#useR #Rstudio
#hydrology #timeseries

Atelier numérique
Workshop

hydRologie avec R
hydRology with R

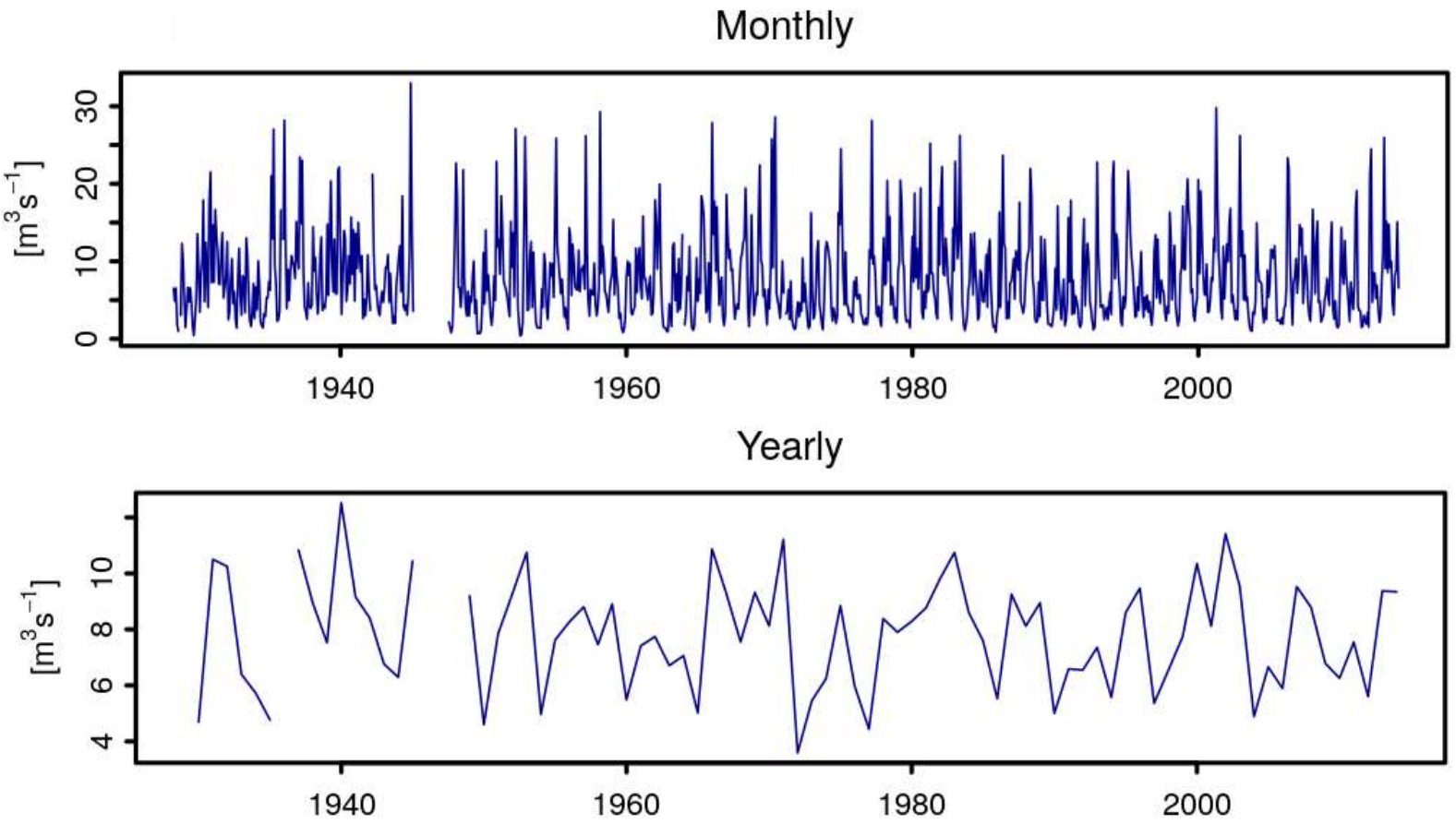
Dr. Pedro RAU
Hydrologue

https://github.com/hydrocodes

UTEC UNIVERSIDAD DE INGENIERÍA Y TECNOLOGÍA
cita CENTRO DE INVESTIGACIÓN Y TECNOLOGÍA DEL AGUA · UTEC
ESCUELA DE POSGRADO UTEC
FURIA DE LOS RIOS
Young Hydrologic Society

1. Introduction aux séries temporelles hydrologiques

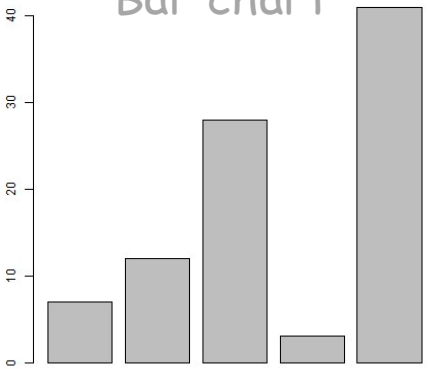
1.1 ¿Comment analyser une serie temporelle?



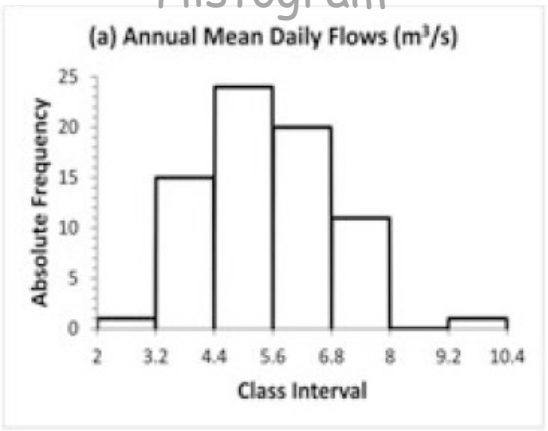
1.2 Analyse préliminaire de données hydrologiques

a. Types de répresentations graphiques

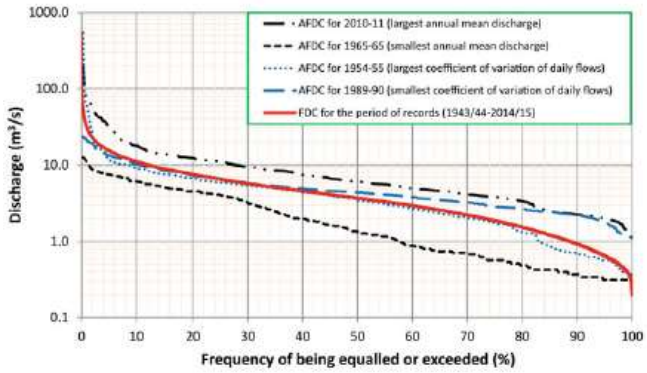
Diagramme à barres
Bar chart



Histogramme
Histogram



Courbe de duration
Duration curves



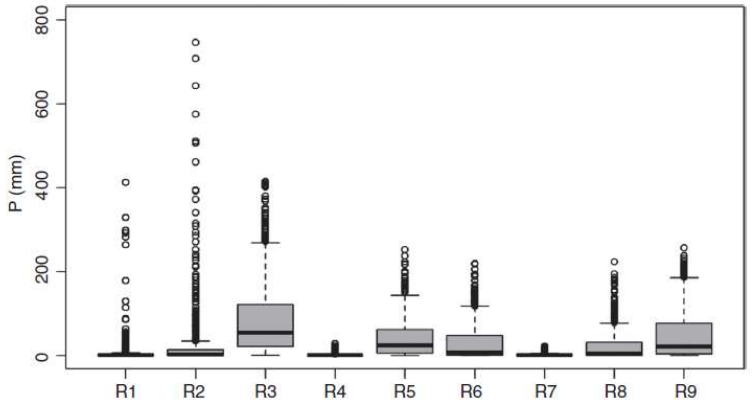
b. Statistiques descriptifs

Parámetro de la población	Estadística de la muestra
1. *Punto medio*	
Media aritmética	
$\mu = E(X) = \int_{-\infty}^{\infty} x f(x)\,dx$	$\bar{x} = \frac{1}{n}\sum_{i=1}^{n} x_i$
Mediana	
x tal que $F(x) = 0.5$	Valor de la información en el 50o. percentil
Media geométrica	
antilog $[E(\log x)]$	$\left(\prod_{i=1}^{n} x_i\right)^{1/n}$
2. *Variabilidad*	
Varianza	
$\sigma^2 = E[(x-\mu)^2]$	$s^2 = \frac{1}{n-1}\sum_{i=1}^{n}(x_i-\bar{x})^2$
Desviación estándar	
$\sigma = \{E[(x-\mu)^2]\}^{1/2}$	$s = \left[\frac{1}{n-1}\sum_{i=1}^{n}(x_i-\bar{x})^2\right]^{1/2}$
Coeficiente de variación	
$CV = \frac{\sigma}{\mu}$	$CV = \frac{s}{\bar{x}}$
3. *Simetría*	
Coeficiente de asimetría (oblicuidad)	
$\gamma = \frac{E[(x-\mu)^3]}{\sigma^3}$	$C_s = \frac{n\sum_{i=1}^{n}(x_i-\bar{x})^3}{(n-1)(n-2)s^3}$

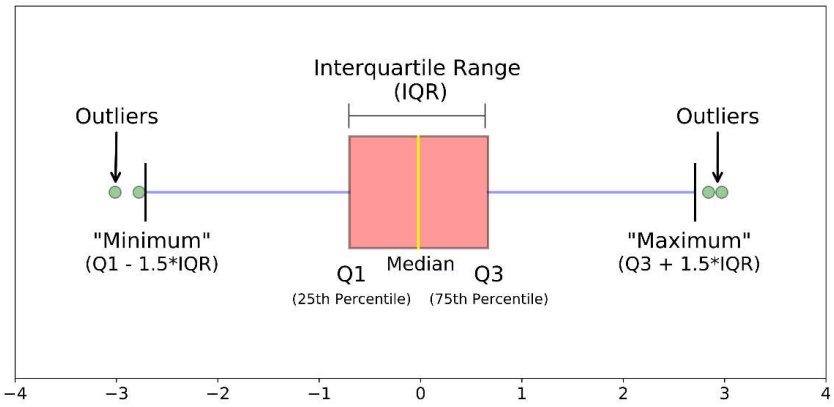
Chow (1994)
Naghettini (2017)

c. Méthodes exploratoires

Boîte à moustaches
Box plot



Rau et al (2017)



Naghettini (2017)

d. Association de données

Nuages de points
Scatter plot

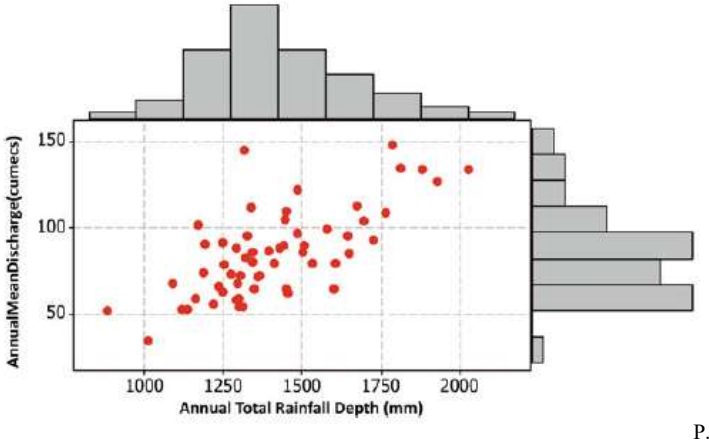
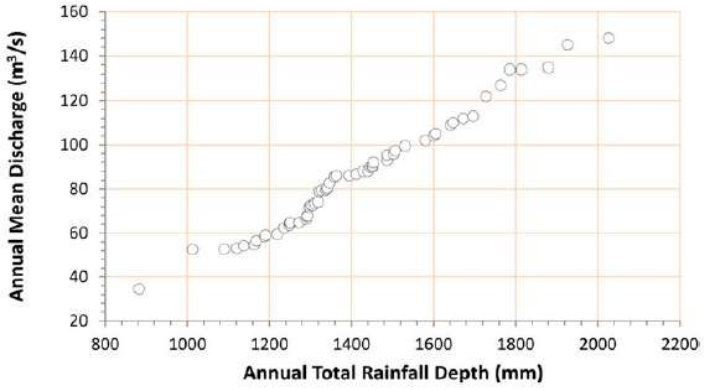
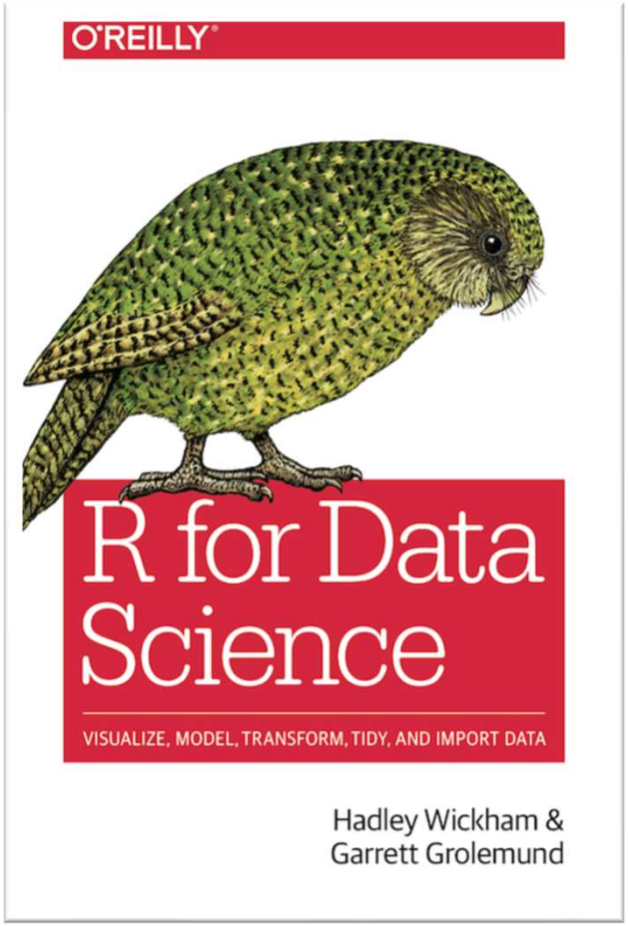


Diagramme Quantile-Quantile
Empirical Quantile-Quantile Diagram
Q-Q Plot



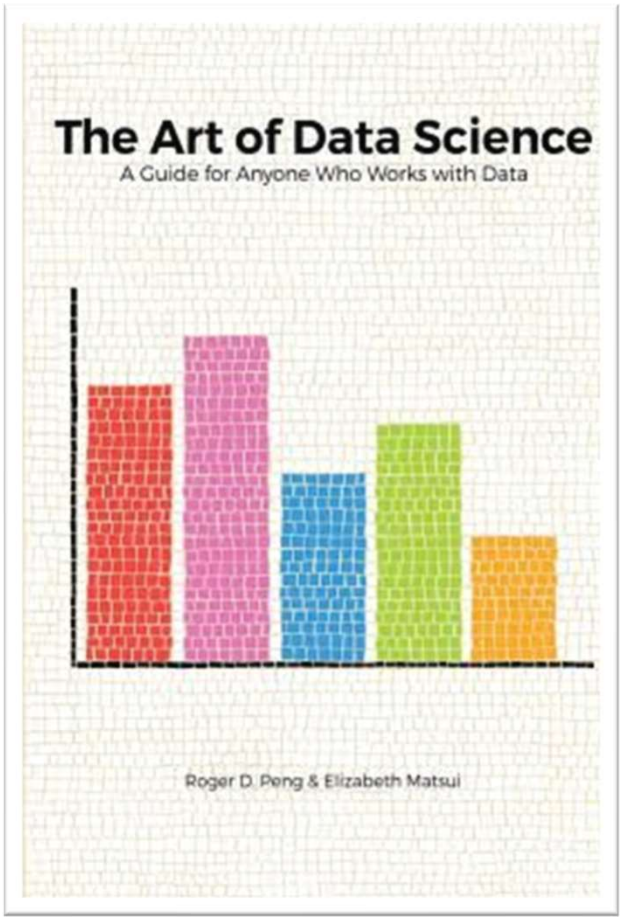
2. Environnement R et Rstudio



https://r4ds.had.co.nz/

https://es.r4ds.hadley.nz/

https://github.com/jrnold/r4ds-exercise-solutions/blob/master/README.md



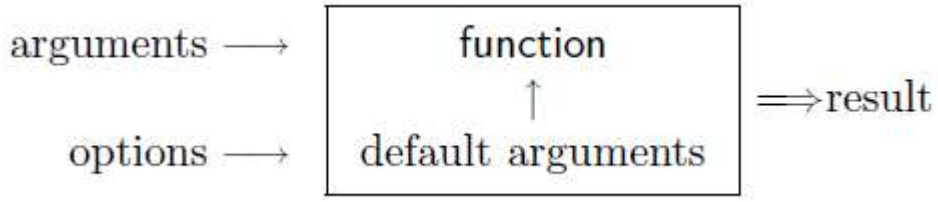
https://bookdown.org/rdpeng/artofdatascience/

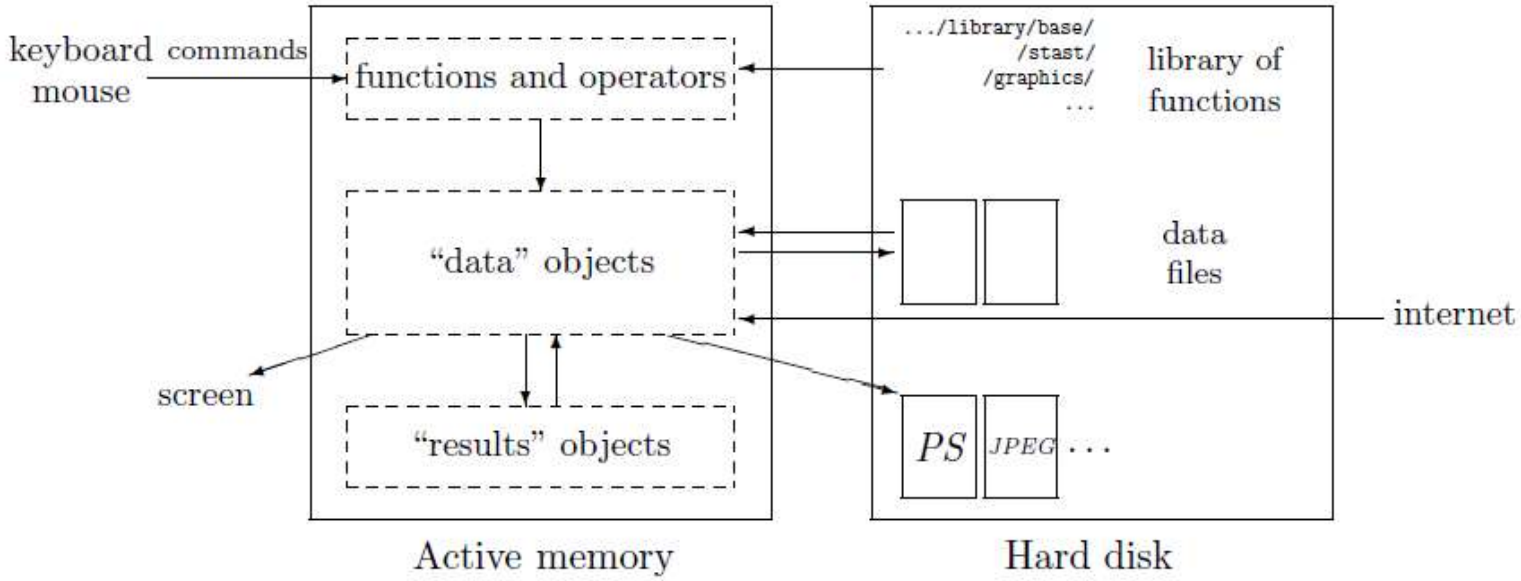
https://github.com/waldronlab/The-Art-of-Data-Science/blob/master/README.md

R-4.2.3 pour Windows (32/64 bit) – mars 2023

https://cran.r-project.org/bin/windows/base/

- ✓ Système pour l'analyse statistique et de graphiques (Ihaka y Gentleman, 1996).
- ✓ Langage interprété, ce n'est pas un langage compilé.
- ✓ Facile et intuitive, « ce n'est pas un langage de programmation » strictement.



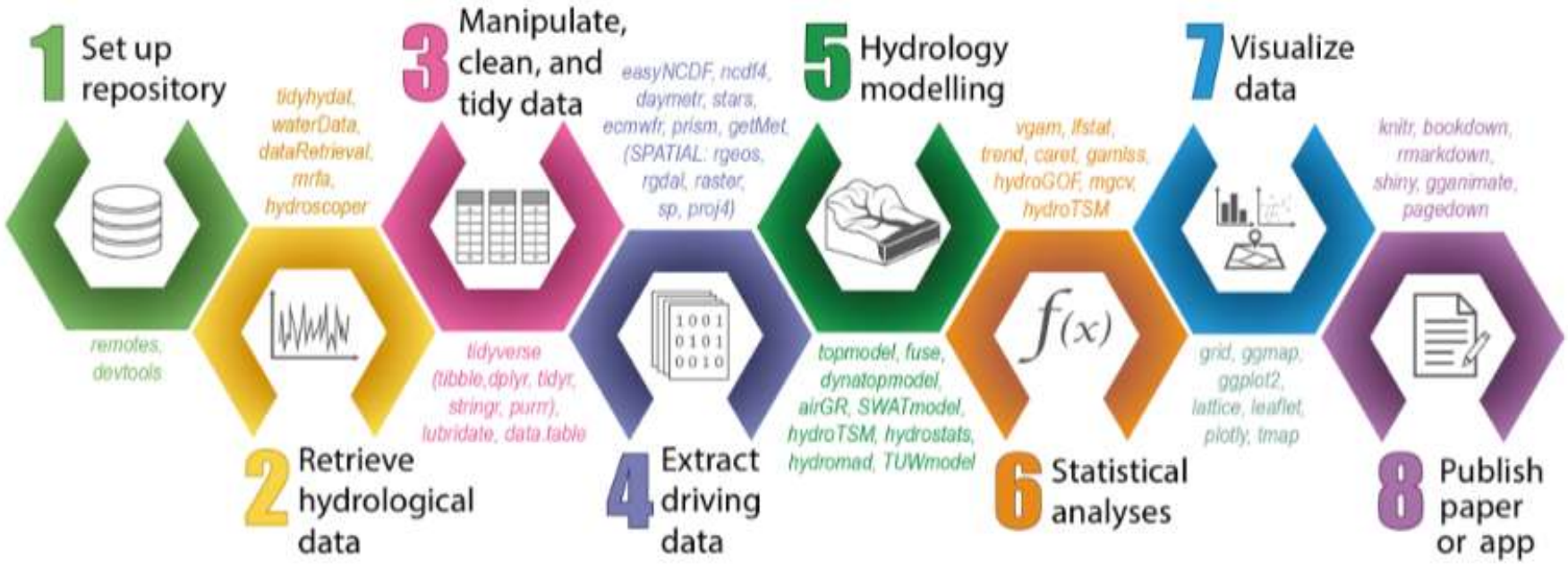


Paradis (2002)

The Comprehensive R Archive Net-work (CRAN)

https://cran.r-project.org

Flux de travail typique en R en hydrologie



Slater et al (2019)

<-

"Classes" d'objets basiques ou "atomic" en R:

- numeric (nombres réels)
- integer
- complex
- logical (True/False)

( )

Types d'objets en R pour représenter les donnéés

object	modes	several modes possible in the same object?	
vector	numeric, character, complex *or* logical	No	c()
factor	numeric *or* character	No	factor()
array	numeric, character, complex *or* logical	No	array()
matrix	numeric, character, complex *or* logical	No	matrix()
data frame	numeric, character, complex *or* logical	Yes	data.frame()
ts	numeric, character, complex *or* logical	No	ts()
list	numeric, character, complex, logical, function, expression, ...	Yes	list()

Paradis (2002)

#

Conversion entre classes d'objets

Conversion to	Function	Rules
numeric	`as.numeric`	`FALSE` → 0
		`TRUE` → 1
		`"1"`, `"2"`, ... → 1, 2, ...
		`"A"`, ... → `NA`
logical	`as.logical`	0 → `FALSE`
		other numbers → `TRUE`
		`"FALSE"`, `"F"` → `FALSE`
		`"TRUE"`, `"T"` → `TRUE`
		other characters → `NA`
character	`as.character`	1, 2, ... → `"1"`, `"2"`, ...
		`FALSE` → `"FALSE"`
		`TRUE` → `"TRUE"`

Fonctions de probabilité

Distribución/función	**función**
Gausse (normal)	`rnorm(n, mean=0, sd=1)`
exponencial	`rexp(n, rate=1)`
gamma	`rgamma(n, shape, scale=1)`
Poisson	`rpois(n, lambda)`
Weibull	`rweibull(n, shape, scale=1)`
Cauchy	`rcauchy(n, location=0, scale=1)`
beta	`rbeta(n, shape1, shape2)`
'Student' (t)	`rt(n, df)`
Fisher–Snedecor (F)	`rf(n, df1, df2)`
Pearson (χ^2)	`rchisq(n, df)`
binomial	`rbinom(n, size, prob)`
geométrica	`rgeom(n, prob)`
hypergeométrica	`rhyper(nn, m, n, k)`
logística	`rlogis(n, location=0, scale=1)`
lognormal	`rlnorm(n, meanlog=0, sdlog=1)`
binomial negativa	`rnbinom(n, size, prob)`
uniforme	`runif(n, min=0, max=1)`
Estadístico de Wilcoxon's	`rwilcox(nn, m, n)`, `rsignrank(nn, n)`

Opérateurs en R

Operators					
Arithmetic		Comparison		Logical	
`+`	addition	`<`	lesser than	`! x`	logical NOT
`-`	subtraction	`>`	greater than	`x & y`	logical AND
`*`	multiplication	`<=`	lesser than or equal to	`x && y`	id.
`/`	division	`>=`	greater than or equal to	`x | y`	logical OR
`^`	power	`==`	equal	`x || y`	id.
`%%`	modulo	`!=`	different	`xor(x, y)`	exclusive OR
`%/%`	integer division				

`plot(x)`	plot of the values of `x` (on the y-axis) ordered on the x-axis
`plot(x, y)`	bivariate plot of `x` (on the x-axis) and `y` (on the y-axis)
`sunflowerplot(x, y)`	id. but the points with similar coordinates are drawn as a flower which petal number represents the number of points
`pie(x)`	circular pie-chart
`boxplot(x)`	"box-and-whiskers" plot
`stripchart(x)`	plot of the values of `x` on a line (an alternative to `boxplot()` for small sample sizes)
`coplot(x~y | z)`	bivariate plot of `x` and `y` for each value (or interval of values) of `z`
`interaction.plot (f1, f2, y)`	if `f1` and `f2` are factors, plots the means of `y` (on the y-axis) with respect to the values of `f1` (on the x-axis) and of `f2` (different curves); the option `fun` allows to choose the summary statistic of `y` (by default `fun=mean`)
`matplot(x,y)`	bivariate plot of the first column of `x` *vs.* the first one of `y`, the second one of `x` *vs.* the second one of `y`, etc.
`dotchart(x)`	if `x` is a data frame, plots a Cleveland dot plot (stacked plots line-by-line and column-by-column)
`fourfoldplot(x)`	visualizes, with quarters of circles, the association between two dichotomous variables for different populations (`x` must be an array with `dim=c(2, 2, k)`, or a matrix with `dim=c(2, 2)` if $k = 1$)
`assocplot(x)`	Cohen–Friendly graph showing the deviations from independence of rows and columns in a two dimensional contingency table
`mosaicplot(x)`	'mosaic' graph of the residuals from a log-linear regression of a contingency table
`pairs(x)`	if `x` is a matrix or a data frame, draws all possible bivariate plots between the columns of `x`
`plot.ts(x)`	if `x` is an object of class `"ts"`, plot of `x` with respect to time, `x` may be multivariate but the series must have the same frequency and dates
`ts.plot(x)`	id. but if `x` is multivariate the series may have different dates and must have the same frequency
`hist(x)`	histogram of the frequencies of `x`
`barplot(x)`	histogram of the values of `x`
`qqnorm(x)`	quantiles of `x` with respect to the values expected under a normal law
`qqplot(x, y)`	quantiles of `y` with respect to the quantiles of `x`
`contour(x, y, z)`	contour plot (data are interpolated to draw the curves), `x` and `y` must be vectors and `z` must be a matrix so that `dim(z)=c(length(x), length(y))` (`x` and `y` may be omitted)
`filled.contour (x, y, z)`	id. but the areas between the contours are coloured, and a legend of the colours is drawn as well
`image(x, y, z)`	id. but the actual data are represented with colours
`persp(x, y, z)`	id. but in perspective
`stars(x)`	if `x` is a matrix or a data frame, draws a graph with segments or a star where each row of `x` is represented by a star and the columns are the lengths of the segments
`symbols(x, y, ...)`	draws, at the coordinates given by `x` and `y`, symbols (circles, squares, rectangles, stars, thermometres or "boxplots") which sizes, colours, etc, are specified by supplementary arguments
`termplot(mod.obj)`	plot of the (partial) effects of a regression model (`mod.obj`)

Tracer des graphiques

add=FALSE if TRUE superposes the plot on the previous one (if it exists)

axes=TRUE if FALSE does not draw the axes and the box

type="p" species the type of plot, "p": points, "l": lines, "b": points connected by lines, "o": id. but the lines are over the points, "h": vertical lines, "s": steps, the data are represented by the top of the vertical lines, "S": id. But the data are represented by the bottom of the vertical lines

xlim=, ylim= species the lower and upper limits of the axes, for example
with xlim=c(1, 10) or xlim=range(x)

xlab=, ylab= annotates the axes, must be variables of mode character

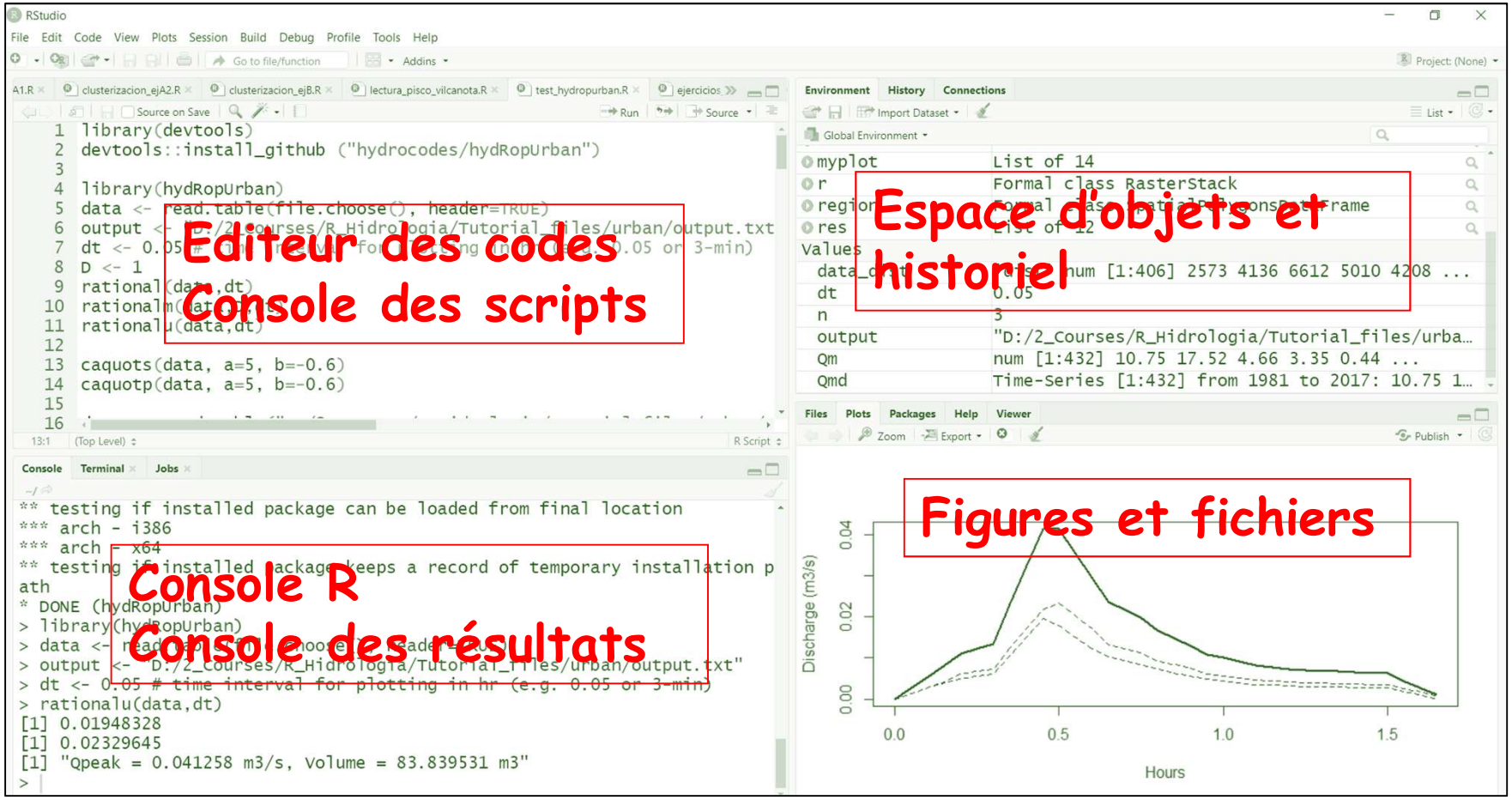
main= main title, must be a variable of mode character

sub= sub-title (written in a smaller font)

RStudio Desktop 2023

https://rstudio.com/products/rstudio/download/#download

Environnement de développement intégré pour le langage de programmation R



A. Commandes et codes dans l'environnement Rstudio

Exercice A.1

Dans la **console de résultats**, effectuer les opérations suivantes:

1) $3^2 - 5 * 9 * (25 - 18)$
2) Créer une série consécutive de 1 al 8.
3) Créer une série consécutive de 8 lettres en commençant par "a".
4) Attribuer l'opération $8 * 5^2$ à l'objet *value*.
5) *Visualiser l'objet value crée.*
6) *Créer l'objet p, en stockant une séquence depuis 5 jusqu'a 15 de 2 en 2.*
7) Visualiser *p*.
8) Stocker les valeurs suivantes de pluie annuelle (mm/an) de differents stations pluviométriques d'un bassin hydrographique: 200, 210.2, 490, 100.5, 150.1, 190, 310 y 200.2 dans l'objet *rain*.
9) Visualiser *rain*.
10) Obtenit le 5éme element de *rain*.
11) Stocker les valeurs suivantes d'Elevation (msnm) : 3200, 3500, 4500, 3050, 3100, 2800, 3800, 3500 dans l'objet *elev*.
12) Visualiser *elev*.

Réponses A.1

```
> 3^2-5*9*(25-18)
[1] -306
> 1:8
[1] 1 2 3 4 5 6 7 8
> letters[1:8]
[1] "a" "b" "c" "d" "e" "f" "g" "h"
> value<-8*5^2 #Attribuer une opération à l'objet value
> value #Visualiser l'objet "value"
[1] 200
> p<-seq(from=5, to=15, by=2) #sequence depuis 5 jusqu'à 15 de 2 en 2
> print (p)
> rain<-c(200, 210.2, 490, 100.5, 150.1, 190, 310, 200.2) #Emploi de la fonction c(combiner)
> rain #Visualiser l'objet "rain"
[1] 200 210.2 490 100.5 150.1 190 310 200.2
> rain[5] #Obtenir le 5eme element de l'objet rain
[1] 150.1
> elev<-c(3200, 3500, 4500, 3050, 3100, 2800, 3800, 3500) # Vecteur des elevations
> elev #Visualiser l'objet "elev"
[1] 3200 3500 4500 3050 3100 2800 3800 3500
```

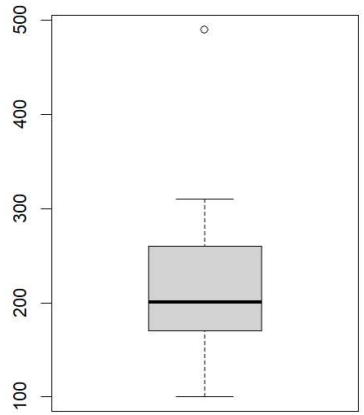
Exercice A.2

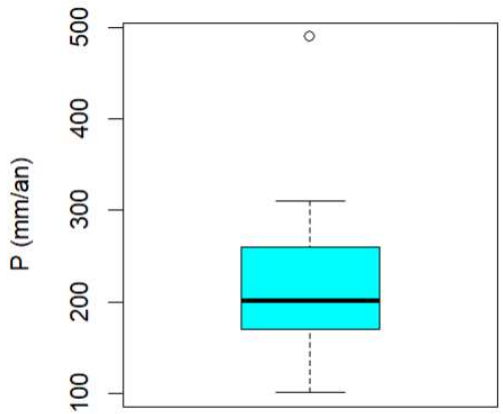
Dans la **console de résultats**, effectuer les opérations suivants:

1) Obtenir la moyenne, médiane, l'écart-type, variance de l'objet *rain* (de l'exercice A.1).
2) Visualiser un résumé des principaux statistiques.
3) Tracer une boîte à moustaches avec les valeurs de l'objet *rain* et interpréter.
4) Ajouter au dernier graphique une couleur de remplissage «cyan» avec une étiquette dans l'axe verticale "P (mm/an)"

Réponses A.2

```
> mean(rain)
[1] 231.375
> median(rain)
[1] 200.1
> sd(rain)
[1] 120.0709
> var(rain)
[1] 14417.03
> summary(rain)
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
  100.5   180.0   200.1   231.4   235.2   490.0
> boxplot(rain)
> boxplot(rain, ylab="P (mm/año)", col="cyan")  #Boite à moustaches avec l'etiquette dans la verticale
```





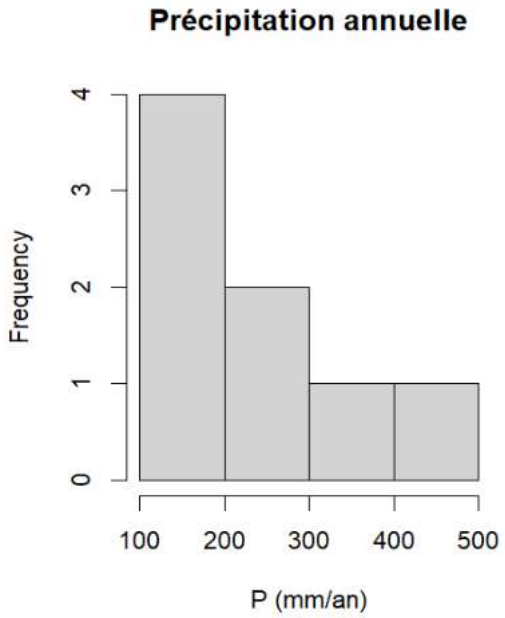
Exercices A.3

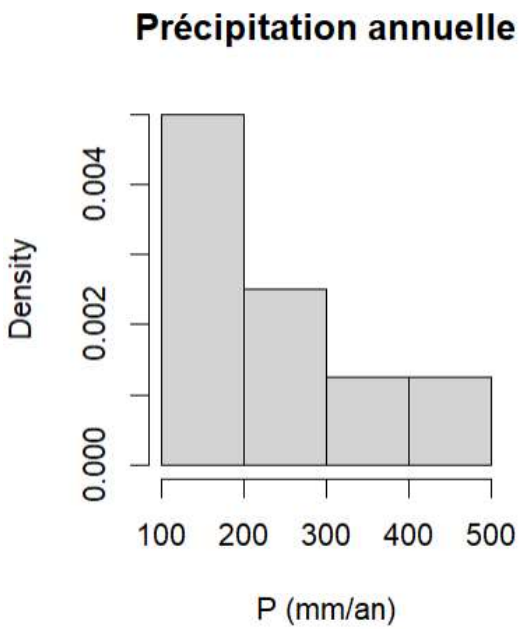
Dans la **console de résultats**, efectuar les opérations suivantes:

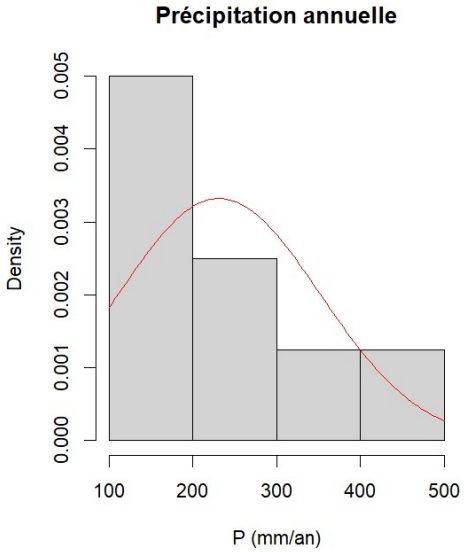
1) Tracer une histogramme de fréquences pour l'objet *rain*, utilisez la fréquence.
2) Tracer une histogramme de fréquences pour l'objet *rain*, utilisez la densité.
3) Ajoutez une courbe de distribution normale dans le graphique précèdent avec un couleur rouge avec la moyenne et l'écart-tyoe estimées précédemment.
4) Ajoutez dans un seule graphique (1filas x 2 colonnes), la boîte à moustache et l'histogramme.
5) Séparez les graphique précédents.
6) Calculez la covariance entre la pluie et l'élévation.
7) Calculez le coefficient de corrélation entre la pluie et l'élévation.
8) Calculez l'équation de régression linéaire entre la pluie et l'élévation.
9) Tracer une nuage de points entre la pluie et l'élévation.
10) Ajoutez une ligne de tendance dans le graphique précèdent.

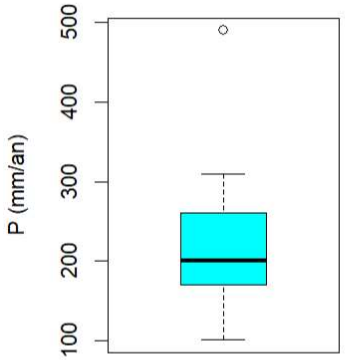
Réponses A.3

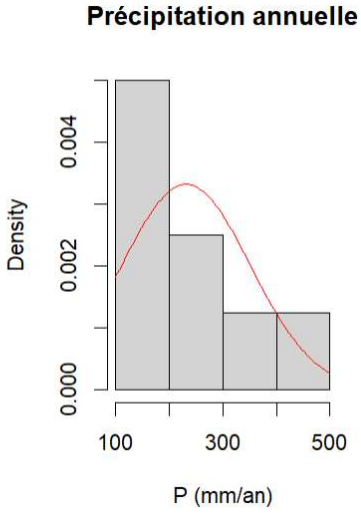
```
> hist(rain, main="Précipitation annuelle", xlab="P (mm/an)", freq=F) # Histogramme, F(False) pour n'est pas utiliser la fréquence
> curve(dnorm(x, mean(rain, na.rm = T), sd(rain, na.rm = T)), add = TRUE, col="red") # Ajouter une courbe de distribution normale en rouge
>split.screen(c(1,2)) #Fonction pour créer une espace des figures, e.g. 2 figures dans 1 ligne et  2 colonnes
>screen(2) #Utiliser cette fonction pour préciser la position avant de tracer la figure
>close.screen(all=TRUE) # fonction pour retourner aux conditions initiales d'une figure par fenêtre
> cov(rain,elev) #covariance
[1] 58855.89
> cor(rain,elev) #coefficient de corrélation r
[1] 0.9182557
> lm(elev ~ rain) #équation de régression linéaire
Call:
lm(elev ~ rain)
Coefficients:
(Intercept)        rain
   2486.688       4.082
> plot(rain,elev,main="Elevation-Rainfall relationship",xlab="Rainfall (mm/yr)", ylab="Elevation (masl)")
> abline(lm(elev~rain))
```

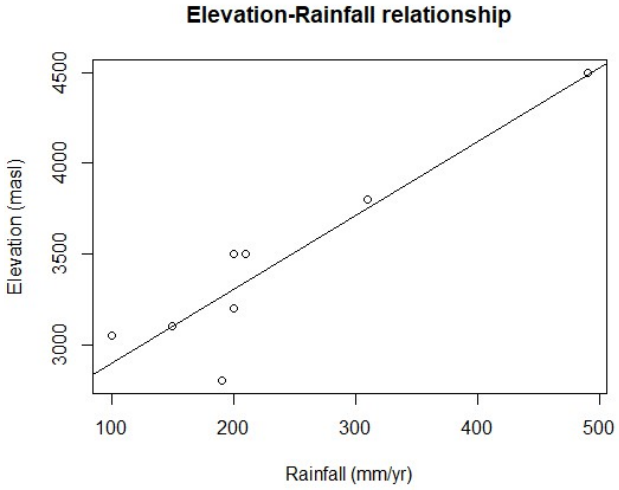

Précipitation annuelle
Frequency
0
1
2
3
4
100
200
300
400
500
P (mm/an)


Précipitation annuelle
Density
0.000
0.002
0.004
100
200
300
400
500
P (mm/an)


Précipitation annuelle
Density
0.000
0.001
0.002
0.003
0.004
0.005
100
200
300
400
500
P (mm/an)


P (mm/an)
100
200
300
400
500


Précipitation annuelle
Density
0.000
0.002
0.004
100
300
500
P (mm/an)


Elevation-Rainfall relationship
Elevation (masl)
3000
3500
4000
4500
100
200
300
400
500
Rainfall (mm/yr)

3. Analyse exploratoire et traitement de données

read.table(file, header = FALSE, sep = "", quote = "\"'", dec = ".", row.names, col.names, as.is = FALSE, na.strings = "NA", colClasses = NA, nrows = -1, skip = 0, check.names = TRUE, fill = !blank.lines.skip, strip.white = FALSE, blank.lines.skip = TRUE, comment.char = "#")

read.csv(file, header = TRUE, sep = ",", quote="\"", dec=".", fill = TRUE, ...)

Lecture de fichiers, base de données

`file`	the name of the file (within `""` or a variable of mode character), possibly with its path (the symbol \ is not allowed and must be replaced by /, even under Windows), or a remote access to a file of type URL (http://...)
`header`	a logical (`FALSE` or `TRUE`) indicating if the file contains the names of the variables on its first line
`sep`	the field separator used in the file, for instance `sep="\t"` if it is a tabulation
`quote`	the characters used to cite the variables of mode character
`dec`	the character used for the decimal point
`row.names`	a vector with the names of the lines which can be either a vector of mode character, or the number (or the name) of a variable of the file (by default: `1`, `2`, `3`, ...)
`col.names`	a vector with the names of the variables (by default: `V1`, `V2`, `V3`, ...)
`as.is`	controls the conversion of character variables as factors (if `FALSE`) or keeps them as characters (`TRUE`); `as.is` can be a logical, numeric or character vector specifying the variables to be kept as character
`na.strings`	the value given to missing data (converted as `NA`)
`colClasses`	a vector of mode character giving the classes to attribute to the columns
`nrows`	the maximum number of lines to read (negative values are ignored)
`skip`	the number of lines to be skipped before reading the data
`check.names`	if `TRUE`, checks that the variable names are valid for R
`fill`	if `TRUE` and all lines do not have the same number of variables, "blanks" are added
`strip.white`	(conditional to `sep`) if `TRUE`, deletes extra spaces before and after the character variables
`blank.lines.skip`	if `TRUE`, ignores "blank" lines
`comment.char`	a character defining comments in the data file, the rest of the line after this character is ignored (to disable this argument, use `comment.char = ""`)

Paradis (2002)

Configurer des Séries temporelles

ts(data = NA, start = 1, end = numeric(0), frequency = 1, deltat = 1, ts.eps = getOption("ts.eps"), class, names)

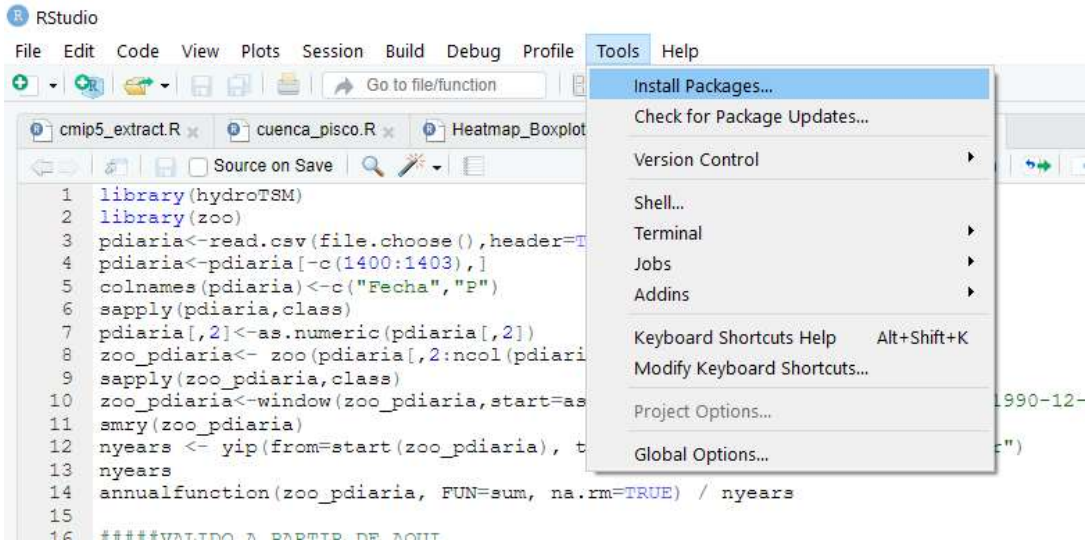
data	a vector or a matrix
start	the time of the rst observation, either a number, or a vector of two integers (see the examples below)
end	the time of the last observation specied in the same way than start
frequency	the number of observations per time unit
deltat	the fraction of the sampling period between successive observations (ex. 1/12 for monthly data); only one of frequency or deltat must be given
ts.eps	tolerance for the comparison of series. The frequencies are considered equal if their dierence is less than ts.eps
class	class to give to the object; the default is "ts" for a single series, and c("mts", "ts") for a multivariate series
names	a vector of mode character with the names of the individual series in the case of a multivariate series; by default the names of the columns of data, or Series 1, Series 2, . . .

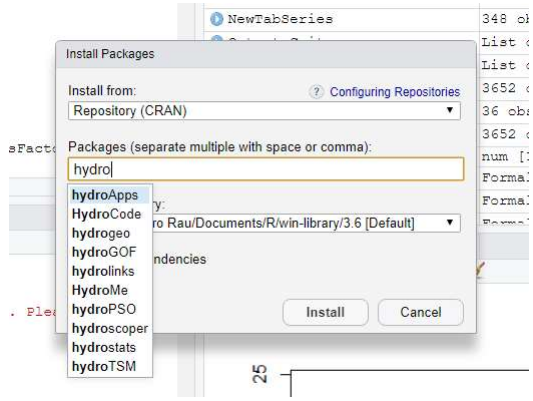
Ex: ts(donnees, start=c(1964, 1), end=c(2007, 12), frequency=12)

B. Explorer dans les données de précipitations mensuelles stockées dans un fichier d'une station

Utilisation des packages: lattice, zoo, hydroTSM

- Installer les packages avec: TOOLS - INSTALL PACKAGES: lattice, zoo, hydroTSM
- Télécharger le fichier "pmensual.txt"
  https://github.com/hydrocodes/Series_hidro_R/blob/master/Tutorial_files/pmensual.txt
- Révision du fichier de type texte (generé depuis MS Excel)





Exercice B

Dans la console de codes ou scripts, creer un code qui fait le suivant:

1) Lecture du fichier de précipitations mensuelles "pmensuelle.txt" stockés en format table avec 13 colonnes (année mois1 mois2 …. mois12) y 45 lignes (nom valeur1 valeur2 …).
2) Tracer tout la série temporelle.
3) Ajouter une courbe de tendance du type lowess.
4) Tracer une carte de chaleur ou "heatmap"

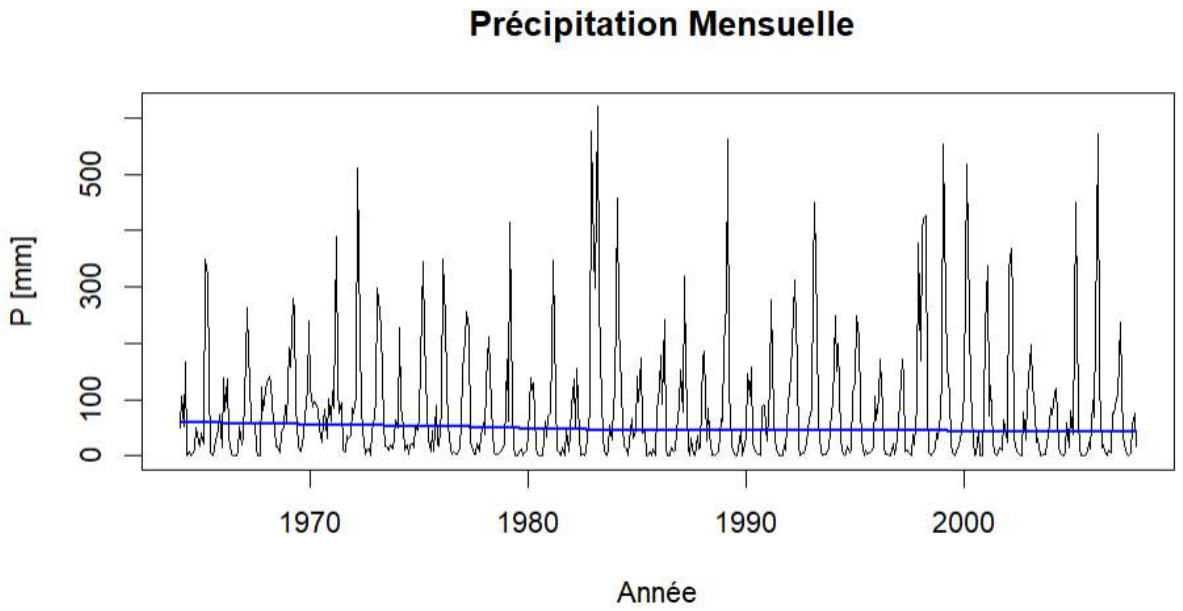
Sauver le script avec le nom: Analyse_mensuelle.R

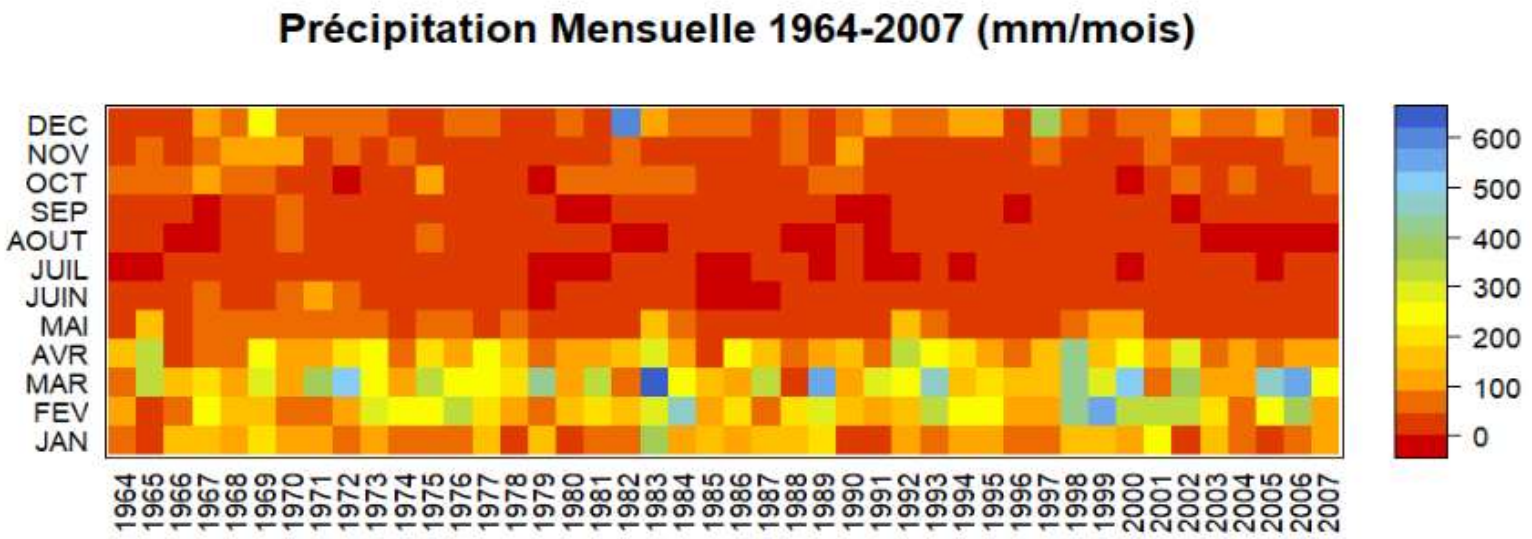
Réponses B

```
library("lattice") #Appeler le package lattice
library("hydroTSM") # Appeler le package hydroTSM
pmensual<-read.table(file.choose(), header = F, check.names = F) #lecture du fichier avec l'option fenêtre en saillie, chercher la route du fichier et sélectionner. « Header=F" dans le cas où l'entêté ne contient pas de données sinon que de noms

#Visualiser la serie temporelle
datos<-pmensual[2:45,2:13]
datos_vector<-as.vector(t(datos)) #convertir en vecteur
datos_ts<-ts(datos_vector, start=c(1964, 1), end=c(2007, 12), frequency=12)
plot.ts(datos_ts, col="black", main=" Précipitation Mensuelle", ylab="P [mm]", xlab="Année")
lines(lowess(time(datos_ts), datos_ts), col="blue", lwd=2) #ajouter la courbe de tendance

#Carte de chaleur ou HeatMap:
lluvia<-pmensual[2:45,2:13] #lecture de données de pluies sans etiquettes et mois (depuis 1964 à 2007)
meses<-pmensual[1:1,2:13] #lecture de l'entêté des mois dans la premiere ligne
colnames(lluvia)<-unlist(meses) #desagregation des noms de mois et attribuer à la matrice
rownames(lluvia)<-pmensual[2:45,1:1] #désagrégation des dates dans le lignes et attribuer à la matrice
matrixplot(lluvia, ColorRamp="Precipitation", main="Précipitation Mensuelle 1964-2007 (mm/mois)")
```


Précipitation Mensuelle
P [mm]
0
100
300
500
1970
1980
1990
2000
Année


Précipitation Mensuelle 1964-2007 (mm/mois)
DEC
NOV
OCT
SEP
AOUT
JUIL
JUIN
MAI
AVR
MAR
FEV
JAN
1964 1965 1966 1967 1968 1969 1970 1971 1972 1973 1974 1975 1976 1977 1978 1979 1980 1981 1982 1983 1984 1985 1986 1987 1988 1989 1990 1991 1992 1993 1994 1995 1996 1997 1998 1999 2000 2001 2002 2003 2004 2005 2006 2007
600
500
400
300
200
100
0

Références

Aybar C, Fernández C, Huerta A, Lavado W, Vega F, Felipe-Obando O. 2020. Construction of a high-resolution gridded rainfall dataset for Peru from 1981 to the present day. Hydrol. Sci. J. 65, 770–785.

Ihaka R. & Gentleman R. 1996. R: a language for data analysis and graphics. Journal of Computational and Graphical Statistics 5: 299-314.

Paradis E. 2002. R for beginners. Institut de Sciences de l'evolution. Université de Montpellier. France

Rau P, Bourrel L, Labat D, Melo P, Dewitte B, Frappart F, Lavado W, Felipe O, 2017. Regionalization of rainfall over the Peruvian Pacific slope and coast. International Journal of Climatology 37(1):143-158.

RStudio Team (2015). RStudio: Integrated Development for R. RStudio, Inc., Boston, MA

Slater L, Thirel G, Harrigan S et al. 2019. Using R in hydrology: a review of recent developments and future directions. Hydrol. Earth Syst. Sci., 23, 2939–2963