A guide to flood frequency analysis using floodnetRfa

Martin Durocher

2020-04-09

Introduction

One objective of FloodNet is to provide Canadian engineers and hydrologists with a set of tools that allows them to perform flood frequency analysis (FFA) efficiently and accurately. To this end, existing methods in FFA were investigated and implemented in the R-package CSHShydRology. Another tool available to the Canadian water science community is the R-package HYDAT that simplifies the communication between R and a local version of the National Water Data Archive. The R-package `floodnetRfa` is built on the top of these two R-packages and aims to create en coherent environment for applying Floodnet recommendations. The package includes instructions that can be invoked directly in the R terminal or via a graphical interface.

In this document, we assume that the reader is familiar with the basic concepts of FFA, and we show the general capabilities of `floodnetRfa` within the terminal interface. The station 01AF009, located on the Iroquois River in New-Brunswick, serves as a case study to illustrate practical situations. The code below initiates a working environment where it loads two datasets containing meta-information about existing hydrometric stations. These companion datasets are provided “as is” and are available here. Their content is described later in the document.

```
library(floodnetRfa)

## Studied station
mysite <- '01AF009'

## Global variable that represent the path of the HYDAT database
## Must be set by the user.
hydat <- ".../Hydat.sqlite3"

gaugedSites <- read.csv(".../gauged_site.csv")
descriptors <- read.csv(".../.csv")
```

Reading data

There are three main types of function that works together in `floodnetRfa` to performs FFA. The first type of function are data extracting functions that prepare the hydrometric data in the correct format. They are normally followed by modelling functions that fit the desired statistical models. Finally, the last type of functions are interpreting functions that assess and extract the component of the fitted models.

All modelling functions require that the hydrometric data be organized in a dataset (`data.frame`) with three columns: `site`, `date` and `value`. In particular, the column `date` must be in a standard `Date` format. In most situations, an extracting function exists to import directly hydrometric data from HYDAT in the correct format. For instance, the function `AmaxData` and `DailyDate` extracts annual maximum discharges and daily streamflows. Nevertheless, if the data comes from a different source, it is recommended to use the function `SequenceData` to convert the data into the desired format. In the examples below, the function `SequenceData` prepares a series of datasets from existing numerical vectors.

```
## Example of a yearly sequence, starting today.
mydata <- rnorm(3)
SequenceData(mydata, site = 'mysite')
##     site       date     value
## 1 mysite 2020-04-09 0.3627749
## 2 mysite 2021-04-09 1.0057996
## 3 mysite 2022-04-09 0.5941100
```

```
## Example of a daily sequence starting in 2000-01-01.
SequenceData(mydata, freq = 'days', sdate = '2000-01-01')
##    site       date     value
## 1 site1 2000-01-01 0.3627749
## 2 site1 2000-01-02 1.0057996
## 3 site1 2000-01-03 0.5941100
```

```
## Example of an irregular sequence.
mydate <- as.Date(c('2000-01-01','2005-05-05','2010-10-10'))
SequenceData(mydata, mydate)
##    site       date     value
## 1 site1 2000-01-01 0.3627749
## 2 site1 2005-05-05 1.0057996
## 3 site1 2010-10-10 0.5941100

## Create a testing set
SequenceData(3, site = c('s1','s2'))
##   site       date    value
## 1   s1 2020-04-09 120.0420
## 2   s1 2021-04-09 106.7599
## 3   s1 2022-04-09 101.6659
## 4   s2 2020-04-09 120.7080
## 5   s2 2021-04-09 114.7318
## 6   s2 2022-04-09 101.8610
```

Note that when the input data is a single integer `n`, the function `SequenceData` simulates a Gumbel distribution of size `n` with mean 100 and standard deviation 30. This option can be useful in creating testing datasets quickly.

A particular type of hydrometric data are exceedances, *i.e.* values above a given threshold. In this case, the prepared dataset must have the same format, except that the meta-information related to the thresholds must be added. In the example below, the function `PeaksMeta` links the thresholds and exceedance rates, expressed in an average number of peaks per year (PPY), to the dataset of exceedances. As shown, the required format for the meta-information is a dataset with exactly three columns: site, thresh and ppy. One can see that the function `PeaksMeta` is also able to extract the meta-information from a dataset of exceedances.

```
## Create hydrometric data from two sites
set.seed(1)
stn <- c('s1','s2')
xd <- SequenceData(3, site = stn)

## Define and add meta-information
```

```r
meta <- data.frame(site = stn, thresh = c(175,160), ppy = c(1.5,2.5) )

PeaksMeta(xd) <- meta

## See the results
head(xd)
##   site       date     value
## 1   s1 2020-04-09  79.89636
## 2   s1 2021-04-09  86.76828
## 3   s1 2022-04-09 100.18125
## 4   s2 2020-04-09 141.24417
## 5   s2 2021-04-09  75.48905
## 6   s2 2022-04-09 138.74228
PeaksMeta(xd)
##    thresh ppy
## s1    175 1.5
## s2    160 2.5
```

Flood frequency analysis using annual maxima

The two main steps to carry out FFA are to fit a statistical distribution of extreme events and to evaluate flood levels associated with given probabilities. For a classical AMAX analysis, the extreme events are annual maxima, while in the peaks over threshold (POT) approach, the extreme events are exceedances. Flood risk is commonly quantified in terms of a return period T defined as the expected waiting time between two extreme events. When assuming that the exceeding probability is constant over time, the evaluation of a return period is equivalent to estimate the flood quantiles of probability $1 - 1/T$ for AMAX and $1 - (\lambda T)^{-1}$ for POT where λ is the exceedance rate. The example below extracts annual maxima from HYDAT using the function `AmaxData` and performs FFA using the function `FloodnetAmax`. The argument `period` specifies the return periods that are estimated.

```r
an <- AmaxData(hydat, mysite)
fit <- FloodnetAmax(an, period = c(10,100))
```

The function `FloodnetAmax` fits a list of candidate distributions using L-moments, choose one according to the Akaike Information Criterion (AIC) and evaluate the flood quantiles. In the end, a parametric bootstrap method measures the uncertainty of the fitted model. If the user does not provide a list of distributions, the distribution with the lowest AIC is identified among the Generalized Extreme Value (GEV), Generalized Logistic (GLO), Generalized Normal (GNO) and Pearson type III (PE3). The identified distribution is then compared to the GEV and preferred only if the difference of AIC is greater than 2. The idea behind this procedure is that for a difference of less than 2, the two distributions have similar fits. Therefore GEV is preferred as it represents the asymptotic distribution of sample maxima (Coles, 2001).

The pipe operator `%>%` can be employed to obtain a syntax familiar to the [tidyverse](). The rest of the document adopts this syntax because it is easier to read and avoid the creation of an intermediate variable. The example below produce the same results as the earlier code.

```r
set.seed(1)
fit <-  hydat %>%
    AmaxData(mysite) %>%
    FloodnetAmax(period = c(10, 100))
```

The output of a modelling function is an S3 object of the class `floodnetMdl` for which interpreting functions are available to access the various component of the fitted model. The function `print` displays only a brief description of the fitted model, while `summary` adds information about the flood quantiles and model parameters. For the 100-year flood quantile (Q100), the AMAX method estimates a discharge of 98 m^3/s with a standard deviation of 27.

```
summary(fit)
##
## Flood Frequency Analysis
##
## Method: amax
## Site: 01AF009
## Distribution: gev
## Return Period: 10 100
##
## Quantiles:
##       pred     se lower  upper
## 0.90 59.10  6.319 48.25  72.98
## 0.99 98.05 27.176 61.75 166.82
##
## Parameters:
##         param     se
## xi     33.774 2.0404
## alpha   9.262 1.7327
## kappa -0.168 0.1653
```

Another way to access this information is to convert the output into a dataset using the function `as.data.frame`. The argument `type = 'p'` is used to return the model parameters instead of the flood quantiles (`type = 'q'`). In this form, it is easy to merge into a single dataset the results from multiple fitted models.

```
## Data flood quantiles
head(as.data.frame(fit), 4)
##      site method distribution period variable     value
## 1 01AF009   amax          gev     10 quantile 59.103900
## 2 01AF009   amax          gev    100 quantile 98.047204
## 3 01AF009   amax          gev     10       se  6.318847
## 4 01AF009   amax          gev    100       se 27.176258

##
sim <- SequenceData(50) %>%
    FloodnetAmax(distr = 'pe3', nsim = 0, verbose = FALSE) %>%
    as.data.frame('p')

## Merged results
rbind(as.data.frame(fit, 'p'), sim)
##       site method distribution period variable      value
## 1   01AF009   amax          gev     10 quantile  59.103900
## 2   01AF009   amax          gev    100 quantile  98.047204
## 3   01AF009   amax          gev     10       se   6.318847
## 4   01AF009   amax          gev    100       se  27.176258
## 5   01AF009   amax          gev     10    lower  48.247181
## 6   01AF009   amax          gev    100    lower  61.753217
## 7   01AF009   amax          gev     10    upper  72.976422
```

```
## 8  01AF009   amax          gev    100     upper 166.819450
## 9    site1   amax          pe3      2 quantile  92.110366
## 10   site1   amax          pe3      5 quantile 123.515251
## 11   site1   amax          pe3     10 quantile 144.817486
## 12   site1   amax          pe3     20 quantile 165.108125
## 13   site1   amax          pe3     50 quantile 190.973414
## 14   site1   amax          pe3    100 quantile 210.045353
## 15   site1   amax          pe3      2       se   4.870441
## 16   site1   amax          pe3      5       se   7.297641
## 17   site1   amax          pe3     10       se  10.383746
## 18   site1   amax          pe3     20       se  14.291395
## 19   site1   amax          pe3     50       se  20.195837
## 20   site1   amax          pe3    100       se  25.010963
## 21   site1   amax          pe3      2    lower  82.564476
## 22   site1   amax          pe3      5    lower 109.212137
## 23   site1   amax          pe3     10    lower 124.465719
## 24   site1   amax          pe3     20    lower 137.097505
## 25   site1   amax          pe3     50    lower 151.390300
## 26   site1   amax          pe3    100    lower 161.024766
## 27   site1   amax          pe3      2    upper 101.656256
## 28   site1   amax          pe3      5    upper 137.818365
## 29   site1   amax          pe3     10    upper 165.169254
## 30   site1   amax          pe3     20    upper 193.118744
## 31   site1   amax          pe3     50    upper 230.556527
## 32   site1   amax          pe3    100    upper 259.065939
```

The function `FloodnetAmax` is built on top of the function `FitAmax` of the R-package `CSHShydRology`. If desired, the output of the underlying function is joint to the output of `FitAmax` when using the `out.model = TRUE`.

```
fit0 <- hydat %>%
    AmaxData(mysite) %>%
    FloodnetAmax(out.model = TRUE)

fit0$fit
##
## At-site frequency analysis
##
## Distribution: gev
## AIC: 215.2
## Method: lmom
## Estimate:
##      xi   alpha   kappa
## 33.774   9.262  -0.168
##
## Std.err:
##     xi alpha kappa
## 2.106 1.765 0.174
##
## Lmoments:
##       l1     lcv     lsk     lkt
## 1 40.95 0.1879 0.2825 0.2539
```

Trend tests

The methods developed in the `floodnetRfa` package are not adapted to the estimation of flood quantiles in a changing environment. Therefore, it is important to verify the presence of trends in the data. The function `floodnetAmax` automatically performs the Mann-Kendall's and Pettitt's test to examine the likelihood of a trend or change points (Helsel and Hirsch, 2002) and issues warnings when it fails. Such warnings indicate that the model assumptions must be further verified.

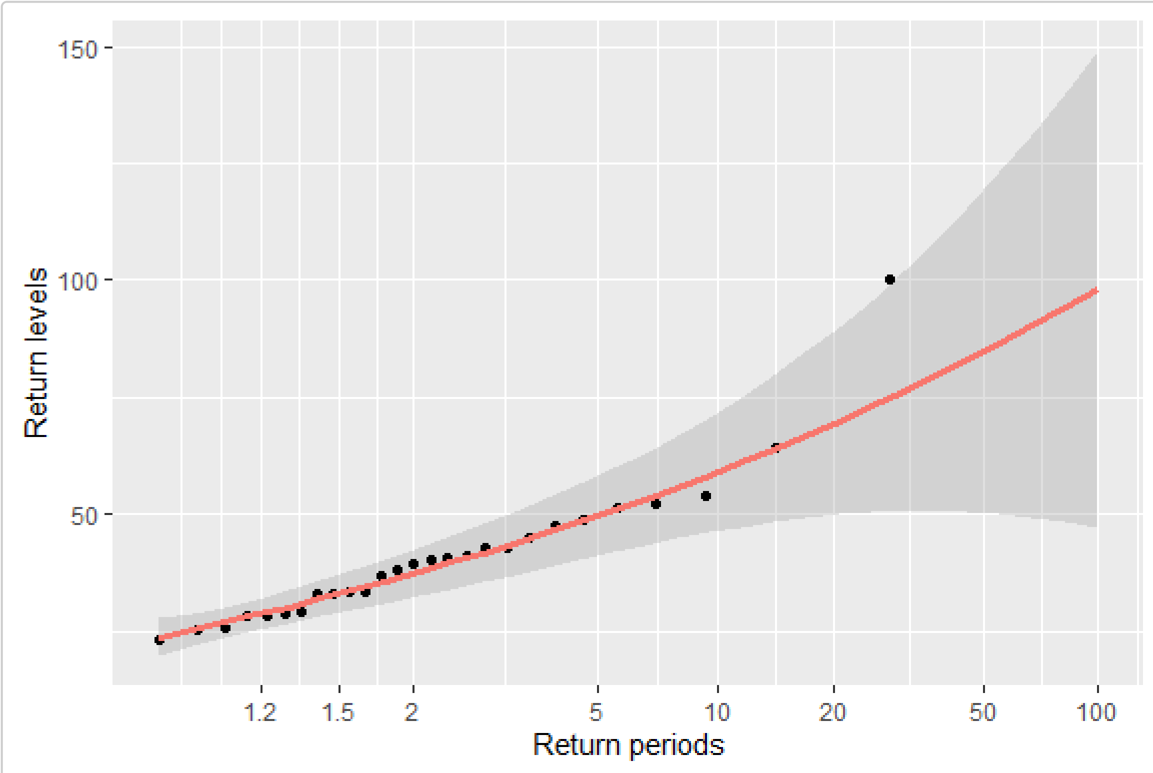
```
## Create a change point to existing data
set.seed(2)
an <- SequenceData(100)
mid <- 1:nrow(an) > 50
an$value[mid] <- an$value[mid] + 50

## Perform FFA
out <- FloodnetAmax(an)
## Warning in FloodnetAmax(an):
## There may be a trend in the data.
## Warning in FloodnetAmax(an):
## There may be a change point in the data.
```

However, if the user is confident in the validity of the selected procedure, these warnings can be silenced by setting the argument `verbose = FALSE`.

Diagnostic plots

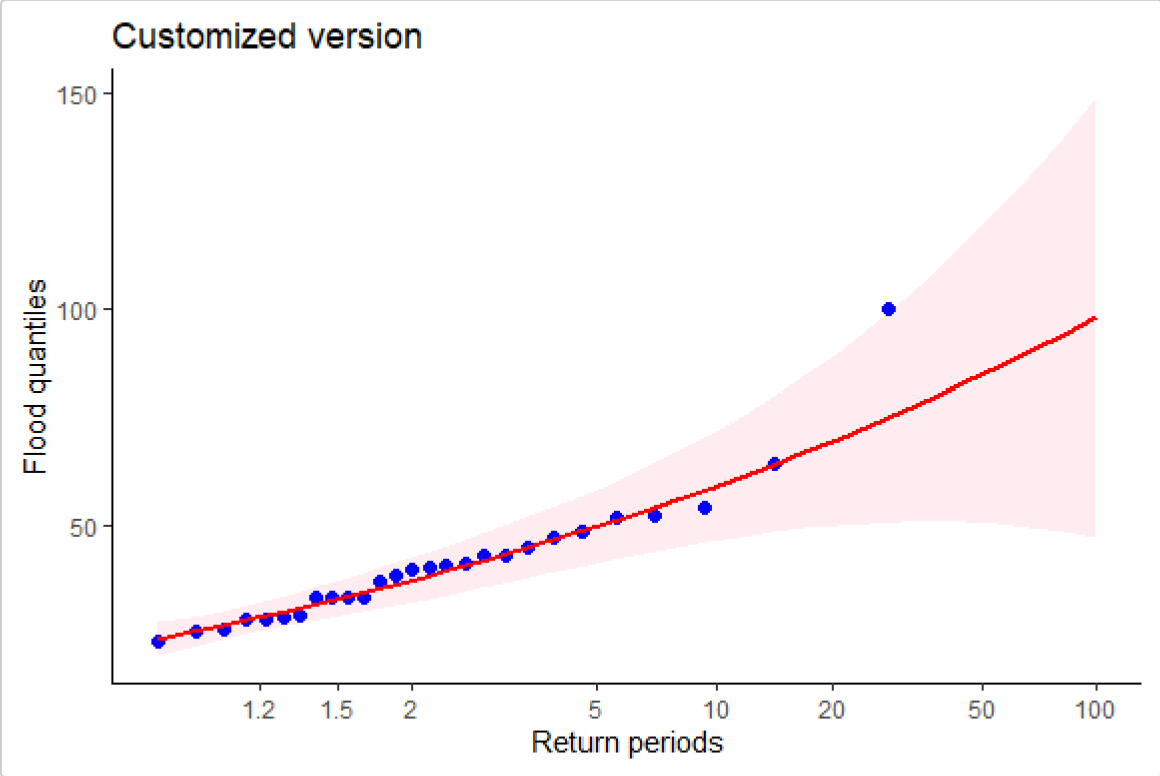
The user has access to various graphics that are generated from the `floodnetMdl` object using the function `plot`, which he can use to assess the fitted models. Under the hood, the package`ggplot2` creates graphics that be further customized by other functions. By default, the return level plot is returned to compare the fitted flood quantiles (red line) with the observations.

```
plot(fit)
```



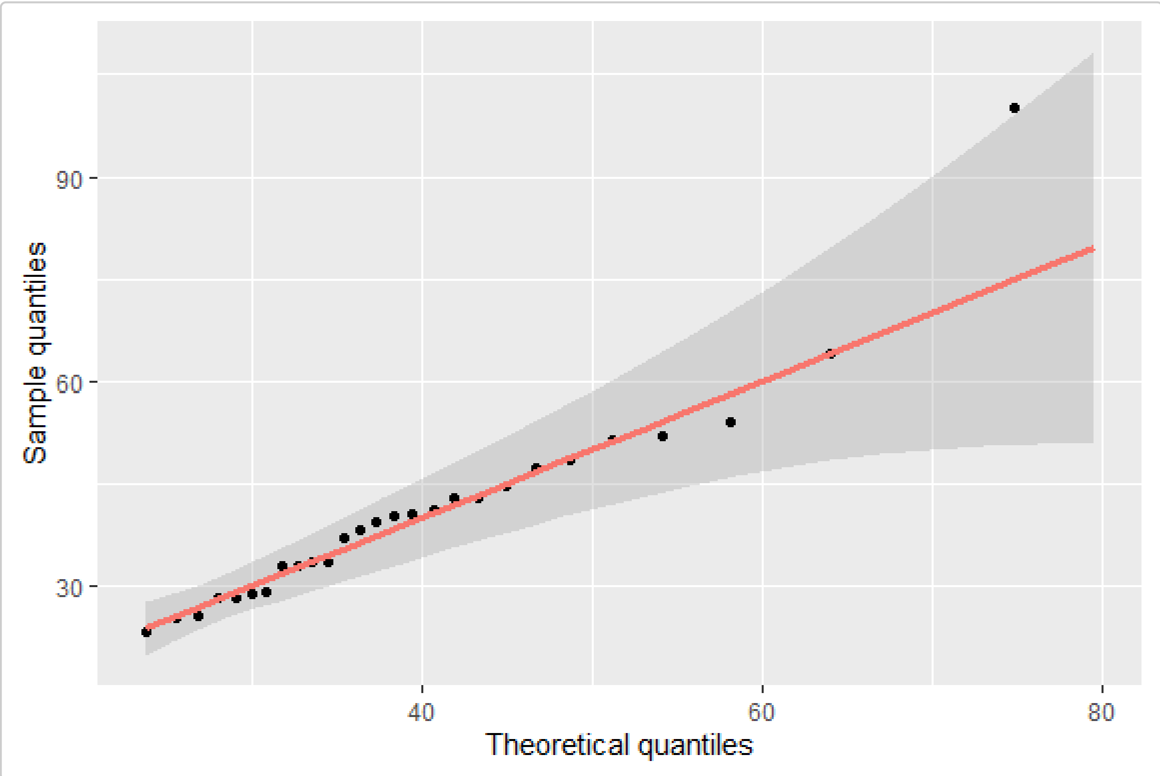
The components of the plot, like the lines (`geom_line`) and points (`geom_point`), can be modified by passing a list of arguments to the underlying geometry. The following examples illustrate how to modify some elements of the previous graphics.

```
library(ggplot2)

plot(fit, line.args = list(colour = 'red', size = 1),
         point.args = list(colour = 'blue', size = 2),
         ribbon.args = list(fill = 'pink'),
         ylab = 'Flood quantiles') +
    theme_classic() + labs(title = 'Customized version')
```



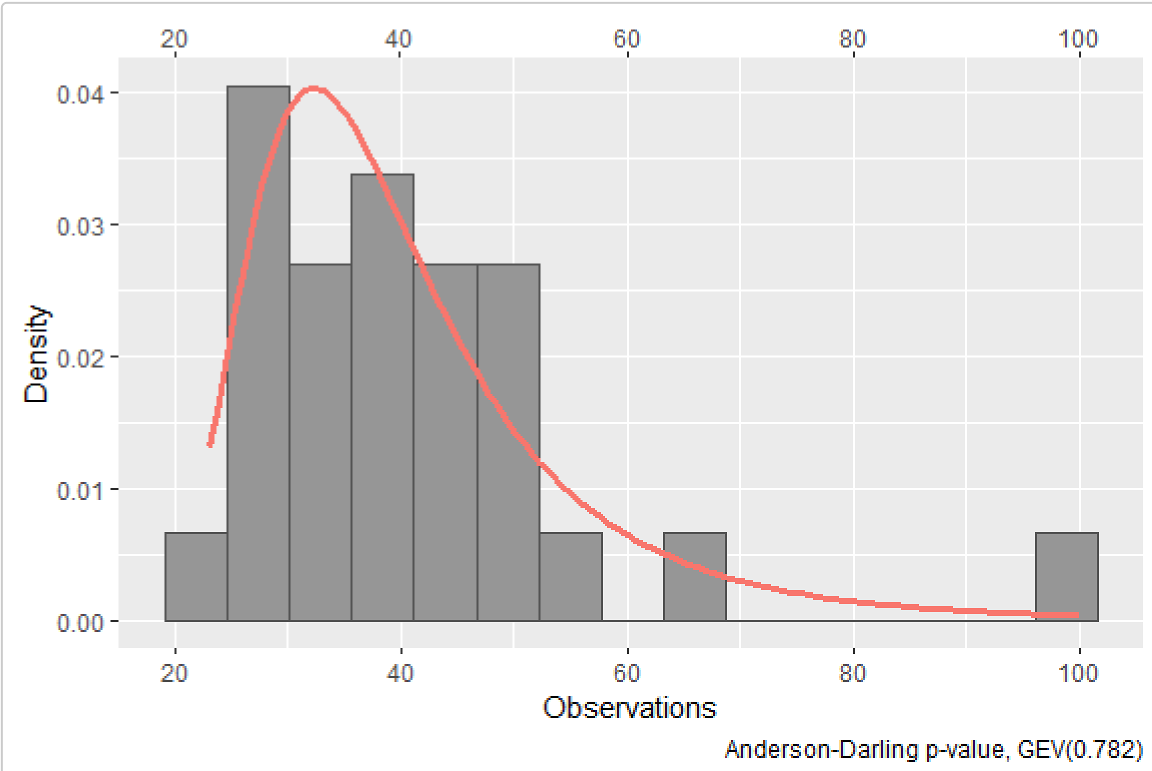
Another useful plot for assessing the fitting of a distribution is the QQ-plot that displays the theoretical versus the sample quantiles of a distribution. With a proper model, the QQ-plot differs from the return level plot by using a different x-axis that results in a linear relationship.

```
plot(fit, 'qq')
```



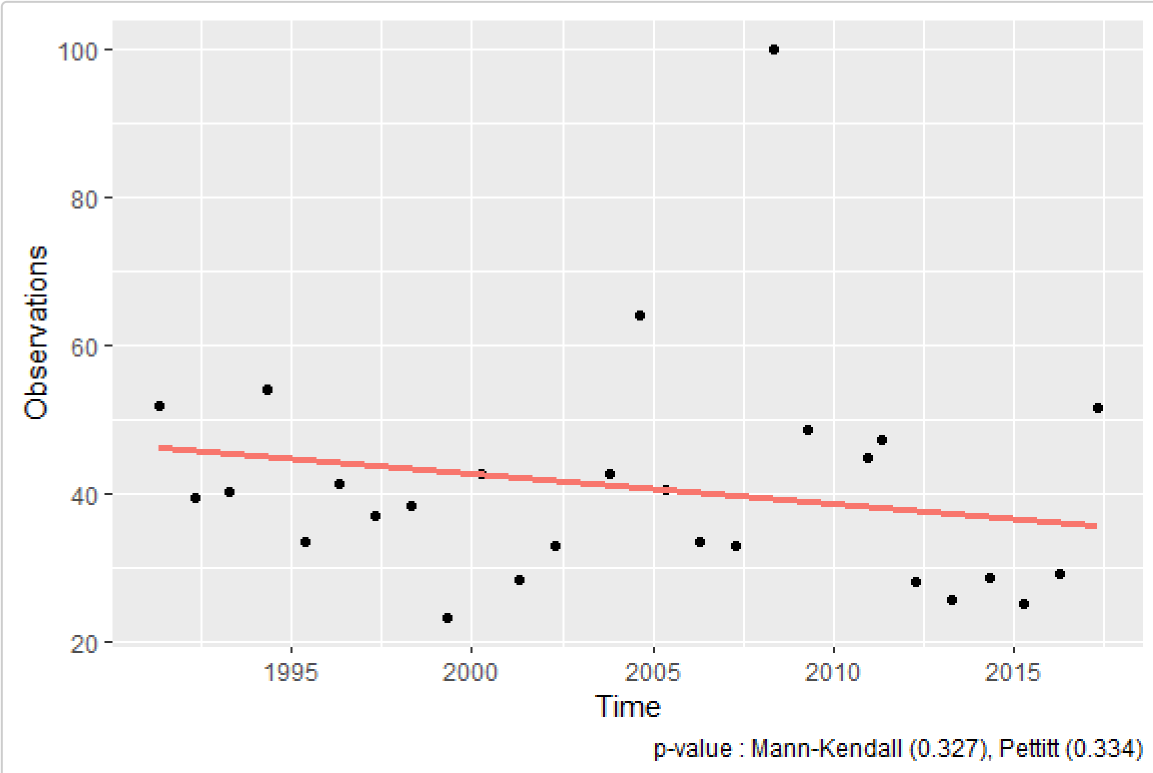
The function `hist`, or `plot` with argument `type = 'h'`, returns a histogram of the observed data and the fitted density. The bottom of the histogram reports the p-value of the goodness-of-fit test of Anderson-Darling as an additional measure of the model validity. For station 01AF009, the hypothesis of a GEV distribution cannot be rejected.

```
hist(fit, histogram.args = list( bins = 15))
```



As mentioned, warnings are issued when the modeling function detects trends and change points. To help with this examination, the function `plot` with the argument `type = 't'` displays the Sen's slope and includes the p-value the tests of Mann-Kendall and Pettitt. For station 01AF009, the two tests do not suggest a significant trend in the data.

```
plot(fit, 't')
```



Flood frequency analysis using peaks over threshold

Similarly to `AmaxData`, the function `DailyData` extracts daily streamflow data to carry out POT analysis with the help of function `FloodnetPot`. This modelling function calls the function `FitPot` of the R-package `CSHShdRology` to extract exceedances and estimate flood quantiles.

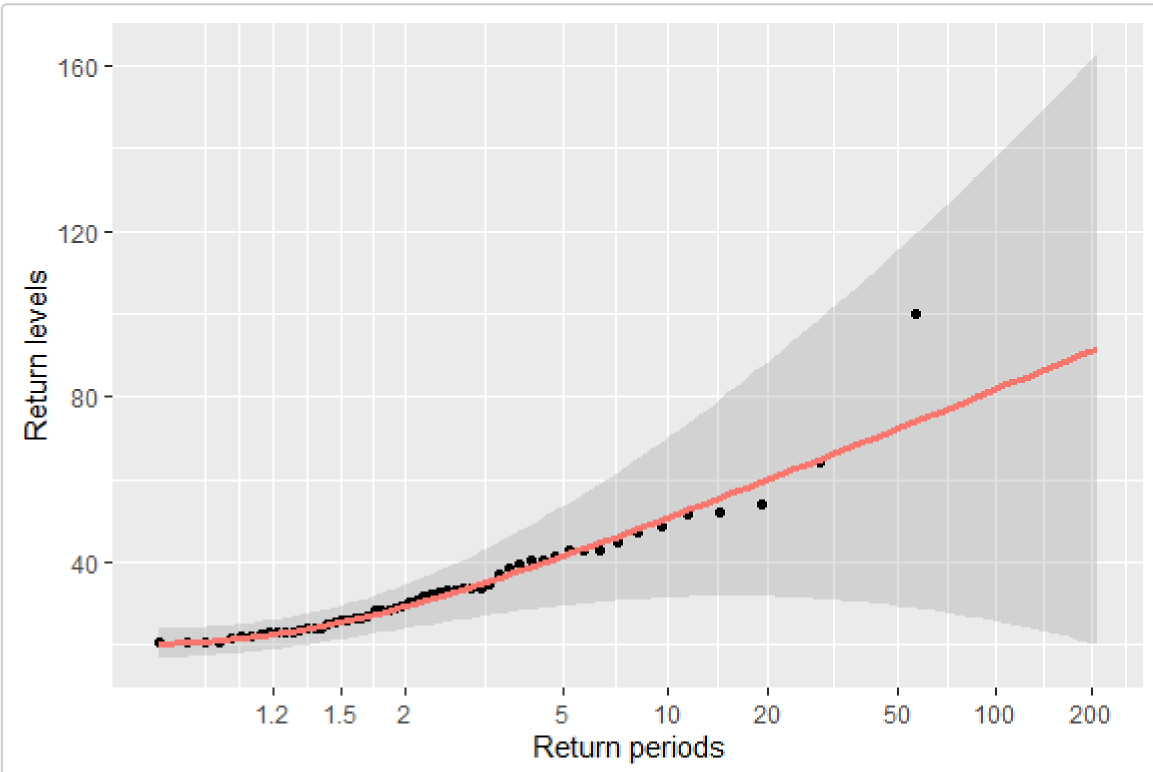
```
set.seed(1)

fit <- hydat %>%
    DailyData(mysite) %>%
    FloodnetPot(period = 100, u = 20, area = 184)
```

In comparison to the AMAX methodology, the distribution of the POT method is known and corresponds to a Generalized Pareto (GPA) distribution. The model has two parameters and also requires a threshold `u` above which independent peaks are extracted. The minimal separating time between independent peaks is determined by the drainage area of the basin, as suggested by the US Water Resources Council (USWRC) (lang et al., 1999). For station 01AF009, the drainage area is 184 km^2 and a threshold $u = 20$ was selected. In the end, a parametric bootstrap strategy evaluates the uncertainty of the flood estimates.

Identically to `floonetAmax`, the output of `floodnetPot` is a `floodnetMdl` object. Therefore, the same interpreting functions extract the model information and create the graphics.

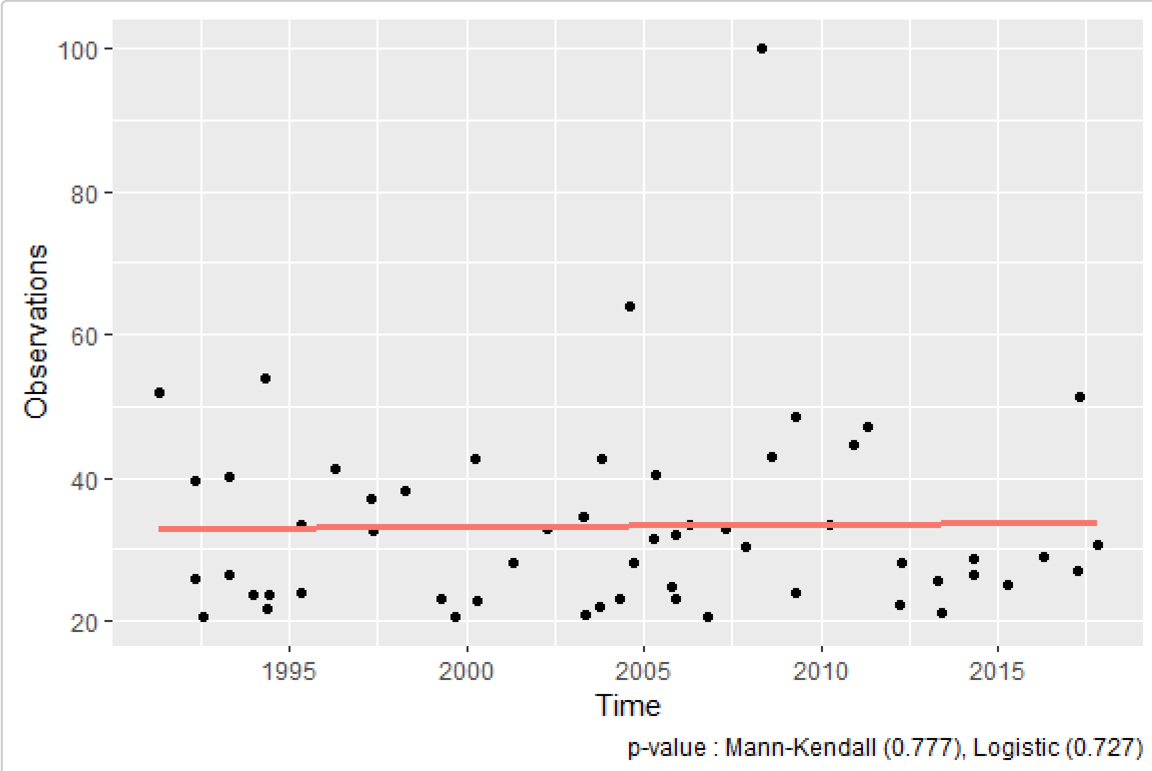
```
summary(fit)
##
## Flood Frequency Analysis
##
## Method: pot
## Site: 01AF009
## Distribution: gpa
## Return Period: 100
```

```
##
## Quantiles:
##        pred    se lower upper
## Q100 91.48 20.14  58.7 136.4
##
## Parameters:
##            param    se
## alpha 13.320249 2.922
## kappa -0.002228 0.158
plot(fit)
```



Before fitting the POT model, Mann-Kendall's test examines the presence of a trend trends in the exceedances. Similarly, a logistic regression with time as a covariate look for a linear change in the exceedance rate from a t-tests that evaluates the significance of the slope parameter.

```
plot(fit, 't')
```



Automatic selection of the threshold

If a threshold is not provided to the function `FloodnetPot`, one will be selected automatically based on the p-value of the goodness-of-fit test of Anderson-Darting (Durocher et al. 2018b). The table `gaugedSites` contains information collected about 1114 stations were identified as having a natural flow regime and at least 20 years of observations. In particular, it includes candidate thresholds. The column `auto` represent the thresholds obtained from the automatic selection method. For the station 01AF009, the automatic threshold selected a threshold of 15.9, which is lower than the one previously used. The other thresholds are available and are associated with specific PPY. For instance, the column `ppy175` is a threshold associated with 1.75 PPY.

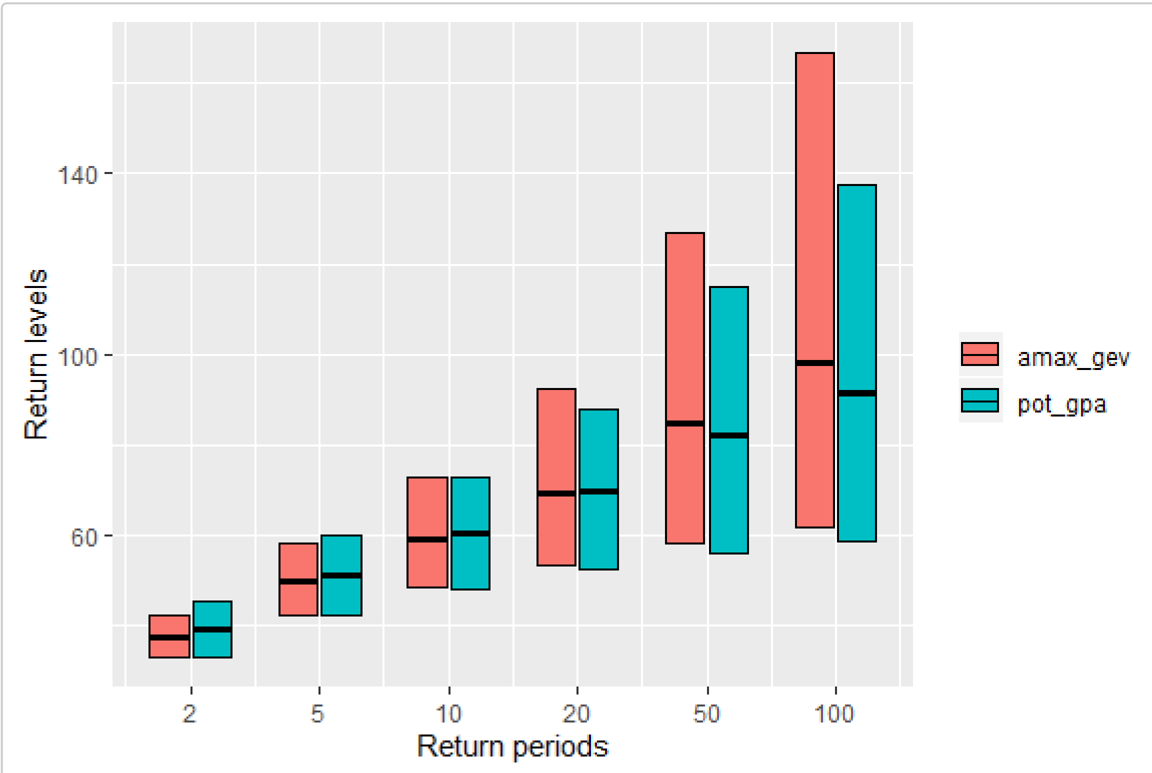
```
gaugedSites[5, c('station','description', 'area','auto','ppy250')]
##   station                          description area auto ppy250
## 5 01AF009 IROQUOIS RIVER AT MOULIN MORNEAULT  182 15.9   15.9
```

Comparison plots

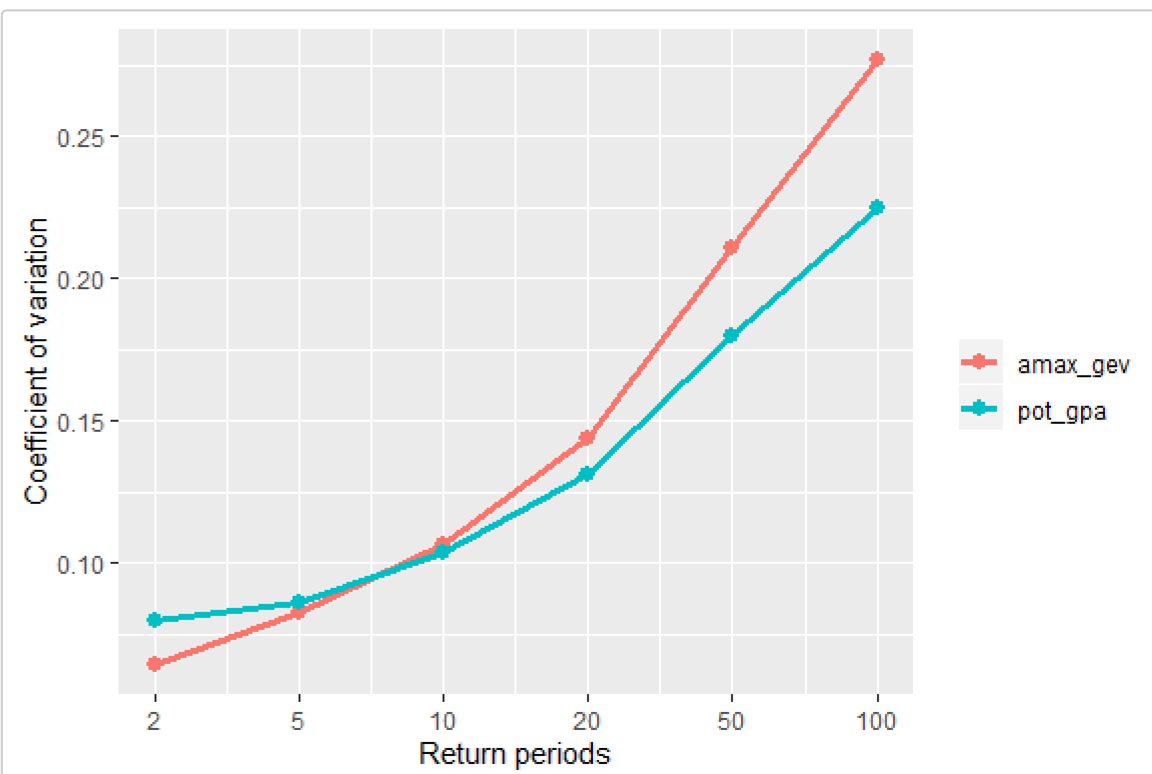
To understand the performance of a model, it is generally a good idea to compare it with others. The examples below show how the function 'CompareModel' allows to easily creates plots that compare the confidence intervals and coefficient of variation of the AMAX and POT results. In this case, we are seeing that for return periods longer than ten years; POT is more accurate than AMAX. The opposite is also true for equal or shorter return periods.

```
set.seed(1)
fit.amax <-  hydat %>%
    AmaxData(mysite) %>%
    FloodnetAmax()

fit.pot <- hydat %>%
```

```
  DailyData(mysite) %>%
  FloodnetPot(u = 20, area = 184)

lst.fit <- CompareModels(fit.amax, fit.pot)

plot(lst.fit)
```




```
plot(lst.fit, 'cv')
```



Super regions

The variability of flood quantile estimates for longer return periods depends heavily on the tail of the selected distribution. Most of the distributions used in FFA possess a shape parameter that allows controlling that aspect of the distribution. However, when only a few years of data are available at the site of interest, the uncertainty associated with this parameter may be substantial. Regional Frequency Analysis (RFA) is recommended to reduce this variability by transferring information from nearby stations with similar properties.

The strategy recommended by FloodNet is to use pooling groups delineated by a similarity measure that accounts for the regularity and timing of the annual maximum flood peaks (Mostofi Zadeh and Burn, 2019). The procedure selects sites forming a pooling group from an initial set of stations, or super region, that are relevant to the analysis. Although it is common to consider administrative boundaries(*e.g.* provinces), more hydrologically relevant super regions can be built by regrouping stations with similar hydrological properties.

The dataset `gaugedSites` contains pre-delineated super regions based on four widely available catchment descriptors:

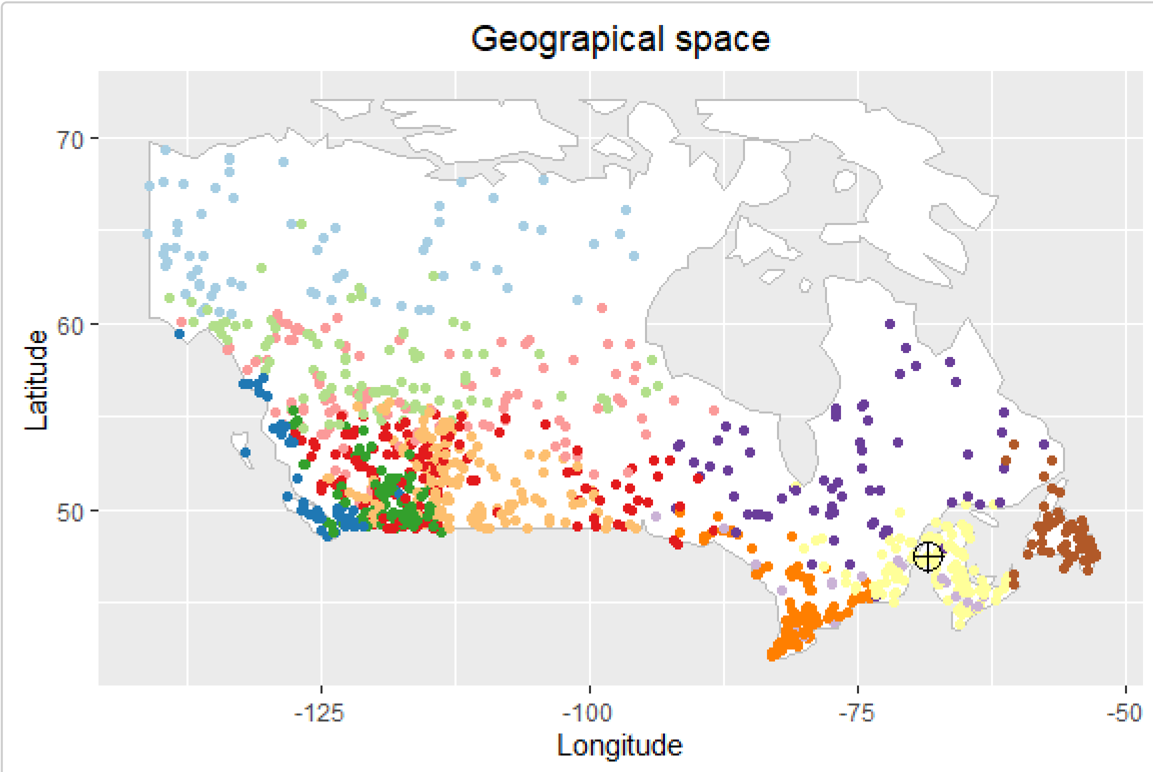
```
* Drainage area
* Mean annual precipitation (MAP)
* Longitude
* Latitude
```

In the rest of the document, we consider the super region proposed in column `supreg_km12` that results from a division of the stations into 12 super regions by the k-means algorithm. The figures below present these super regions in the geographical, seasonal and descriptor spaces. The function `MapCA` and `SeasonPlot` are provided as utility functions in `floodnetRfa` to simplify the creation of a simple map in the respective spaces.

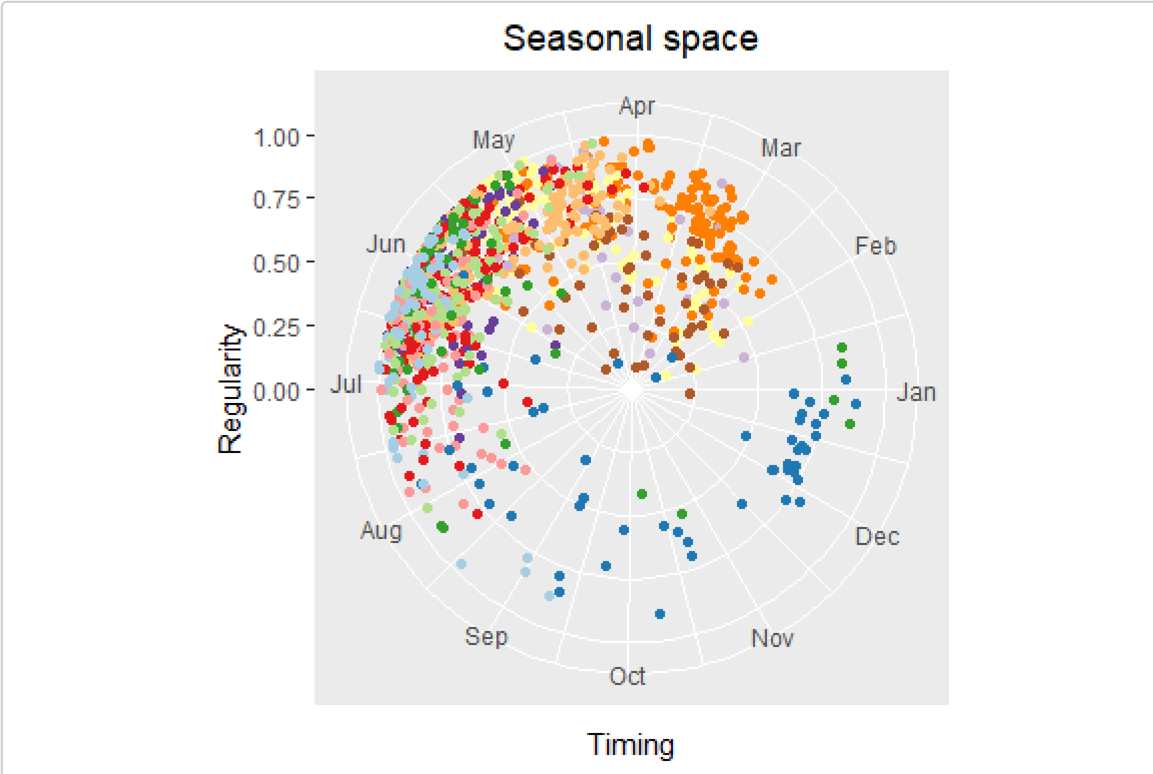
```
## Extract data
xd <- gaugedSites[, c('station', 'lon', 'lat')]

xd$area <- log(gaugedSites$area)
xd$map <- log(gaugedSites$map)
xd$region <- as.factor(gaugedSites$supreg_km12)
xd$theta <- gaugedSites$season_angle
xd$r <- gaugedSites$season_radius

## Function that customize graphics
FormatPlot <- function(plt, main) {

    ## Define a custom palette
    mycolors = c('#a6cee3','#1f78b4','#b2df8a','#33a02c','#fb9a99','#e31a1c',
                      '#fdbf6f','#ff7f00','#cab2d6','#6a3d9a','#ffff99','#b15928')

    plt + scale_colour_manual(values = mycolors) +
      theme(legend.pos = '', plot.title = element_text(hjust = 0.5)) +
        ggtitle(main)
}

## Maps
plt <- MapCA(polygon.args = list(fill = 'white', colour = 'grey')) +
    geom_point(data = xd, aes(x = lon, y = lat, colour = region)) +
    geom_point(data = xd[5,], aes(x = lon, y = lat),
                      colour = 'black', size = 5, shape = 10)
```

```
## Loading required package: sp
## Regions defined for each Polygons
```

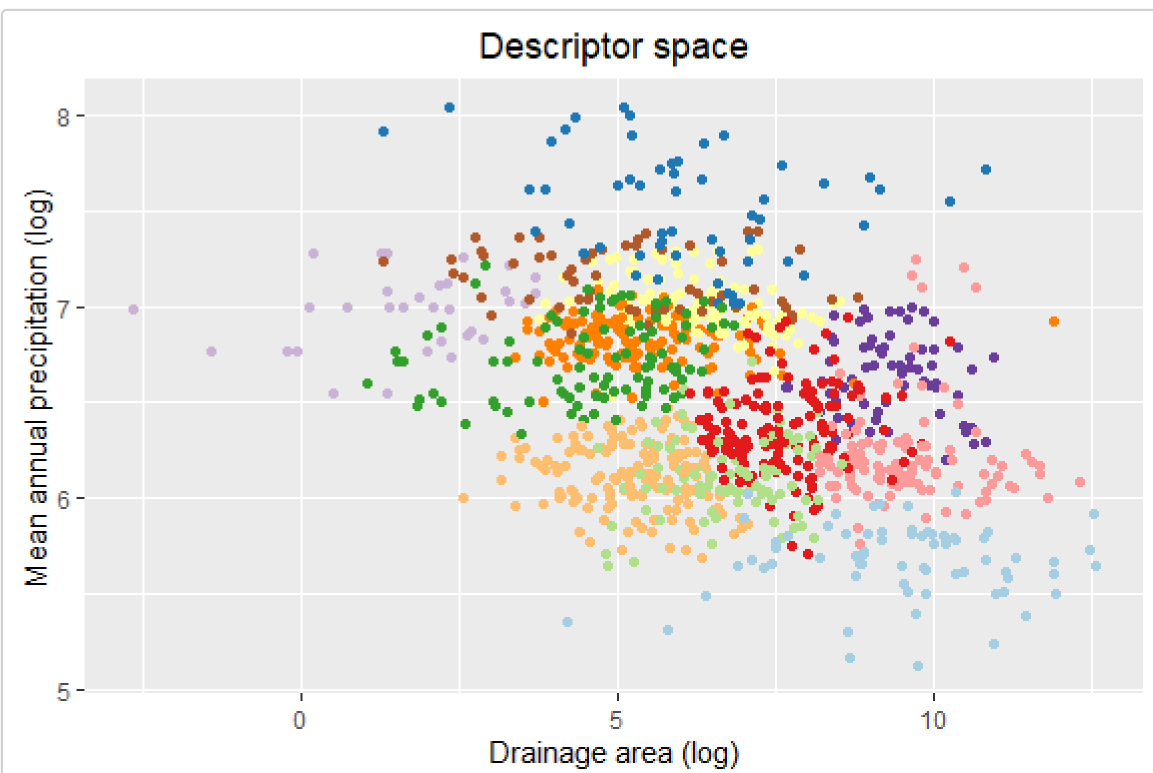
```
FormatPlot(plt, 'Geograpical space')
```




```
## Seasonal plot
plt <- SeasonPlot() +
    geom_point(data = xd, aes(x = theta, y = r, colour = region))

FormatPlot(plt, 'Seasonal space')
```




```r
## Descriptor space
plt <- ggplot() +
    geom_point(data = xd, aes(x = area, y = map, colour = region)) +
    xlab('Drainage area (log)') + ylab('Mean annual precipitation (log)')

FormatPlot(plt, 'Descriptor space')
```



Another information contained in the dataset `gaugedSites` is the p-values of standard trend tests. For AMAX, it includes the result of Mann-Kendall's (`trend_mk`) and Pettitt's (`trend_pt`) tests. For POT, it includes the test of Mann-Kendall on the exceedances (`trend_mx`), and the logistic regression procedure on the exceedance rates (`trend_lg`). Overall, the dataset `gaugedSites` contains sufficient information to select a super region that is hydrologically relevant to the target site and for which the assumption of constant flood risk over time appears reasonable.

```
## Super regions of 01AF009
cond.sreg <-  with(gaugedSites, supreg_km12 == supreg_km12[5])

## with AMAX  stationary sites
cond.trend <- with(gaugedSites, (trend_mk >= 0.05) & (trend_pt >= 0.05))
sreg.amax <- gaugedSites[cond.sreg & cond.trend, 'station']
sreg.amax <- as.character(sreg.amax)

## with POT AMAX stationary sites
cond.trend <- with(gaugedSites, (trend_mx >= 0.05) & (trend_lg >= 0.05))
sreg.pot <- gaugedSites[cond.sreg & cond.trend, c('station', 'auto', 'area')]
```

Regional flood frequency analysis

The function `FloodnetPool` performs RFA and requires hydrometric data in the same format as the previous modelling function. When the argument `target` is passed to `AmaxData`, the function extracts only the data of the desired pooling group. By default, the recommend similarity measure based on the seasonality of the annual maxima serves in the delineation of the pooling group. However, a user can pass an alternative vector of distance to the target to the parameter `distance`. In this case, the target site is identified as the unique site having a distance of zero.

```
set.seed(1)

## Using seasonal distance
fit <- hydat %>%
    AmaxData(sreg.amax, target = mysite) %>%
    FloodnetPool(target = mysite, verbose = FALSE)
```

```
## Using Euclidean distance
sid <- gaugedSites$station %in% sreg.amax
euclid <- gaugedSites[sid, c('area','map')]
euclid <- dist(scale(log(euclid)))
euclid <- as.matrix(euclid)[5,]

hydat %>%
    AmaxData(sreg.amax, distance = euclid) %>%
    FloodnetPool(target = mysite, verbose = FALSE)
```

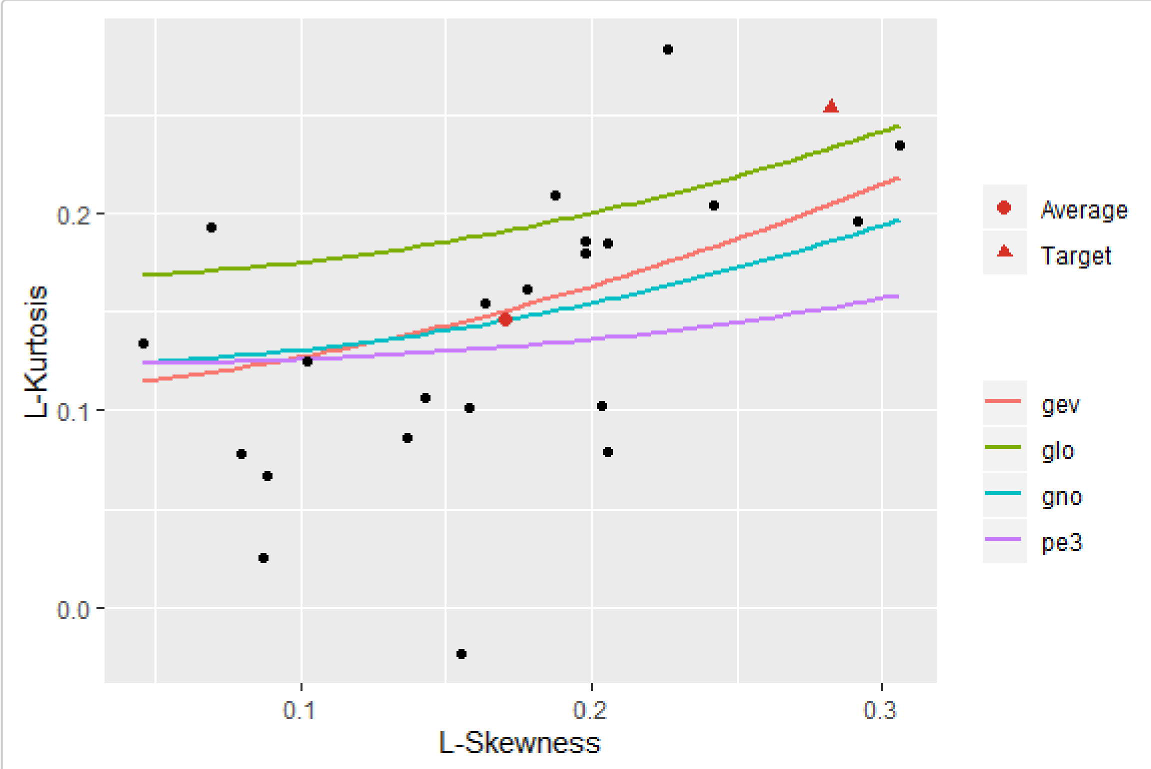
Following the same strategy as the previous modelling functions, the function `FloodnetPool` call the function `FitRegLmom` of the R-package `CSHShydRology` to fit an index-flood model (IFM) using the L-moment algorithm (Hosking and Wallis, 1997). IFM assumes that inside a homogenous region, all distributions are proportional up to a scaling factor that corresponds to the sample average. In particular, this assumption implies that the coefficient of variation of all sites must be the same. The heterogeneous measure H of a pooling group represents the variability of the L-coefficient of variation (LCV) and serves as a criterion for judging the correctness of the IFM hypothesis. If $H > 2$, the pooling group is said to be "most likely heterogeneous" and consequently should be updated. To this end, the modelling function removes in turns each neighbouring site and reevaluate H. At the end of the loop, the station leading to the largest improvement is permanently removed. The process repeated the previous step until successfully updating the pooling group or that it reaches a stopping criterion. In this case, the modelling function stops the procedure if less than 5T station-years are present in the pooling group, where T is the desired return period (Robinson and Reed, 1999). If the user has passed multiple return periods, the function `FloodnetPool` uses the largest return period requested to evaluate that stopping criterion. For example, to evaluate a 100-year return period, at least 500 station-years are necessary. In the end, the function issues a warning if it fails to encounter a pooling group respecting $H \leq 2$.

The standard deviation and the confidence intervals of the flood quantile are estimated using parametric bootstraps. The procedure simulates samples from a multivariate normal distribution (MVN) and transforms the marginal distributions to represent the at-site estimates. Outside the diagonal, the correlation matrix of the MVN has constant coefficients that represent the average of the pairwise correlations. If not specified, the best distribution is selected using the Z-statistic (Hosking and Wallis, 1997) among GEV, GLO, GNO and PE3. This criterion identifies the distribution having the theoretical L-kurtosis that best matches the theoretical one, knowing the sample L-skewness.

In addition to the previous interpreting function, the function `plot` produces an L-moment ratio diagram when used with the argument `type = 'l'`.

```
summary(fit)
##
## Flood Frequency Analysis
##
## Method: pool_amax
## Site: 01AF009
## Distribution: gno
## Return Period: 2 5 10 20 50 100
##
## Quantiles:
##         pred  rmse lower  upper
## 0.5000 38.57 2.804 33.71  44.60
## 0.8000 51.39 3.796 44.72  59.38
## 0.9000 59.76 4.492 51.77  69.37
## 0.9500 67.72 5.207 58.35  78.67
## 0.9800 77.98 6.219 66.56  90.98
## 0.9900 85.69 7.051 72.57 100.04
##
## Parameters:
##          param       se
## IF    40.9481 2.979526
## xi     0.9420 0.007259
## alpha  0.3195 0.009786
## kappa -0.3518 0.038789
```

```
plot(fit, 'l')
```



If the input data are exceedances, RFA is performed using a GPA distribution and accounts for the threshold. The function `ExtractPeaksData` can be used to extract independent peaks from multiples stations. However, if the hydrometric data are coming from HYDAT, it is easier to use the function `DailyPeaksData`. The latter performs as a single instruction the task of importing the hydrometric data (`DailyData`) and extracting the independent peaks (`ExtractPeaksData`). The function also allows limiting the imported data to the desired pooling group.

```
fit <- hydat %>%
    DailyPeaksData(info = sreg.pot, target = mysite) %>%
    FloodnetPool(target = mysite, verbose = FALSE)

summary(fit)
##
## Flood Frequency Analysis
##
## Method: pool_pot
## Site: 01AF009
## Distribution: gpa
## Return Period: 2 5 10 20 50 100
##
## Quantiles:
##          pred  rmse lower  upper
## 0.8015 40.00 2.596 35.23  45.32
## 0.9206 51.90 3.881 44.79  59.87
## 0.9603 60.15 4.801 51.38  69.97
## 0.9801 67.82 5.693 57.34  79.77
## 0.9921 77.10 6.847 64.61  91.63
## 0.9960 83.54 7.711 69.50 100.18
##
## Parameters:
##          param      se
## IF    14.6603 1.57891
```

```
## xi     0.0000 0.02393
## alpha  1.1078 1.57891
## kappa  0.1078 0.02393
```

Prediction at ungauged basins

When there is no hydrometric data at the site of interest, FFA cannot be done simply by fitting a distribution. Instead, we adopt the quantile regression techniques (QRT) to estimate flood quantiles based on available catchment descriptors. The dataset `descriptors`, contains meteorological and physical characteristics of 770 stations, also found in `gaugedSites`. This information can be used to fit a QRT model when the same descriptors are available at the ungauged site.

In this section, we treat the station 01AF009 as ungauged. The objective is to evaluate the 100-year flood quantile (Q100) according to its descriptors. First, we extract and transform six common descriptors: Drainage area (AREA), mean annual precipitation (MAP), percentage of water bodies (WB), stream density (STREAM), elevation (ELEV) and slope (SLOPE). Here, we apply the logarithmic transformations to reshape the data in approximately normal distributions.

```
gauged <- with(descriptors,
  data.frame(
    site = station,
    area = log(area),
    map  = log(map_ws),
    wb   = log(.01 + wb),
    stream = log(.01 + stream),
    elev = elev_ws,
    slope = log(.01 + slope)))

## Separating the target from the other stations
target.id <- which(gauged$site == '01AF009')

target <- gauged[target.id,]
gauged <- gauged[-target.id,]
```

The function `FloodnetRoi` is built on the top of the function `FitRoi` in `CSHShydRology` and estimates the flood quantiles of one, or more, ungauged sites by the QRT method. In addition to the catchment descriptors, the function requires the annual maxima of all gauged sites. In the example below, the function `AmaxData` provides the hydrometric data.

```
set.seed(1)
fit <- hydat %>%
    AmaxData(gauged$site) %>%
    FloodnetRoi(target = target, sites = gauged, period = 100, size = 30)
```

To explain the procedure briefly, it starts by evaluating the desired flood quantiles $\mathbf{q} = (q_1, \ldots, q_n$ for each gauged station using the AMAX method and passes the resulting at-site estimate to a locally log-linear model

$$\log(\mathbf{q}) = \mathbf{X}\beta + \mathbf{e}$$

where $\mathbf{X}$ is a design matrix of descriptors, β is a vector of parameters and $\mathbf{e}$ is a term of errors. For each target site, the procedure fits a regression model using a weighted least-squares approach that gives more weights to the nearest stations. The Euclidean distance between standardized descriptors serves to evaluate these weights according to the Epanechnikov kernel. The local regression model requires the calibration of a bandwidth parameter that controls the weight decay that creates regions of influence (ROI), or pooling groups outside of which the weights are zero. The `FloodnetRoi` expresses the bandwidth parameter as the rank of the nearest gauged sites to the target, which corresponds to the size of the pooling groups.

Similarly to the outputs of the gauged analyses, interpreting functions exist to extract information from the output of the ungauged analyses. The function `print` can display the estimated flood quantiles, or the user can use the function `as.data.frame` to convert the model to a dataset. Remember that we estimated Q100 equal to 98 m^3/s using the AMAX method. Here we predict a Q100 of 89 m^3/s based on the catchment descriptors.

```
print(fit)
##
## Predictions at ungauged sites
##
## Method: qrt
## Site: 01AF009
## Pool size: 30
## Return Period: 100
##
## Quantiles:
##         quantile
## 01AF009    88.84
as.data.frame(fit)
##      site method size period variable  value
## 1 01AF009    qrt   30    100 quantile 88.837
```

After the fitting step, the modelling function uses a 10-fold cross-validation resampling strategy to assess the quality of the results. The strategy consists of dividing the set of gauged sites into ten groups of approximately equal sizes. In turn, it treats each validation group as ungauged and estimates its flood quantiles. Note that the validation groups are divided at random.
Consequently, the outcome of the cross-validation procedure will differ for each function call.

The Nash-Sutcliffe criterion or NASH is a prediction skill score that rescales the mean square error into a unitless measure that quantifies the model performance with respect to the sample average. A NASH of 1 indicates a perfect score, while a score close to zero indicates poor performance. To account for the scale of the watersheds, the NASH is applied here to the logarithm values.
The same rescalling strategy is applied to derive the following skill score

$$SKILL = 1 - \frac{\sum_{i=1}^{n} |l_i - \hat{l}_i|}{\sum_{i=1}^{n} |l_i - \bar{l}_i|}$$

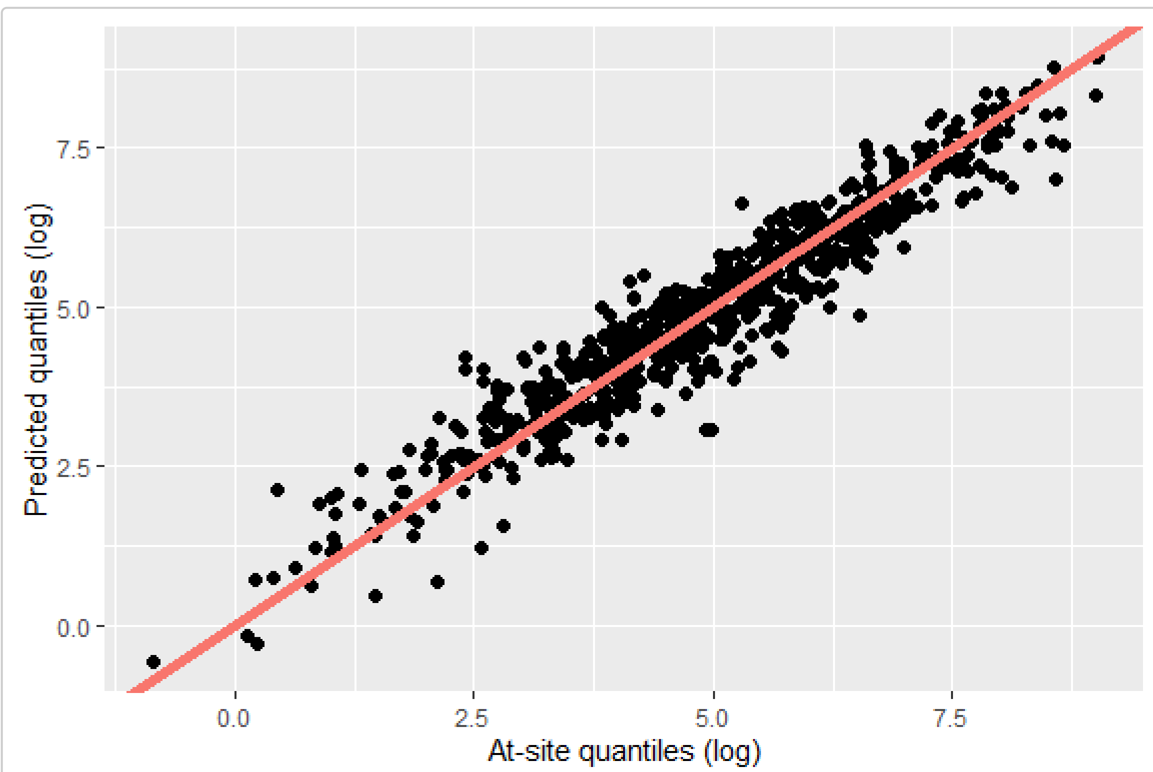
where $l_i = \log(q_i)$, $\hat{l}_i$ is a predicted value and $\bar{l}_i$ is the sample average. This criterion has the same forms as the NASH but replaces the square differences by absolute deviations. If the user provides more than one pooling group size to `FloodnetRoi`, the function determines the optimal size by cross-validation based on the SKILL criterion, which is equivalent to minimize the Mean Absolute Deviation (MAD). The function `summary` displays the results of the selection procedure and the assessment of the final model by cross-validation. Note in the example below, the small discrepancy between the two `nash` criteria. This difference results from using different cross-validation samples for the calibration and the evaluation of the QRT model.

```
fit <- hydat %>%
    AmaxData(gauged$site) %>%
    FloodnetRoi(target = target, sites = gauged, period = 100,
                            size = seq(20, 120, 10), verbose = FALSE)
```

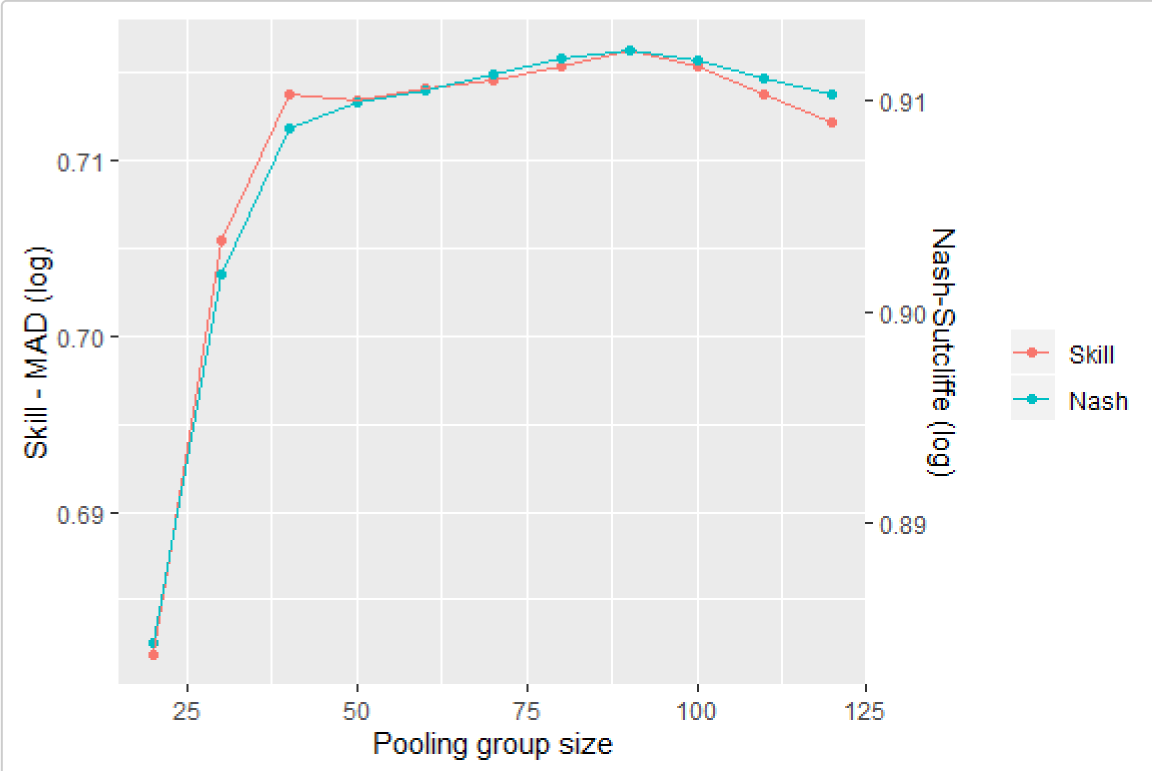
```r
as.data.frame(fit)
##      site method size period variable    value
## 1 01AF009    qrt   90    100 quantile 109.6906
summary(fit)
##
## Best Cross Validation Scores:
##   size   nash  skill
## 8   90 0.9123 0.7163
## 7   80 0.9120 0.7154
## 9  100 0.9119 0.7154
## 6   70 0.9112 0.7145
## 5   60 0.9105 0.7140
##
## Cross Validation Scores:
##    rmse   mad  rrmse   rmad   lmse   lmad   nash  skill
## 1 410.7 149.5 0.6199 0.4104 0.4934 0.3806 0.9126 0.7156
```

The two following graphics assess the estimated flood quantiles and the calibration of the pooling group size. The first plot shows the at-site estimates of Q100 versus the values predicted for each cross-validation samples. A deviation from the unitary line suggests systematic errors in the QRT model. The second plot shows the evolution of the two cross-validation scores with respect to the pooling group size.

```r
plot(fit)
```




```r
plot(fit, 'cv')
```



To estimate the uncertainty of the QRT model, the function `FloodnetRoi` uses a hybrid bootstrap resampling technique. As done for RFA, the method first simulates annual maxima from the proper transformation of multivariate normal distribution. Next, it estimates the flood quantiles $\mathbf{q}_i^*$ at each gauged site i. Independently, the method samples prediction errors $\mathbf{e}_i^*$ (balance bootstrap) from the residuals of the QRT model based on the prediction obtained during the cross-validation. The final bootstrap sample is composed of the flood quantiles values $\mathbf{q}_i^{**} = \mathbf{q}_i^* + \mathbf{e}_i^*$ that account for both the modeling error and the sampling error. Close sites are more likely to have correlated annual maxima. To use a correlation matrix in the bootstrap procedure that represents that reality, the function `IntersiteCorData` evaluates such correlation matrix according to an exponential correlation.

```
## Estimation of the intersite correlation matrix
icor <- IntersiteCorData(hydat, gauged$site, smooth = 0.6, distance.max = 300)

## Ungauged analysis with bootstrap
set.seed(1)
fit <- hydat %>%
    AmaxData(gauged$site) %>%
    FloodnetRoi(target = target, sites = gauged, period = 100, size = 90,
                            nsim = 30, corr = icor)
##
## [Bootstrapping]
## ================================================================================

print(fit)
##
## Predictions at ungauged sites
##
## Method: qrt
## Site: 01AF009
## Pool size: 90
## Return Period: 100
```

```
##
## Quantiles:
##          quantile    se lower upper
## 01AF009    109.7 24.37 79.72 170.2
```

The catchment descriptors in the dataset `descriptors` represent only partial information about the available catchment. In particular, several missing descriptors could be spatially distributed, which may lead to spatially correlated residuals. To account for residual spatial correlation among the estimated flood quantiles, the user can provide to `FloodnetRoi` the sites coordinate to performs simple kriging.

The example below uses multidimensional scaling to project the geographical coordinates into a Cartesian space that approximately preserves the great-circle distance and passes the coordinates to the modelling functions. Here, it shows an improvement of the criterion SKILL from 0.72 to 0.75.

```
library(CSHShydRology)

## Extract the coordinates
coord <- descriptors[, c('lon', 'lat')]
coord <- as.data.frame(cmdscale(GeoDist(coord), 2))

target.coord <- coord[target.id,]
coord <- coord[-target.id,]

## Predict target using ROI + simple kriging
set.seed(1)
fit <- hydat %>%
    AmaxData(gauged$site) %>%
    FloodnetRoi(sites = gauged, target = target, sites.coord = coord, target.coord = target.coord,
              size = 300, period = 100, verbose = FALSE)

summary(fit)
##
## Cross Validation Scores:
##    rmse   mad rrmse   rmad   lmse   lmad  nash  skill
## 1 391.7 141.7 0.623 0.3671 0.4511 0.3366 0.927 0.7485
```

Conclusion

In summary, this document showed how the R-package `floodnetRfa` can perform FFA using the hydrometric data from the HYDAT database. Data extraction functions `AmaxData`, `DailyData` and `DailyPeakData` served to extract from HYDAT the hydrometric data in the correct format. The outputs were passed to the modelling functions `FloodnetAmax`, `FloodnetPot` and `FloodnetPool`that carried out FFA based on AMAX, POT, and RFA methods. Finally, interpreting functions `print`, `summary`, `as.data.frame` and `plot` extract the estimated flood quantiles and assess the fitted models. Similarly, the function `FloodnetRoi` estimated flood quantiles at ungauged sites.

References

- Coles, S. (2001). An introduction to statistical modeling of extreme values. Springer Verlag.

- Durocher, M., Burn, D. H., & Mostofi Zadeh, S. (2018a). A nationwide regional flood frequency analysis at ungauged sites using ROI/GLS with copulas and super regions. Journal of Hydrology, 567, 191–202. https://doi.org/10.1016/j.jhydrol.2018.10.011
- Durocher, M., Zadeh, S. M., Burn, D. H., & Ashkar, F. (2018b). Comparison of automatic procedures for selecting flood peaks over threshold based on goodness-of-fit tests. Hydrological Processes. https://doi.org/10.1002/hyp.13223
- Durocher, M., Burn, D. H., Zadeh, S. M., & Ashkar, F. (2019). Estimating flood quantiles at ungauged sites using nonparametric regression methods with spatial components. Hydrological Sciences Journal, 64(9), 1056–1070. https://doi.org/10.1080/02626667.2019.1620952
- Helsel, D. R., & Hirsch, R. M. (2002). Statistical Methods in Water Resources. In Techniques of Water-Resources Investigations of the United States Geological Survey. Retrieved from http://water.usgs.gov/pubs/twri/twri4a3/
- Hosking, J. R. M., & Wallis, J. R. (1997). Regional frequency analysis: An approach based on L-moments. Cambridge Univ Pr.
- Lang, M., Ouarda, T. B. M. J., & Bobée, B. (1999). Towards operational guidelines for over-threshold modeling. Journal of Hydrology, 225(3), 103–117. https://doi.org/10.1016/S0022-1694(99)00167-5
- Mostofi Zadeh, S., & Burn, D. H. (2019). A Super Region Approach to Improve Pooled Flood Frequency Analysis. Canadian Water Resources Journal / Revue Canadienne Des Ressources Hydriques, 0(0), 1–14. https://doi.org/10.1080/07011784.2018.1548946
- Robson, A., & Reed, D. (1999). Flood estimation handbook. Institute of Hydrology, Wallingford.