**R functions for frequency analysis**

v0.2.4

Guénolé Choné - guenole.chone@concordia.ca

This document provides information on how to use the functions in the file *frequencyanalysis.R*, to do frequency analyses in R

**Loading the script functions and the required data**

The file *frequencyanalysis.R* need to be added in the R environment by opening it (*File -> Open* File in R studio), and then clicking the *Source* icon

A screenshot of a computer

Description automatically generated

The functions to do the frequency analyses require the data (annual maximum flow) to be provided as a vector in R.

Example data:

flow **<-** c**(**428,572,518,501,535,631,453,609,473,422,603,462,493,660,827,674,580,558,521,436,538,858,515,422,464,472,993,897,592,704,1310,380,683,651,379,432,757, 1410,543,470,851,486,827,548,540,559,1490,515,595,504,748,473,880**)**

If the data are in a csv file, they can be easily loaded into a vector:

alldata **<-** read.csv**(**"D:/R programming/AnalyseFrequentielle/QC\_AMS\_23Stations.csv"**)**

flow **<-** alldata**[!**is.na**(**alldata**[**,4**])**,4**]** # take the 4th column of the csv file, removing potential empty cells.

**Do a frequency analysis**

The function *FA\_fit* fits a statistical distribution into the provided data. The supported distributions include:

**exp** = Exponential distribution (two parameters)

**gam** = Gamma distribution (two parameters)

**gev** = Generalized Extreme Value distribution (three parameters)

**gum** = Gumbel distribution (two parameters)

**nor** = Normal distribution (two parameters)

**lno =** Log-Normal distribution (two parameters)

**pe3** = Pearson Type III distribution (three parameters)

**lpe** = Log-Pearson Type III distribution (three parameters)

**wei** = Weibull distribution (three parameters)

**gno** = Generalized Normal distribution (three parameters)

The function used mainly the R package *lmomco*, that supports other statistical distributions. A complete list of implemented distributions is available at: <https://www.rdocumentation.org/packages/lmomco/versions/1.4.3/topics/Introduction>

Example of frequency analysis using the GEV or the log-Pearson type III distribution:

fa **<-** FA\_fit**(**series**=**flow, distribution**=**"gev"**)**

fa **<-** FA\_fit**(**series**=**flow, distribution**=**"lp3"**)**

Parameters of the distribution are estimated by default using the L-moments method. This can be changed for the maximum likelihood estimation instead, or for the moment matching estimation for the log-Pearson type III distribution only:

fa **<-** FA\_fit**(**series**=**flow, distribution**=**"gev", method**=**"mle"**)**

fa **<-** FA\_fit**(**series**=**flow, distribution**=**"lp3, method**=**"mme"**)**

NB: the moment matching estimation for the log-Pearson type III distribution is the method used by the USGS (bulletin 17 method).

To get estimated flows for return periods, the function *FA\_getvalues* can be used:

FA\_getvalues**(**fa**)**

nep rp estimate

1 0.5000000 2 558.8506

2 0.9500000 20 1076.3260

3 0.9900000 100 1690.6935

4 0.9971429 350 2428.3316

Default return periods are 2, 20, 100 and 350 years. This can be changed by providing a vector of probability of exceedance (probability of exceedance = 1 – 1/return period) to *FA\_getvalues*:

FA\_getvalues**(**fa, nep**=**c**(**0.5, 0.9, 0.95, 0.99, 1**-**1**/**350, 1**-**1**/**500**))** #adding the 10-yr and 500-yr return periods

**Comparing frequency analyses with different statistical distributions**

The object returned by the function *FA\_fit* contains 5 different assessment of the goodness of fit: Chi-square test (providing a Chi-square score and a p-value), the Akaike Information Criterion (AIC) and its variant the Bayesian Information Criterion (BIC), the Kolmogorov-Smirnov test (providing the Kolmogorov-Smirnov score and a p-value) and the Anderson-Darling test (also providing a score and a p-value).

Chi-square score: lower is better

Chi-square p-value: higher is better – a p<0.05 means that the distribution does not fit the data.

AIC: lower is better.

BIC: lower is better.

Kolmogorov-Smirnov score: lower is better

Kolmogorov-Smirnov p-value: higher is better – a p<0.05 means that the distribution does not fit the data.

AIC, BIC and Anderson-Darling test cannot always be computed for distributions with parameters assessed by the L-moments method (Inf value)

The function *FA\_comparisons()* helps visualize these results by providing a dataframe, for several statistical distributions:

FA\_comparisons**(**flow**)**

distribution chi\_square chi\_square\_rank chi\_square\_p AIC AIC\_rank

1 wei 2.756571 4 0.43069915 **Inf** 4.5

2 gev 1.770760 2 0.62131843 695.6035 1.0

3 gam 11.764920 5 0.01918774 715.2205 3.0

4 lp3 1.879503 3 0.59778891 **Inf** 4.5

5 gno 1.721214 1 0.63222784 697.5824 2.0

(NB: Example result table is truncated to display only chi-square and AIC results)

The default distributions list and the parameters estimation method can be modified this way:

FA\_comparisons**(**flow, method**=**"mle", list\_distr **=** c**(**"wei", "gev", "gam", "exp"**))**

*FA\_comparisons()* also output a graphic of the fitted statistical distribution with the histogram of the input data. The default number of bins of the histogram can be changed by the argument *nb\_bins*:

FA\_comparisons**(**flow, method**=**"mle", nb\_bins = 20)

The results table can be easily exported in a csv file if needed:

comparison **<-** FA\_comparisons**(**flow**)**

write.csv**(**comparison, "D:/R programming/AnalyseFrequentielle/Export\_comparison.csv"**)**

**Compute confidence intervals**

The function *BootstrapCI()* add the confidence interval, which can be displayed using *FA\_getvalues()*

fa **<-** BootstrapCI**(**fa**)**

FA\_getvalues**(**fa**)**

nep rp lower estimate upper

1 0.5000000 2 517.9754 558.8506 609.4706

2 0.9500000 20 844.4567 1076.3260 1377.7236

3 0.9900000 100 1064.0038 1690.6935 2829.4551

4 0.9971429 350 1234.5620 2428.3316 5275.9352

Default lower and upper values are for a 95% confidence interval. This can be changed with the *ci* parameter:

FA\_getvalues**(**fa, ci=0.9**)** #using the 90% confidence intervals instead of 95%

The function *BootstrapCI()* works by bootstrapping, a Monte-Carlo approach. The number of iterations can be modified (default is 25000):

fa **<-** BootstrapCI**(**fa, n.resamples **=** 1000**)**

The bootstrapping is done using the L-moments method for computational efficiency, even if the original frequency analysis was computed using the maximum likelihood estimation or the moment matching estimation. Bootstrapping using the original frequency analysis can be forced:

fa **<-** BootstrapCI**(**fa, n.resamples **=** 1000, use\_lmom **=** **FALSE)**

Note that bootstapping with the maximum likelihood estimation or the moment matching estimation can be long. Lowering the number of iterations is recommended in this case.

**Visualization of a frequency analysis**

A graph of the resulting function can be plotted using *frequencyPlot()*. The input for this function can be either the result of *FA\_fit* or the result of *BootstrapCI()*.

frequencyPlot(fa)

![A graph with a line

Description automatically generated]()