdf and df of the standard normal distribution respectively. Some discussion on plotting positions is given in Benard and Bos-Levenbach (1953) and in many other papers.

But as modern software allows plotting much more information than the sole plotting positions $\hat{u}_r$ or $\tilde{u}_r$ or some other alternative, the plotting positions will become only a curiosity if all the other information is plotted. Fig. 2 is an example of a normal quantile-quantile plot where the ordered data are plotted against standard normal distribution quantiles.

The return level plot (fig. 3) is described here in a little bit more detail, because it may be unfamiliar to many readers. In return level plots the ordered data (return levels) are plotted against Gumbel(0,1) distribution quantiles even though these aren't always marked on the axis as the return period is the more important information in extreme value analysis. The (average) return period is the reciprocal of the upper tail probability *1-F* or *1-H* when the probabilities are expressed per unit of time.

Coles (2001) shows, how the return level plot can be used in the context of different kinds of extreme value analyses. The traditional (pre computer era) usage is as follows. The data used is a sample of extremes, where one extreme (the largest observation) is sampled per unit of time. The extremes are plotted on the return level plot, and a straight line is fitted using the plotting positions. The fitted line can be used to extrapolate the magnitude of possible future extremes that might occur at longer return periods than the available observational period. Adding the inaccuracy information as described in this article gives much more information about the probability of extreme values and helps to see whether the largest of the observed extreme values is an outlier or not. When using sole plotting positions the largest of the extremes very often look like an outlier, but the uncertainty of the return period for the exceedance of the largest observed case is very large as seen from fig 3.

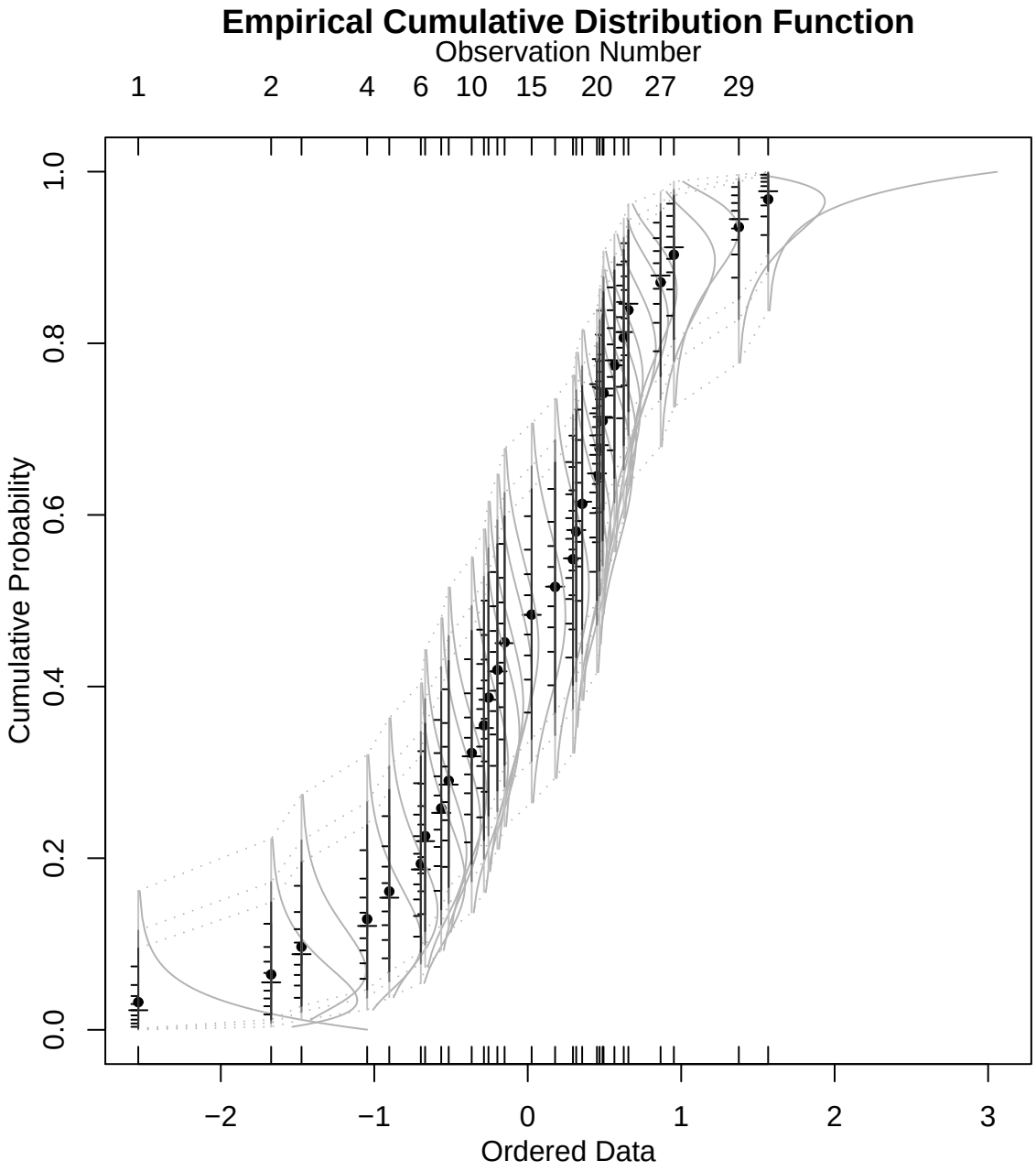

Fig. 1. The empirical distribution function: median estimates are shown by "+" signs, and expectations by dots. The median "+" signs look like longer tickmarks in the small axes representing observations, the tickmarks in the axes are the quantiles 0.1,0.2,…,0.9 of the value of $F(x_r)$, r=1,2,..,30. The dotted grey lines and the corresponding changes of the greylevels of the axes show the 0.005, 0.025, 0.05 and 0.95, 0.975, 0.995 quantiles. The grey lines drawn according to $\beta_{r,30-r+1}(F)$ show how the probability for the value of $F(x_r)$ is distributed along the F axis.

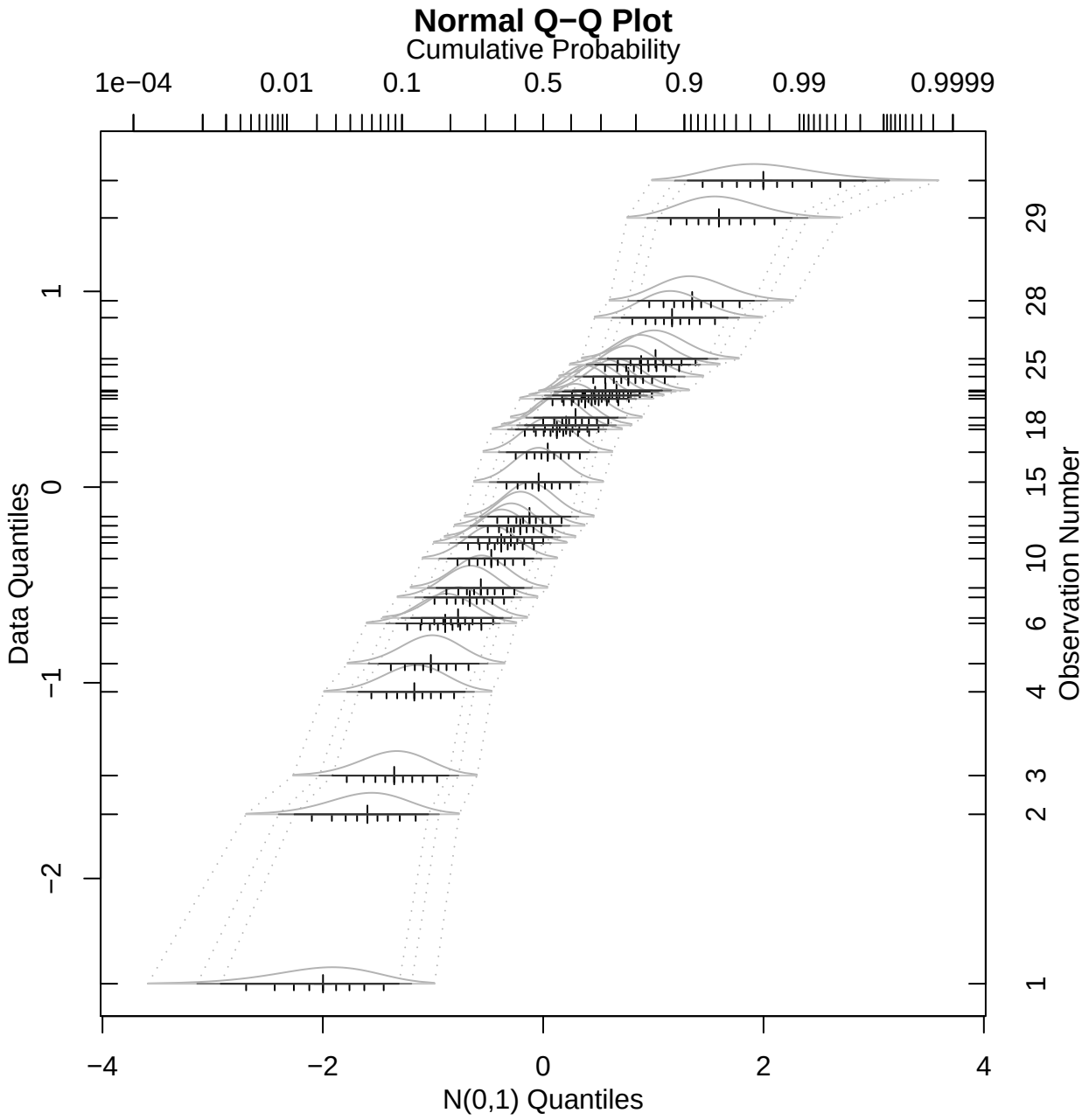

Fig. 2. Normal quantile-quantile plot showing the same 30 observations generated from the N(0,1) distribution as in fig. 1. The "+" signs are the medians of the N(0,1) order statistics.

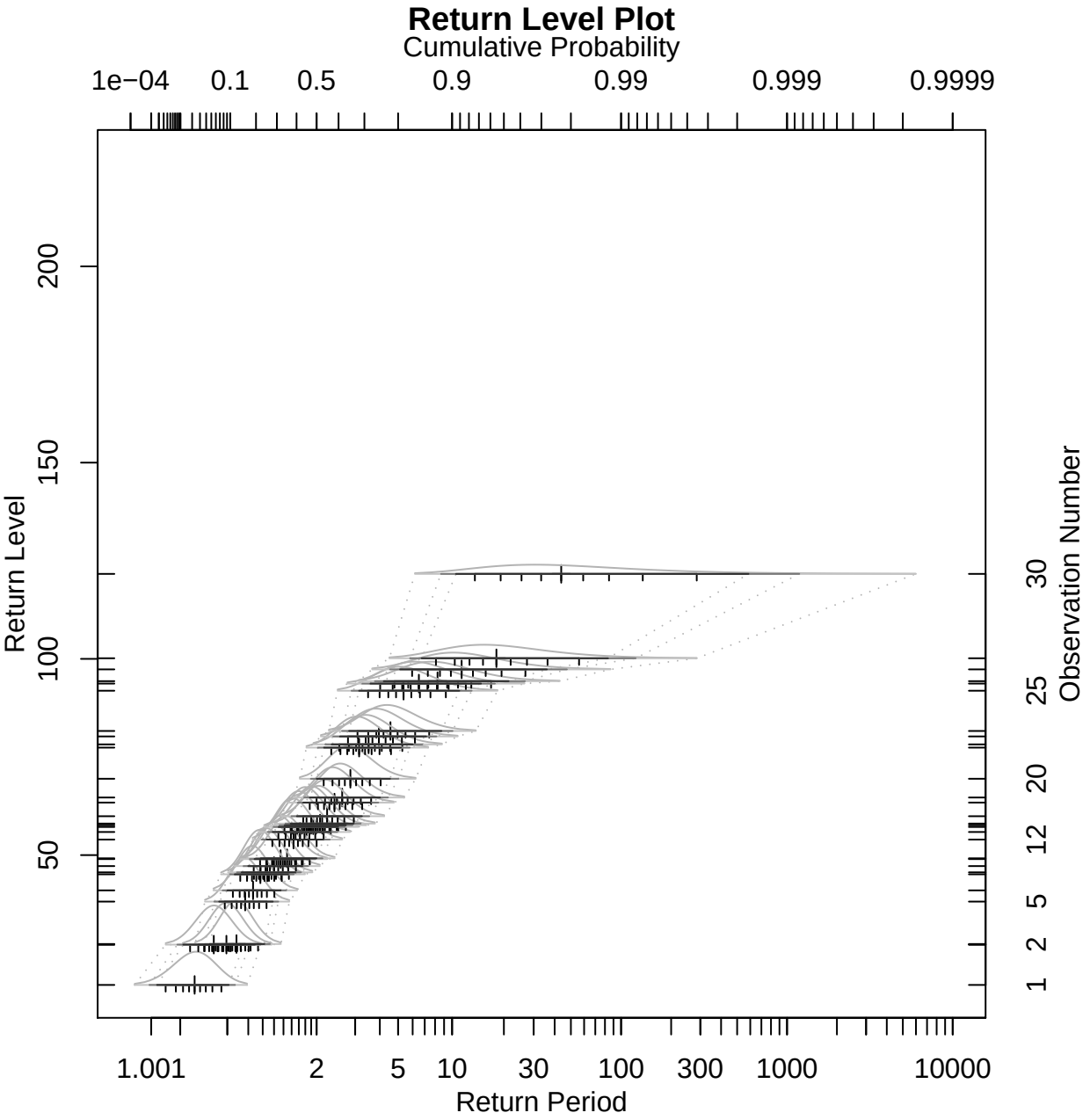

Fig. 3. Return level plot showing 30 observations generated from the Gumbel(50,20) distribution. The "+" signs are located according to the medians of the Gumbel(0,1) order statistics even though the axis is hidden and there is the return period axis instead. Extrapolations can be added by hand fitting or by using extreme value models.

References:

Benard, A. and Bos-Levenbach, E. C. (1953): Het uitzetten van waarnemingen op waarschijnlijkheids-papier. Statistica Neerlandica, Vol. 7 pp. 163-173. English translation by Schop, R. (2001): The Plotting of Observations on Probability Paper. Report SP 30 of the Statistical Department of the Mathematics Centrum, Amsterdam. http://www.barringer1.com/wa.htm

Coles, S. (2001): An Introduction to Statistical Modeling of Extreme Values. Springer Series in Statistics, Springer-Verlag, 208 pp.

Jacquelin, J. (1993): A Reliable Algorithm for the Exact Median Rank Function. IEEE Transactions on Electrical Insulation, Vol. 28, No 2, pp. 168-171.

Mischke, C. R. (1979): A Distribution-Independent Plotting Rule for Ordered Failures. An ASME publication 79-DET-112, American Society of Mechanical Engineers, 9 pp. http://www.barringer1.com/wa.htm

R Development Core Team (2003): R: A language and environment for statistical computing. R Foundation for Statistical Computing, Vienna, Austria. http://www.R-project.org/

Royston, J. P. (1982): Algorithm AS 177: Expected Normal Order Statistics (Exact and Approximate). Applied Statistics, Vol. 31, No. 2, pp.161-165.

Appendix 1. An R function for drawing ecdf and probability plots

Description:

The function prbplot can draw the empirical cumulative distribution function (ecdf) and a few kinds of probability plots (currently only the normal quantile-quantile plot and the return level (Gumbel) plot are implemented). The function prbplot is capable to draw not only a point for each observation but also the inaccuracy related to the probability (or transformed probability) axis.

Arguments:

`plottype`: the following types are available: "ecdf" (the default), "qqnorm" (normal quantile-quantile plot) and "gum" (the return level or Gumbel plot).
`main`: the main header of the plot.
`axlabels`: a vector of length four containing the labels for the axes.
`ecdflim`: a vector of length two, defining the lower and upper limit of the plotting region in terms of the cumulative probability axis. Default c(0.0001,0.9999).
`mdtype`: determines how the median plotting positions should be plotted. The same options are available as in type, see the help page for plot.
`etype`: determines how the expected ecdf plotting positions should be plotted. The same options are available as in type, see the help page for plot. Suitable only for plots with untransformed probability axis (ecdf).
`ppointstype`: determines how the ppoints based plotting positions should be plotted. See the help page for ppoints. The same options are available as in type, see the help page for plot. Probably suitable only for plots having the normal quantile function (also called inverse normal cdf) transformed probability axis (qqnorm).
`cltype`: determines, how the quantiles set by prob should be plotted, The same options are available as in type in the help page for plot.
`prob`: a vector of lower tail probabilities. The quantiles corresponding to these and the corresponding upper tail probabilities 1-prob will be drawn as determined by cltype. In addition there will be greyscale change in the axes marked for the observations. Can be used to draw confidence limits for the ecdf or the corresponding order statistic.
`oaxes`: logical. Should an axis be plotted for each observation showing the quantiles of the ecdf or corresponding order statistic? The range of shown quantiles corresponds to the probabilities min(prob),1-min(prob).
`oticks`: logical. Should the tickmarks be drawn at the 10,20,…,30 % quantiles on the axes drawn by oaxes=TRUE.
`densiscale`: a numeric scaling factor which determines the scale at which the probability densities describing the uncertainty in the ecdf or the order statistic will be drawn. A value of zero or FALSE causes no densities drawn.
`transpose`: logical. If FALSE (the default), the probability or transformed probability axis will be vertical.
…: some graphical parameters will work.

Examples:

```
#generation of random and nonrandom N(0,1) distributed data
Nrandom   <- rnorm(30)
Nnonrandom<- qnorm(qbeta(0.5,1:30,30:1))

#generation of random and nonrandom Gumbel(50,20) distributed data
Grandom   <--20*log(-log(runif(30)))+50
Gnonrandom<--20*log(-log(qbeta(0.5,1:30,30:1)))+50
```

```
#ecdf
prbplot(Nrandom,etype="p")
prbplot(Nnonrandom,etype="p")

#normal quantile-quantile plot
prbplot(Nrandom,plottype="qqnorm",densiscale=2,transpose=T)
prbplot(Nnonrandom,plottype="qqnorm",densiscale=2,transpose=T)

for(i in 1:100){
prbplot(rnorm(30),plottype="qqnorm",oaxes=F,densiscale=0,transpose=T,ylim=c(-4,4))
abline(0,1)
}

#return level plot (Gumbel plot)
#the ylim is chosen so that there is space for extrapolation in the upper part of the plot
prbplot(Grandom,plottype="gum",densiscale=5,transpose=T,ylim=c(min(Grandom),max(Grandom)+diff(range(
Grandom))))
prbplot(Gnonrandom,plottype="gum",densiscale=5,transpose=T,ylim=c(min(Gnonrandom),200))

for(i in 1:100){
prbplot(-20*log(-log(runif(30)))+50,plottype="gum",oaxes=F,densiscale=0,transpose=T,ylim=c(0,200))
abline(50,20)
}
```

The source code of the function prbplot:

```
prbplot<-function(x
,plottype="ecdf"
,main=switch(plottype
     ,ecdf="Empirical Cumulative Distribution Function"
     ,qqnorm="Normal Q-Q Plot"
     ,gum="Return Level Plot")
,axlabels=switch(plottype
     ,ecdf=c("Ordered Data","Cumulative Probability","Observation Number","")
     ,qqnorm=c("Data Quantiles","N(0,1) Quantiles","Observation Number"
               ,"Cumulative Probability")
     ,gum=c("Return Level","Return Period","Observation Number"
               ,"Cumulative Probability"))
,ecdflim=c(0.0001,0.9999)
,mdtype="p"
,etype="n"
,ppointstype="n"
,cltype="l"
,prob=c(0.005,0.025,0.05)
,oaxes=T
,oticks=T
,densiscale=1
,transpose=F
,...){

#functions for doing transposed plots
tplot <-function(x,y,tr=transpose,...){if(tr) plot(y,x,...)else plot(x,y,...)}
tlines<-function(x,y,tr=transpose,...){if(tr)lines(y,x,...)else lines(x,y,...)}
tpoints<-function(x,y,tr=transpose,...){if(tr)points(y,x,...)else points(x,y,...)}
taxis <-function(side,tr=transpose,...){
 if(tr){side=c(2,1,4,3)[side];axis(side,...)}else axis(side,...)
}
tmtext<-function(side,tr=transpose,...){
if(tr){side=c(2,1,4,3)[side];mtext(side,...)}else mtext(side,...)
}


#functions for transformations
loc<-function(p,pltype){#location (transformed positions of p)
switch(pltype
, ecdf=p
, qqnorm=qnorm(p)
, gum=-log(-log(p))
, stop("Invalid option plottype=",pltype)
)
}
dtrf<-function(x,pltype){#density transformation
switch(pltype
, ecdf=1
, qqnorm=dnorm(x)
, gum=exp(-x-exp(-x))
)
}
```

```
densiloc<-function(xasc,densiscale,n,pq,rr,pqloc,plottype){
xasc[rr]+2*densiscale*(xasc[n]-xasc[1])/(n^1.5)*dbeta(pq,rr,n-rr+1)*dtrf(pqloc,plottype)
}


#definitions of some variables
n<-length(x)  #number of observations
r<-1:n
xasc<-sort(x) #observations in ascending order
LP<-length(prob)

#define the plotting region
pq1<-qbeta(c(seq(prob[1],1-prob[1],length=100)),1,n)
pqn<-qbeta(c(seq(prob[1],1-prob[1],length=100)),n,1)
pqloc1<-loc(pq1,plottype)
pqlocn<-loc(pqn,plottype)
op<-par(mar=c(5,4,4,3)+0.1,pty="s")
tplot(#c(xasc[1],xasc[n])
c(min(densiloc(xasc,densiscale,n,pq1,1,pqloc1,plottype))
 ,max(densiloc(xasc,densiscale,n,pqn,n,pqlocn,plottype)))
 ,loc(ecdflim,plottype),type="n",axes=F,xlab="",ylab="",...)
box()

#Titles
title(main=main,line=3)
tmtext(text=axlabels,side=1:4,line=2)

#axes
if(plottype %in% c("qqnorm","gum")){
ptickval <-
c(0.0001,0.001,0.002,0.002,0.003,0.004,0.005,0.006,0.007,0.008,0.009
,0.01,0.02,0.03,0.04,0.05,0.06,0.07,0.08,0.09
,0.1,0.2,0.3,0.4,0.5,0.6,0.7,0.8
,0.9,0.91,0.92,0.93,0.94,0.95,0.96,0.97,0.98,0.99
,0.991,0.992,0.993,0.994,0.995,0.996,0.997,0.998,0.999
,.9991,.9992,.9993,.9994,.9995,.9996,.9997,.9998,.9999)
ptickval2<-c(0.0001,0.001,0.01,0.1,0.5,0.9,0.99,0.999,0.9999)
pticklоc <-loc(ptickval,plottype)
pticklоc2<-loc(ptickval2,plottype)
taxis(4, at=pticklоc, labels=F)
taxis(4, at=pticklоc2, labels=ptickval2)
}#if
if(plottype=="gum"){
Ttickval <-
c(1.001,1.01,1.1,1.2,1.3,1.4,1.5,1.6,1.7,1.8,1.9
,2,3,4,5,6,7,8,9,10,20,30,40,50,60,70,80,90
,100,200,300,400,500,600,700,800,900
,1000,2000,3000,4000,5000,6000,7000,8000,9000
,10000,20000,30000,40000,50000,60000,70000,80000,90000,100000)
Ttickval2 <- c(1.001,1.01,1.1,2,3,5
,10,30,100,300,1000,3000,10000,100000)
Ttickloc <-loc(1-1/Ttickval,plottype)
Ttickloc2<-loc(1-1/Ttickval2,plottype)
taxis(2, at=Ttickloc, labels=F)
taxis(2, at=Ttickloc2, labels=Ttickval2)}
else{taxis(2)}
taxis(1, at=x, labels=F, tcl=0.5)
taxis(1)
taxis(3, at=xasc, labels=r, tcl=0.5)

#the ecdf
p<-qbeta(0.5,r,rev(r)) #median ecdf
ploc<-loc(p,plottype)
tpoints(xasc,ploc,type=mdtype,pch=3)
#tlines(c(xasc[1],xasc[n]),c(ploc[1],ploc[n]))
p<-r/(n+1)             #expected ecdf
ploc<-loc(p,plottype)
tpoints(xasc,ploc,type=etype,pch=20)
#tlines(c(xasc[1],xasc[n]),c(ploc[1],ploc[n]))
p<-ppoints(n)
ploc<-loc(p,plottype)
tpoints(xasc,ploc,type=ppointstype,pch=1)
#tlines(c(xasc[1],xasc[n]),c(ploc[1],ploc[n]))

#a distribution for each observation
if(densiscale){
for(rr in r) {
pq<-qbeta(c(seq(prob[1],1-prob[1],length=100)),rr,n-rr+1)
pqloc<-loc(pq,plottype)
```

```
#tlines(xasc[rr]+2*densiscale*(xasc[n]-xasc[1])/(n^1.5)
# *dbeta(pq,rr,n-rr+1)*dtrf(pqloc,plottype),pqloc
# ,col=grey(0.7))
tlines(densiloc(xasc,densiscale,n,pq,rr,pqloc,plottype),pqloc
 ,col=grey(0.7))
}#for
}#if
#envelope
for(pr in c(prob,1-prob)) {
 clloc<-loc(qbeta(pr,r,rev(r)),plottype)
 tlines(xasc,clloc,col=grey(0.7),lty="dotted",type=cltype)
}#for(pr)
#a probability axis with tickmarks for each observation
if(oaxes){
if(oticks){
for(rr in r){
 tckloc<-loc(qbeta(seq(0.1,0.9,by=0.1),rr,n-rr+1),plottype)
 taxis(2,at=tckloc,pos=xasc[rr],labels=F,tcl=-0.2)
}#for(rr)
}#if(oticks)
#extensions for the axes (grey)
for(i in 1:LP) {
 for(rr in r){
  chgrloc<-loc(qbeta(c(prob[i],1-prob[i]),rr,n-rr+1),plottype)
  tlines(rep(xasc[rr],times=2),chgrloc
  ,col=grey((LP-i+0.6)/(LP+0.4)))
 }#for(rr)
}#for(i)
}#if(oaxes)
par(op)
}
```