the **Renext** package
user guide

Yves Deville

June 22, 2015, Renext version 3.0.0

Contents

Abstract

The `Renext` package has been specified by IRSN. The main goal is to implement the statistical framework known as "méthode du renouvellement". This is similar to the Peaks Over Threshold (POT) method but the distribution of the excesses over the threshold is not restricted to GPD. Data Over Threshold can be completed by historical data. Some utility functions of the package are devoted to event analysis or to graphical analysis.

Chapter 1

Introduction

This document was produced using **Renext 3.0.0**. Function calls may have changed in subsequent versions of the package. More information on the **Renext** project can found at the URL `https://gforge.irsn.fr/gf/project/renext`.

Acknowledgments

We gratefully acknowledge the BEHRIG[1] members for their major contribution to designing, documenting and testing programs or datasets: Claire-Marie Duluc, Lise Bardet, Laurent Guimier and Vincent Rebour. We also gratefully acknowledge Yann Richet who encouraged this project from its beginning and provided assistance and useful advice.

1.1 Goals

The **Renext** package has been specified and implemented by the french *Institut de Radioprotection et de Sûreté Nucléaire* (IRSN). The main goal is to implement in the R environment (R Development Core Team 2010) the statistical framework known within the community of french-speaking hydrologists as *Méthode du Renouvellement* and partly devoted to Extreme Values (EV) problems. This methodology appeared during the years 1980 and was well-accepted both by practitioners and researchers. The lack of freely available software may have limited its applicability, but this method is still in use or referred to. The book in french by Miquel (1984) is a frequently cited reference, while Parent and Bernier (2007) give a more recent presentation. Although some connections exist with the theory of Renewal Processes (Cox 1962), it must be said that the standard application of the "Renouvellement" relies on the much simpler Homogeneous Poisson Process (HPP) (Cox and Isham 1980), and is then similar to the famous Peaks Over Threshold (POT) method (Davison and Smith 1990).

POT methods are widespread and are described e.g. in the book of Coles (2001) or that of Embrecht, Klüppelberg, and Mikosch (1996). There are several nice R packages devoted to POT or extreme values: `extRemes` (Gilleland, Katz, and Young 2004), `ismev` (Heffernan and Stephenson 2012), `evd` (Stephenson 2002), `POT` Ribatet (2009), `evir` Pfaff and McNeil (2012), `evdbayes` (Stephenson and Ribatet 2008) among others. The package `nsRFA` (Viglione 2009) also contains useful functions for Extreme Values modelling.

Yet Another POT package?

- Contrary to most POT packages, the distribution of the excesses over the threshold is not in **Renext** restricted to be in the Generalised Pareto Distributions (GPD) family and can be chosen within half a dozen of classical distributions including Weibull or gamma. Though theory says that GPD will be adequate for large enough thresholds, this is not a counter indication to the use of other distributions. Fitting e.g. Weibull or gamma excesses might seem preferable to some practitioners and give good results for reasonably large return levels.

[1] IRSN *Bureau d'Expertise Hydrogéologique, Risques d'inondation et géotechnique.*

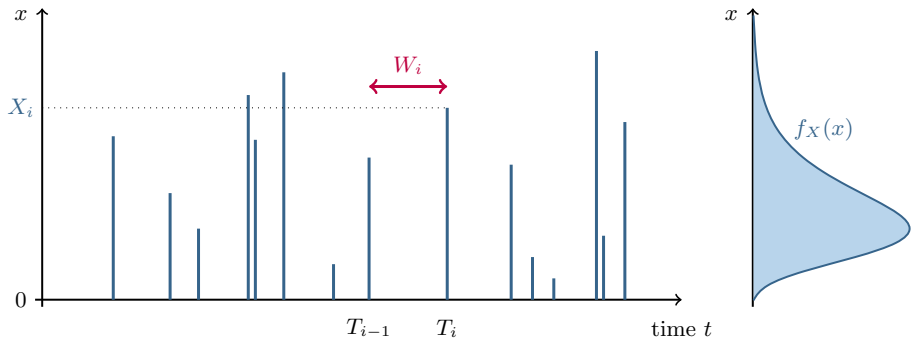

Figure 1.1: Events and levels. The random variable. $W_i = T_i - T_{i-1}$ can be called interevent.

- The package allows the use of *historical data* as explained in section 3.4. Such data can have considerable importance in practical contexts since fairly large periods can be concerned.

Unlike most R packages, **Renext** was not designed to implement innovative techniques arising from recent research in statistics but rather well accepted ones, as used by practitioners. The present document is not intended to be a manual of extreme values modelling but a presentation of the implemented tools with a limited statistical description of these.

The general framework for estimation is *Maximum Likelihood* (ML) and a black-box maximisation can be used with a quite arbitrary distribution of excesses. For the sake of generality, the inference mainly relies on the approximate *delta method*. The present version does not allow the use of covariables. The package allows extrapolation to fairly large return periods (centuries). Needless to say, such extrapolations must be handled with great care.

1.2 Context and assumptions

1.2.1 Assumptions

The general context is the modelling of a *marked point process* $[T_i, X_i]$. Events (e.g. floods) occur at successive random times T_i when a random variable "level" X_i is observed (e.g. flow). We assume that only *large* values of the level X are of interest. Thus even if the data are recorded on a regular basis (e.g. daily) the data can be soundly pruned to remove small or even moderately large values of X.

Under some general assumptions the times T_i corresponding to large enough levels X_i should be well described by an *Homogeneous Poisson Process*. Recall that for HPP events the number N of events on a time interval of length w has a Poisson distribution with mean $\mu_N = \lambda \times w$. Moreover the numbers of T_i corresponding to disjoint intervals are independent. The parameter $\lambda > 0$ is called the *rate* and has the physical dimension of an inverse time: it will generally be given in inverse years or events by year. Another important property of the HPP is that the interevent random variables $W_i = T_i - T_{i-1}$ are independent with the same exponential distribution with mean $1/\lambda$.

Unless explicitly stated otherwise, we will make the following assumptions about the marked process.

1. Events T_i occur according to a Homogeneous Poisson Process with rate λ.
2. Levels X_i form a sequence of independent identically distributed random variables with continuous distribution function $F_X(x)$, survival function $S_X(x) = 1 - F_X(x)$ and density $f_X(x)$.
3. The levels sequence and events sequence are independent.

The distribution $F_X(x)$ will be chosen within a parametric family and depends on a vector of parameters $\boldsymbol{\theta}_X$. This dependence can be enlightened using the notation $F_X(x; \boldsymbol{\theta}_X)$ when needed, the same convention applying to the density and the survival functions. The survival function can often in POT be preferred to the distribution function. The parameters of the whole model consist in λ and a vector $\boldsymbol{\theta}_X$.

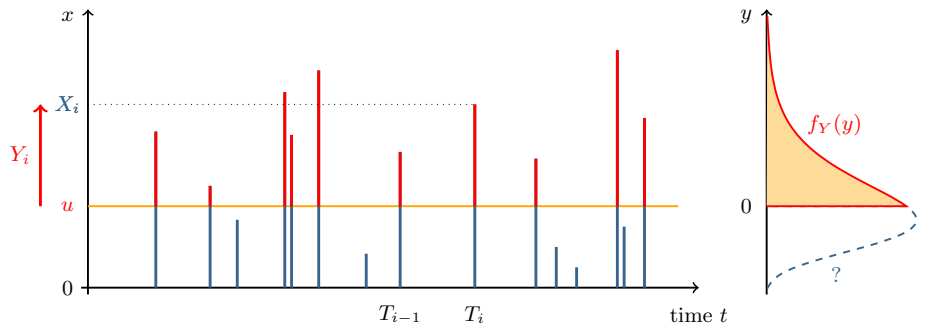

Figure 1.2: In POT, only the levels X_i with $X_i > u$ are modeled through the excesses $Y_i = X_i - u$. The lower part $x < u$ of the distribution $F_X(x)$ remains unknown.

1.2.2 Return period, return level

The *return period* of a given level x is the mean time between two events T_i with levels exceeding x, that is with $X_i > x$. Under the assumptions above, it is given by

$$T(x) = \frac{1}{\lambda\, S_X(x)}. \tag{1.1}$$

Indeed the probability of $\{X_i > x\}$ is $S_X(x)$, and the events with level exceeding x also form an HPP[2] (thinned HPP) with rate $\lambda\, S_X(x)$. The mean interevent is the inverse rate.

Conversely, for a given period $T > 0$ the *return level* $x(T)$ is the level value x having the return period T. It is given by

$$x(T) = q_X(p), \quad p := 1 - \frac{1}{\lambda T} \tag{1.2}$$

where q_X is the quantile function. The period T must be greater than $1/\lambda$. The limit of $x(T)$ for large periods is the upper end-point of the distribution, which can be finite in some cases.

In practice, the interest is often focused only on large return levels or periods.

1.2.3 Peaks Over Threshold (POT)

The POT framework

In the POT approach, only the upper part of the distribution $F_X(x)$ is modelled. More precisely, the interest is on the part $X > u$ where u is a *threshold*. The steps are

- Fix a suitable threshold u,
- Consider only the observations with level X_i greater than u i.e. with $X_i > u$,
- Estimate the rate of the events $X_i > u$ and fit a distribution for the *excesses* $Y_i = X_i - u$.

The distribution of X conditional on $X > u$ is deduced from that of the excess Y by translation.

The threshold will often be chosen above the mode of X, leading to a decreasing density for the excess Y as suggested on figure 1.2. The distribution of Y typically has two parameters.

Generalised Pareto Distribution

The POT approach usually retains a GPD for the excess Y or equivalently a GPD for the level X conditional on the exceedance $X > u$. This choice is supported for a large threshold u by the Pickands-Balkema-de Haan theorem (see B.1.3) or by the related *POT-stability* property of the GPD (see B.3.2). The family of GPDs with a given shape parameter ξ can be said to be POT-stable: if for a given threshold u the distribution of X conditional on $X > u$ is a GPD with shape ξ, then for any another threshold $v > u$ the distribution of X conditional on $X > v$ is still a GPD with the same shape ξ. By selecting a

[2]The is due to the independence of the two sequences X_i and T_i.

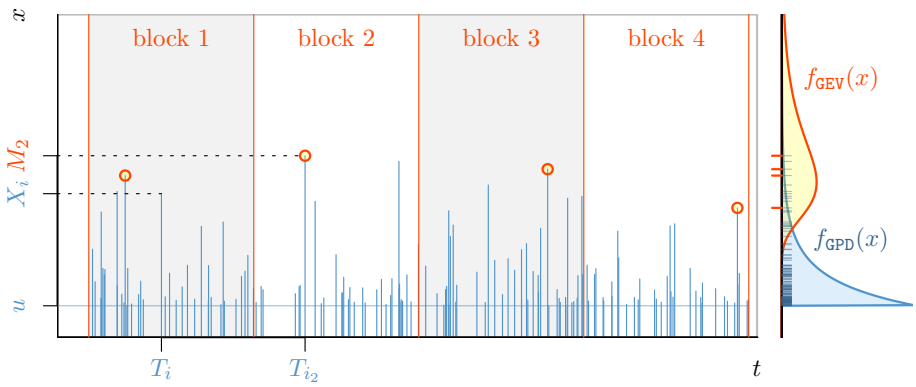

Figure 1.3: Block maxima for the marked process. If the marks X_i follow a GPD, then for a constant block duration the block maxima $M_b := X_{i_b}$ follow a GEV distribution (provided that they exist).

threshold $v > u$ in POT, the estimation will use a smaller set of X_i but the underlying distribution of X conditional on exceedance is the same in the two cases.

The choice of the threshold is a well-known difficulty in classical POT where only GPD excesses are used (Davison and Smith 1990). The situation is much more complex with non-GPD excesses, because POT stability no longer holds. For instance if the excesses over u are Weibull with shape $\alpha > 0$ and scale $\beta = 1$ i.e.

$$\Pr\{X > x \mid X > u\} = \exp\{-(x-u)^\alpha\} \qquad x > u$$

then the conditional distribution of the excess $X - v \mid X > v$ *is not* Weibull; it is a shifted version of the *Left Truncated Weibull* (LTW), see B.3.10.

1.2.4 Link with Block Maxima

Alternative approaches to POT for univariate Extreme Values modelling use time *blocks* of, say, one year and related by-block data. Numerous observations of the variable of interest X are assumed to exist in each block, and only the largest of them are retained in the analysis. Popular examples are

- **block maxima**: for each block, only the maximal value is used in the analysis.
- r **largest**: for each block the largest r observations (i.e. the r largest order statistics) are recorded. The number r may vary across the blocks.

Block maxima is the special case of r largest for $r = 1$, and using $r > 1$ largest observations when available leads to a better estimation. The r largest analysis is described in chap. 3 of the book of Coles (2001). The distribution retained for the maxima or the r largest is based on asymptotic considerations. The block maxima are usually assumed independent and to follow a Generalised Extreme Values (GEV) distribution. From the Fisher-Tippett-Gnedenko theorem, this corresponds to the situation where n independent and identically distributed X_i are found in each block, the number n being large – see section B.1.

Interestingly, the assumptions concerning the marked point process as stated before in 1.2.1 provide a framework to derive the distribution of the maxima or that of the r largest observations over non-overlapping blocks, without any asymptotic consideration. Given B such blocks $b = 1, 2, \ldots, B$ with known duration w_b, the maximum M_b for block b is the maximum of N_b levels X_i with T_i falling in that block, where N_b has a Poisson distribution with mean λw_b. Moreover the maxima M_b are independent across blocks. The distribution of the M_b can be related to the distribution of the marks: see appendix page 50. Note however that the marked process can lead to blocks b with no observation, especially when the block duration w_b is not large relative to the mean interevent $1/\lambda$. Similarly, the joint distribution of the r largest in a block is easily derived, see section 3.4 later.

When the distribution of the marks X_i is assumed to be GPD and the blocks have the same duration w_1 (e.g. one year), the block maxima M_b are independent and follow a GEV distribution, as is usually

assumed for block maxima. It can be shown as well that the distribution of the r largest observations X_i is then the distribution used in the r largest analysis Coles (Coles 2001, chap. 3), provided that at least r observations X_i exist. The notion of *return period* for the blocks framework differs from the one given above see discussion A.3 page 51. However, the difference between the two notions is confined to the small return periods context.

To summarise: maxima or r largest observations can be viewed as *partial observations* of the marked process, or as the result of a *temporal aggregation* of this process. When the result of such an aggregation (i.e. maxima or r largest) is known for one or several blocks with large duration, say decades or centuries, we may speak of *historical data*.

Although **Renext** primarily uses the original data $[T_i, X_i]$ as described in 1.2.1 above, it is possible to make use of supplementary block data in a quite flexible fashion. Maxima and r largest observations within block(s) can also be used, as well as the marks exceeding some known auxiliary threshold as sometimes called a *perception threshold*. A typical use of these possibilities is for historical data.

1.2.5 Declustering

Most of Extreme Values problems concern a continuous time process: discharge flow, temperature, sea surge, etc. POT modelling most often requires a *declustering* step leading to independent events: floods, heat or cold waves, storms, etc. **Renext** does not currently provide any declustering function, which can be found in the other POT packages cited above.

1.3 Heterogeneous data

1.3.1 Remarks

Model fitting functions in R usually have a formal argument specifying data with a *data frame* object, the model being typically given by a *formula*. Due to the presence of heterogeneous types of data within a given "dataset", the arguments of **Renext** functions will take a slightly more complex form. For instance, it will generally be necessary to specify a duration or several block durations in complement to the vector of levels, to specify where missing periods (gaps) occurred, etc.

Some of the package functions require the use of `POSIX` objects representing date and time. R base package provides versatile functions to manage date/time or timestamps. See for instance the help of the `strptime` function.

As most R packages do, **Renext** comes with a few datasets taken from relevant literature or from real data examples. These datasets are given as lists objects with hopefully understandable element names. Some datasets have an S3 class named `"Rendata"` and can as such be used as the first formal argument of popular S3 methods: `plot`, `summary` and more.

1.3.2 OT data

The data used in POT will mainly consist in recorded levels X_i or levels exceeding a reasonably low known threshold $u_\star$, with possibly $u_\star = -\infty$. Such data will be called *OT data* for "Over Threshold". The POT modelling will typically use a higher suitably chosen threshold $u > u_\star$.

The data `Brest` contain sea surge heights at high tide for the Brest gauging station. Only values exceeding $u_\star = 30\,\text{cm}$ are retained. More details about these data are provided in the package manual. The data are provided as a list with several parts.

```
> library(Renext)
> names(Brest)
[1] "info"      "describe"  "OTinfo"    "OTdata"    "OTmissing"
```

As their names may suggest the list elements contain Over Threshold (OT) data and information.

```
> head(Brest$OTdata, n = 4)
```

```
        date  Surge comment
1 1846-01-14 35.989
2 1846-01-21 59.987
3 1846-01-24 45.986
4 1846-01-28 39.985
```

```
> str(Brest$OTinfo)
```

```
List of 4
 $ start      : POSIXct[1:1], format: "1846-01-01"
 $ end        : POSIXct[1:1], format: "2009-01-01"
 $ effDuration: num 148
 $ threshold  : num 30
```

The `OTdata` element is a data frame giving the T_i (in time order) and the corresponding levels X_i. Note that the time part of the `POSIX` object may not be relevant. Here only the date part makes sense and the time part is by convention `"00:00"` with the time zone set to `"GMT"` to use Coordinated Universal Time (UTC). Of course, the observations were made at a different time.

The `OTinfo` list mentions an *effective duration*. This is less than the time range which can be computed using the methods `range` and `diff` from the **base** package

```
> End <- Brest$OTinfo$end; Start <- Brest$OTinfo$start
> Dur <- as.numeric(difftime(End,  Start, units = "days"))/365.25
> Dur
```

```
[1] 162.9979
```

```
> Dur - as.numeric(Brest$OTinfo$effDuration)
```

```
[1] 15.37795
```

The difference – more than 15 years – is due to gaps or *missing periods*. The missing periods are described in the element `OTmissing`.

The `Brest` dataset has class `"Rendata"`. This is an S3 class defined in **Renext** to describe objects containing `OTdata` and possibly some extra information on missing periods or historical data. It has a `summary` method

```
> class(Brest)
```

```
[1] "Rendata"
```

```
> summary(Brest)
```

```
o Dataset Surge Heights at Brest (France)
   data 'Brest', variable 'Surge' (cm)

o OT data (main sample) from  1846-01-01  to  2009-01-01  (eff. dur. 147.62 years)

      n    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
1289.00   30.02   33.65   38.31   41.76   46.58  143.90

o missing 'OT' periods, total 15.38 years

        n      Min.   1st Qu.    Median      Mean   3rd Qu.      Max.
43.000000  0.002738  0.016430  0.038330  0.357600  0.086240  8.419000

o no 'MAX' historical data

o no 'OTS' historical data
```

The displayed information concerns the levels in the main OT sample and the possible gaps in this sample: number, duration (in years). A `plot` method also exists

```
> plot(Brest)
```

which produces the plot on the left of figure 1.4.

1.3.3 Missing periods or gaps

A common problem in POT modelling is the existence of gaps within the observation period. These can result from many causes: damage or failure of the measurement system, human errors, strikes, wars, ...

Renext uses a natural description of the gaps within a dataset. They are stored as rows of a data.frame with two `POSIX` columns `start` and `end`

```
> head(Brest$OTmissing, n = 4)
```

```
       start        end comment
1 1846-01-01 1846-01-04
2 1847-01-01 1847-01-21
3 1852-01-21 1852-02-08
4 1857-05-31 1859-11-24
```

Missing periods must be taken into account in the analysis. They should be displayed on timeplots showing events, since it is important to make a distinction between periods with no events and gaps, see figure 1.4. An important prerequisite to modelling is to ensure that the gaps occur independently from measured variables. For instance, storms can damage gauging systems for wind or sea level thus leading to *endogenously missing* observations forming an endogenous gap. This may be considered as a form of censoring.

1.3.4 Aggregated (block) data

Motivation

In a `Rendata` object, the ordinary data provided in the `OTdata` element can be completed by some data observed in blocks with known duration. This possibility is often required to use historical information. Two types of block data are currently supported under the names "MAX" and "OTS" data. These can be regared as the two types of censored data: type I for OTS and type II for MAX, and are described more precisely in section 3.4.1 page 25.

MAXdata

As a first possible complement to `OTdata`, we may have `MAXdata` that is: r largest observations over one or several blocks. Such data require a complementary information: the block duration(s) which must be given in years.

The dataset `Garonne` is taken from Miquel (1984) where it is described. The data concern the french river *La Garonne* at the gauging station named *Le Mas d'Agenais* where many floods occurred during the past centuries. The data consist in both OT data and historical data. The variable is the river discharge flow in m^3/s as estimated from the river level using a rating curve. The precision is limited and many ties are present among the flow values. The OT data contain flow values over the threshold $u = 2500$ m^3/s.

The historical data in `Garonne` are simply the 12 largest flows for a period of about 143 years and will be used later.

```
> names(Garonne)
```

```
[1] "info"      "describe"  "OTinfo"    "OTdata"    "OTmissing" "MAXinfo"
[7] "MAXdata"
```

```
> Garonne$MAXinfo
```

```
       start        end duration
1 1770-01-01 1913-01-01   143.09
```

```
> head(Garonne$MAXdata, n = 4)
```

```
  block date Flow  comment
1     1 <NA> 7500 1 (1875)
2     1 <NA> 7400 2 (1770)
3     1 <NA> 7000 3 (1783)
4     1 <NA> 7000 4 (1855)
```

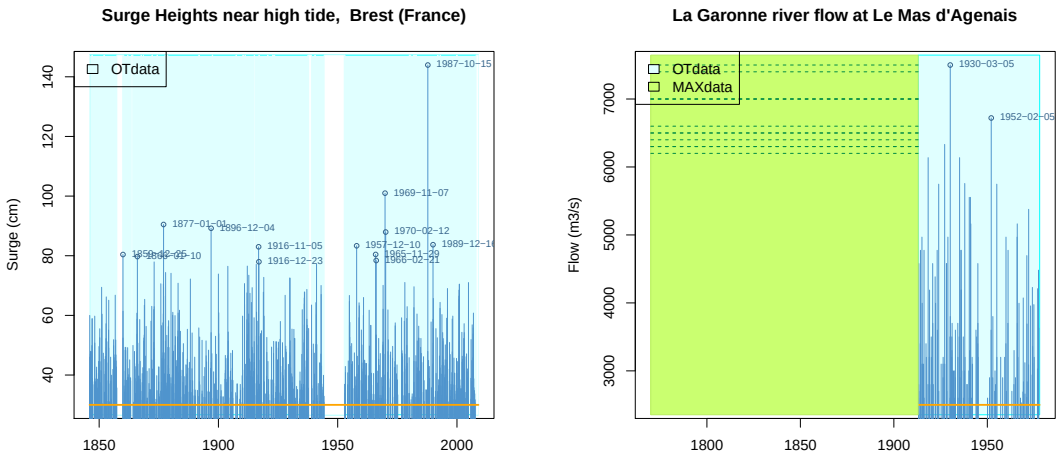

Figure 1.4: Graphics produced using the `plot` method of the `"Rendata"` class. On the left, the `Brest` object contains missing periods that are shown. On the right, the `Garonne` dataset contains information about an *historical period*, displayed as a green rectangle.

The `Garonne` dataset has class `"Rendata"`. The `plot` method for this class

```
> plot(Garonne)
```

produces a graphic displaying the historical period as on the right panel of figure 1.4. Here the dates of the historical events are not known exactly and thus are provided here as `NA POSIXct` objects. The historical levels are thus displayed as horizontal segments, while vertical segments would be used for known dates. The `plot` method for the class `Rendata` has a `showHist` logical formal argument telling that historical periods should be shown (default value `TRUE`) or not.

Note that the function `OT2MAX` can be used to compute the r largest values in blocks of one year from observations $[T_i, X_i]$ of a marked process. This function can be used to compare a POT approach to block maxima or r largest, see 4.3.

It can be remarked here that working with the original unit leads to observations with a quite large order of magnitude. This can be a problem in some numerical evaluations such as the determination of a hessian. Although the `Renouv` function used later internally scales the data, it could be preferable to rescale the data e.g. by dividing them by 1000.

OTSdata

The other type of block data involves a number of B non-overlapping blocks. For each block $b = 1, 2, \ldots, B$ the duration w_b is assumed to be known as well as a threshold u_b. We then assume to be given all the observations with level X_i exceeding the threshold u_b i.e. with $X_i > u_b$. It is assumed that no OTS threshold u_b is smaller than the OT threshold u. In some cases the times T_i are known and can be provided in the `date` column of the data frame `OTSdata`. Unlike with `MAXdata`, one block can be empty because no level X_i exceeding u_b was found. The block will then appear in the `OTSinfo` data frame but not in `OTSdata`.

1.3.5 Overview

The general structure of a `Rendata` object is described in table 1.1.

The `readXML` function (still experimental) can be used to read such heterogeneous data from an XML file and possibly linking to csv files. Some examples are shipped with the package, see help with `?readXML`.

element	class	content
`info` ($\star$)	`list`	general information: variable name, units, ...
`describe`	`character`	optional description
`OTinfo` ($\star$)	`list`	`start`, `end`, `duration` w, `threshold` u
`OTdata` ($\star$)	`data frame`	`date` T_i, *level* X_i and `comment`
`OTMissing`	`data frame`	`start`, `end`, `comment`
`MAXinfo`	`data frame`	`start`, `end`, `duration` w_b
`MAXdata`	`data frame`	`block`, `date`, *level*, `comment`
`OTSinfo`	`data frame`	`start`, `end`, `duration` w_b, `threshold` u_b
`OTSdata`	`data frame`	`block`, `date`, *level*, `comment`

Table 1.1: Structure of a `Rendata` object. The required elements are marked with a star ($\star$). The threshold in `OTinfo` can be set to `-Inf`, thus allowing the computations of the excesses $X - u$ for any threshold u.

1.3.6 Simulating heterogeneous data

Heterogeneous data generated by a Monte-Carlo simulation are of great help in POT-based analysis. For instance, simulated data can be used to assess the bias of an estimate, or to compare several plotting positions. It also helps in getting familiar with the random variations in the estimates or in the return level plots. The `rRendata` function can be used to generate a `RenData` object with a specific design: duration of the main sample, number and durations of MAX data or OTS blocks.

Suppose that we use a main sample of (default) duration 100 years and the default distribution the standard exponential. We can enhance the data by adding three MAX blocks of say 40, 50 and 30 years. By default, only the maximum observation will be kept in each block.

```
> set.seed(1234)
> RD1 <- rRendata(MAX.effDuration = c(40, 50, 30))
> plot(RD1)
```

See left of figure 1.5. The three MAX blocks are by convention located before the start of the main sample since in practice such blocks often represent historical data. We can similarly add 3 OTS blocks with 3 choosen durations and thresholds.

```
> RD2 <- rRendata(effDuration = 30,
                  distname.y = "GPD",
                  par.y = c(scale = 1, shape = 0.1),
                  OTS.effDuration = c(40, 50, 30), OTS.threshold = c(3, 4, 2))
> plot(RD2)
```

Note that we used here a non-default `"GPD"` distribution for the excesses Y_i, and we gave the values of the parameters. For now, the `rRendata` function can not generate random missing periods.

1.3.7 Aggregated data and gaps

A difficulty with aggregated data such as block data concerns the treatment of missing data or gaps. There is usually no reason that missing periods should correspond to full blocks (e.g. years), and most often a fraction of some blocks is missing. Excluding all blocks with missing data leads to a severe loss of information, while ignoring gaps in blocks may cause a bias. The use of aggregated data will be illustrated later in the section 2.3 about `barplotRenouv`. The problem of gaps in blocks will be also be discussed when describing the `OT2MAX` function in section 4.3 p. 38.

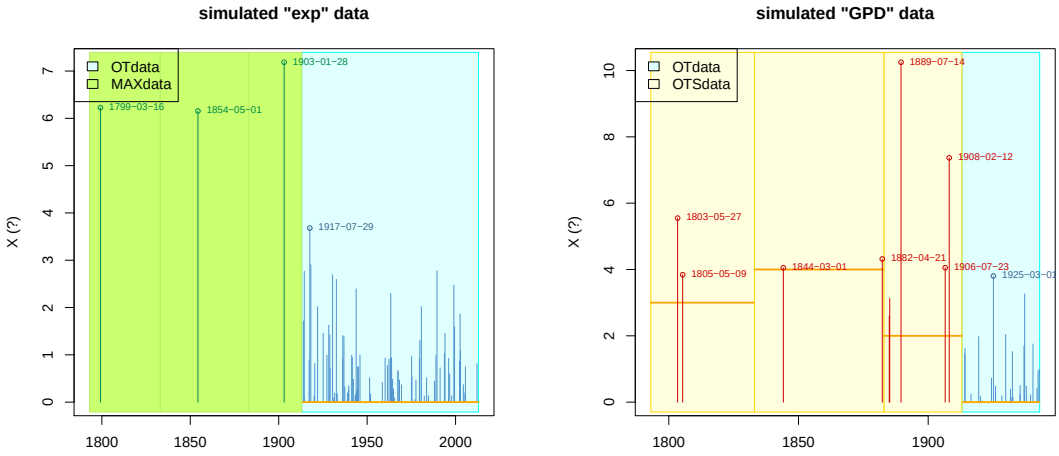

Figure 1.5: Two randomly generated `Rendata` objects. The distribution of the marks is exponential on the left, and GPD on the right. Three MAX blocks are used on the left, and three OTS blocks are used on the right.

Chapter 2

Descriptive tools

Some functions of **Renext** have been designed to check the assumptions relative to the stationnarity of the events or to the distribution of the levels. The analysis of the events can cope with gaps as are often met in practice. Although of less importance, the case where counts are used in place of events is also considered.

2.1 Functional plots

2.1.1 Principles

Widespread graphical tools in statistics are *functional plots*, such as exponential plot, Weibull or Gumbel plots. In all cases, the plot is designed so that the theoretical distribution curve (exponential/Weibull/Gumbel) shows as a straight line. For instance the relations for distribution functions F

$$\begin{aligned} -\log\left[1 - F(x)\right] &= (x - \mu)/\sigma \quad \text{(exponential)} \\ -\log\left[-\log F(x)\right] &= (x - \mu)/\sigma \quad \text{(Gumbel)} \end{aligned}$$

both show a linear relation between x and a transformed version $\phi(F)$ of $F(x)$, e.g. $\phi(F) = -\log\left[1 - F\right]$ for the exponential case. The functional plots are obtained by plotting $[x, \phi(F)]$ still using the values of the probability F to display the unevenly spaced graduations on the y-axis. The Weibull plot is similar but also uses a (log) transformation of x.

With a sample X_i of size n one uses non-parametric estimates $\widetilde{F}_i$ of the values $F(Z_i)$ of the distribution function at the order statistics Z_i with $Z_1 > Z_2 > \cdots > Z_n$. The n resulting points with ordinates $\widetilde{F}_i$ can be plotted with the transformed scale on the y-axis. A classical choice for the plotting positions is implemented in the `ppoints` function of the **stats** package

$$\widetilde{F}(Z_{n+1-i}) = \frac{i - a}{n - 2a + 1}, \tag{2.1}$$

where a is a parameter typically in the interval $[0, 1]$. The right hand side is the expectation of the random variable $F(Z_{n+1-i})$ for $a = 0$ and an approximation of its median for $a = 0.3$.

As some other packages do, **Renext** provides exponential and Weibull plotting functions, namely `expplot` and `weibplot`

```
> expplot(x = Brest$OTdata$Surge, main = "expplot for \"Brest\"")
```

```
> weibplot(x = Brest$OTdata$Surge-30, main = "weibplot for \"Brest\" (surge - 30)")
```

producing the two plots on figure 2.1.

Note that the transformation $\phi(F)$ must not depend on unknown parameters. Therefore the Weibull plot produces a theoretical line only for the version with two parameters (shape and scale), and not for the three parameter one (with location).

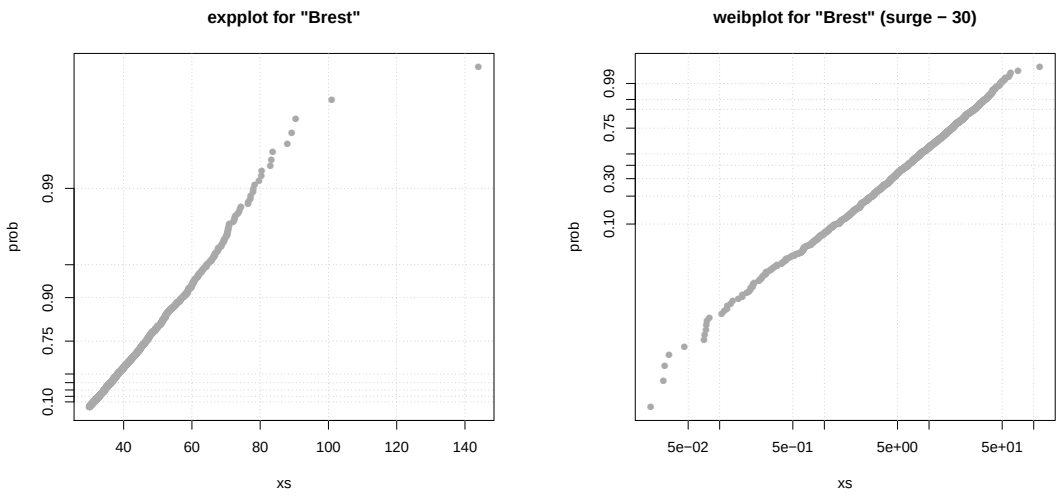

Figure 2.1: Exponential and Weibull plot for the Brest data. The variable `Surge` is used for the exponential plot. The threshold 30 cm is subtracted from `Surge` for the Weibull plot. The later uses a log-scale for `x`.

2.1.2 Exponential vs Gumbel

While hydrologists often favour Gumbel plots, the exponential plot may also be used. The latter is better suited to the use of "OTdata" i.e. data where only values over a threshold u are kept. Even if the original observations X_i are Gumbel, the conditional distribution $X_i \mid X_i > u$ will be close to an exponential for u large enough, see B.1.3. This can be illustrated with a few simple R commands

```
> library(evd); set.seed(136)
> X <- rgumbel(400); X <- X[X > 0.6]           ## X is truncated Gumbel
> n <- length(X);
> Zrev <- sort(X); F <- (1:n) / (n + 1)          ## distribution function
> y.exp <- -log(1 - F); y.gum <- -log(-log(F))
> plot(Zrev, y.exp, col = "red3", main = "exponential plot")
> plot(Zrev, y.gum, col = "SteelBlue3", main = "Gumbel plot")
```

The two plots are shown on figure 2.2. The difference between exponential and Gumbel plots is restricted to the small values.

2.2 Events and stationarity

2.2.1 Simple plots

The simplest plot for checking stationarity has points $[T_i, X_i]$ and can be obtained with R functions of the `graphics` package. The T_i and X_i will typically be available as two vectors of the same length or as two columns of a same data.frame object. For the example datasets of **Renext**, the T_i and X_i are given as two columns of the `OTdata` data frame

```
> plot(Flow ~ date, data = Garonne$OTdata, type = "h", main = "Flows > 2500 m3/s")
```

The graphic shows that several successive years had no exceedance over 2500 m^3/s during the second half of the 1940-1950 decade. This could lead to further investigations using the `subset` function

```
> subset(Garonne$OTdata, date >= as.POSIXct("1945-01-01") & date <= as.POSIXct("1950-01-01"))
         date Flow comment
96 1945-01-29 3200
```

The graphics can be enhanced using the `text` function in the `graphics` package to annotate special events or periods.

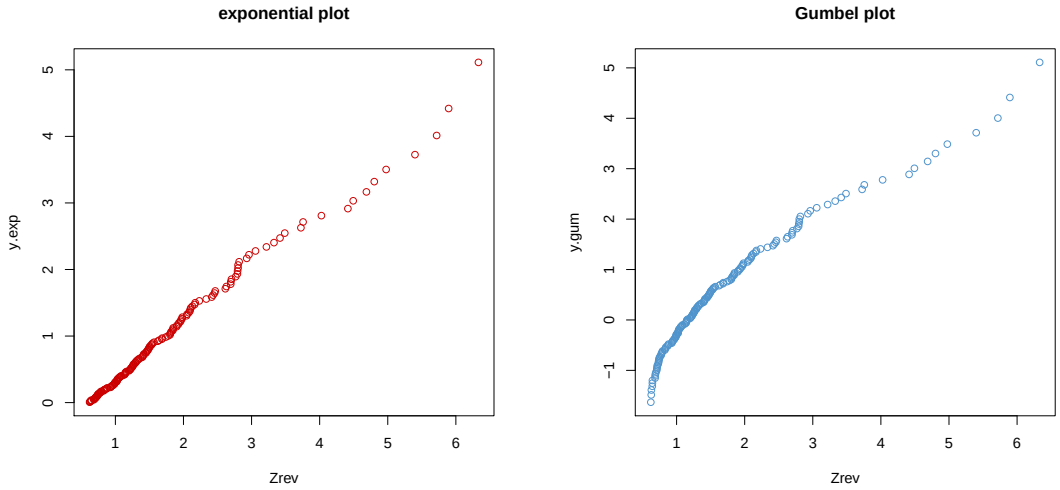

Figure 2.2: Truncated or "thresholded" Gumbel random sample. Due to the truncation, the sample distribution is close to an exponential. The graduations for the y-axis are not in probability-scale.

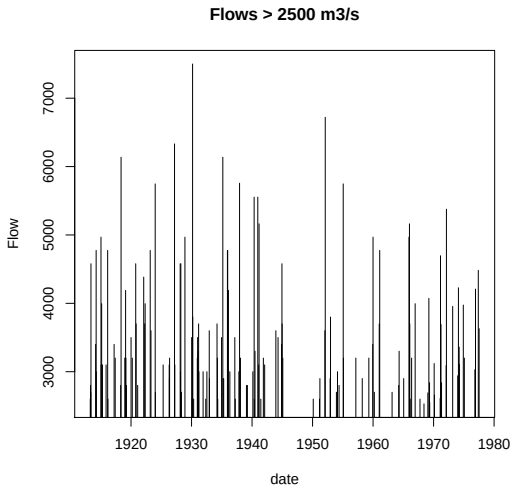

Figure 2.3: Simple plot of events for the `Garonne` data.

2.2.2 Uniformity

The `gof.date` function performs some tests to check the (conditional) uniformity of the events T_i as implied by the HPP assumption. It is based on the fact that for a given interval of time (s, t) the events T_i falling in the interval are jointly distributed as are the order statistics of a sample of the uniform distribution on (s, t). The sample size n is then random. Alternatively, the n events falling in an interval (T_k, T_{n+k+1}) also have this joint conditional distribution. In both cases a Kolmogorov-Smirnov (KS) test is well suited to check the uniformity.

The `gof.date` function mainly works with a `POSIX` object containing the events T_i as in

```
> gof.date(date = Garonne$OTdata$date)
```

which produces the plot on the left of figure 2.4. The empirical cumulative distribution function (ECDF) is compared to the uniform and the KS distance D_n is shown as a vertical segment. The displayed KS p-value tells that uniformity should be rejected at the significance level of 0.1%. Though less clearly than above, the plot points out that the years 1940-1950 had fewer events.

The `gof.date` function has optional args `start` and `end` to specify (and possibly restrict) the period on which the test is performed. By default these are taken as the first and last event in `date` and therefore only inner events are used in the ECDF.

2.2.3 Interevents

An important property of the HPP concerns the interevents $W_i = T_i - T_{i-1}$: the W_i are independent and have exponential distribution with rate λ. Thus an exponentiality test might be performed to check the HPP assumption for observed data.

The `interevt` function computes the interevents W_i as numbers of days. The function returns a list with a `interevt` data.frame element containing the W_i in the `duration` column which can be used to check exponentiality. This can be done either with a plot - see figure 2.4 or with the test of exponentiality of the function `gofExp.test`

```
> ie <- interevt(date = Garonne$OTdata$date)
> names(ie)
[1] "interevt" "noskip"
> d <- ie$interevt$duration
> expplot(d, main = "Exponential plot for interevents")
> bt <- gofExp.test(d)
> bt
$statistic
[1] 193.9517

$df
[1] 149

$p.value
[1] 0.01560322

$method
[1] "Bartlett gof for exponential"
```

It seems unlikely to obtain a good exponential fit as far as events occurrence shows seasonality as is the case here. A seasonality can no longer result from another distribution of interevents – that is from a non-Poisson stationary renewal process. Increasing the threshold might improve the adequacy with the assumptions.

2.2.4 Missing periods or gaps

In practice the situation is somewhat more complex due to the possible existence of missing (or skipped) periods where no events have been recorded. Event rates should then be computed using *effective duration* that is: the total duration of measurement *ignoring missing periods*.

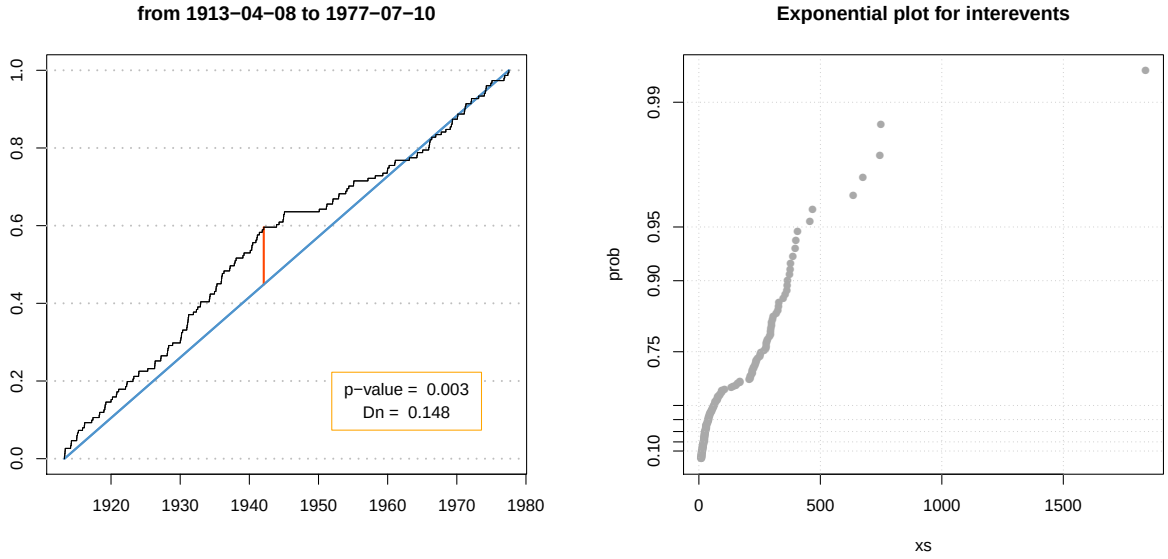

Figure 2.4: Analysis of the events for the `Garonne` data set (OTdata). Left panel: test for the uniformity of events with the KS distance shown as a vertical segment. Right panel : exponential plot for the interevents.

The functions `gof.date` and `interevt` can take this problem into consideration. The `gof.date` plot can display the missing periods or "gaps" provided that a suitable `skip` arg is given. For instance the following commands produce the plot on the left of figure 2.5

```
> gof.Brest  <- gof.date(date = Brest$OTdata$date, skip = Brest$OTmissing,
                         start = Brest$OTinfo$start, end = Brest$OTinfo$end)
> print(names(gof.Brest))
 [1] "effKS.statistic" "effKS.pvalue"    "KS.statistic"    "KS.pvalue"
 [5] "effnevt"         "nevt"            "rate"            "effrate"
 [9] "duration"        "effduration"     "noskip"
```

As their name may suggest, the returned list elements give the effective duration and the effective rate based on the true non-missing periods. The `noskip` element contains detailed information about each non-skipped period

```
> head(gof.Brest$noskip, n = 2)
       start        end duration nevt      rate        Dn         KS
1 1846-01-04 1847-01-01 0.991102   17 17.152624 0.2586935 0.17172882
2 1847-01-21 1852-01-21 4.999316   48  9.601314 0.2057777 0.02929104
```

For each period the rate has been computed as well as a KS test of uniformity. The power of the test is obviously limited for periods containing only a few events.

The preceding call to `gof.date` corresponded to the default value of `plot.type`, namely `"skip"`. A drawback of the plot and KS test is that the comparison with the uniform is biased by the gaps. The KS distance D_n between the empirical and theoretical distributions can be amplified by the gaps when there are too few events or, on the contrary, be reduced by gaps when there are too much events. These two phenomena can be seen by comparing the two plots of figure 2.5 although the two KS statistics and p-value are here nearly identical. The right panel plot was produced using the non-default choice for the `plot.type` arg i.e. `plot.type= "omit"`, missing periods can be omitted on the plot and in the KS test computation.

```
> gof.Brest2  <- gof.date(date = Brest$OTdata$date,
                          skip = Brest$OTmissing, plot.type = "omit",
                          start = Brest$OTinfo$start, end = Brest$OTinfo$end)
```

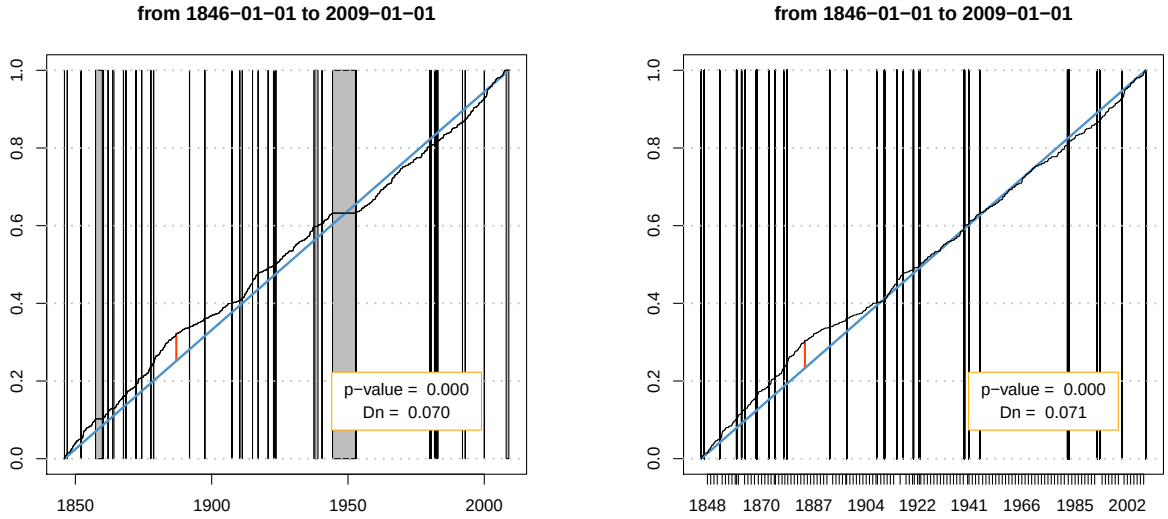

Figure 2.5: Using the `plot.type` arg of `gof.date` leads to the left panel (default value or `"skip"`), or the right one (value `"omit"`). Each missing period appears as a grey rectangle on the left, and is flattened as a line on the right. Though graphically unevenly spaced, the tickmarks of time axis on the right show the beginning of years.

The time axis now has *unevenly* spaced ticks since it is obtained by concatenating the successive non-missing periods. More precisely, each retained time interval k begins at the first event T_{f_k} of a continuous observation period and ends at its last event T_{ℓ_k}. Each of the vertical lines shows an interval $(T_{\ell_k}, T_{f_{k+1}})$, which covers a missing period and is cut out as shown on figure 2.6. The displayed information on the right panel of figure 2.5 concerns `effKS.pvalue` and `effKS.statistic` of an "effective" KS test performed on non-missing periods. Provided that observation gaps occur independently from the events T_i, the interevents for couples of successive events falling in the same non-missing period can be used in a modified KS test. In the HPP case, these interevents should be independent and identically distributed with exponential distribution, thus concatenating them should produce an HPP hence an uniform conditional distribution of events.

For the `Brest` example, the test tells us that the uniformity of events should be rejected while the plot indicates that there were more events during the XIXth century than during the XXth (the events have been shown on the left of figure 1.4). Since large surges tend to occur more frequently in winter, further investigation of the gaps distribution would be useful. The `OT2MAX` function can help, see section 4.3.2 page 40. Since the interest in on high surge levels, we can select the events exceedances over the threshold $u := 50$ cm.

```
> gof.Brest3 <- gof.date(date = subset(Brest$OTdata, Surge > 50)$date,
                         skip = Brest$OTmissing, plot.type = "omit",
                         start = Brest$OTinfo$start, end = Brest$OTinfo$end)
> c(gof.Brest3$KS.pvalue,  gof.Brest3$effKS.pvalue)
[1] 0.6017242 0.1963612
```

The test now tells that the uniformity is accepted; the second p-value ≈ 0.2 is computed by omitting the gaps and is thus more reliable than the first. The plot is not shown.

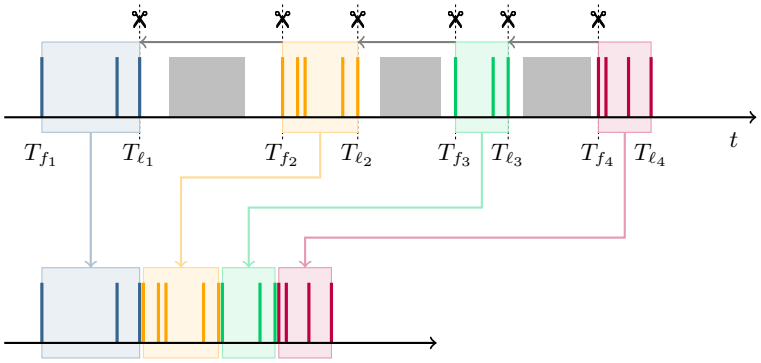

Figure 2.6: With `plot.type = "omit"`, the plot of `gof.date` only considers interevents for couples falling in the same non-missing period and concatenates them. The time interval $(T_{\ell_k}, T_{f_{k+1}})$ between the last event T_{ℓ_k} of the non-missing period k and the first event $T_{f_{k+1}}$ of the following non-missing period is "cut out". The two events T_{ℓ_k} and $T_{f_{k+1}}$ collapse into *one* event of the new Point Process. Note that a non-missing period with less than two events is cut out since it contains no valid interevent.

2.3 Aggregated (counts) data

2.3.1 Counts

The `barplotRenouv` function draws a barplot for counts data and performs a few tests adapted to this context where events are unknown, or when interevents can no longer be used. The data used are n counts N_i for $i = 1, 2, \ldots, n$. These counts must be on *disjoint intervals* or "blocks" with the *same duration*, e.g. one year. If events occur according to an HPP the N_i form a sample of a Poisson distribution. The barplot compares the empirical (or observed) frequencies to their theoretical counterparts i.e. the expectations. The theoretical distribution is estimated using the sample mean as Poisson parameter (Poisson mean).

The `Brest.years` object contains aggregated data for one-year blocks. Some blocks are incomplete and are listed in `Brest.years.missing` which can be used in `barplotRenvouv`

```
> data(Brest.years); data(Brest.years.missing)
> bp40 <- barplotRenouv(data = Brest.years, threshold = 40,
             na.block = Brest.years.missing, main = "threshold = 40 cm")
```

produces the graphic at the left of figure 2.7. Increasing the threshold

```
> bp50 <- barplotRenouv(data = Brest.years, threshold = 50,
             na.block = Brest.years.missing, main = "threshold = 50 cm")
```

we get a barplot for the smaller sample at the right of figure 2.7. Note that the function guesses that the first column represents a block indication which may not be true with other data. Therefore the normal use would specify the `blockname` and `varname` formal arguments of `barplotRenouv`.

Great care is needed when the data contain missing periods since the number of events is then biased downward.

2.3.2 Goodness-of -fit

A popular test for Poisson counts is called *overdispersion test*. It is based on the fact that expectation and variance are equal in a Poisson distribution. The test statistic is

$$I = (n-1)\, S^2/\bar{N}$$

where $\bar{N}$ and S^2 are the sample mean and variance. Under the null hypothesis I is approximately distributed as $\chi^2(n-1)$. The statistic I tends to take large values when the observations N_i come from an overdispersed distribution such as the negative binomial. A one-sided test can therefore be used for a negative binomial alternative.

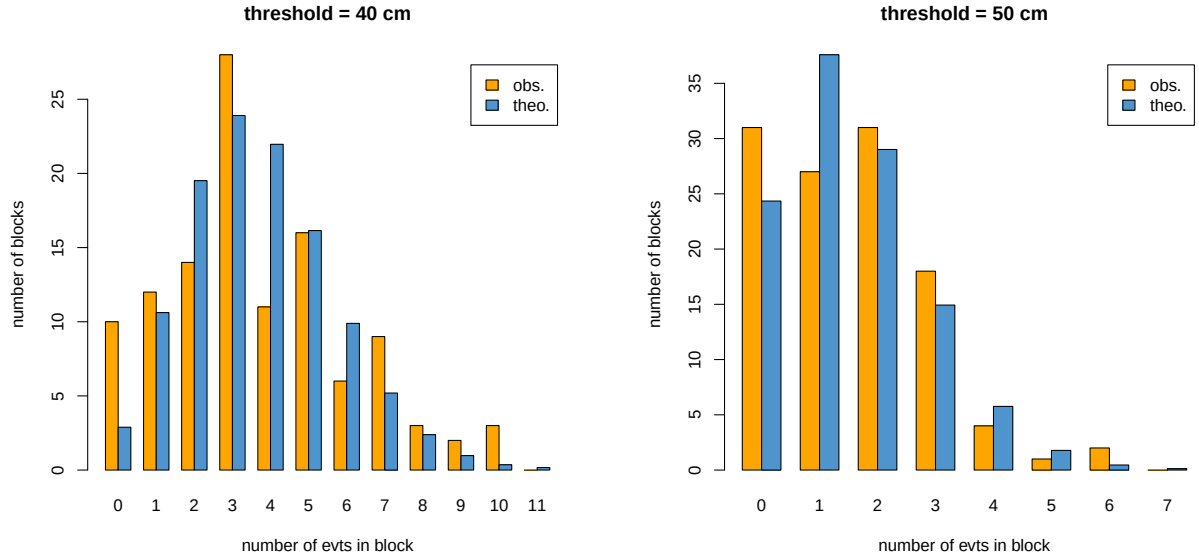

Figure 2.7: The two barplots produced with `barplotRenouv`. A bar height represents a number of blocks (here years) with the number of events given in abscissa.

A Chi-square Goodness-of-fit test is also available to check the goodness-of-fit of the N_k to a Poisson distribution. In this test, the counts values N_k are summarized in a tabular format retaining m distinct values or group of adjacent values, together with the corresponding frequencies. The test statistic is

$$D^2 = \sum_{k=1}^{m} (O_k - E_k)^2 / E_k$$

where O_k and E_k are the observed and expected frequencies for the class k. For instance, the first class $k = 1$ can be $N = 0$ meaning that O_1 and E_1 are the number of intervals with no events recorded. Asymptotically (for large n)

$$D^2 \sim \chi^2(m - p - 1)$$

where p is the number of parameters estimated from data, here $p = 1$ (for the mean of N). A one-sided test will reject the Poisson hypothesis when D^2 is too large[1].

A classical drawback of this test is that classes with a small expected count E_i should be grouped, in order to reach a minimal total of (say) 5.

```
> bp40$tests
     statistic  df      p.value
disp  181.4726 113 4.652672e-05
chi2   21.5105   5 6.485040e-04
> bp50$tests
      statistic  df   p.value
disp 131.022727 113 0.1181542
chi2   5.722912   3 0.1258975
```

For the dataset `Brest.years`, using a threshold of 50 cm leads to acceptable tests (at the 10% level), while 40 cm seems too small. For the chi-square test, more details (e.g. grouping) are available.

```
> bp50$freq
```

[1] That is: $D^2 > \chi^2_\alpha$

```
  obs.      theo. group
0   31 24.3452997     1
1   27 37.5857258     2
2   31 29.0135427     3
3   18 14.9309460     4
4    4  5.7628213     5
5    1  1.7793974     5
6    2  0.4578567     5
7    0  0.1244104     5
```

The values of N have been grouped in odrer to reach a minimal expected number of 5 for each group.

Note that for a fairly high threshold, the statistic N will generally take only the two values 0 and 1. Then the chi-square test which requires at least three classes will not be available.

Chapter 3

Renouv objects

Fitted POT models are in **Renext** considered as objects of a `"Renouv"` S3 class, and can be used with methods `coef`, `vcov`, `plot`, `predict`, ... Such models are usually created by a ML estimation using the creator function `Renouv`. This function can carry out the usual estimation from observations X_i of the marked process. It can also cope with heterogeneous data including blocks such as `MAXdata` or `OTSdata` described in previous chapters, e.g. to use historical information. In some rare cases, a `Renouv` object can also be created with `RenouvNoEst` if all coefficients are known.

3.1 Fitting POT for La Garonne

For the dataset `Garonne`, the OT data contain flow values over the threshold $u_\star = 2500$ m^3/s. We can fit a POT model with any threshold $u \geqslant 2500$. As in Miquel (1984) we fit an exponential and a two-parameter Weibull distribution using OT data only. The `Renouv` function needs on input the *levels* given in a vector `x`, the *effective duration* `effDuration` – normally in years – and the *threshold*

```
> fit.exp <- Renouv(x = Garonne$OTdata$Flow, effDuration = 65, threshold = 3000,
                    distname.y = "exponential", main = "exponential")
Special inference for the exponential case without history
> class(fit.exp)
[1] "Renouv"
```

The result is an object with (S3) class `"Renouv"`. A few S3 methods are available for this class:

```
> methods(class = "Renouv")
 [1] AIC     anova   BIC     coef    lines   logLik  nobs    plot    PPplot
[10] predict print   QQplot  summary vcov
see '?methods' for accessing help and source code
```

The method `coef` extracts the vector of estimated coefficients

```
> coef(fit.exp)
      lambda         rate
1.5076923077 0.0009695003
```

The first element named `"lambda"` is the event rate expressed in *events by year*. The other elements are the ML estimates of the distribution for excesses, with names corresponding to the probability functions – here one name `"rate"` for the exponential distribution parameter. The ubiquitous `plot` method can be used to re-draw a return level plot from the fitted object. The `summary` method can be used to display the results. The `predict` method can be used to compute return levels corresponding to given return periods as illustrated later.

A `Renouv` object is mainly a list within which the `estimate` element gives the maximum likelihood estimates returned by `coef`. Many other results are returned.

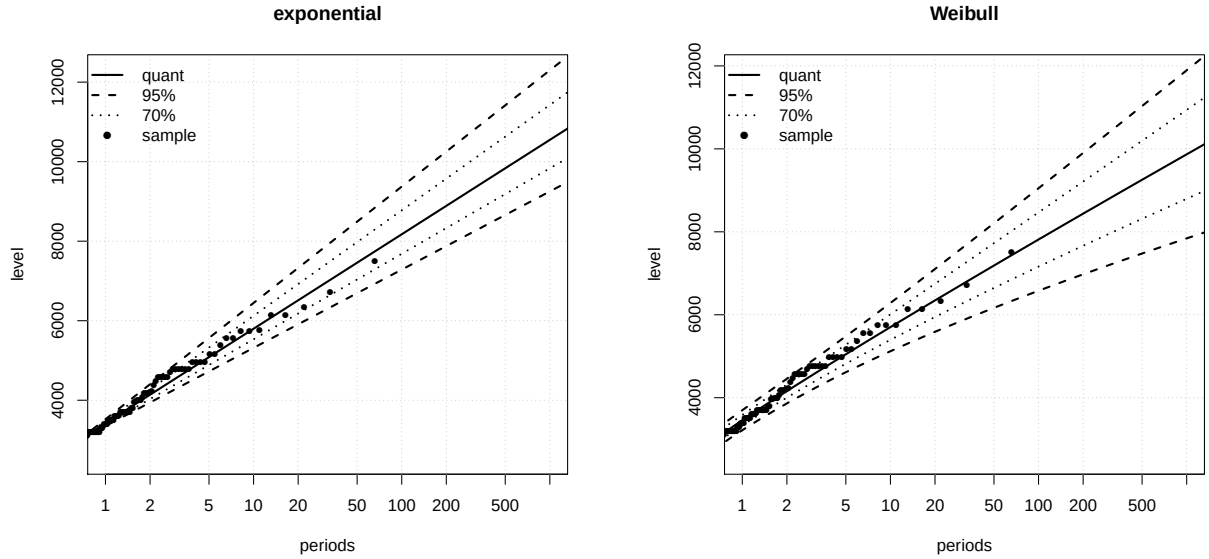

Figure 3.1: Return level plots for the example `Garonne` with two distributions for the excesses.

```
> head(names(fit.exp), n = 24)
 [1] "call"         "x.OT"         "y.OT"         "nb.OT"        "effDuration"
 [6] "threshold"    "distname.y"   "p.y"          "parnames.y"   "fixed.y"
[11] "trans.y"      "est.N"        "cov.N"        "est.y"        "cov.y"
[16] "corr.y"       "estimate"     "fixed"        "df"           "nobs"
[21] "p"            "opt"          "logLik"       "sigma"
```

This shows the 24 first elements in the list. The `sigma` element gives the vector of standard deviations for the estimates.

The `distname.y` formal in `Renouv` is used to change the distribution for the excesses $Y_i = X_i - u$.

```
> fit.weibull <- Renouv(x = Garonne$OTdata$Flow, effDuration = 65, threshold = 3000,
                        distname.y = "weibull", main = "Weibull")
> coef(fit.weibull)
     lambda       shape       scale
   1.507692    1.063710 1057.298215
> fit.weibull$sigma
      lambda        shape        scale
  0.15229992   0.08377919 105.72097002
```

The estimated parameters of the Weibull distribution and their standard deviation (list item `sigma`) show that the shape is close to 1.0, which corresponds to the exponential distribution. The two fits produced return level plots shown on figure 3.1.

3.2 Return level plot

3.2.1 Description

Renext uses a return level plot which may be qualified as *exponential*, and differs from the usual one which uses *Gumbel* scales. The main difference is that the exponential plot uses a log scale for return periods while the Gumbel plot uses a log-log scale. In both cases, the theoretical return level curve (exponential/Gumbel) shows as a straight line.

The difference between the two plots is restricted to the small levels/return periods, since the exponential and Gumbel distribution functions are close for large values. As it was advocated in the discussion

about functional plots page 11, the exponential return level plot is better suited to the use of "OTdata" i.e. data where only values over a threshold are kept, even if the the original observations X_i are Gumbel see B.1.3.

The return level plot is similar to the classical exponential plot of the previous chapter, *but with the two axes x, y exchanged*. A concave (downward) RL plot indicates a distribution with a tail "lighter than the exponential" or even with finite end-point such as GPD with $\xi < 0$.

The displayed confidence limits are in all cases pointwise and bilateral, and correspond to the confidence percents displayed which can be changed in the call. In most cases the confidence limits are approximate and obtained by using the *delta method* briefly described later. For some special cases with exponential distribution an exact inference is possible and used. The `infer.method` element in the list returned by `Renouv` provides information about this.

3.2.2 Plot method for `Renouv` objects

Once created with the `Renouv` function, an object of class `"Renouv"` can be used to (re)draw a return level plot and change some options. Useful changes concern the main title using the `main` argument, or axes labels `xlab`, `ylab`. Axis limits can also be set. For the return levels, this is done using the usual `ylim` argument. For the return periods, the limits are set using `Tlim` or `problim`. The first possibility works with a vector containing two return periods (in years); the second possibility requires a vector with two probabilities.

The two following code chunks produce the return level plots shown on figure 3.2. On left panel, we change the return periods axis limits.

```
> plot(fit.weibull,  Tlim = c(1, 100), main = "return periods from 0 to 100 years")
```

On the right panel we change both axes and the confidence level.

```
> plot(fit.weibull, Tlim = c(1, 100), ylim = c(3000, 10000), pct.conf = 95,
       main = "return levels and 95% limits")
```

The chosen percentage for the confidence limits `pct.conf = 95` must correspond to a value available in the object description. Otherwise, it is necessary to force a new prediction by passing a suitable `pct.conf` argument along with `predict = TRUE` in the call to the `plot` method. The shown elements of the `Renouv` object can be selected, see chapter 5 p. 42 for more details.

When only OT data are used as here, the default plotting positions use a return period at the order statistics $Z_1 > Z_2 > \cdots > Z_n$ estimated by $1/\widehat{T}(Z_i) = \widehat{\lambda}\,\widetilde{S}(Z_i)$, where $\widehat{\lambda}$ is the natural estimate of the rate (see below) and $\widetilde{S}(Z_i) = 1 - \widetilde{F}(Z_i) = i/(n+1)$. Alternatively, the `ppoints` formula (2.1) for $a \neq 0$ or Nelson's formula (Nelson 2000) can be specified with `plotOptions` passed to the `SandT` function. The difference between the different choices can be important for the largest order statistics (i.e. for small i).

3.3 Computational details

3.3.1 Maximum Likelihood theory

Estimation and inference in **Renext** mainly rely on the Maximum Likelihood (ML) theory. A relevant presentation can be found in Coles (2001, chap. 2) or in the *Further reading* references given there.

The standard application context of ML is when an ordinary sample i.e. n independent random variables X_i with the same distribution depending of an unknown vector $\boldsymbol{\theta}_X$ with density $f_X(x; \boldsymbol{\theta}_X)$. The likelihood function L is the joint density of the sample i.e.

$$L = \prod_{i=1}^{n} f_X(X_i;\, \boldsymbol{\theta}_X)$$

and the estimator $\widehat{\boldsymbol{\theta}}_X$ is the value of $\boldsymbol{\theta}_X$ maximising L. In some special cases the maximisation of L can have an explicit solution, but a numerical optimisation will generally be required. The ML theory

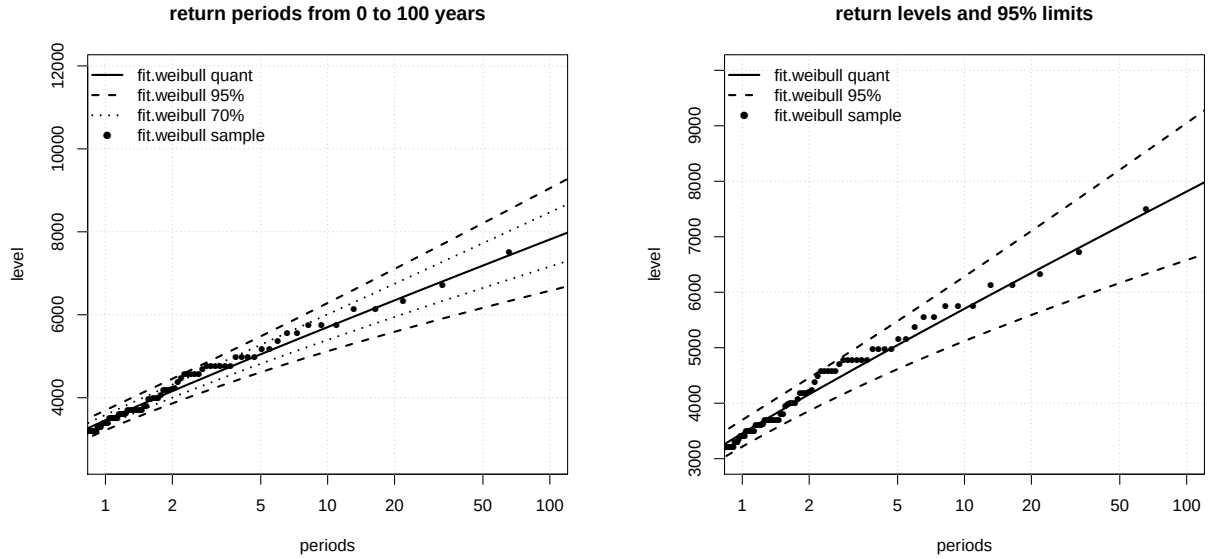

Figure 3.2: Changing the settings of the return level plot. Left and right: the limits of the x-axis are set using `Ylim`. Right: `ylim` and `pdc.conf` are used and only the specified 0.95 confidence level is shown.

warrants[1] the *asymptotic unbiasedness* and *asymptotic normality*: when n is large $\widehat{\boldsymbol{\theta}}_X$ has its expectation approximately equal to the true unknown $\boldsymbol{\theta}_X$, and it is approximately normally distributed.

The ML theory applies to more general situations where observations are no longer independent or can have different marginal distributions. This occurs when order statistics are used in the estimation, e.g. with historical data.

The general principle of the `Renouv` function is to allow a large choice of distributions, yet trying to take advantage of the specific distribution/independence when possible. In most cases the maximisation of the likelihood is obtained using `optim` function of the `stats` package. When historical data are used they are considered as a complement to the ordinary data (excesses) and two optimisations might be used.

3.3.2 Estimation and inference

The model uses a parameter vector $\boldsymbol{\theta} = [\lambda, \boldsymbol{\theta}_X^\top]^\top$ of length p formed with the HPP rate λ and the parameter vector $\boldsymbol{\theta}_X$ for the levels distribution.

When only OT data are used, the observed data consist in N events $[T_i, X_i]$ on a given period. Since events T_i and levels X_i are independent the likelihood is

$$L_{\texttt{OT}} = \underbrace{\frac{(\lambda w)^N}{N!} e^{-\lambda w}}_{\text{events}} \times \underbrace{\prod_{i=1}^{N} f_X(X_i; \boldsymbol{\theta}_X)}_{\text{levels}}$$

where w is the time-length (i.e. the effective duration), and the log-likelihood is

$$\log L_{\texttt{OT}} = N \log(\lambda w) - \lambda w - \log(N!) + \sum_{i=1}^{N} \log f_X(X_i; \boldsymbol{\theta}_X). \tag{3.1}$$

The ML estimation consists in two simple ML estimations: one for the events (rate estimation) and the other for levels. The ML estimate of the unknown rate λ is

$$\widehat{\lambda} = \frac{N}{w} = \frac{\text{number of events}}{\text{duration}},$$

[1] Under suitable *regularity conditions*.

its variance is $\mathrm{Var}[\widehat{\lambda}] = \lambda/w \approx \widehat{\lambda}/w$. Note that the number of events N is a *sufficient statistic* for λ: the events T_i are not used and the whole information they provide about λ is contained in N. The "X-part" of ML concerns an ordinary sample. The ML estimate $\widehat{\boldsymbol{\theta}}_X$ is available in closed form in some cases (e.g. exponential) or can be computed by using a specific method (e.g. GPD, Weibull, gamma), see Deville (2015).

When only OT data are used, it can be said that λ and $\boldsymbol{\theta}_X$ are orthogonal parameters. This is no longer true when block data (e.g. historical data) are also used: the likelihood then takes a slightly more complex form given below.

In a few cases with only OT data and favourable distribution (e.g. Weibull), it is possible to use the *expected* information matrix. But the general treatment in **Renext** is based on the *observed* information and the numerical derivatives. More precisely, the information matrix is obtained as the numerical hessian at convergence. The hessian can either be the element `hessian` returned by the `optim` function, or result from the use of the `hessian` function from the `numDeriv` package: see the manual for more details.

3.3.3 Delta method

The *delta method* can be used to infer about a function[2] $\psi = \psi(\boldsymbol{\theta})$ of the parameter $\boldsymbol{\theta}$. For instance $\psi(\boldsymbol{\theta})$ can be the return period of a given level x (see 1.1). The transformed parameter estimate is $\widehat{\psi} = \psi(\widehat{\boldsymbol{\theta}})$. As a general result in the ML framework, the transformed parameter estimate is asymptotically unbiased $\mathbb{E}[\widehat{\psi}] \approx \psi(\boldsymbol{\theta})$ and asymptotically normal with variance

$$\mathrm{Var}[\widehat{\psi}] \approx \boldsymbol{\delta}^\top \mathrm{Var}[\widehat{\boldsymbol{\theta}}]\, \boldsymbol{\delta}$$

where $\boldsymbol{\delta}$ is the gradient vector

$$\boldsymbol{\delta} = \frac{\partial \psi}{\partial \boldsymbol{\theta}} = \left[\frac{\partial \psi}{\partial \theta_1}, \frac{\partial \psi}{\partial \theta_2}, \ldots, \frac{\partial \psi}{\partial \theta_p}\right]^\top$$

evaluated at $\widehat{\boldsymbol{\theta}}$, see Coles (2001, chap. 2).

Renext uses this approach[3] with ψ taken as the level (or quantile) $x(T)$ corresponding to a given return period T, given by (1.2) in section 1.2.2. Using chain rule, the derivative with respect to the rate λ is

$$\frac{\partial}{\partial \lambda} x(T) = \frac{1}{\lambda^2 T f_X}$$

where the density f_X is evaluated at $x(T)$. In practice, the uncertainty on λ has a minor impact for large return periods and can optionally be ignored in the computations. The gradient of the quantile function with respect to $\boldsymbol{\theta}_X$ is computed numerically using a finite difference approximation.

3.3.4 Goodness-of-fit

As a general tool to assess the fit, the Kolmogorov-Smirnov (KS) test is computed in all cases.

The KS test normally requires a *completely specified* distribution for the null hypothesis while the *fitted* distribution is used here – thus generating a bias. In some special cases (normal, exponential) the bias could be corrected using an adaptation depending on the distribution as in Lilliefors test for the normal. However since the number of estimated parameters is small (usually 1 or 2 for the "excesses part") the bias will be small provided that the number of exceedances is large enough, say 50 or more.

For some distributions such as exponential a specific test may be available. In the current version distribution-specific tests are in `Renouv` limited to Bartlett's test of exponentiality.

Rounded measurements often lead to ties in the sample, which would without precaution generate a warning in the KS test. This can be avoided by "jitterising" i.e. adding a small random noise to the observed values.

The graphical analysis of the fit using the return level plot is generally instructive. For exponential or Weibull excesses, classical exponential or Weibull plot can also be drawn using the `expplot` and `weibplot` functions.

[2] Smooth enough.

[3] In the `predict` method for `Renouv` objects.

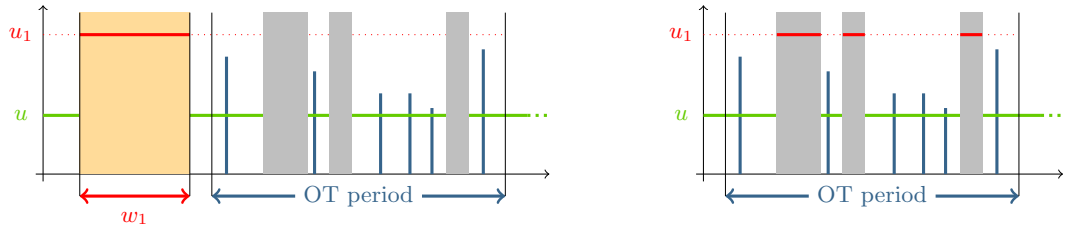

Figure 3.3: An unobserved level can provide information on an historical period (left) or on missing periods (right). In the second case, one would use a virtual block with its duration w_1 set to the sum of all gaps lengths.

When block data (e.g. historical data) are given, they are used during the estimation but not included in the empirical distribution in the KS test. In this case, the interpretation of the test needs further investigations.

3.4 Using heterogeneous data

3.4.1 Two types of block data

Beside OT data, `Renouv` and other **Renext** functions can use two other sorts of data: MAX data which are r largest, and OTS data for "Over Threshold Supplementary" data[4]. In both cases, the data are structured in blocks and can be used only as complement to the main OT data.

MAX data contain r largest blocks. Each block corresponds to a time interval of known duration w during which the r largest values are available. Blocks are assumed to be mutually disjoint and disjoint from the OT period. Neither the duration of blocks nor the number r of observations are assumed to be constant; hence each block b has a specified duration w_b and a number r_b of largest values.

OTS data contain Over Threshold blocks with known duration, exceedances and levels (or excesses). Again, blocks are assumed to be mutually disjoint and disjoint from the OT period and other blocks. For each such block b with known duration w_b, we must have a threshold u_b and all observations with levels exceeding u_b. The number r_b of such observations may be zero, in which case we may say that u_b is an *unobserved level*. The threshold u_b can not be smaller than the main threshold.

In the context of historical information, the threshold u_b of an OTS block can be called a *perception* threshold. Unobserved levels (empty OTS blocks) occur when it is granted, or at least believed, that u_b was never exceeded during a period of time. For instance it can be granted that a river never flooded over a given benchmark level during the last five centuries, or that the arch of a bridge was never reached since its construction. Such information has obviously a great potential impact on the estimation since it typically concerns very long periods, much longer than the observation period. Note that the unobserved level can concern missing periods for OT data: although no data are available we may still know that no very high level occurred, see figure 3.3.

3.4.2 Likelihood

Global likelihood

In the general setting of heterogeneous data described above, the log-likelihood takes the form

$$\log L = \log L_{\texttt{OT}} + \log L_{\texttt{MAX}} + \log L_{\texttt{OTS}},$$

because the OT period and MAX or OTS blocks are assumed to correspond to disjoint time intervals and hence to independent observations. Similarly the log-likelihood for MAX or OTS data are sums of

[4] Or Over ThresholdS.

contributions arising from independent blocks, so

$$\log L = \log L_{\texttt{OT}} + \sum_{b \in \{\texttt{MAX blocks}\}} \log L_b + \sum_{b \in \{\texttt{OTS blocks}\}} \log L_b.$$

The log-likelihood for a MAX or OTS block is given below.

MAX data

Consider a MAX block b with duration w_b. Let $Z_{b,1} \geqslant Z_{b,2} \geqslant \cdots \geqslant Z_{b,r_b}$ be the r_b largest observations. The log-likelihood for the block can be proved to be

$$\log L_b = r_b \log(\lambda w_b) + \sum_{i=1}^{r_b} \log f_X(Z_{b,i};\, \boldsymbol{\theta}_X) - \lambda w_b\, S_X(Z_{b,r};\, \boldsymbol{\theta}_X) \tag{3.2}$$

up to an unimportant additive constant.

OTS data

The likelihood for an OTS block with threshold u_b is simpler to derive. According to the POT assumptions[5], the levels greater than u_b occur according to an HPP *thinning* the original HPP. This thinned process has rate $\lambda S_X(u_b)$ because at each OT event, the level u_b can be exceeded with probability $S_X(u_b)$. Let w_b be as before the block duration, and let $Z_{b,1} \geqslant Z_{b,2} \geqslant \cdots \geqslant Z_{b,r_b}$ be the r_b observations, with possibly $r_b = 0$. Up to an additive constant, the log-likelihood is

$$\log L_b = r_b \log(\lambda w_b) + \sum_{i=1}^{r_b} \log f_X(Z_{b,i};\, \boldsymbol{\theta}_X) - \lambda w_b\, S_X(u_b;\, \boldsymbol{\theta}_X). \tag{3.3}$$

This expression is identical to (3.2) with the block threshold u_b replacing the minimum observed value Z_{b,r_b}.

When an OTS block b contains no observation i.e. when $r_b = 0$, the log-likelihood (3.3) is simply

$$\log L_b = -\lambda w_b\, S_X(u_b;\, \boldsymbol{\theta}_X). \tag{3.4}$$

This is easily checked: on a period of length w_b, the number of levels $> u_b$ is Poisson with mean $\mu := S_X(u_b) \times \lambda w_b$. Hence the probability to observe no level $> u_b$ is: $e^{-\mu}\mu^0/0! = e^{-\mu}$.

Remarks

Assume that we have OT data, and consider the impact of using one extra block. Some special cases arise.

1. When only one OTS block with no observation and with u_b equal to the main threshold, its contribution to the global log-likelihood is $-\lambda w_b$ since then $S_X(u_b) = 1$ in 3.4). Up to an unimportant constant, the resulting global log-likelihood is identical to the one which would result from simply adding w_b to the effective duration w of the main OT sample in (3.1).

2. Assume that we have only one historical MAX block which only contains the maximum $Z_{b,1}$ i.e. has $r_b = 1$. The contribution of the block to the log-likelihood (3.2) is

   $$\log L_b = \log(\lambda w_b) + \log f_X(Z_{b,1};\, \boldsymbol{\theta}_X) - \lambda w_b\, S_X(Z_{b,1};\, \boldsymbol{\theta}_X).$$

   At the right hand side, the third term is identical to (3.4) with an unobserved level $u_b = Z_{b,1}$ and a period length w_b. The sum of the two first terms at right hand side is the extra contribution that would be added to the log-likelihood of the OT data if a new OT observation with level $Z_{b,1}$ had been added without changing the main OT period duration. Therefore, the same likelihood/results are obtained in the two following approaches.

   - Specify an historical MAX block of length w_b with $r_b = 1$ and level $Z_{b,1}$.
   - Join the observed maximum $Z_{b,1}$ to the OT levels X_i, and specify that the level $u_b := Z_{b,1}$ was never reached during a OTS block of length w_b.

   The second approach might seem natural to practitioners.

[5] See section 1.2.1 page 2.

Likelihood maximisation

When heterogeneous data are used, the ML estimators of λ and $\boldsymbol{\theta}_X$ are found by numerically maximising the log-likelihood. It can be shown that this likelihood function can be concentrated with respect to the rate λ, thus leading to the maximisation of a function $\log L_c(\boldsymbol{\theta}_X)$ depending on $\boldsymbol{\theta}_X$ only, see Deville (2015).

The numerical maximisation relies on the `optim` function. Like most EV packages do, **Renext** uses an unconstrained optimisation, while most distributions would normally require the use of inequality constraints. For instance with GPD excesses, constraints should be imposed because the maximal likelihood is otherwise infinite, see B.3.2. In practice, this is not really a concern because the log-likelihood will take the value `NA` or `NaN` rather than a large value near the boundary of the parameter domain, and `optim` copes quite well with a `NA` value of the objective.

By default, the initial values for the estimation with heterogeneous data are obtained as the ML estimates based on the OT data only. The reason is that MAX or OTS data were regarded as complementary data in the initial conception of **Renext**. Moreover, the ML estimation based on OT data is simplified for most of the distributions used in practice.

3.4.3 Example: using Garonne historical MAX data

As seen in chapter 1, the `Garonne` dataset contains historical data of type MAX, which can be used in the estimation. The data are described in the section 1.3.4 page 7. The historical part corresponds here to one block, and the following levels

```
> Garonne$MAXdata$Flow
 [1] 7500 7400 7000 7000 7000 6600 6500 6500 6400 6300 6300 6200
```

The duration is given in `Garonne$MAXinfo$duration` with value 143.09 years.

As a general rule, the MAX or OTS data must in `Renouv` be passed as a *list* of numeric vectors, each vector corresponding to one block. The (effective) durations are given as a numeric vector with the *same length as the list*. For the MAX case, the formal arguments to use are `MAX.data` (list) and `MAX.effDuration` (numeric vector).

Since the data corresponds here to one block, the list `MAX.data` contains only one vector and the vector `MAX.effDuration` is of length one. The two following fits produce the return level plots shown in figure 3.4.

```
> fit.exp.H <- Renouv(x = Garonne$OTdata$Flow,
                      effDuration = 65, threshold = 3000,
                      MAX.data = list(Garonne$MAXdata$Flow),
                      MAX.effDuration = Garonne$MAXinfo$duration,
                      distname.y = "exponential",
                      main = "Garonne data, \"exponential\" with MAXdata")


> fit.weib.H <- Renouv(x = Garonne$OTdata$Flow,
                       effDuration = 65, threshold = 3000,
                       MAX.data = list(Garonne$MAXdata$Flow),
                       MAX.effDuration = Garonne$MAXinfo$duration,
                       distname.y = "weibull",
                       main = "Garonne data, \"Weibull\" with MAXdata")
```

The exponential fit is only slightly modified by the use of historical data. As said before, the parameter λ and $\boldsymbol{\theta}_X$ are no longer orthogonal when historical data are used

```
> fit.exp.H$corr
          lambda      rate
lambda 1.0000000 0.2054784
rate   0.2054784 1.0000000
```

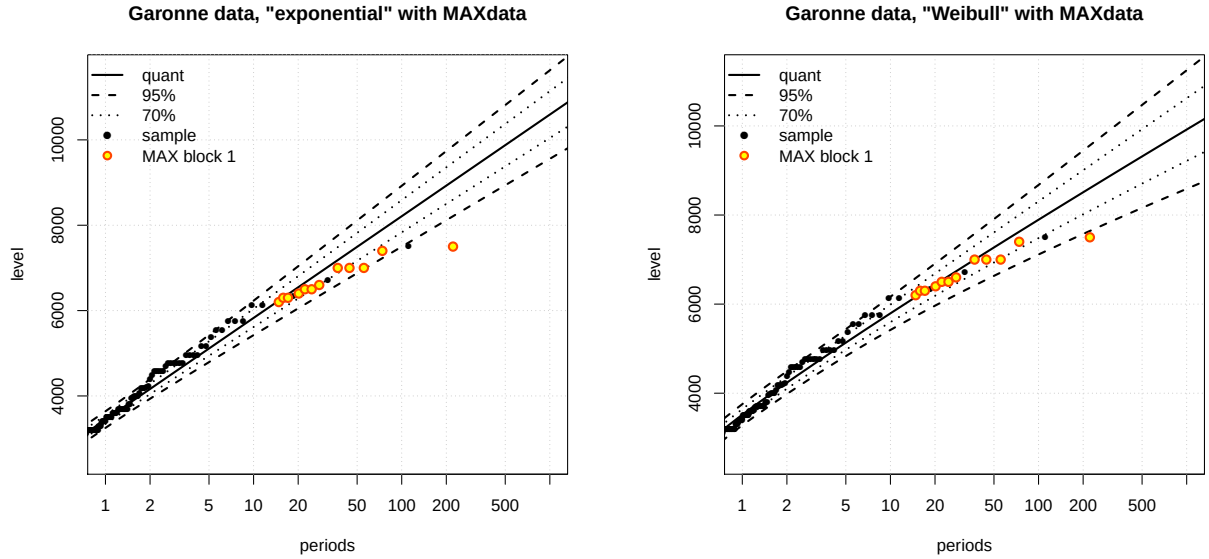

Figure 3.4: Return level plots for the example `Garonne` with two distributions for the excesses and historical data. Specific plotting positions are used to take into account the historical observations. The twp plots can be compared to those of figure 3.1.

3.4.4 Plotting positions

To be displayed on the return level plot (see figure 3.4), MAX or OTS data require suitable plotting positions. Naive plotting positions based on predictions where used in former versions of **Renext**. They are now replaced by more elaborated ones arising from some relevant literature on censored data (Millard and Neerchal 2001). The principle is most easily understood for OTS data.

- If there is only one OTS block with threshold $u_1 > u$ and duration w_1, then we can easily estimate the return period of the level u_1 by counting the total number of exceedances over u_1 (including those during the OT period). The product $\lambda S_X(u_1) = 1/T(u_1)$ is estimated as the number of exceedances divided by the duration $w + w_1$. The plotting positions for the observations above u_1 are determined as usual, see section 3.2.2 above, with the estimated rate $\widehat{\lambda}$ replaced by $1/\widehat{T}(u_1)$. The number of exceedances over u is then estimated by assuming that the observations with levels in (u, u_1) occurred in the w_b years of the block with the known rate for the w years of the OT data. We thus can estimate the return period $T(u)$ of u and then those of the observations with level between u and u_1 using an interpolation.

- When B OTS blocks exist, the threshold u and the B thresholds u_b can without loss of generality be assumed to be ordered as $u_0 < u_1 < \cdots < u_B$ with $u_0 := u$, and thus define $B+1$ slices of levels (u_b, u_{b+1}) for $0 \leqslant b \leqslant B$ with $u_{B+1} := \infty$. The previous computation still applies for the upper slice which correspond to levels in $> u_B$. Starting from this highest slice, one can then estimate by recursion the number of observations falling in each of the slices (u_b, u_{b+1}) for $b = B, B-1, \ldots, 0$, and thus the return periods $T(u_b)$. The computation is similar to that described by Hirsch and Stedinger (1987) for the survival. The plotting positions for the observations in a slice result from an interpolation.

For MAX blocks, the plotting positions are computed by considering a MAX block as an OTS block with it its threshold set near to the smallest observation in the block, i.e. $u_b := Z_{b,r_b} - \epsilon$ in the notations used in (3.2) where ϵ is small. The chosen value of ϵ depends on the data.

For instance, for the data `Garonne` with $B = 1$, the MAX block can be considered as an OTS block with threshold $u_1 = 6200 - \epsilon = 6195$ m^3/s. The total number of observations in the upper slide (u_1, ∞) is $3 + 12 = 15$ (in the OT sample and the MAX block), during $w + w_1 = 65 + 143.09 = 208.09$ years.

So the return period of u_1 is $208.09/15 = 13.9$ years. Now the number of observations in the next slide (u_0, u_1) is estimated by using the rate of such observations during the OT period.

The computations are carried out by the `SandT` function which estimates both the survival S and the return periods T. Details are provided in Deville (2015).

3.4.5 Fitting from Rendata objects

Recall that a S3 class `"Rendata"` is defined in **Renext** in order to represent heterogeneous data with optional block or historical data. An object of class `"Rendata"` contains an OT sample, but also embeds useful pieces of information such as the effective duration for the OT sample or the variable name. It seems sensible to use these indications in a POT model by simultaneously passing them as formal arguments to the fitting function. For instance, when the OT sample of a `"Rendata"` object is used in a fit, the effective duration could consistently be taken from this object. `Renouv` can indeed be used by giving an `x` formal with class `"Rendata"` instead of a numeric vector.

```
> fitWithObj <- Renouv(x = Garonne)
```

Note that the threshold is taken from the `Rendata` object's `OTdata` part, and will generally be too small for a POT modelling. It can be changed simply

```
> fitWithObj1 <- Renouv(x = Garonne, threshold = 3000)
```

Similarly, the effective duration of the object can be shortcut by providing the `effDuration` formal argument in the call. The distribution of the excesses can be set in the usual way. In all cases, the `summary` method should be invoked on the fitted object.

Using `"Rendata"` objects passed as `x` formals can simplify the task of fitting many datasets files if these are read with the `readXML` function.

3.5 GPD excesses

3.5.1 Standard POT

Of course, the `Renouv` function can be used with a GPD for the excesses.

```
> fit.GPD <- Renouv(x = Garonne$OTdata$Flow, effDuration = Garonne$OTinfo$effDuration,
                    threshold = 3000, distname.y = "GPD",
                    main = "Garonne data, \"GPD\"")
> coef(fit.GPD)
```

```
      lambda        scale        shape
  1.5076923 1160.1536041   -0.1226653
```

The fitted distribution has a negative shape $\widehat{\xi} = -0.12$, hence has a finite upper end-point, which makes a major difference with the Weibull fit. The maximal level is thus estimated as $u - \widehat{\sigma}/\widehat{\xi} = 12458$.

As before, we can use the historical information in `Garonne`

```
> fit.GPD.H <- Renouv(Garonne, threshold = 3000, distname.y = "GPD",
                      main = "Garonne data, \"GPD\" with MAXdata")
> coef(fit.GPD.H)
```

```
      lambda        scale        shape
  1.5547065 1321.6227580   -0.1853906
```

The maximal level is now estimated as 10129.

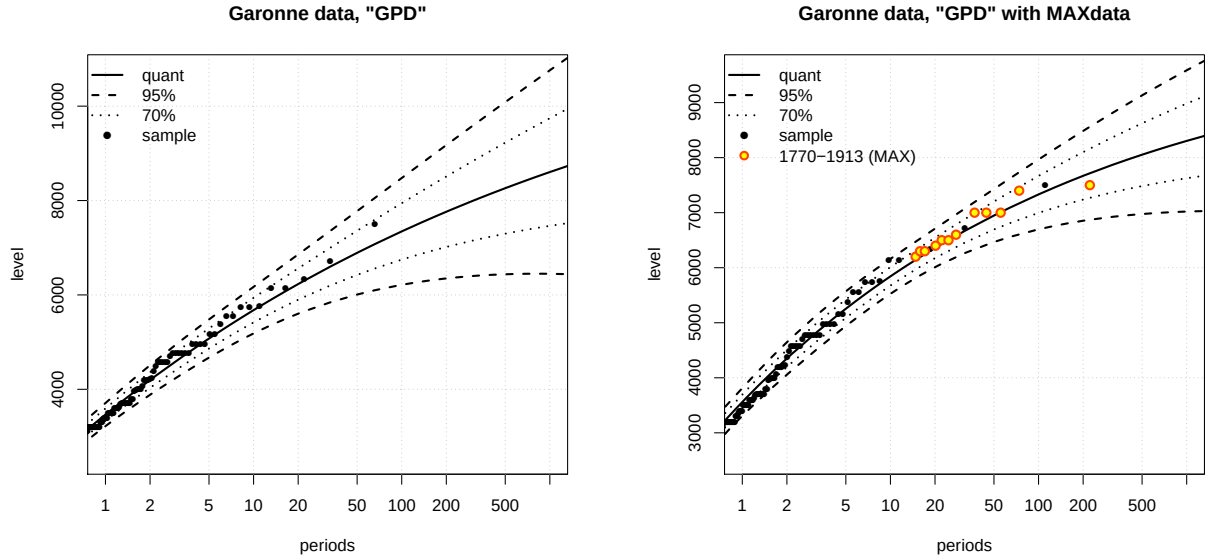

Figure 3.5: Using GPD excesses for `Garonne`.

3.5.2 Several parameterisations

The Lomax and maxlo[6] distributions are re-parameterisations of the GPD with shape $\xi > 0$ and $\xi < 0$ respectively, see B.3.7 and B.3.8. In both cases, the distribution involves a scale parameter $\beta > 0$ and a shape parameter $\alpha > 0$. The exponential corresponds to a limit when $\alpha \to \infty$ and $\beta \to \infty$ while β/α tends to a finite limit $\sigma > 0$.

```
> fit.maxlo <- Renouv(x = Garonne$OTdata$Flow,
                      effDuration = Garonne$OTinfo$effDuration,
                      threshold = 3000, distname.y = "maxlo")
> coef(fit.maxlo)
     lambda       shape       scale
   1.507692    8.152266 9457.880896
```

The scale parameter of the maxlo distribution is upper end-point for the excess, hence the upper end-point for the level is estimated as $u + \widehat{\beta} = 12458$ as it was with the `GPD` distribution.

Choosing the Lomax distribution would here give an error

```
> trylomax <- try(fmaxlo <- Renouv(x = Garonne$OTdata$Flow,
                                   effDuration = Garonne$OTinfo$effDuration,
                                   threshold = 3000, distname.y = "lomax"))
> class(trylomax)
[1] "try-error"
> cat(trylomax)
Error in flomax(x = y, info.observed = FALSE) :
  CV < 1. Estimation impossible for "lomax"
```

When only OT data are used in the estimation, only one of the two distributions Lomax and maxlo can be fitted with `Renouv` without producing an error. Indeed, a finite ML estimator exists for the Lomax distribution if and only if the empirical coefficient of variation $\widehat{\mathrm{CV}}$ is greater than 1, while a finite ML estimator exists for the maxlo if and only $\widehat{\mathrm{CV}} < 1$. Note that for a sample of size n of a GPD with

[6] We use this new name for an important yet apparently unnamed distribution. While the Lomax distribution is named after K.S. Lomax, no Mrs or Mr Maxlo seems famous yet for having used it, hence the name does not require a capital letter.

$\xi > 0$ small, the probability that $\widehat{\mathrm{CV}} < 1$ hence that the ML estimation of the Lomax is impossible is not always negligible. For $\xi = 0$ the probability that $\widehat{\mathrm{CV}} < 1$ is computed by the `pGreenwood1` function.

Again, when MAX or OTS data are used only one of the two distributions Lomax and maxlo can be fitted

```
> fit.maxlo.H <- Renouv(Garonne, threshold = 3000, distname.y = "maxlo")
> coef(fit.maxlo.H)
     lambda       shape       scale
  1.554721    5.391524 7126.255255
```

The situation can be quite confusing when MAX or OTS data are used because it is possible then that the sign of the ML estimator of the GPD shape ξ differs depending on whether the block data (MAX and OTS) are used or not. In such a case, it is simpler to directly use `GPD`.

The return levels for the GPD or its re-parameterisation as Lomax or maxlo are identical. Inasmuch the delta method is used, the confidence intervals on the RL are identical as well, up to small numerical differences.

```
> predict(fit.GPD, newdata = c(100, 200))
    period    quant     L.95     U.95     L.70     U.70
100    100 7345.885 6214.780 8476.990 6747.754 7944.016
200    200 7762.568 6350.062 9175.073 7015.632 8509.504
> predict(fit.maxlo, newdata = c(100, 200))
    period    quant     L.95     U.95     L.70     U.70
100    100 7345.885 6214.779 8476.991 6747.754 7944.016
200    200 7762.568 6350.063 9175.072 7015.632 8509.503
```

3.6 Fixing parameter values

3.6.1 Problem

In some situations one may want to fix one or several parameters in the distribution of excesses and still perform a ML estimation for the remaining parameters. For instance, the `shape` of a Weibull distribution can be fixed while the `scale` is to be estimated.

The `Renouv` function supports fixed parameters, with some limitations. In the current version, the HPP rate parameter λ *can not be fixed*, and *at least one parameter must be estimated in the excesses part*. Thus the full model must have at least two non-fixed parameters.

The specification of the fixed parameter is done using the `fixed.par.y` formal argument in `Renouv`. Its value must be a named vector list with names in the distribution parnames. As a general rule[7], the non-fixed (estimated) parameters must be given using the `start.par.y` arg with a similar list value.

3.6.2 Example

The fixed parameter option can work with or without historical data in the same manner.

```
> fit.weib.fixed.H <-
    Renouv(x = Garonne$OTdata$Flow,
           effDuration = 65, threshold = 3000,
           MAX.data = list(Garonne$MAXdata$Flow),
           MAX.effDuration = Garonne$MAXinfo$duration,
           distname.y = "weibull",
           fixed.par.y = c(shape = 1.4),
           start.par.y = c(scale = 2000),
           trace = 0,
           main = "Garonne data, \"Weibull\" with MAXdata and fixed shape")
```

[7] In some special cases, this is unnecessary but harmless.

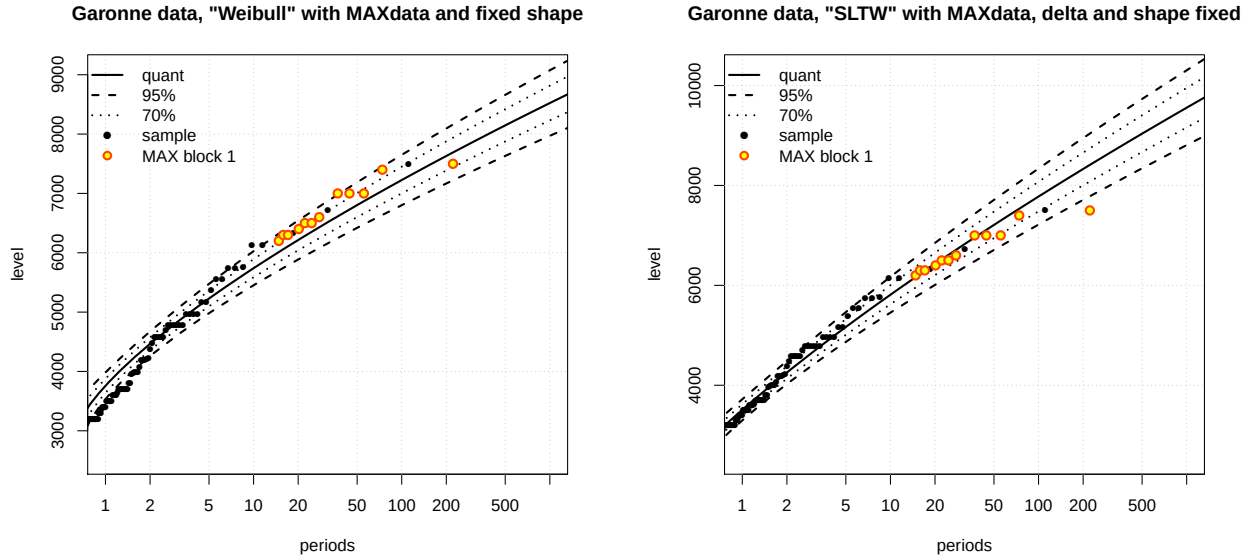

Figure 3.6: Return level plots for the example `Garonne` with two distributions with **fixed parameters** (and historical data).

```
> fit.weib.fixed.H$estimate
```

```
    lambda      shape      scale
  1.579748   1.400000 1326.602872
```

With some distributions such as the SLTW some parameters *must* be fixed. Here the shift parameter `delta` is fixed to $\delta = 2800$ m^3/s meaning that we believe that excesses over $u - \delta = 500$ are Weibull, even if we only know excesses over the threshold $u = 3000$ m^3/s.

```
> fit.SLTW.H <-
    Renouv(x = Garonne$OTdata$Flow,
           effDuration = 65, threshold = 3000,
           MAX.data = list(Garonne$MAXdata$Flow),
           MAX.effDuration = Garonne$MAXinfo$duration,
           distname.y = "SLTW",
           fixed.par.y = c(delta = 2800, shape = 1.4),
           start.par.y = c(scale = 2000),
           main = "Garonne data, \"SLTW\" with MAXdata, delta and shape fixed")
```

When some parameters are fixed the covariance contains structural zeros, and consequently the correlation matrix contains non-finite coefficients.

```
> fit.SLTW.H$cov
```

```
           lambda delta shape       scale
lambda  0.02270571     0     0   -2.950236
delta   0.00000000     0     0    0.000000
shape   0.00000000     0     0    0.000000
scale  -2.95023609     0     0 9881.428891
```

3.6.3 All parameters known

Using `Renouv` with its `fixed.par` argument is possible when some of the parameters are known, but not all of them. The `RenouvNoEst` function can be used to create a `Renouv` object with all its parameters known, including the Poisson rate λ. This can be useful to use `plot` or `predict` with known parameters. See the help `?RenouvNoEst` for an example.

3.7 Likelihood Ratio tests

3.7.1 Using the anova method

In section 3.1, two POT models were fitted using the same `Garonne` data with exponential and Weibull distributions for the excesses. The ML estimate for the Weibull shape was $\widehat{\alpha} = 1.06$, and since the exponential distribution corresponds to Weibull with shape $\alpha = 1$, it seems natural to test the hypothesis $H_0 : \alpha = 1$ against the alternative $H_1 : \alpha \neq 1$.

More generally, we can consider two *nested models*: the null hypothesis H_0 imposes some restrictions[8] on the parameter vector $\boldsymbol{\theta}$ which is unrestricted under the alternative H_1. The Likelihood Ratio (LR) statistic is obtained by maximising the restricted and unrestricted likelihoods as

$$\text{LR} := \frac{\text{maximal likelihood under } H_0}{\text{maximal likelihood under } H_1},$$

with values $\text{LR} \leqslant 1$. It is often convenient to use the test statistic $W := -2 \log \text{LR}$, which takes values $W \geqslant 0$ and is the difference of the *deviances* $D := -2 \log L$. A large value for W tells that H_0 should be rejected. Under some general conditions, it can be proved that W has asymptotic distribution $\chi^2(r)$ where r is the number of independent scalar restrictions imposed by the null hypothesis, so $r = 1$ in the exponential vs Weibull case.

LR tests are often made available in R packages through the `anova` method which must be implemented for the class of fitted models that we wish to test. The `anova` methods compare nested models fitted *with the same data*. In a general context, it is enough to extract the log-likelihood and the number of parameters for each model. The statistic W above is computed and compared to its asymptotic distribution. In **Renext**, the `anova` method was implemented for the `Renouv` class. For instance using two `Renouv` objects created before

```
> anova(fit.exp, fit.weibull)
Models:
  o 'fit.exp' with exceedances dist. "exponential"
  o 'fit.weibull' with exceedances dist. "weibull"

Method used:  asymptotic approximation

Analysis of Deviance Table

            df deviance       W Pr(>W)
fit.exp      2   1556.0
fit.weibull  3   1555.4 0.58725 0.4435
```

which tells that the null hypothesis of an exponential distribution must here be rejected at any level $\alpha \leqslant 0.44$ and accepted for larger values of α. So in practice we would here accept H_0.

Note that the same test statistic W could have been used to test H_0 against the thin-tailed Weibull alternative $H_1 : \alpha > 1$, which seems in better accordance with the data. But the asymptotic distribution of W would then no longer be χ^2, and be that of the product BC of two independent random variables with Bernoulli and chi-square distributions $B \sim \texttt{Ber}(1/2)$ and $C \sim \chi^2(1)$. The reason is that the tested value of the parameter is now on the boundary of parameter domain, and the statistic W takes the value 0 when $\widehat{\alpha} < 1$. For $w > 0$ we have $\Pr\{BC > w\} = \Pr\{B = 1\} \Pr\{C > w\}$, so the p-value for $H_1 : \alpha < 1$ is about 0.22 and the exponentiality would thus still be rejected.

3.7.2 LR test for the GPD family

In the standard POT framework, excesses are assumed to follow a GPD, say $\texttt{GPD}(0, \sigma, \xi)$. Depending on the sign of ξ, very different tail behaviour and return levels will be obtained. Not infrequently, the ML estimate $\widehat{\xi}$ is close to zero thus suggesting to test the null hypothesis $\xi = 0$ corresponding to the

[8] Equality constraints.

exponential distribution. Three (composite) alternative hypotheses are often of interest

$$\mathrm{H}_0 : \xi = 0 \,(\text{exponential}) \quad \text{against} \quad \mathrm{H}_1 : \begin{cases} \xi \neq 0 & (\text{GPD}), \\ \xi > 0 & (\text{Lomax}), \\ \xi < 0 & (\text{maxlo}). \end{cases}$$

The LR still can be used as test statistic. However, a well-known problem is that the distribution of the LR ratio has a *very slow convergence* to its asymptotic distribution in the present context[9]. More than 100 excesses are typically needed to obtain a p-value with a two-digit precision. For instance, with the Lomax alternative, the probability to obtain $W = 0$ under H_0 is the probability that $\widehat{\mathrm{CV}} < 1$ and is given by the `pGreenwood1` function related to the Greenwood's statistic. For $n = 50$, we get $\Pr\{W = 0\} \approx 0.6$ while the asymptotic probability mass is 0.5.

To overcome this problem of slow convergence, the distribution of the test statistic for a given sample size n has been computed by Monte-Carlo simulations and a statistical model was fitted to allow a more precise evaluation of the distribution of W (Deville 2015). This approximation is used in the `LRexp.test` and also in the `anova` method for the `Renouv` class as far as no MAX or OTS data are used.

```
> anova(fit.exp, fit.GPD)
Models:
  o 'fit.exp' with exceedances dist. "exponential"
  o 'fit.GPD' with exceedances dist. "GPD"

Method used:  numerical approximation

Analysis of Deviance Table

        df deviance       W Pr(>W)
fit.exp  2     1556
fit.GPD  3     1555 0.99712 0.3444
> anova(fit.exp, fit.maxlo)
Models:
  o 'fit.exp' with exceedances dist. "exponential"
  o 'fit.maxlo' with exceedances dist. "maxlo"

Method used:  numerical approximation

Analysis of Deviance Table

          df deviance       W Pr(>W)
fit.exp    2     1556
fit.maxlo  3     1555 0.99712 0.2343
```

So none of the two tests would reject exponentiality.

The LR test also works with heterogeneous data. However the usual asymptotic approximation will still be used as soon as the compared fits used MAX or OTS block data.

```
> anova(fit.exp.H, fit.GPD.H)
Models:
  o 'fit.exp.H' with exceedances dist. "exponential"
  o 'fit.GPD.H' with exceedances dist. "GPD"

Method used:  asymptotic approximation

Analysis of Deviance Table

          df deviance      W Pr(>W)
fit.exp.H  2   1714.2
fit.GPD.H  3   1710.0 4.1515  0.318
```

[9] See e.g. Kozubowski, Panorska, Qeadan, Gershunov, and Rominger (2009)

```
> anova(fit.exp.H, fit.maxlo.H)

Models:
  o 'fit.exp.H' with exceedances dist. "exponential"
  o 'fit.maxlo.H' with exceedances dist. "maxlo"

Method used:  asymptotic approximation

Analysis of Deviance Table

            df deviance      W Pr(>W)
fit.exp.H    2   1714.2
fit.maxlo.H  3   1710.0 4.1515  0.159
```

So when the historical data are used, the exponentiality hypothesis is still accepted against the $\xi < 0$ alternative, but the p-value is now smaller.

Concerning the specific application to the `Garonne` data, it must be said that the tests give pretty different results when the threshold varies in the range [2500, 3500]. For instance, the exponentiality is rejected at the 5% level when the threshold is taken as 3200.

3.7.3 Other tests for the exponential-GPD context

As far as only OT data are used, two other tests of **Renext** can be used to test the exponential $\xi = 0$ against the Lomax alternative. These tests use the squared coefficient of variation and the Jackson's statistic and named CV^2 *test* (or *WE test*, for Wilk's Exponentiality test) and *Jackson's test*. As for the LR test, the distribution of the test statistic for both of these tests is approximated thanks to a statistical model fitted on simulated values. In accordance with the results of Kozubowski *et al.* (2009), the Jackson's test was found on simulations to be nearly as poweful as the LR test, both having greater power than the CV^2 test.

We can use these two tests for the excesses of the `Garonne` example, although the Lomax alternative $H_1 : \xi > 0$ is clearly not well supported by the data

```
> X <- Garonne$OTdata$Flow
> Y <- X[X > 3000]
> c(CV2 = CV2.test(Y)$p.value, Jackson = Jackson.test(Y)$p.value)

    CV2 Jackson
      1       1

> Y <- X[X > 3300]
> c(CV2 = CV2.test(Y)$p.value, Jackson = Jackson.test(Y)$p.value)

    CV2 Jackson
  0.581   1.000
```

So both tests accept $H_0 : \xi = 0$ against the Lomax alternative. The computed p-value in these tests can be 1 (exactly), which may seem unusual. The reason is that the p-value is computed with a precision which is not greater than two digits and is rounded. This is not a concern for large p-values (≈ 1).

Chapter 4

POT and block data

Although devoted to POT, **Renext** can be used for some analyses involving block data: block maxima and r largest. These possibilities are restricted to models arising form the marked process with a distribution of levels in the GPD family, including exponential, Lomax or maxlo distributions.

4.1 Example: Venice data

Consider the *Venice* data, concerning the sea-level at Venice (in cm). This dataset is used as example 1.5 in Coles' book, where it used in section 3.5.3 for a r largest analysis. Variants of this dataset are provided by several CRAN packages[1]. We will use here the *data frame* object named `venice` from **evd**.

```
> head(venice, n = 3)
       1   2   3   4   5  6  7  8  9 10
1931 103  99  98  96  94 89 86 85 84 79
1932  78  78  74  73  73 72 71 70 70 69
1933 121 113 106 105 102 89 89 88 86 85
> range(venice, na.rm = TRUE)
[1]  69 194
```

We have 51 years of data from 1931 to 1981 and for each year the r largest observations for $r \leqslant 10$, given in descending order. Missing observations are present in year 1935, and then given as `NA`.

We may regard the observations as arising from a POT model, and hence use them as MAX data blocks, all with duration equal to one year. For that aim, we need to build a list with one element by year: a numeric vector with the r largest values observed that year, e.g. $r = 5$.

```
> r <- 5
> MAX.data <- as.list(as.data.frame(t(venice[ , 1:r])))
> MAX.data <- lapply(MAX.data, function(x) x[!is.na(x)])
> MAX.effDuration <- rep(1, length(MAX.data))
> head(MAX.data, n = 2)
$`1931`
[1] 103  99  98  96  94

$`1932`
[1] 78 78 74 73 73
> head(unlist(lapply(MAX.data, length)))
1931 1932 1933 1934 1935 1936
   5    5    5    5    5    5
```

[1] E.g. with a `Year` column in **ismev**.

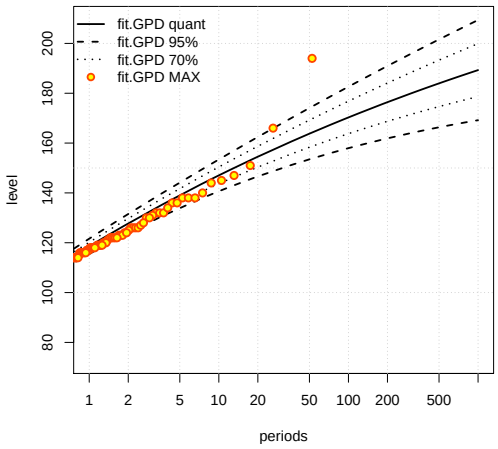

Figure 4.1: Fit using r largest values from the `venice` data set. The plotting positions are obtained as explained in section 3.4.4.

Note that the transposition method `t` returns a *matrix*, and a coercion to `data.frame` is required to get a list.

Since all observations are > 66 cm, we can consider a POT model with $u \leqslant 66$ to use all available information. Then we can the use the `Renouv` function

```
> fit.GPD <- Renouv(x = NULL,
                    MAX.data = MAX.data, MAX.effDuration = MAX.effDuration,
                    distname.y = "GPD", threshold = 66,
                    numDeriv = FALSE, trace = 0, plot = FALSE)
> coef(fit.GPD)
    lambda      scale      shape
27.5229835 18.2785841 -0.0878973
```

We implicitly supposed that for each year the r provided observations are the largest, even when `NA` are found. Since the fitted object has class `"Renouv"`, the `plot` method shows the usual RL plot

```
> plot(fit.GPD)
```

see fig. 4.1. The plotting positions for the points are obtained as explained in section 3.4.4.

For `Renouv` objects using a distribution in the GPD family, a "translation" of the parameters to GEV parameters for block maxima is provided in the `MAX` element of the result.

```
> fit.GPD$MAX
$distname
[1] "gev"

$blockDuration
[1] 1

$estimate
        loc       scale       shape
118.5643750  13.6583175  -0.0878973

$sigma
       loc      scale      shape
1.56605550 0.77530725 0.03293652
```

```
$cov
              loc      scale        shape
loc    2.452529835 0.74260886 -0.005367611
scale  0.742608862 0.60110134  0.012711216
shape -0.005367611 0.01271122  0.001084814
```

The translation provides the estimated parameters of a GEV distribution $\texttt{GEV}(\mu^\star, \sigma^\star, \xi^\star)$ and must not be confused with those of the $\texttt{GPD}(u, \sigma, \xi)$ for the excesses of the POT model. The estimated values of the shape parameters ξ and $\xi^\star$ are the always the same, but the estimated scale parameters differ and $\mu^\star$ is not equal to the threshold u. The GEV distribution can be used as usual in the r largest context. If a different threshold had been used, e.g. $u = 50$ the POT parameters would have been very different, but the GEV parameters would have been the same.

4.2 Using `fGEV.MAX`

Beside `Renouv`, the `fGEV.MAX` function was added to **Renext** to perform the estimation of a GEV distribution $\texttt{GEV}(\mu^\star, \sigma^\star, \xi^\star)$ from block maxima or from r largest observations, using in both cases blocks having *the same duration*. This function uses the representation of the GEV as the distribution of the block maxima in a POT model with GPD excesses. The distribution is no longer a formal argument, and nor is the threshold u which is chosen depending on the data. In the likelihood L, the POT rate λ has been concentrated out, so L depends only on the two parameters σ and ξ of the POT model. Although computation time is not really a concern here, the optimisation is much faster than the usual one which uses the three GEV parameters $\mu^\star$ $\sigma^\star$ and $\xi^\star$. The hessian at the optimum is computed using analytical derivatives rather than numerical derivatives. Details can be found in Deville (2015).

```
> fit.GEV <- fGEV.MAX(MAX.data = MAX.data, MAX.effDuration = MAX.effDuration)
> fit.GEV$estimate
         loc        scale        shape
7.286931e+01 1.675720e+01 3.469447e-18
> require(ismev)
> fit.GEVref <- rlarg.fit(venice, show = FALSE)
> fit.GEVref$mle
[1] 120.5479027  12.7840265  -0.1129418
```

Note that `fGEV.MAX` returns a simple `list` object and the methods such as `coef` can not be used.

4.3 Computing the r largest observations

4.3.1 Coping with gaps

Given observations $[T_i, X_i]$ of the marked process, it seems quite easy to compute block maxima or r largest – in other words, to aggregate the data. However, in some cases gaps are present and must be taken into account in the aggregation. When known gaps exist in the data, they should be carefully inspected to assess their possible impact on the estimation.

The `OT2MAX` function was designed to compute the r largest observations as well as some diagnostics when known missing periods exist. The formal argument `OTdata` of this function corresponds to a data frame with two columns: a `date` column and a column containing the variable X, as in the `OTdata` element of an object of the `"Rendata"` class. The `maxMissingFrac` gives the maximum fraction (between 0 and 1) of gap within a block. When this fraction is exceeded in a block, the returned block observations are `NA`. By default, the function produces a plot as shown in figure 4.2.

The `Dunkerque` data set used here is similar to `Brest`: it also concern sea surge and embeds missing periods, but the data cover a smaller period of time.

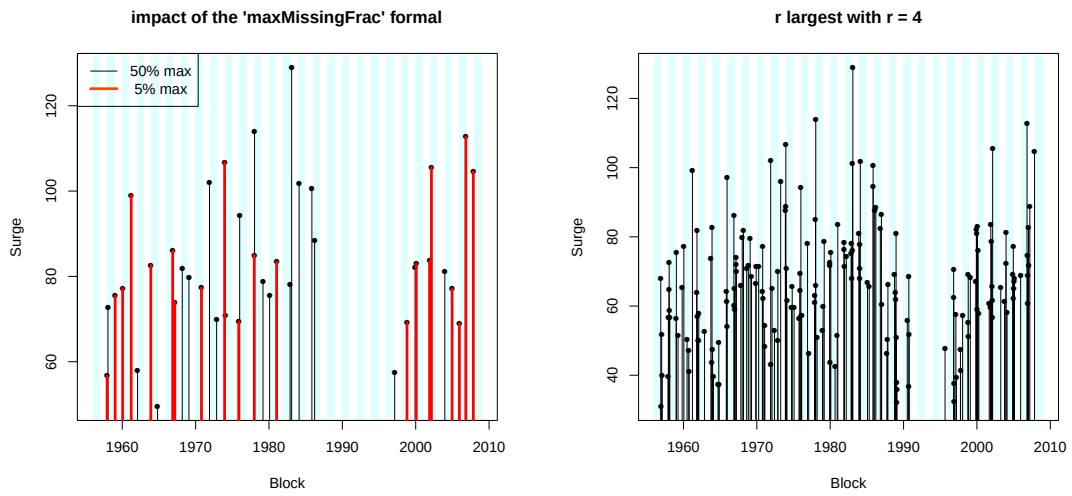

Figure 4.2: Left: block maxima for `Dunkerque` with `maxMissing` set to `0.5` and `0.05`. Right: the r largest observations for $r = 4$. Each annual block can contain up to 4 observations.

```
> RD <- Dunkerque
> OTdata <- RD$OTdata; OTmissing <- RD$OTmissing
> ## allow up to 50% of gap within each block, or only 5%?
> MAX1 <- OT2MAX(OTdata = OTdata, OTmissing = OTmissing,
                 maxMissingFrac = 0.5,
                 main = "impact of the 'maxMissingFrac' formal")
> MAX2 <- OT2MAX(OTdata = OTdata, OTmissing = OTmissing, dataFrames = TRUE,
                 prefix = "Max", maxMissingFrac = 0.05, plot = FALSE)
> lines(MAX2$MAXdata$date, MAX2$MAXdata$Surge, type = "h", col = "red", lwd = 3)
> legend("topleft", lw = c(1, 3), col = c("black", "orangered"),
         legend = c("50% max", " 5% max"))
```

The `OTmissing` element of `Dunkerque` reports quite large gaps in the nineties (e.g. from October of 1992 to July of 1995). With the larger value `maxMissingFrac = 0.5`, up to 50% of a block can be a gap, and fewer `NA` block observations result than when a small value of `maxMissingFrac` is used.

```
> ## r largest obs for r = 4
> MAX3 <- OT2MAX(OTdata, OTmissing = OTmissing, MAX.r = 4,
                 maxMissingFrac = 0.9,
                 dataFrames = FALSE, trace = TRUE,
                 main = "r largest with r = 4")
Number of events by block
1956 1957 1958 1959 1960 1961 1962 1963 1964 1965 1966 1967 1968 1969 1970 1971
   2    7   11   13   12   13   11   15    6    9   18   22   29   18   17   11
1972 1973 1974 1975 1976 1977 1978 1979 1980 1981 1982 1983 1984 1985 1986 1987
  13   16    9   14   11   15   10   17   14   29   29   29   21   17   17   13
1988 1989 1990 1991 1992 1993 1994 1995 1996 1997 1998 1999 2000 2001 2002 2003
  17    4    5   NA   NA   NA   NA    1    7    8   14   16   19   28   22   17
2004 2005 2006 2007 2008
  24   19   20   31   NA
> ## restrict the period
> MAX4 <- OT2MAX(OTdata, OTmissing = OTmissing, MAX.r = 4,
                 start = "1962-01-01",
                 end = "1990-01-01",
                 maxMissingFrac = 0.9,
                 dataFrames = FALSE, trace = TRUE,
                 main = "r-largest with r = 4 with given 'start' and 'end'")
```

```
Number of events by block
1962 1963 1964 1965 1966 1967 1968 1969 1970 1971 1972 1973 1974 1975 1976 1977
  11   15    6    9   18   22   29   18   17   11   13   16    9   14   11   15
1978 1979 1980 1981 1982 1983 1984 1985 1986 1987 1988 1989
  10   17   14   29   29   29   21   17   17   13   17    4
> ## use in a block maxima analysis, as if there were no gaps.
> fitDunk <- fGEV.MAX(MAX.data = MAX3$data,
                      MAX.effDuration = rep(1, length(MAX3$effDuration)))
```

4.3.2 Diagnostics for gaps

A quite common problem with gaps is that they can conceal a seasonal effect: the probability that a randomly selected time T falls in a gap can differ according to the season of T. Even if the gaps are really exogenous, this may cause a bias in models, either POT or block maxima. For example severe storm surges occur mainly in winter, so a gap with a six month duration will probably lead to loose more of large observations when it is located in winter rather than in summer. This can be controlled by estimating the probability that T falls in a gap according to its location in the year. The `plotType` argument of `OT2MAX` provides an useful related diagnostic.

```
>  ## plot the gap rate
> MAX5 <- OT2MAX(OTdata = OTdata, OTmissing = OTmissing,
                 maxMissingFrac = 0.5,
                 main = "probability of being in a  gap",
                 plotType = "gap")
```

The plot (fig 4.3, left) shows that the probability of falling in a gap does not have a very strong variation along one year, and broadly ranges from 1/5 to 1/3. The horizontal segments in gray show jitterised versions of the gap rates for all the year $\times$ month pairs. Many of these rates are equal to 0 (no gap in the month) and several of them are equal to 1 (fully missing month).

A complementary diagnostic is obtained by plotting a yearly time series for each month of a year (as in fig 4.3 right), thus showing the evolution of the gap fraction in a given month.

```
> require(lattice)
> xyplot(MAX5$monthGapTS[ , c(1:3, 10:12)], type = "h", lwd = 2, ylim = c(0, 1))
```

The **lattice** base package (Sarkar 2008) used here provides nice plots for multiple time series.

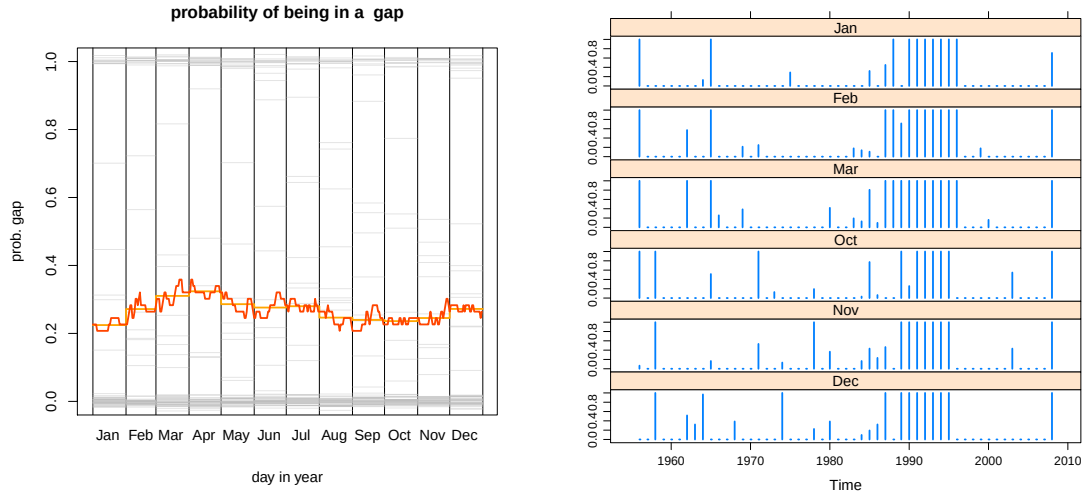

Figure 4.3: Controlling the possible impact of gaps. Left: the probability that a day in the year falls in a gap is shown in orange. The horizontal segments in gray show a jitterised version of the gap fraction for a year/month combination. Right: yearly time series of gap fraction for six months.

Chapter 5

Renext graphics

Renext graphics are based on the **graphics** package and can hence be customised as usual. However, adding points to a RL plot is not always easy: when several types of data exist, the determination of the plotting positions requires quite technical computations as performed by the `SandT` function.

A number of supplementary functions are provided to facilitate the most frequent modifications of RL plots. A widespread practice is showing on a same RL plot the data (sample points) and some elements of a fitted model: quantile line, confidence bounds. When several kinds or sources of data are used in the fit, it is important to display them in such a way that the different sources are readily identified. It arose from users practice that representing *several* fits on the same RL plot through a `lines` or `points` method is often a valuable option, provided that the fits can be identified by colour or line type, and that a legend is shown: the `RLpar` function and the `RLlegend*` functions have been designed for these two tasks.

5.1 The `plot` and `lines` methods

The `plot` and `lines` methods can be used to build return level plots showing several elements: quantile line (or return level line), sample points, ... A plot can be obtained by adding elements to an existing plot with the `lines` method. Recall that the dispatch mechanism of S3 applies when the first argument (here `x`) of the generic function (here `lines`) is an object of a class `"Renouv"` for which a method has been implemented (here `"Renouv"`).

```
> fitG <- Renouv(Garonne, distname.y = "GPD", plot = FALSE)
> ## specify pch background color for MAX block #1
> plot(fitG, show = list(OT = TRUE, MAX = FALSE), main = "use plot, then lines")
> lines(fitG, show = list(MAX = TRUE))
```

The `show` argument of `plot` and `lines` is used to select the elements in `"Renouv"` object passed in `x` (here `Garonne`) that will be shown. This is a named list having logical vectors as its elements. By playing with the `show` formal, we can build a plot in several steps as here: first plot without `MAX` blocks, then add them to the plot. Note that the legend is not updated when adding elements to the graph, motivating the `RLlegend*` mechanism described later in section 5.3.

5.2 The `RLpar` function

5.2.1 Basics

The `RLpar` function is used to change some of the graphical parameters such as colours, line types or plotting characters. It returns a hierarchical list designed to be used as a value of the `par` formal argument of the `plot` and `lines` methods. The hierarchical structure of this list can be shown using `str`, but this would take too much space here, so we will use `names`

```
> names(RLpar())
```

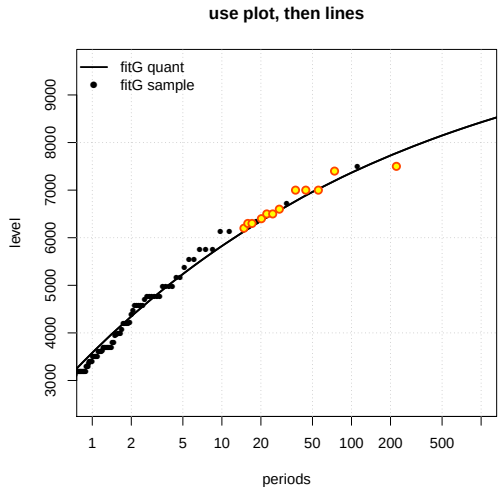

Figure 5.1: Adding historical information with `lines`. Note that the points added using `lines` are not described in the legend.

```
[1] "quant" "OT"    "conf"  "MAX"   "OTS"
> str(RLpar()$quant)
List of 4
 $ type: chr "l"
 $ col : chr "black"
 $ lwd : num 2
 $ lty : chr "solid"
> names(RLpar()$MAX)
 [1] "block1"  "block2"  "block3"  "block4"  "block5"  "block6"  "block7"
 [8] "block8"  "block9"  "block10"
```

The hierarchical structure is displayed in table 5.1. The list can be flattened by using `unlist`, producing element names as shown in the last column of table 5.1.

```
> ## display 10 names
> head(names(unlist(RLpar())), n = 10)
 [1] "quant.type"     "quant.col"      "quant.lwd"      "quant.lty"
 [5] "OT.col"         "OT.pch"         "OT.cex"         "OT.bg"
 [9] "conf.conf1.lty" "conf.conf1.col"
```

So `unlist` coerces the hierarchical list into a character vector with named elements. In the elements names, the dot `.` indicate the hierarchical levels that have been flattened. For instance, the element `quant.type` is the coercion of the `quant$type` element of the hierarchical list `RLpar()`. Using this dot separated format, we can easily change the value of any graphical parameter appearing in the list.

```
> newPar <- RLpar("quant.col" = "azure")
> unlist(newPar$quant)
   type     col     lwd     lty
    "l" "azure"     "2" "solid"
```

The use of `RLpar` is not totally unlike that of the `par` function of the **graphics** package; however `RLpar` does not alter the value of a variable outside of the global environment as `par` does. The normal use of `RLpar` is as a value for the `par` formal argument within a call to `plot` or `lines` methods, with the aim of encapsulating the graphical parameters settings. Here is an example.

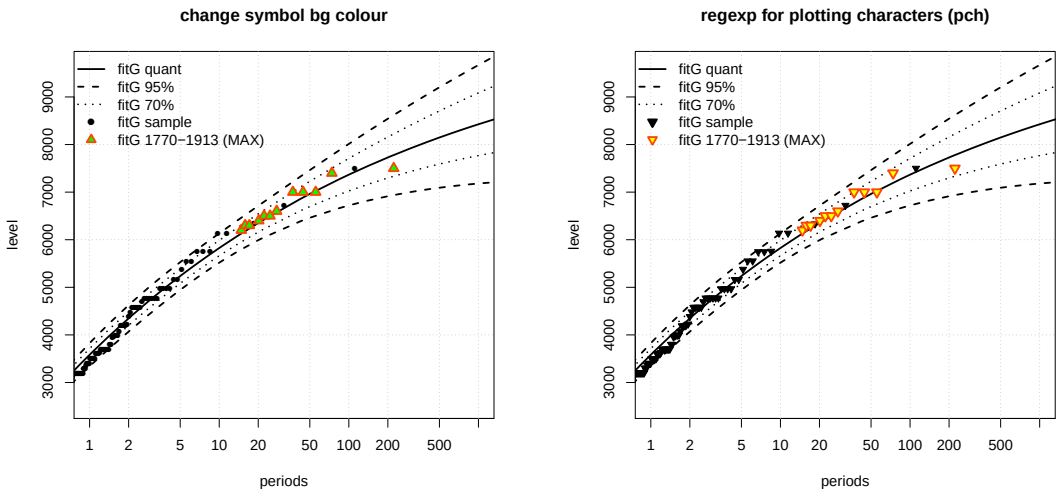

Figure 5.2: Using a `par` formal with `RLpar`. Note that on the plot at the right all the plotting characters have been changed.

```
> ## specify pch background colour for MAX block #1
> plot(fitG, par = RLpar(MAX.block1.bg = "green", MAX.block1.pch = 24),
       main = "change symbol bg colour")
```

The given values for the parameters must be chosen with care since they are not controlled. For instance, giving the value `"blue"` for a `pch` parameter will cause no error or warning but will most probably lead to an unwanted result. Note that as seen in table 5.1, the graphical parameters can be numeric or character[1]. Character values for plotting characters (e.g. in `pch = "+"`) should not be used, because they are likely to create problems in legends. They can be replaced by an equivalent numeric (e.g. in `pch = 3`).

With a package version `>= 2.2-0`, regular expressions can be used as well to change *several* graphical parameters. For instance, in

```
> newPar <- RLpar("OTS.block[0-9]+.col" = "red")
> newPar$OTS$block1$col
[1] "red"
```

we turn to red the colour of *all* the symbols used for the OTS blocks. We can as well use only "nabla" triangles (∇, `pch = 25`) as plotting characters with

```
> plot(fitG, par = RLpar("*.pch" = 25), main = "regexp for plotting characters (pch)")
```

which produces the plot at the right of figure 5.2.

By combining the two formals `show` and `par` of the `plot` and `lines` methods, we can easily change the styles of the elements of a plot, see section 5.4 later.

5.3 The `RLlegend*` functions

A `plot` statement can contain directives to plot several graphic elements: quantile lines, sample points, ... each generating a line in the legend provided that the `legend` formal is `TRUE`. To a certain extend, the text labels in the legend can be changed by using named elements in the lists or vector.

The `RLlegend*` functions are used to add a legend to a return level plot which is built by steps via `lines`.

[1] To a certain extend, they also can be R language to be evaluated. E.g. `rgb(0.1, 0.2, 0.9)` can be used to specify a colour.

level 1	level 2	level 3	value	full name
quant				
	"type"		"l"	quant.type
	"col"		"black"	quant.col
	"lwd"		2	quant.lwd
	"lty"		"solid"	quant.lty
OT				
	"col"		"black"	OT.col
	"pch"		16	OT.pch
	"cex"		0.8	OT.cex
	"bg"		"black"	OT.bg
conf				
	conf1			
		"lty"	2	conf.conf1.lty
		"col"	"black"	conf.conf1.col
		"lwd"	2	conf.conf1.lwd
	conf2	(list)		
	⋮			
	conf6	(list)		
MAX				
	block1			
		"col"	"orangered"	MAX.block1.col
		"pch"	21	MAX.block1.pch
		"cex"	1.1	MAX.block1.cex
		"lwd"	2	MAX.block1.lwd
		"bg"	"yellow"	MAX.block1.bg
	block2	(list)		
	⋮			
	block10	(list)		
OTS				
	block1			
		"col"	"orangered"	OTS.block1.col
		"pch"	21	OTS.block1.pch
		"cex"	1.1	OTS.block1.cex
		"lwd"	2	OTS.block1.lwd
		"bg"	"yellow"	OTS.block1.bg
	block2	(list)		
	⋮			
	block10	(list)		

Table 5.1: The RLpar() hierarchical list. The hidden structures are similar to those shown, e.g. within MAX, the block2 has the same structure as block1.

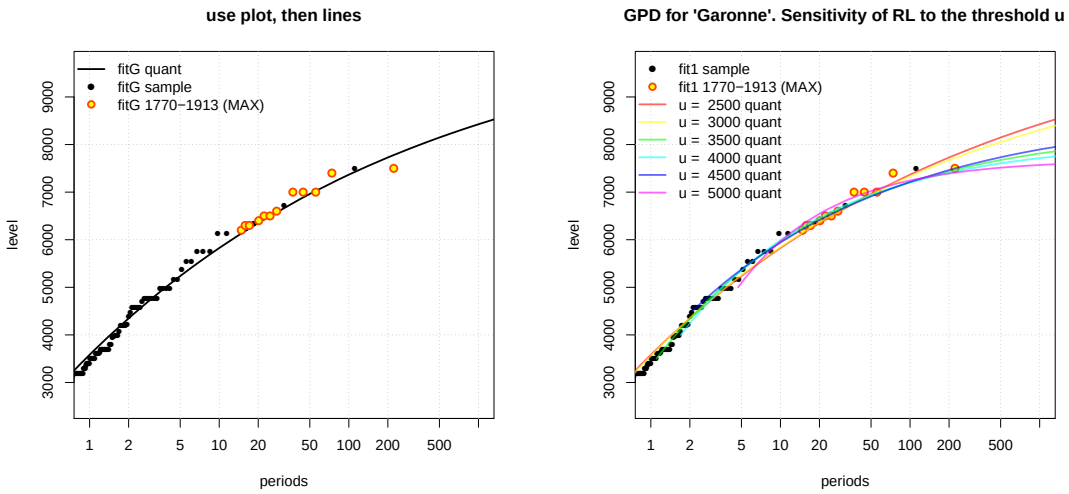

Figure 5.3: Left: building a RL plot with legend by steps. Right: Fits using different thresholds u.

1. A call to the `RLlegend.ini` function initialises a special variable which can be thought of as global[2].
2. One call to the `plot` method creates the plot, and subsequent calls to `lines` add elements to it. For these statements, the `legend` formal argument must be turned to `FALSE` in order to delay the construction of the legend[3].
3. `RLlegend.show` adds the legend to the plot on the active device.

Consider again the first example of this chapter.

```
> RLlegend.ini()
> plot(fitG, show = list(OT = TRUE, MAX = FALSE),
       main = "use plot, then lines", legend = FALSE)
> lines(fitG, show = list(OT = FALSE, quant = FALSE, MAX = TRUE), legend = FALSE)
> RLlegend.show()
```

The elements added with lines are now duly described in the legend as shown on figure 5.3. Note that the name of the R object used in the `x` argument of `plot` or `lines` (here `fitG`) is used as prefix. This can be changed by specifying a value for the `label` argument.

5.3.1 Example: sensitivity to the choice of the threshold

The next example shows a situation in which the gradual construction of a RL plot can be useful. We want to compare RL plots for different fits of the same data but using different thresholds u. We use again `Garonne`, including its historical information. Since the fit lines differ only by their colour, we can use the standard palette[4] `rainbow`. We also make colours translucent (i.e. semi-transparent) for clarity.

```
> u <- seq(from = 2500, to = 5000, by = 500)
> fit1 <- Renouv(Garonne, threshold = u[1], distname.y = "GPD", plot = FALSE)
> cols <- translude(rainbow(length(u)), alpha = 0.6)
> RLlegend.ini()
> ## plot with no lines or points.
> plot(fit1,
       main = "GPD for 'Garonne'. Sensitivity of RL to the threshold u",
       show = list(quant = FALSE, OT = TRUE, conf = FALSE, MAX = TRUE),
```

[2] To be exact, this variable is stored in an environment bound to the package.
[3] Without this precaution, the same element will be shown several times in the legend.
[4] A better solution would use a sequential palette from the **RColorBrewer** package.

```
       legend = FALSE)
> for (i in 1L:length(u)) {
      fiti <- Renouv(Garonne, threshold = u[i], distname.y = "GPD", plot = FALSE)
      lines(fiti, legend = FALSE,
            label = paste("u = ", u[i]),
            show = list(OT = FALSE, conf = FALSE, quant = TRUE, MAX = FALSE),
            par = RLpar(quant.col = cols[i]))
  }
> RLlegend.show()
```

The plot is shown on the right of figure 5.3. It shows that choosing $u \geqslant 3500$ will lead to much smaller return levels for the return period $T = 1000$.

5.4 Block data

5.4.1 One style per block?

When a `Renouv` object contains block data (MAX or OTS), these can be shown on the RL plot in a quite flexible way. As explained above, the graphical parameters can be set with `RLpar`, although a limited number of styles is imposed for the blocks.

- A different plotting style can be used or not for each block, depending on the `byBlockStyle` formal argument.
- For each of the two block types, one can select the blocks shown by using a logical or character vector as a `MAX` or `OTS` element of the `show` list formal.

Consider the following fictive example with 4 OTS blocks. We begin with a basic call to `plot` producing the plot on the left of figure 5.4.

```
> fitSim <- Renouv(x = rexp(100), effDuration = 100, threshold = 0,
                   OTS.data = list("deluge" = c(1.2, 2.4, 6.2, 3.1),
                       "dryness1" = c(0.2, 0.3),
                       "dryness2" = numeric(0),
                       "dryness3" = numeric(0)),
                   OTS.effDuration = c(60, 100, 20, 30),
                   OTS.threshold = c(1.0, 0.1, 0.3, 0.1),
                   plot = FALSE)
> plot(fitSim, main = "simulated data, by Block", label = "")
```

Each of the four blocks uses a different style and is shown in the legend. By using the `byBlockStyle` argument, we can change this default behaviour, see the plot at the right of figure 5.4. Note that when `byBlockStyle` is `TRUE`, the common plotting characteristics can be changed as would be the first block `"block1"`, even if the block with number `1` is not shown on the plot – this is just a matter of convention. To specify a common style for all `OTS` blocks we use

```
> plot(fitSim, main = "simulated data", label = "", byBlockStyle = c("OTS" = FALSE))
```

Since there are no `MAX` blocks here, it is not necessary to specify the `MAX` element. Obviously, the elements of `byBlockStyle` must be named; yet we could have used as well a *list* as in `list("OTS" = FALSE)`, instead of the character (atomic) vector `c("OTS" = FALSE)`.

5.4.2 Enlightening one block

We now consider a more elaborated example using the same `Renouv` object as before, namely `fitSim`. Assume that we want all blocks to be shown with the same style, except one. We can use a logical *vector* in the considered element of `show`, i.e. in `showOTS`. This vector must have its length equal to the number of blocks, and its elements tell if the corresponding block (in the same order) is shown or not.

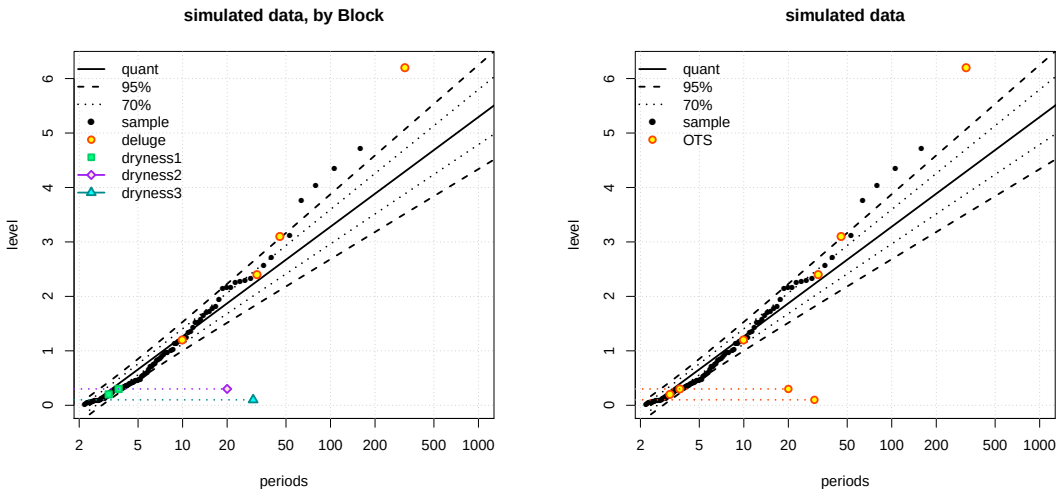

Figure 5.4: Using a different style for each block (left) or one common style for all (right).

```
> RLlegend.ini()
> plot(fitSim, main = "grouping blocks", label = "",
       show = list("OTS" = FALSE),              ## IMPORTANT!
       legend = FALSE)
> ## add dryness blocks. Note that the label is used as prefix for all elements.
> lines(fitSim, label = "dyryness",
        byBlockStyle = c("OTS" = FALSE),
        show = list("quant" = FALSE, "OTS" = c(FALSE, TRUE, TRUE, TRUE)),
        par = RLpar(OTS.block1.pch = 22,
            OTS.block1.col = "red", OTS.block1.bg = "gold"),
        legend = FALSE)
> ## add deluge block
> lines(fitSim, label = "",
        byBlockStyle = c("OTS" = TRUE),
        show = list("quant" = FALSE, "OTS" = c(TRUE, FALSE, FALSE, FALSE)),
        par = RLpar(OTS.block1.col = "SteelBlue3", bg = "darkcyan"),
        legend = FALSE)
> RLlegend.show()
```

As said before, we use `OTS.block1` to select the plotting symbol and its properties, although the block with number `1` (named `"deluge"`) is not displayed by the corresponding call to `lines` (since the first element of `show$OTS` is `FALSE`).

Instead of a logical vector, each of the list elements named `"MAX"` and `"OTS"` in `show` can be a *character string* used to select the wanted elements. This is useful when the names of the blocks are relevant for the selection. The following code produced the plot shown at the right of figure 5.5.

```
> RLlegend.ini()
> plot(fitSim, main = "char. in 'show'", label = "", show = list("OTS" = "dryness"))
> RLlegend.show()
```

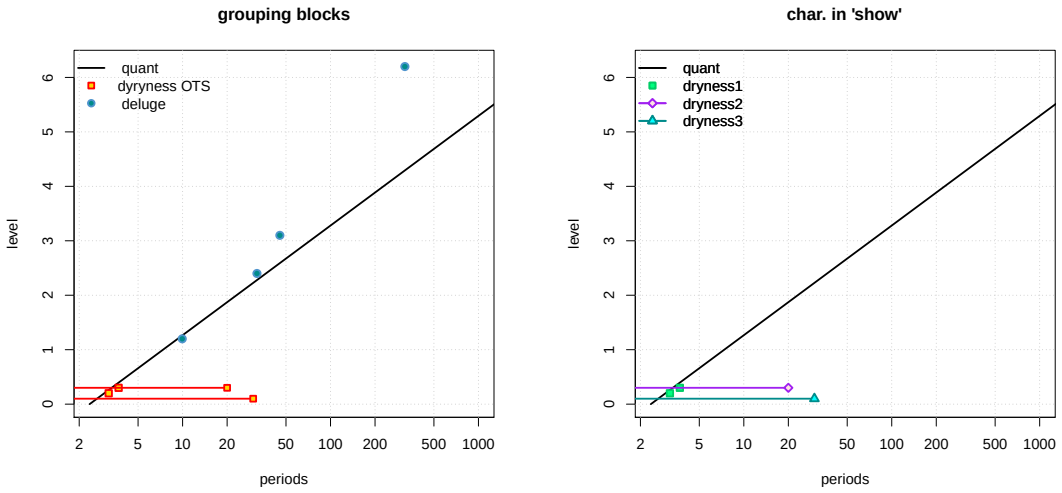

Figure 5.5: Using a common style for all blocks except one (left). Using a character value in `show` to select some blocks by their name (right).

Appendix A

The "renouvellement" context

A.1 Marked point process

The *méthode du renouvellement* uses a quite general marked process $[T_i,\, X_i]$ for events and levels. As in 1.2.1 the two sequences "events" and "levels" are assumed to be independent, and the X_i are assumed to be independent and identically distributed with continuous distribution $F_X(x)$.

An alternative equivalent description of the events occurrence is through the associated *counting process* $N(t)$. This describes the joint distribution for the the numbers of events $N(t_k) - N(s_k)$ on an arbitrary collection of disjoint intervals $(s_k,\, t_k)$. Although the most important and clearest context is the HPP, the theory can be extended to cover non-poissonian Lévy counting processes $N(t)$ e.g. Negative Binomial. However, the Negative Binomial Lévy Process implies the presence of multiple (simultaneous) events.

A.2 Maxima

A.2.1 Compound maximum

Consider an infinite sequence of independent and identically distributed random variables X_k with continuous distribution $F_X(x)$. The maximum

$$M_n = \max(X_1, X_2,\, \ldots,\, X_n)$$

has a distribution function given by $F_{M_n}(x) = F_X(x)^n$. Now let N be a random variable independent of the X_k sequence and taking non-negative integer values. The "compound maximum"

$$M = \max(X_1, X_2,\, \ldots,\, X_N)$$

is a random variable with a mixed type distribution: it is continuous with a probability mass corresponding to the $N = 0$ case which can be considered as leading to the certain value $M = -\infty$. The distribution of M can be derived from that of X_k and N. Using $\Pr(M \leqslant x \mid N = n) = F_X(x)^n$ and the total probability formula we get

$$F_M(x) = \sum_{n=0}^{\infty} F_X(x)^n \Pr\{N = n\} = h_N[F_X(x)] \tag{A.1}$$

where $h_N(z) = \mathbb{E}\left(z^N\right)$ is the generating function of N.

When N has a Poisson distribution with mean $\mu_N = \lambda w$ the generating function is given by $h_N(z) = \exp\{-\mu_N\,[1 - z]\}$ and

$$F_M(x) = \exp\{-\lambda w\,[1 - F_X(x)]\} = \exp\{-\lambda w\, S_X(x)\}. \tag{A.2}$$

When $F_X(x)$ is GPD, it can be shown that M is[1] GEV see later.

[1] Up to its probability mass.

For large return levels x, we have $F_X(x) \approx 1$. The generating function $h_N(z)$ for $z = 1$ has a value $h_N(z) = 1$ and a first derivative $h'_N(z) = \mathbb{E}(N)$, leading to

$$1 - F_M(x) \approx \mathbb{E}(N)\,[1 - F_X(x)]\,, \tag{A.3}$$

or equivalently

$$F_M(x) \approx F_X(x)^{\mathbb{E}(N)} \tag{A.4}$$

which tells that for large return levels, the distribution of M is approximately that of the maximum of $\mathbb{E}(N)$ independent X_k. Both formula (A.3) and (A.4) tell that the distribution of N only influences large return periods through its expectation. Consequently there is little point in choosing a non-Poisson distribution for N as far as the interest is focused on large return periods.

From formula (A.4) and the asymptotic behaviour of the maximum of n independent and identically distributed random variables (see B.1 later), it appears that when $\mathbb{E}(N)$ is large the distribution of M will generally be close to a suitably scaled GEV distribution.

A.2.2 Special cases

A case with special interest is when N is Poisson with mean $\mu_N = \lambda w$ and X has a Generalised Pareto Distribution (GPD). Then M follows[2] a Generalised Extreme Values (GEV) distribution as is usually assumed .

Consider first the exponential case $S_X(x) = e^{-(x-\mu)/\sigma}$ for $x \geqslant \mu$. Then (A.2) writes as

$$F_M(x) = \exp\left\{-\lambda w\, e^{-(x-\mu)/\sigma}\right\}$$

which using simple algebra can be identified as the Gumbel distribution function with parameters $\mu^\star = \mu + \sigma \log(\lambda w)$ and $\sigma^\star = \sigma$.

In the general case where $F_X(x)$ corresponds to the GPD, we have $S_X(x) = [1 + \xi(x-\mu)/\sigma]^{-1/\xi}$ for $x \geqslant \mu$, hence

$$F_M(x) = \exp\left\{-\lambda w\,[1 + \xi(x-\mu)/\sigma]^{-1/\xi}\right\}$$

which can be identified as $\texttt{GEV}(\mu^\star, \sigma^\star, \xi)$ with parameters $\mu^\star$ and $\sigma^\star$ depending on μ and σ, see Deville (2015). Note that the shape parameter ξ is the same for the GPD and the GEV distribution.

A.3 Return periods

In the general marked process context described above, the return period of a given level x can be defined using the thinned process $[T_i,\, X_i]$ of events with level exceeding x i.e. with $X_i > x$. The return period will be the expectation $T_X(x)$ of the interevent in the thinned process. In the rest of this section, we assume that events occur according to a HPP with rate $\lambda > 0$. Due to the independence of events and levels, the thinned event process also is an HPP with rate $\lambda(x) = \lambda\, S_X(x)$. The return period is then given by

$$T_X(x) = \frac{1}{\lambda\, S_X(x)}.$$

Actually the interevent distribution is exponential with expectation $1/\lambda(x)$.

Still using the same probabilistic framework, we may consider the sequence of annual maxima or more generally the sequence M_n of maxima for successive non-overlapping time blocks with the same duration $w > 0$. The random variables M_n are independent with a common distribution $F_M(x)$ that can be determined as it was done in the previous section. In this "block" context, the return period of a level x naturally expresses as a (non-necessarily integer) multiple of the block duration. Thus if $F_M(x) = 0.70$ i.e. if the level x is exceeded with 30% chance within a block, the return period is $1/0.3 \approx 3.33$ expressed in block duration unit. More generally, the *block* return period of the level x will be computed as

$$T_M(x) = \frac{w}{1 - F_M(x)} = \frac{\text{block duration}}{\text{prob. that } M \text{ exceeds } x}. \tag{A.5}$$

[2] Up to its probability mass in $-\infty$.

A major difference between the two return periods $T_X(x)$ and $T_M(x)$ is that the level x can be exceeded several times within the same block, especially for small x. This difference may make ambiguous some statements about yearly return periods or yearly risks. For instance, the level x with a 100 years return period $T_X(x)$ is very likely to be exceeded twice or more within a given century[3].

Using the relation (A.2) between the distributions $F_X(x)$ and $F_M(x)$, the relation (A.5) becomes

$$T_M(x) = \frac{w}{1 - \exp\{-\lambda w\,[1 - F_X(x)]\}}. \tag{A.6}$$

In practice, the interest will be focused on large levels x. In the expression at the denominator we may then use the approximation $1 - e^{-z} \approx z$ for small z, leading to $T_M(x) \approx T_X(x)$. Moreover the inequality $1 - e^{-z} \leqslant z$ for $z \geqslant 0$ shows that $T_M(x) \geqslant T_X(x)$ for all x. Using $1 - e^{-z} \approx z - z^2/2$, we even find a better approximation for moderately large levels x

$$T_M(x) \approx T_X(x) + \frac{w}{2}.$$

The presence of the half-block length $w/2$ can be viewed as a rounding effect.

[3]Within a given century, the number $N(x)$ of events with levels $X_i > x$ is then Poisson with mean 1. Thus $\Pr\{N(x) = 0\} \approx 0.37$ and $\Pr\{N(x) > 1\} \approx 0.26$.

Appendix B

Distributions

B.1 Asymptotic theory

B.1.1 An important theorem

The following conventions or definitions are used.

- Two probability distributions $F(x)$ and $G(x)$ are of same type when $G(x) = F(ax + b)$ for some constants $a > 0$ and b. All distributions of a given type are often written as $F_0([x - \mu]/\sigma)$ where $F_0(z)$ is a chosen member of the type, μ (location) and $\sigma > 0$ (shape) are parameters. The parameters μ and σ are not necessarily the mean nor the standard deviation.
- The notation z_+ is for the positive part of a number z, that is $z_+ = \max(z, 0)$.

A central result of Extreme Values theory is the following

Theorem (**Fisher-Tippett-Gnedenko**). *Let X_n be a sequence of independent and identically distributed random variables, and let $M_n = \max(X_1, X_2, \ldots, X_n)$. If there exists two sequences b_n and $a_n > 0$ such that $(M_n - b_n)/a_n$ has a non-degenerate limiting distribution $G(z)$, then that limiting distribution must be one of the following three types*

$$\begin{array}{ll} G(z) = \exp\{-e^{-z}\} & \textit{Gumbel or type I} \\ G(z) = \exp\{-z_+^{-\alpha}\} & \textit{Fréchet or type II} \\ G(z) = \exp\{-(-z)_+^{\alpha}\} & \textit{Weibull (reversed) or type III} \end{array}$$

where $\alpha > 0$ is a parameter for types II or III.

For each type, the distribution depends on μ and $\sigma > 0$ and possibly of $\alpha > 0$. E.g. the general Gumbel distribution is

$$G(x) = \exp\left\{-\exp\left[-(x - \mu)/\sigma\right]\right\}.$$

The third distribution corresponds to values $z \leqslant 0$ and is often called Weibull. This may create a confusion with the ordinary Weibull described below. A preferable appellation is *reversed Weibull*.

Each of the three possible limiting distributions is *max-stable* i.e. is closed for the maximum of independent and identically distributed random variables. For example if the X_i are independent with the same Gumbel distribution, then their maximum M_n is also of Gumbel type.

The three possible limit distributions are fairly different. Some mathematical criteria allow to say whether a given distribution of X_k is in the *domain of attraction* of Gumbel, Fréchet or (reversed) Weibull (Embrecht *et al.* 1996, chap. 3). Some usual examples are found in the book of Kotz and Nadarajah (2005, chap. 1) and table B.1 gives the domains of attraction for the main distributions used in **Renext**. Broadly speaking, distributions with exponentially decaying upper tail (such as normal, exponential, gamma) fall in the domain of attraction of Gumbel. The Fréchet domain attracts heavy-tailed distributions (Pareto, Cauchy).

distribution of X_i	limit of M_n
exponential	Gumbel
Weibull	Gumbel
gamma	Gumbel
GPD $\xi = 0$	Gumbel
GPD $\xi > 0$	Fréchet
GPD $\xi < 0$	reversed Weibull
log-normal	Gumbel
finite mixture of exponentials	Gumbel
Pareto	Fréchet
Cauchy	Fréchet

Table B.1: Domain of attraction of some classical distributions.

B.1.2 The Generalised Extreme Values distribution

The three types of the theorem above can be considered as special cases of the *Generalised Extreme Value* distribution depending of a shape parameter ξ

$$G(z) = \exp\left\{-\left[1+\xi\, z\right]_+^{-1/\xi}\right\}.$$

The sign of the shape parameter ξ is essential. When $\xi > 0$ we retrieve the Fréchet above up to a translation of z. For $\xi < 0$ we get the reversed Weibull up to a translation of z. When $\xi = 0$ the power $[1+\xi\, z]^{-1/\xi}$ is to be replaced by its limit for $\xi \to 0$ which is e^{-z} and $G(z)$ is the Gumbel distribution function above.

Using a linear transform $z = (x-\mu)/\sigma$ with arbitrary μ and $\sigma > 0$ all distributions of the GEV type are obtained as

$$G(x) = \exp\left\{-\left[1+\xi\,\frac{(x-\mu)}{\sigma}\right]_+^{-1/\xi}\right\}. \tag{B.1}$$

This distribution is named GEV with location parameter μ and scale parameter $\sigma > 0$, and it will be denoted as $\texttt{GEV}(\mu, \sigma, \xi)$. It is defined on the set of values x for which the bracketed expression within [] in (B.1) is non-negative that is

$\xi < 0$	$\xi = 0$	$\xi > 0$
$-\infty < x \leqslant \mu - \sigma/\xi$	$-\infty < x < +\infty$	$\mu - \sigma/\xi \leqslant x < +\infty$

so the GEV distribution has a finite upper end-point for $\xi < 0$.

A distribution $F(x)$ satisfying the conditions Fisher-Tippett-Gnedenko theorem can be said to be in the domain of attraction of the GEV with shape ξ; the sign of the shape parameter and ist value characterize the tail behaviour of the distribution.

Grouping the three distributions of the theorem into one GEV family may be thought of as a purely formal trick. However, since the GEV distribution is regular at $\xi = 0$ we have a parametric family in the usual sense, with a parameter ξ. Thus it makes sense to estimate the parameter ξ without specifying its sign, or to give a confidence interval including the value $\xi = 0$. Note that the support of the distribution depends on the parameters and thus that Maximum Likelihood (ML) theory must be invoked with care.

B.1.3 POT

The Fisher-Tippett-Gnedenko theorem suggests that the GEV distribution should be systematically used to describe block maxima. A comparable result holds for the POT context.

Theorem (**Pickands-Balkema-de Haan**). *Assume that the distribution F_X is in the domain of attraction of the GEV distribution with shape ξ, and let $x^\star$ denote its upper end-point. Then there exists a positive function $a(u) > 0$ such that for any z with $z > 0$ and $1 + \xi z > 0$*

$$\lim_{u \to x^\star} \Pr\left[X - u > a(u)z \mid X > u\right] = \left[1+\xi z\right]^{-1/\xi}.$$

See theorem 4.1 in Coles (2001) or theorem 3.4.5 in Embrecht *et al.* (1996).

The implication in POT and the marked process context is that when a large enough threshold u is chosen, the scaled excess $Z := Y/a(u)$ with $Y := X-u$ has approximately the survival $S(z) = [1+\xi z]^{-1/\xi}$, meaning that Y approximately follows a two-parameter Generalised Pareto Distribution (GPD) with shape ξ and scale $a(u)$, see B.3.2.

B.2 Probability distributions in POT

B.2.1 Levels vs excesses

POT methods fit a distribution to the excesses $Y_i = X_i - u$ over a fixed threshold u. The excesses are positive by construction and might contain small values since the threshold will generally be taken greater than the mode of X.

In the rest of this section the letter X will be used for a level while Y is used for a positive excess random variable. The densities and distribution functions of X will be denoted as $f_X(x)$ and $F_X(x)$ while the Y subscript is used for Y. Thus

$$f_X(x) = f_Y(x-u), \qquad f_Y(y) = f_X(y+u).$$

For the distribution fitted in POT the threshold u *is not a parameter* to be estimated. Yet the probability functions for level X can have a location parameter. R functions used for Y can also have a location parameter with suitable default value for it.

B.2.2 Coefficient of variation

The *coefficient of variation* CV of a positive random variable Y is the ratio of the standard deviation to the mean

$$\mathrm{CV} = \sqrt{\mathrm{Var}(Y)}/\mathbb{E}(Y). \tag{B.2}$$

Comparing this theoretical CV to its empirical equivalent $\widehat{\mathrm{CV}}$ is often instructive, keeping in mind that $\widehat{\mathrm{CV}}$ is subject to sampling fluctuation. For an exponential distribution we have CV = 1; a mixture of several exponentials corresponds to CV > 1. When fitting distributions from the Pareto families, comparing $\widehat{\mathrm{CV}}$ to 1 will often be essential, see B.3.2 page 59 below.

B.2.3 Some useful probability functions

Several probability functions provide useful insights about the upper tail of a given distribution. Their name is related to *survival analysis* where the random variable of interest is the lifetime Y of a subject or item. The relation with POT is: increasing the POT threshold u is equivalent to selecting subjects still alive at "time" u.

The *survival function* value $S(y)$ is the probability $\Pr\{Y > y\} = 1 - F(y)$. The *hazard function* $h(v)$ is defined by

$$h(v)\,\mathrm{d}v = \Pr\left[v < Y \leqslant v + \mathrm{d}v \mid Y > v\right], \qquad v \geqslant 0$$

corresponding to the notion of instantaneous death rate. An usual equivalent definition is $h(v) = f(v)/S(v)$. In survival analysis, hazards are usually non-decreasing since a decreasing hazard would mean a "rejuvenation" effect. Yet in POT modelling, distributions often have decreasing hazards. A decreasing hazard will correspond either to an exponential tail behaviour if the limit of $h(y)$ for $y \to \infty$ is positive (as for the gamma distribution ot the finite mixture of exponentials), or to a long-tail hence heavy-tail behaviour otherwise.

The *mean residual life* MRL (or mean excess life) is defined as

$$\mathrm{MRL}(v) := \mathbb{E}\left[Y - v \mid Y > v\right], \qquad v \geqslant 0.$$

While a decreasing MRL(v) may seem natural, a distribution with long tail such as GPD can have an increasing mean residual life. One can show that an increasing hazard rate implies an increasing mean residual life.

dist. name	ini.	spec. ML	par. name	note
`exponential`	n	y	`rate`	GPD with $\xi = 0$
`gpd`	n	y	`scale`, `shape`	from the **evd** package
`GPD`	n	y	`scale`, `shape`	`NaN` with bad parameters
`lomax`	n	y	`scale`, `shape`	GPD with $\xi > 0$
`maxlo`	n	y	`scale`, `shape`	GPD with $\xi < 0$
`weibull`	n	y	`scale`, `shape`	
`gamma`	n	y	`scale`, `shape`	
`lnorm`	n	y	`meanlog`, `sdlog`	
`mixexp2`	n	n	`prob1`, `rate1`, `delta`	
`SLTW`	y	n	`delta`, `scale`, `shape`	

Table B.2: Distributions in `Renouv`. The **ini.** column indicates whether or not initial values are always required on input. The **spec. ML** columns indicates if a specific ML estimation is used. However the special ML is used when only OT data are used.

Another meaningful function is the *cumulative hazard* $H(y)$

$$H(y) = -\log S(y) = \int_0^y h(z)\,\mathrm{d}z, \qquad y \geqslant 0.$$

Increasing and decreasing hazards $h(y)$ are respectively equivalent to convex and concave cumulative hazards $H(y)$. When the distribution function $F(y)$ is plotted on an exponential plot, the ordinate used is in fact $H(x)$, see page 11. The concavity of the resulting curve is that of $H(y)$, and hence is related to the variation of $h(y)$. Distributions with increasing hazard $h(y)$ will give a convex (upward concave) curve on the exponential plot while a decreasing $h(y)$ leads to a concave (downward) one. The same effect is observed for the exponential return level plot but with axes exchanged hence with opposite concavity.

An alternative to the quantile function $q_X(p)$ of X is the following *return level function*, sometimes called *tail quantile function*. Consider an independent and identically distributed sequence X_i with survival $S_X(x)$; for a given $m > 1$ the value x_m that is exceeded on average once every m observations is given by the equation

$$S_X(x_m) = 1/m \qquad (m > 1) \tag{B.3}$$

and it can be called the return level with period m (or m-return level). This is an increasing function of m with limit for large m the upper end-point of the distribution of X. For many distributions the solution of (B.3) exist in closed form. In the POT context where levels X_i are observed on a rate of λ events by years, the value of m in (B.3) is to be divided by the rate λ to obtain the corresponding period T. Then x_m is the return level corresponding to period $T := m/\lambda$.

Since $1/m = S_X(x_m)$, we have $\log m = H_X(x_m)$. Thus plotting points $[\log m,\, x_m]$ i.e. points $[m,\, x_m]$ with a log scale for the first axis (return periods) is equivalent to plotting points $[x,\, H_X(x)]$, but with the two axes exchanged.

B.3 Distributions in Renext

B.3.1 Exponential

Definition

The exponential distribution has a survival function $S(y)$ and a density $f(y)$ given by

$$S(y) = e^{-\nu y}, \qquad f(y) = \nu\, e^{-\nu y}, \qquad y \geqslant 0 \tag{B.4}$$

where $\nu > 0$ is a parameter called *rate*.

Properties

The equation $S(y) = 1/m$ giving the "m years return level" has the explicit solution $y_m = \log(m)/\nu$.

The exponential distribution has constant hazard rate – a fact known as the "memorylessness property". It therefore also has a constant mean residual life.

The exponential is a special case of several families: Weibull (with shape $\alpha = 1$), GPD (with shape $\xi = 0$) and gamma (with shape $\alpha = 1$). For these three families, the shape parameter is in one-to-one relation with the coefficient of variation CV which can take values smaller or larger than 1. Within the three families, the exponential is characterized by CV = 1.

The exponential distribution is closely related to Gumbel distribution. If Y is exponential then $V = -\log Y$ is Gumbel.

Estimation and inference

The exponential distribution has a well known ML inference from an ordinary sample Y_i of size n.

The ML estimator for ν is the inverse of the sample mean $\widehat{\nu} = 1/\bar{Y}$. Up to a scaling factor the exponential distribution is nothing but the $\chi^2(2)$ with two degrees of freedom. More precisely $2\nu\, Y_i \sim \chi^2(2)$. Multiplying the sum $\sum_i Y_i = n\,\bar{Y}$ by 2ν gives a "pivotal" quantity $V = 2\nu \times n\,\bar{Y}$ having a $\chi^2(2n)$ distribution. Since $V = 2n\,\nu/\widehat{\nu}$, an exact confidence interval at the level $1-\alpha$ for ν is obtained as

$$\frac{\chi^2_{1-\alpha/2}}{2n} \times \widehat{\nu} \leqslant \nu \leqslant \frac{\chi^2_{\alpha/2}}{2n} \times \widehat{\nu}$$

where χ^2_α is the upper quantile for the $\chi^2(2n)$ distribution[1]. Exact confidence intervals are similarly derived for the distribution $F(y)$ with given y or for a m-return level y_m with m given.

Goodness-of-fit

A specific goodness-of-fit test for the exponential distribution is sometimes called Bartlett's (or Moran's) test of exponentiality. The test statistic B_n involves the sample mean $\overline{Y}$ as well as the sample mean $\overline{\log Y}$ of the logged Y_i

$$B_n = b_n \times \left\{\log \bar{Y} - \overline{\log Y}\right\}, \qquad b_n = 2n \times \{1 + (n+1)/(6n)\}^{-1}.$$

Under the null hypothesis we have approximately $B_n \sim \chi^2(n-1)$ and a two-sided test is in order.

Remind that the goodness-of-fit can also be evaluated using a graphical analysis with an exponential plot.

Use in Renext

The exponential can be used in `Renouv` under the two names `"exponential"` and `"exp"`. In both cases, the rate parameter ν of (B.4) is named `rate`. In the `Renouv` function, the choice of the distribution name among the two possible ones for the exponential has consequences.

- Using `distname.y = "exponential"` (which corresponds to the default value), the estimation and inference will be specific to the exponential. The test of exponentiality is computed and displayed by the `summary` method for the fitted object. When no historical data are used, the exact inference described above is used both for the parameter and the return levels.
- Using `distname.y = "exp"`, the distribution of the **stats** package is used in black-box mode, as it would be with any other available distribution. Thus the inference on the parameter and the return levels is based on the asymptotic normality and the delta method.

The first possibility should obviously be preferred. In the second case, the likelihood is maximised numerically, and an initial value must be given using the `start.par.y` argument.

[1] $\Pr\left\{\chi^2(2n) > \chi^2_\alpha\right\} = \alpha$

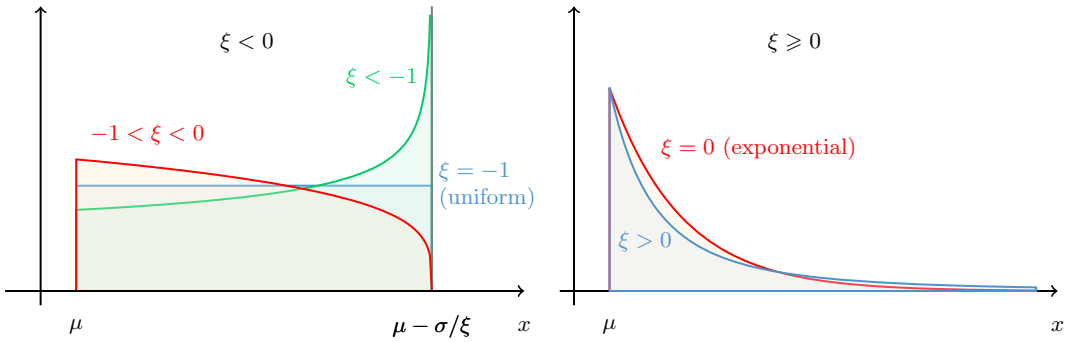

Figure B.1: GPD densities for $\xi < 0$ (left) and $\xi \geqslant 0$ (right). In the $\xi < 0$ case, the parameters are chosen in order to give the same support, i.e. μ and $-\sigma/\xi$ are kept constant.

B.3.2 Generalised Pareto GPD

Definition

The Generalised Pareto Distribution (GPD) depends on three parameters μ (location), $\sigma > 0$ (scale) and ξ (shape). When $\xi \neq 0$, the survival function $S(y)$ and the density function $f(y)$ are given by

$$S(x) = \left[1 + \xi \frac{(x-\mu)}{\sigma}\right]_+^{-1/\xi} \qquad f(x) = \frac{1}{\sigma}\left[1 + \xi \frac{(x-\mu)}{\sigma}\right]_+^{-1/\xi - 1} \qquad x \geqslant \mu \tag{B.5}$$

while the limit for $\xi \to 0$ is to be used for $\xi = 0$

$$S(x) = e^{-(x-\mu)/\sigma} \qquad f(x) = \frac{1}{\sigma} e^{-(x-\mu)/\sigma} \qquad x \geqslant \mu$$

which is a shifted exponential distribution with rate $1/\sigma$.

The distribution is defined for the values x with $x \geqslant \mu$ and $1 + \xi\,(x-\mu)/\sigma \geqslant 0$, that is

$\xi < 0$	$\xi = 0$	$\xi > 0$
$\mu \leqslant x \leqslant \mu - \sigma/\xi$	$\mu \leqslant x < +\infty$	$\mu \leqslant x < +\infty$

Unlike the GEV distribution the support of $\texttt{GPD}(\mu, \sigma, \xi)$ never extends to $-\infty$.

The value of the shape parameter ξ has a very strong impact, see figure B.1.

- When $\xi < 0$ the distribution has a finite upper end-point. As a special case, the uniform distribution is obtained with $\xi = -1$. The density function is decreasing for $-1 < \xi < 0$.
- When $\xi > 0$ the density is decreasing. The distribution tail thickens as ξ increases.

For most practical applications, the range of values for ξ is $(-0.5, 0.5)$.

Properties

The GPD has a finite expectation when $\xi < 1$ and a finite variance when $\xi < 1/2$ then given by

$$\mathbb{E}(X) = \mu + \frac{\sigma}{1-\xi}, \qquad \mathrm{Var}(X) = \frac{\sigma^2}{(1-\xi)^2(1-2\xi)}, \qquad \mathrm{CV}(Y) = \frac{1}{\sqrt{1-2\xi}}.$$

The shape parameter ξ can be related to the coefficient of variation. Note that $\xi > 0$ gives $\mathrm{CV}(Y) > 1$.

For $m > 1$ the return level with period m of (B.3) is

$$x_m = \mu + \sigma\,\left[m^{\xi} - 1\right]/\xi$$

It can be remarked that for any fixed m the value x_m is increasing with respect to each of the three parameters μ, σ and ξ and the same is true for the expectation. Thus increasing any of the three parameters leads to a distribution with greater values.

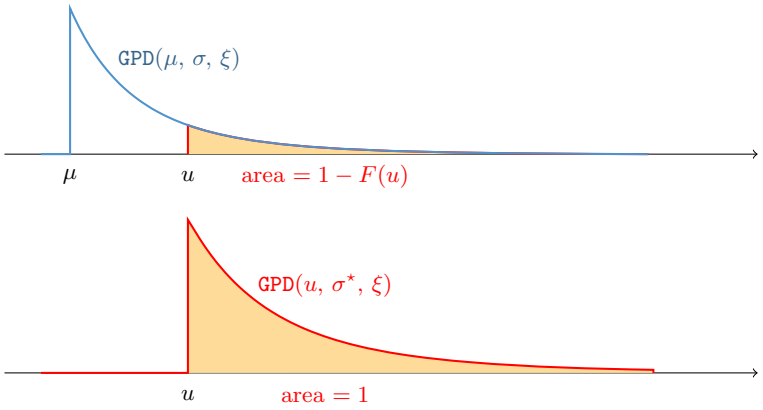

Figure B.2: "POT stability" of the GPD family. When $X \sim \texttt{GPD}(\mu, \sigma, \xi)$ the density of X conditional on $X > u$ is $\texttt{GPD}(u, \sigma^\star, \xi)$ with location u and the shape parameter ξ.

The GPD can be said to be "POT stable" in the following sense. If $X \sim \texttt{GPD}(\mu, \sigma, \xi)$ then for $u \geqslant \mu$

$$X \mid X > u \sim \texttt{GPD}(u, \sigma^\star, \xi)$$

with $\sigma^\star = \sigma + \xi(u - \mu)$. In other words, the upper tail of a GPD density is a (unnormalized) GPD density see figure B.2.

When $\xi < 1$ the GPD corresponds to a linear mean residual life

$$\mathbb{E}\left[X - v \mid X > v\right] = \frac{\sigma + \xi\, v}{1 - \xi}$$

This may be used for the determination of the threshold in POT: replacing the expectation by a sample mean we can check that the mean excess life is linear: see Coles (2001, chap. 4).

From the Pickands-Balkema-de Haan theorem, if X is a random variable with a distribution in the domain of attraction of the GEV distribution with shape ξ, then distribution of $Y = X - u$ conditional on $X > u$ when u is large will be close to a GPD with shape ξ. This property provides a justification for the traditional exclusive use of the GPD for excesses of POT models. A simple illustration for the Gumbel case $\xi = 0$ is given page 12.

The GPD has an infinite variance when $\xi \geqslant 1/2$. In practice, the values used are generally in the range $-0.3 \leqslant \xi \leqslant 0.3$.

Estimation and inference

In the POT context, the parameter μ is known since it is taken as the threshold u. The excesses $Y_i := X_i - u$ are distributed according to the GPD with location $\mu = 0$ and unknown σ and scale ξ.

Given an ordinary sample Y_i of size n, moments estimators for σ and ξ are readily available

$$\widehat{\xi}_{\texttt{mom}} = \frac{1}{2}\left[1 - \widehat{\mathrm{CV}}^{-2}\right], \qquad \widehat{\sigma}_{\texttt{mom}} = \frac{\bar{Y}}{2}\left[1 + \widehat{\mathrm{CV}}^{-2}\right].$$

The ML estimation can rely on a two-dimensional maximisation. Interestingly enough, the sign of the ML estimator $\widehat{\xi}_{\texttt{ML}}$ has a simple relation with the empirical coefficient of variation $\widehat{\mathrm{CV}}$. Provided that a denominator n is used to estimate the variance[2] in (B.2), one can show that $\widehat{\xi} < 0$ is equivalent to $\widehat{\mathrm{CV}} < 1$. In other words, $\widehat{\xi}_{\texttt{mom}}$ and $\widehat{\xi}_{\texttt{ML}}$ have the same sign. This shows that the sign of the ML estimator $\widehat{\xi}_{\texttt{ML}}$ must be interpreted with care since it is not robust to outliers.

It is important to note that when $\xi < 0$ the inequality $-\sigma/\xi > \max\{Y_i\}$ must hold, and also that the likelihood tends to ∞ when $-\sigma/\xi \to \max\{Y_i\}$ with $\xi < -1$. Therefore, a constraint $\xi \geqslant \xi_\star$ with $\xi_\star > -1$ should in theory be imposed in a numerical optimisation, although the limited precision of computations prevents from converging to a boundary parameter vector.

[2] That is $\widehat{\mathrm{Var}}(Y) = \frac{1}{n}\sum_i (Y_i - \overline{Y})^2$.

Use in Renext

In **Renext**, the ML estimation of the two-parameters for an ordinary sample can be done using the `fGPD` function. The estimation is carried out by using either the Lomax or the maxlo re-parameterisation below (see sections B.3.7 and B.3.8), depending on the sign of $\widehat{\mathrm{CV}} - 1$. In both cases, a one-dimensional maximisation is used thanks to a concentration of the likelihood.

The GPD can be used in `Renouv` under the name `"GPD"`. The parameters of (B.5) are named as those of the distribution names `"gpd"` the **evd** package

$$\sigma \leftrightarrow \texttt{scale}, \qquad \xi \leftrightarrow \texttt{shape}.$$

Note that the parameter μ is used with the name `"loc"` in the distribution functions, but should not be used in the POT context: it must then be equal to its default value 0, since the distribution is fitted on the excesses Y_i.

The GPD can also be used under the name `"gpd"` for compatibility reasons and is then taken from the **evd** package. For the ordinary sample (no historical data) case, **Renext** then relies on the `evd` package (Stephenson 2002) and its `fpot` estimation function. As for usual functions related to the distribution (density, distribution, quantile, ...) the difference between `"GPD"` and `"gpd"` is the the former returns `NaN` when an invalid parameter is provided , e.g. a negative value of `scale`, while the later then produces an error. Sine the `optim` function can cope with a `NaN` value for the optimised function, `GPD` is more flexible than `gpd`.

B.3.3 Weibull

Definition

The Weibull distribution has a survival function $S(y)$ and a density function $f(y)$ given by

$$S(y) = e^{-(y/\beta)^\alpha}, \qquad f(y) = \frac{\alpha}{\beta}\left[\frac{y}{\beta}\right]^{\alpha-1} e^{-(y/\beta)^\alpha}, \qquad y \geqslant 0 \tag{B.6}$$

where $\alpha > 0$ is the shape parameter and $\beta > 0$ the scale parameter.

Properties

The Weibull distribution has finite moments at any order with

$$\mathbb{E}(Y) = \beta\,\Gamma(1+1/\alpha), \qquad \mathrm{Var}(Y) = \beta^2\left[\Gamma(1+2/\alpha) - \Gamma^2(1+1/\alpha)\right], \qquad \mathrm{CV}(Y) = \sqrt{\frac{\Gamma(1+2/\alpha)}{\Gamma^2(1+1/\alpha)} - 1}.$$

The coefficient of variation is strictly decreasing with respect to α and takes the value 1 in the exponential case $\alpha = 1$. For $\alpha = 0.2$ the CV is about 15.8 so only values $\alpha > 0.2$ are used in practice.

The properties of the Weibull depend on the shape parameter $\alpha > 0$.

- When $0 < \alpha < 1$, the hazard rate decreases to the limit 0, and the mean residual life MRL is increasing.
- When $\alpha = 1$ the distribution is exponential with constant hazard rate and constant MRL.
- When $\alpha > 1$ the distribution has an increasing hazard rate and decreasing MRL.

See Bagnoli and Bergstrom (2004).

The return level of period $m > 1$ is given by $y_m = \beta\,[\log m]^{1/\alpha}$, confirming that the exponential return level curve $[\log m,\, y_m]$ is convex (concave upwards) for $0 < \alpha < 1$ and (downwards) concave for $\alpha > 1$.

The Weibull distribution is closely related to the exponential. When Y is Weibull with shape α the random variable $Z = Y^{1/\alpha}$ has an exponential distribution. Thus when Y follows a Weibull distribution $V = -\log Y$ has a Gumbel distribution.

Estimation and inference

The ML estimation is carried out by concentrating the scale parameter out of the likelihood. It can be shown that with a suitable re-parameterisation the concentrated likelihood is a log-concave function having an unique maximum easily obtained through a one-parameter maximisation. Moreover the expected information matrix can be given in closed form. These tips are used in **Renext**.

The moment estimators are not available in closed form and they can be obtained only at nearly the same cost as the ML estimators.

Goodness-of-fit

Specific tests exist for Weibull distributions but are not implemented in **Renext**. The fit can be controlled graphically with a *Weibull plot* such as produced by the `weibplot` function.

Use in Renext

The Weibull distribution can be used in **Renext** under the name `"weibull"`. The parameters of (B.6) are named as in the `stats` package from which the distribution functions are taken

$$\beta \leftrightarrow \texttt{scale}, \qquad \alpha \leftrightarrow \texttt{shape}.$$

The ML estimation with likelihood concentration is available in the `fweibull` function.

This distribution can be used in `Renouv` as a special distribution. It is not necessary to provide initial values for the ML estimation since specific initial values are used then in `Renouv`.

B.3.4 Gamma

Definition

The gamma distribution has density

$$f(y) = \frac{1}{\Gamma(\alpha)\,\beta^{\alpha}}\, y^{\alpha-1} e^{-y/\beta} \qquad y \geqslant 0 \tag{B.7}$$

where $\Gamma(\alpha)$ denotes the Euler's gamma function, $\beta > 0$ is the scale parameter and $\alpha > 0$ is the shape parameter. The distribution function $F(y)$ and the survival $S(y)$ do not have a simple expression.

Properties

Expectation, variance and coefficient of variation are given by

$$\mathbb{E}(Y) = \alpha\beta, \qquad \mathrm{Var}(Y) = \alpha\beta^2, \qquad \mathrm{CV}(Y) = \frac{1}{\sqrt{\alpha}}.$$

The shape parameter α is related to the coefficient of variation and $0 < \alpha < 1$ gives $\mathrm{CV}(Y) > 1$.

The properties of the distribution depend on the shape parameter $\alpha > 0$.

- For $0 < \alpha < 1$ the hazard rate decreases to the limit $1/\beta$ and the mean residual life MRL increases to the limit β
- For $\alpha = 1$ the distribution is the exponential with constant hazard and constant MRL,
- For $\alpha > 1$ the hazard rate increases to the limit $1/\beta$ and the MRL decreases to the limit β.

See Bagnoli and Bergstrom (2004).

The gamma distribution is not frequently used to describe extremes, maybe because it nearly boils down to an exponential with rate $1/\beta$ for large return periods. In the decreasing hazard case $0 < \alpha < 1$, it can be considered as a continuous mixture of exponentials with rates $\lambda > 1/\beta$.

It can be shown that the gamma distribution falls in the domain of attraction of the Gumbel distribution. It is a light-tailed distribution.

Estimation

Using an ordinary sample Y_i the moment estimators are readily available

$$\widehat{\alpha}_{\texttt{mom}} = \widehat{\mathrm{CV}}^{-2}, \qquad \widehat{\beta}_{\texttt{mom}} = \bar{X} \times \widehat{\mathrm{CV}}^{2},$$

and these could be used as initial values for a numerical likelihood maximisation.

As in the Weibull case, it is possible to concentrate the likelihood and thus to solve a one-parameter maximisation problem. Moreover, the maximisation can be reduced to that of a concave function, and the *expected* information matrix can be computed.

Use in Renext

The gamma distribution can be used in **Renext** under the name `"gamma"`. The parameters of (B.7) are named as in the `stats` package from which the distribution functions are taken

$$\beta \leftrightarrow \texttt{scale}, \qquad \alpha \leftrightarrow \texttt{shape}.$$

The ML estimation with likelihood concentration is available in the `fgamma` function.

It is not necessary to provide initial values for the ML estimation since specific initial values are used then in `Renouv`.

B.3.5 Log-normal

Definition

The log-normal distribution is the distribution of e^V where V is normal. It has density

$$f(y) = \frac{1}{y\,\sigma\sqrt{2\pi}}\,\exp\left\{-\frac{1}{2\sigma^2}\,\left[\log y - \mu\right]^2\right\} \qquad y > 0, \tag{B.8}$$

where μ and $\sigma > 0$ are the parameter of the normal distribution of $\log Y$. The distribution function $F(y)$ and the survival $S(y)$ do not have simple expression.

Note that these parameters are not the location nor the scale parameter since they are in the logged scale.

Properties

The expectation, variance and coefficient of variation of the log-normal distribution are

$$\mathbb{E}(Y) = e^{\mu+\sigma^2/2}, \qquad \mathrm{Var}(Y) = \left[e^{\sigma^2} - 1\right]\,e^{2\mu+\sigma^2}, \qquad \mathrm{CV}(Y) = \sqrt{e^{\sigma^2} - 1}.$$

For the log-normal distribution neither the hazard $h(y)$ nor the mean residual life $\mathrm{MRL}(y)$ are monotonous functions. The mean residual life $\mathrm{MRL}(y)$ is reputed[3] to be decreasing for large values of y.

Estimation and inference

The ML estimation from an ordinary sample is straightforward using the log transformation which leads to the normal case. Exact inference is also available for the parameters.

However, exact inference for the return levels or return periods is more complicated. Hence the standard numerical "delta method" is used in **Renext**.

Goodness-of-fit

The fit of the log-normal distribution can be assessed using the logged values and a normality test (e.g. Shapiro-Wilk). Since the log-normal is not frequently used in POT, such a test is not in computed in **Renext**.

[3] No proof of this assertion was found.

Use in Renext

The log-normal distribution can be used in **Renext** under the name `"lnorm"`. The parameters of (B.8) are named as in the `stats` package from which the distribution functions are taken

$$\mu \leftrightarrow \texttt{meanlog}, \qquad \sigma \leftrightarrow \texttt{sdlog}.$$

It is not necessary to provide initial values for the ML estimation since specific initial values are used then in `Renouv`.

B.3.6 Finite mixture of exponentials

Definition

The finite mixture of exponentials is a distribution with density (or survival) function obtained as a weighed mean of a finite number of exponential densities (or survivals) with different rates. For a mixture of two exponentials, the survival function $S(y)$ and density $f(y)$ are given by

$$S(y) = \alpha_1\, e^{-\lambda_1 y} + (1-\alpha_1)\, e^{-\lambda_2 y}, \qquad f(y) = \alpha_1\lambda_1\, e^{-\lambda_1 y} + (1-\alpha_1)\lambda_2\, e^{-\lambda_2 y}, \qquad y \geqslant 0 \tag{B.9}$$

and the parameters are α_1, λ_1 and λ_2 must verify

$$0 < \alpha_1 < 1 \qquad 0 < \lambda_1 < \lambda_2. \tag{B.10}$$

It can be preferable to use the alternative parameter vector $[\alpha_1,\, \lambda_1,\, \delta]^\top$ with $\delta := \lambda_2 - \lambda_1$, since the constraint $\lambda_1 < \lambda_2$ is replaced then by the simple constraint $\delta > 0$.

The usual interpretation of a mixture applies: the distribution is that of a random variable that would be randomly chosen from the exponential with rate λ_1 or from the exponential with rate λ_2 the respective probabilities being α_1 and $1-\alpha_1$. In survival analysis the mixture components correspond to two death rates that may result from two causes of mortality or from the existence of two sub-populations.

Properties

The expectation and uncentered moments have a simple form

$$\mathbb{E}(Y^\gamma) = \alpha_1/\lambda_1^\gamma + (1-\alpha_1)/\lambda_2^\gamma$$

for any $\gamma > 0$. The coefficient of variation is always greater than 1.

For large values of y, the survival $S(y)$ only depends on the smallest rate λ_1, since

$$S(y) \underset{y\to+\infty}{\sim} \alpha_1\, e^{-\lambda_1 y}. \tag{B.11}$$

The survival analysis context provides a simple interpretation: after a large time y, the sub-population with smaller death rate λ_1 dominates, and the mean residual life therefore increases.

It can be shown that the hazard rate function $h(y)$ is decreasing with a limit λ_1, and that the mean excess life is increasing with a finite limit $1/\lambda_1$. This "rejuvenation effect" results from the progressive extinction of the population having the highest death rate λ_2. The cumulative hazard $H(y)$ is concave, see figure B.3.

The quantile function is not available in closed form and must be computed numerically.

Estimation and inference

Note that the model would be unidentifiable if the second constraint of (B.10) was omitted since the distribution is invariant under the transformation

$$[\alpha_1,\, \lambda_1,\, \lambda_2] \to [1-\alpha_1,\, \lambda_2,\, \lambda_1].$$

For an ordinary sample Y_i the ML estimation can be done using Expectation-Maximisation (EM) algorithm. In this approach, each data Y_i is associated to a latent variable Z_i with value $z = 1$ or $z = 2$ indicating the group (or sub-population) for observation i and consequently the rate λ_z.

In **Renext** the standard log-likelihood maximisation is used. Initial values are computed using the moments when possible, or using (B.11): regressing $\log S(y)$ against y for large values of y give $-\log\alpha_1$ (intercept) and λ_1 (slope), see figure B.3. Then λ_2 can be deduced from the sample mean. However care is needed since these estimates may not fulfil the constraint requirements.

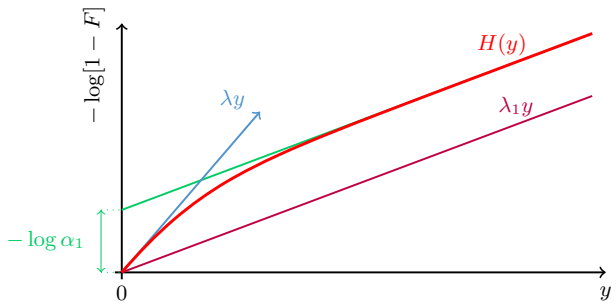

Figure B.3: Exponential plot for the distribution function of a mixture of two exponentials. The curve shows the cumulative hazard $H(y) = -\log[1 - F(y)]$. The slope of the tangent to the curve at the origin is the weighed mean rate $\lambda = \alpha_1\lambda_1 + (1 - \alpha_1)\lambda_2$. The slope of the asymptote is λ_1. Note that $\lambda_1 < \lambda < \lambda_2$.

Generalisation

A mixture of m exponentials ($m \geqslant 2$) can be defined through

$$S(y) = \sum_{i=1}^{m} \alpha_i \, e^{-\lambda_i y}, \qquad f(y) = \sum_{i=1}^{m} \alpha_i \lambda_i \, e^{-\lambda_i y}, \qquad y \geqslant 0$$

with constraints $0 < \alpha_i < 1$, $\sum_i \alpha_i = 1$ and $0 < \lambda_1 < \lambda_2 < \cdots < \lambda_m$ Since the parameter α_m can be dropped as in the $m = 2$ case, the distribution depends on $2m - 1$ free parameters. The behaviour for large y results from (B.11) which still applies.

The mixture of exponentials is sometimes called *hyper-exponential distribution.*

Use in Renext

The mixture of exponential distributions can be used in **Renext** under the name `"mixexp2"`, and is currently limited to $m = 2$ exponentials. The distribution functions (including the quantile function) are provided by **Renext** and use the following names for the parameters of (B.9)

$$\alpha_1 \leftrightarrow \texttt{prob1}, \qquad \lambda_1 \leftrightarrow \texttt{rate1}, \qquad \delta = \lambda_2 - \lambda_1 \leftrightarrow \texttt{delta}.$$

It is not necessary to provide initial values for the ML estimation since specific initial values are used then in `Renouv`.

The ML-based inference for the mixture of exponentials is well known to be difficult, and bayesian inference might be a valuable alternative.

B.3.7 Lomax

Definition

The *Lomax* distribution depends on two parameters $\beta > 0$ (scale) and $\alpha > 0$ (shape) with survival and density functions

$$S(y) = \left[1 + \frac{y}{\beta}\right]^{-\alpha}, \qquad f(y) = \frac{\alpha}{\beta}\left[1 + \frac{y}{\beta}\right]^{-\alpha-1}, \qquad y > 0. \tag{B.12}$$

This distribution is also known as *Pareto distribution of the second kind* (Johnson, Kotz, and Balakrishnan 1994). When Y is a random variable following this distribution, $X = Y + \beta$ is Pareto with minimum $x_0 = \beta$ and shape α, that is

$$S_X(x) = \left[\frac{x_0}{x}\right]^{\alpha}, \qquad x > x_0.$$

The Pareto distribution with minimum x_0 and shape α is a special case of `GPD`(μ, σ, ξ) with location $\mu = x_0$, shape $\xi = 1/\alpha$ (positive) and the extra constraint $\sigma/\xi = x_0$. The Lomax distribution is the

special case of the Generalised Pareto $\texttt{GPD}(\mu, \sigma, \xi)$ with $\mu = 0$, $\sigma = \beta/\alpha$ and $\xi = 1/\alpha$, thus implying a positive shape parameter ξ.

We can rewrite the distribution function of Y in the form (B.15) below, with $\phi_\alpha(z) \equiv \log z$, i.e. with the Box-Cox transformation (B.16) for $\alpha = 0$. Therefore, the Lomax distribution can be considered as a limit case of the Shifted Left Truncated Weibull SLTW. We may speak of *log-exponential distribution* although the expression is ambiguous.

Properties

The quantile function is available in closed form. The expectation is finite only for $\alpha > 1$ and the variance is finite only for $\alpha > 2$. In this case

$$\mathbb{E}(Y) = \frac{\beta}{\alpha - 1}, \qquad \mathrm{Var}(Y) = \frac{\alpha\,\beta^2}{(\alpha-1)^2(\alpha-2)}, \qquad \mathrm{CV}(Y) = \sqrt{\frac{\alpha}{\alpha-2}} > 1.$$

Only the cases with $\alpha > 2$ seem practicable. Then $\mathrm{CV}(Y)$ will be close to 1 for a large shape α.

The Lomax distribution has a decreasing hazard rate and a linearly increasing Mean Residual Life.

If both α and β tend to ∞ with α/β tending to $\lambda > 0$ then the Lomax distribution tends to the exponential with rate λ.

It can be shown that this distribution is a (continuous) gamma mixture of exponentials. More precisely, the survival of (B.12) can be written as

$$S(y) = \int_0^{+\infty} g(\lambda)\, e^{-\lambda y}\, \mathrm{d}\lambda$$

where $g(\lambda)$ is the density of the gamma distribution with shape $\alpha_{\texttt{gam}} := \alpha$ and scale $\beta_{\texttt{gam}} := 1/\beta$. The survival $S(y)$ is thus the weighed mean of the exponential survivals $e^{-\lambda y}$ with the weight function $g(\lambda)$. Contrary to the finite mixture of exponentials which behaves for large return periods as does its component with the smallest rate (B.11), this continuous mixture is heavy tailed. The reason is that $g(\lambda)$ weights small rates $\lambda \approx 0$, and thus the mixture embeds exponentials with arbitrarily large means $1/\lambda$. The survival function is a *completely monotone* function (Feller 1971).

Estimation

When the two parameters $\beta > 0$ and $\alpha > 0$ are unknown, the ML estimators from an ordinary sample Y_i can be found using a one-dimensional optimisation by concentrating the shape parameter α out of the likelihood. Although the concentrated log-likelihood $\ell_{\texttt{c}}(\beta)$ is not concave, it can be proved to have a maximum[4] when the sample CV is greater than 1. Moreover the expected information matrix is available in closed form (Giles, Feng, and Godwin 2013). The ML estimates fail to exist when the sample coefficient of variation CV is less than 1. The estimation may also fail when CV is greater than, yet close to 1.

When β is known, the estimation boils down to that of the exponential distribution since $V := \log[1 + Y/\beta]$ then follows an exponential distribution with rate α.

Use in Renext

This distribution is provided in **Renext** under the name `"lomax"`. The names of the formal arguments for the parameters in the probability functions are

$$\beta \leftrightarrow \texttt{scale}, \qquad \alpha \leftrightarrow \texttt{shape}.$$

The ML estimation with likelihood concentration is available in the `flomax` function. This function rescales the data to avoid numerical problems.

This distribution is recognized as special in `Renouv`, thus providing a simple mean to impose the constraint $\xi > 0$ for excesses assumed to follow $\texttt{GPD}(0, \sigma, \xi)$.

Estimation and exact inference are possible in the case where the shift β is taken as the (known) threshold i.e. $\beta = u$. The exponential distribution should then be used with a logarithmic transformation as explained below in B.3.9. The two formal arguments and values to use in the `Renouv` call are `distname.y = "exponential"` and `trans.y = "log"`. Note that α is then obtained with the name `"rate"`, and its estimated value will be greater than 1.

[4] Our proof states that a global maximum exists, but not that it is unique.

B.3.8 Maxlo

Definition

Though very useful in POT models, this distribution does not seem to have deserved its own name yet. We decided to call it "maxlo" as a pun inspired by a kind of symmetry to the Lomax distribution.

The *maxlo* distribution depends on two parameters $\beta > 0$ (scale) and $\alpha > 0$ (shape). The support of the distributions is $[0, \beta]$ and the survival and density functions are

$$S(y) = \left[1 - \frac{y}{\beta}\right]^{\alpha}, \qquad f(y) = \frac{\alpha}{\beta}\left[1 - \frac{y}{\beta}\right]^{\alpha-1} \qquad 0 < y < \beta. \tag{B.13}$$

The maxlo distribution is the special case of the Generalised Pareto $\texttt{GPD}(\mu, \sigma, \xi)$ with $\mu = 0$, $\sigma = \beta/\alpha$ and $\xi = -1/\alpha$, thus implying a *negative shape* ξ.

Properties

The quantile function is available in closed form. This distribution has finite moments of any order and

$$\mathbb{E}(Y) = \frac{\beta}{\alpha + 1}, \qquad \mathrm{Var}(Y) = \frac{\alpha\,\beta^2}{(\alpha+1)^2(\alpha+2)}, \qquad \mathrm{CV}(Y) = \sqrt{\frac{\alpha}{\alpha+2}} < 1.$$

Note that $\mathrm{CV}(Y)$ will be close to 1 for large values of the shape α.

If both α and β tend to ∞ with α/β tending to $\lambda > 0$ then the maxlo distribution tends to the exponential with rate λ.

Estimation

When the two parameters $\beta > 0$ and $\alpha > 0$ are unknown, the ML estimators from an ordinary sample Y_i can be found using a one-dimensional optimisation by concentrating the shape parameter α out of the likelihood. Note that the inequality constraint $\beta > \max\{Y_i\}$ must hold and that the likelihood tends to ∞ when $\beta \to \max\{Y_i\}$ with $\alpha < 1$. So in practice an inequality $\alpha \geqslant \alpha_{\mathtt{L}}$ must be imposed for some $\alpha_{\mathtt{L}} > 1$.

Although the concentrated log-likelihood $\ell_{\mathtt{c}}(\beta)$ is not concave it can be proved to have a maximum[5] when the sample CV is smaller than 1, thus mirroring the property stated for the Lomax distribution. Moreover the expected information matrix is available in closed form. The ML estimates fail to exist when the sample coefficient of variation CV is greater than 1. The estimation may also fail when CV is smaller than yet close to 1.

When β is known, the estimation boils down to that of the exponential distribution since $V := -\log[1 - Y/\beta]$ follows an exponential distribution with rate α.

Use in Renext

This distribution is provided in **Renext** under the name `"maxlo"`. The names of the formal arguments for the parameters in the probability functions are

$$\beta \leftrightarrow \texttt{scale}, \qquad \alpha \leftrightarrow \texttt{shape}.$$

The ML estimation with likelihood concentration is available in the `fmaxlo` function. This function rescales the data to avoid numerical problems.

This distribution can be used in `Renouv`, thus providing a simple mean to impose the constraint $\xi < 0$ for excesses assumed to follow $\texttt{GPD}(0, \sigma, \xi)$.

[5] Our proof states the existence of local maximum.

B.3.9 Transformed Exponential distributions

Definition

This rather informal family of distributions is sometimes used in hydrology. Although we will only consider in practice the two functions $\phi(x) = x^2$ and $\phi(x) = \log x$ both for $x > 0$, a slightly more general framework can be proposed as follows. Let $\phi(x)$ be a regular and strictly increasing function defined for $x > x_0$ and let u be a known value $u > x_0$. When a random variable X is such that

$$\phi(X) - \phi(u) \sim \texttt{Exp}$$

we may say that X has a *transformed exponential* distribution. The values of this distribution are the real numbers x with $x > u$. Note that the transformation needs to be one-to-one, because the distribution of X must be determinable from that of $Z = \phi(X) - \phi(u)$. Then

$$X = \psi\left(Z + \phi(u)\right)$$

where $\psi(z)$ is the reciprocal function of $\phi(x)$. As an example, the square transformation can be applied only for $x > 0$.

The survival function is given by

$$S_X(x) = \exp\left\{-\nu\left[\phi(x) - \phi(u)\right]\right\} \qquad x > u$$

where $\nu > 0$ is the rate of the exponential distribution. The density comes by derivation.

Properties

The properties of the distribution obviously depend on the choice of the transformation.

- For the square transformation $\phi(x) = x^2$ we get a shifted and truncated Weibull distribution as described below. It may be called *square-exponential* or (in french) *loi en carré*.
- With the logarithmic transformation $\phi(x) = \log x$ we get a shifted version of the Pareto (heavy tailed) distribution called Lomax distribution and described above in B.3.7. It may be called *log-exponential*.

The quantile function is available in closed form provided that the reciprocal function $\psi(z)$ is such. This is actually the case for the two transformations considered.

Estimation and inference

As far as an ordinary sample X_i is used, the ML estimator $\widehat{\nu}$ of the rate ν is available using the mean of the transformed random variables $Z_i = \phi(X_i) - \phi(u)$

$$1/\widehat{\nu} = \bar{Z} = \overline{\phi(X)} - \phi(u)$$

Exact inference on ν is deduced from the exponential case.

Use in Renext

The package allows the use of two transformed exponential distributions with the `Renouv` function, where *u is necessarily taken as equal to the threshold*. The value given for the transformation formal argument `trans.y` can be either `"square"` or `"log"`. In both cases, the exponential distribution must be specified by giving the value `"exponential"` to the distribution argument `distname.y`.

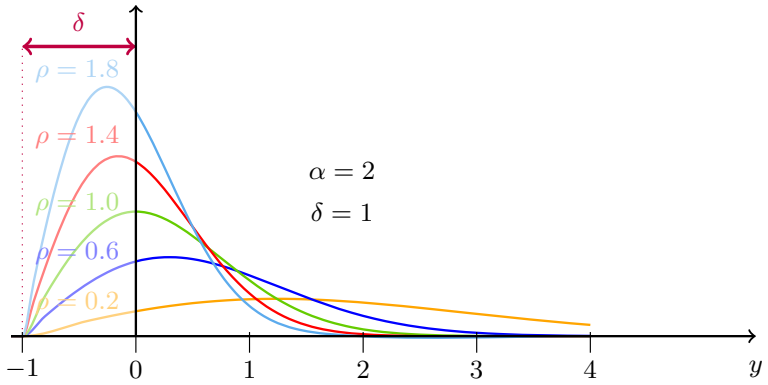

Figure B.4: "Square exponential" densities, i.e. SLTW densities with shape $\alpha = 2$. Only the part $y \geqslant 0$ of the Weibull densities is used and the normalisation is on the interval $y \geqslant 0$.

B.3.10 Shifted Left Truncated Weibull (SLTW) distribution

Definition

We call (shifted) *left truncated Weibull* (SLTW) the following distribution for a random variable $Y > 0$.

It depends on three parameters $\delta > 0$ (shift or location), $\beta > 0$ (scale) and $\alpha > 0$ (shape) and has survival function

$$S(y) = \exp\left\{-\left[\left(\frac{y+\delta}{\beta}\right)^{\alpha} - \left(\frac{\delta}{\beta}\right)^{\alpha}\right]\right\} \qquad y > 0 \tag{B.14}$$

The density comes by derivation. This is the conditional distribution $X - \delta \mid X > \delta$ where X has Weibull distribution with shape α and scale β.

For $\alpha = 2$ we can rewrite the survival as

$$S(y) = \exp\left\{-\nu\left[(y+\delta)^2 - \delta^2\right]\right\} \qquad y > 0$$

thus the distribution is identical to the square-exponential described previously.

This three parameter family can be used for excesses in POT, but in a general framework there is no natural choice for $\delta > 0$ in relation with a physical threshold u, though the two quantities have the same physical dimension. For some applications of POT where the random variable is positive δ is sometimes chosen as the threshold $\delta = u$.

Properties

The three parameter family is (by construction) POT stable for positive thresholds. The moments or even the expectation are not easily computed in the general case.

For $\alpha \leqslant 1$ the mode of Y is always $y = 0$. For $\alpha > 1$ the mode of Y is the positive part $y_+^\star$ of the shifted mode $y^\star$ of the Weibull i.e. $y^\star = (\alpha - 1)^{1/\alpha}\,\beta - \delta$. Thus for a fixed α and δ we can have a mode varying with β.

The quantile function is available in closed form. The hazard and the MRL for this distribution are merely truncations of their equivalent for the Weibull distribution, e.g. the hazard is decreasing for $0 \leqslant \alpha < 1$ and increasing for $\alpha > 1$.

For $\alpha > 0$ and large δ, the distribution is close to the exponential since the Weibull distribution is in the domain of attraction of the Gumbel distribution for which the excesses over a large threshold tend to be exponentially distributed.

Using the notation $\rho = \alpha/\beta^\alpha$ we can rewrite the survival as

$$S(y) = \exp\left\{-\rho\left[\phi_\alpha(y+\delta) - \phi_\alpha(\delta)\right]\right\} \qquad y > 0, \tag{B.15}$$

where $\phi_\alpha(z)$ is the Box-Cox transformation defined for $z > 0$ by

$$\phi_\alpha(z) = \begin{cases} (z^\alpha - 1)/\alpha & \alpha > 0 \\ \log z & \alpha = 0. \end{cases} \tag{B.16}$$

The function $\phi_\alpha(z)$ is strictly increasing with limit $+\infty$ when $z \to +\infty$ and it is regular with respect to α for $\alpha = 0$. Thus if α and β both tend to zero in such way that ρ tends to a limit $\rho^\star > 0$ the distribution tends to the Lomax distribution described above. The limit survival is (B.15) with $\alpha = 0$ and $\rho = \rho^\star$.

Estimation

In most contexts, the shift parameter δ should be known and given.

Note that when both α and δ are known and when the estimation is from an ordinary sample Y_i of size n, the ML estimator $\widehat{\rho} = \alpha/\beta^\alpha$ of ρ is available using the mean of the transformed Y_i

$$1/\widehat{\rho} = \overline{\phi_\alpha(Y + \delta)} - \phi_\alpha(\delta)$$

Exact inference on ρ or on the quantiles is then easily deduced from the exponential case.

Use in Renext

The SLTW distribution is provided in **Renext** under the name `SLTW`. The relevant probability functions share the three following formal arguments for the parameters, in correspondence with (B.14)

$$\delta \leftrightarrow \texttt{delta}, \qquad \alpha \leftrightarrow \texttt{shape}, \qquad \beta \leftrightarrow \texttt{scale}.$$

Note that the parameter named `scale` *is not* a scale parameter in the usual statistical sense; the name only refers to the original Weibull distribution.

No specific inference method is implemented in the **Renext** POT fitting. A special case is when δ is equal to the (known) threshold u and when moreover α is known. Indeed, we then fit an exponential distribution to a transformed version $\phi_\alpha(X)$ of the level $X \equiv Y + u$. We thus can use in the special case where $\alpha = 2$ (square transformation) and the limit case where $\alpha = 0$ (log transformation) as explained above in B.3.9. In the `Renouv` function, one must then use `distname.y = "exponential"`; the transformation argument must be respectively `trans.y = "square"` and `trans.y = "log"`.

B.3.11 Other distributions

It is possible to use a quite arbitrary distribution within the `Renouv` function provided the probability functions[6] are available in R and satisfy the conditions stated in the help of the `Renouv` function.

[6] Density, distribution and quantile functions are required.

Bibliography

Bagnoli M, Bergstrom T (2004). "Log-Concave Probability and Its Applications." University of California Santa Barbara, dpt of Economics. Paper 1989D. URL `works.bepress.com/ted_bergstrom/98`.

Coles S (2001). *An Introduction to Statistical Modelling of Extreme Values*. Springer.

Cox D (1962). *Renewal Theory*. Methuen, London.

Cox D, Isham V (1980). *Point Processes*. Monograph on Applied Probability and Statistics. Chapman and Hall.

Davison A, Smith R (1990). "Models for Exceedances over High Thresholds." *J.R. Statist. Soc. B*, **52**(3), 393–442.

Deville Y (2015). "Renext Computing Details." Institut de Radioprotection et de Sûreté Nucléaire.

Embrecht P, Klüppelberg C, Mikosch T (1996). *Modelling Extremal Events for Insurance and Finance*. Springer.

Feller W (1971). *An Introduction to Probability Theory and its Applications*, volume 2. 2nd edition. Willey.

Giles D, Feng H, Godwin R (2013). "On the Bias of the Maximum Likelihood Estimator for the Two-Parameter Lomax Distribution." *Communications in Statistics - Theory and Methods*, **42**(11), 1934–1950.

Gilleland E, Katz R, Young G (2004). *extRemes: Extreme value toolkit*. R package version 1.59, URL `http://www.assessment.ucar.edu/toolkit/`.

Heffernan J, Stephenson A (2012). *ismev: An Introduction to Statistical Modeling of Extreme Values*. R package version 1.39. Original S functions by J.E. Heffernan and R port and R documentation files by A. Stephenson., URL `http://www.ral.ucar.edu/~ericg/softextreme.php`.

Hirsch R, Stedinger J (1987). "Plotting Positions for Historical Floods and their Precision." *Water Ressources Research*, **23**(4), 715–727.

Johnson N, Kotz S, Balakrishnan N (1994). *Continuous Univariate Distributions*, volume 1. 2nd edition. Wiley.

Kotz S, Nadarajah S (2005). *Extreme Value Distributions, Theory and Applications*. Imperial College Press.

Kozubowski T, Panorska AK, Qeadan F, Gershunov A, Rominger D (2009). "Testing Exponentiality Versus Pareto Distribution via Likelihood Ratio." *Communications in Statistics - Simulation and Computation*, **38**(1), 118–139.

Millard S, Neerchal N (2001). *Environmental Statistics with S-Plus*. CRC Press.

Miquel J (1984). *Guide pratique d'estimation des probabilités de crues*. Eyrolles.

Nelson W (2000). "Theory and Applications Censored Failure Data." *Technometrics*, **42**(1), 12–25.

Parent E, Bernier J (2007). *Le raisonnement bayésien: modélisation et inférence*. Coll. Statistiques et probabilités appliquées. Springer-Verlag.

Pfaff B, McNeil A (2012). *evir: Extreme Values in R*. R package version 1.7-3, URL `http://CRAN.R-project.org/package=evir`.

R Development Core Team (2010). *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria. ISBN 3-900051-07-0, URL `http://www.R-project.org`.

Ribatet M (2009). *POT: Generalized Pareto Distribution and Peaks Over Threshold*. R package version 1.1-0, URL `http://CRAN.R-project.org/package=POT`.

Sarkar D (2008). *Lattice: Multivariate Data Visualization with R*. Springer, New York. ISBN 978-0-387-75968-5, URL `http://lmdvr.r-forge.r-project.org`.

Stephenson A (2002). "evd: Extreme Value Distributions." *R News*, **2**(2), 0. URL `http://CRAN.R-project.org/doc/Rnews/`.

Stephenson A, Ribatet M (2008). *evdbayes: Bayesian Analysis in Extreme Value Theory*. R package version 1.0-7.

Viglione A (2009). *nsRFA: Non-supervised Regional Frequency Analysis*. R package version 0.6-9, URL `http://CRAN.R-project.org/package=nsRFA`.

Index