Time series: Lab 1

Comments on R, RStudio versions and R notebook

- I'm working on RStudio Version 1.0.136 with a R version 3.3.1.
- This is an R Markdown document. When you execute code within the notebook, the results appear beneath the code.
- You can execute the chunks by clicking the *Run* button within the chunk or by placing your cursor inside it and pressing *Cmd+Shift+Enter*.
- Add a new chunk by clicking the *Insert Chunk* button on the toolbar or by pressing *Cmd+Option+I*.
- When you click the **Knit** button a html document will be generated that includes both content as well as the output of any embedded R code chunks within the document.
- Check here http://rmarkdown.rstudio.com/authoring_basics.html for authoring basics.
- If **you experience trouble with the encoding**: go the *file* menu and choose *reopen with encoding*, then choose UFT-8.

Comments on the Lab

- There are 6 exercises:
  - Exercise 0 Basics on how to deal with time series in R. **Do not include** this exercise to your submission...
  - Exercise 1 Simulations in the T+S+X model and decomposition
  - Exercise 2 Simulations in different, computation of AFCs and PACFs, which a first step towards model selection
  - Exercise 3 AR(2) processes and their ACFs
  - Exercise 4 Prediction in an AR(1) model
  - Exercise 5 Prediction in a MA model via the innovations algorithm
- You **can** send me your lab results (Exercise 1 to 5) at the end of this lab at agathe.guilloux@math.cnrs.fr,
  - this will **add up to 2 points on your final grade**.
  - **Caution: I'll only accept a .pdf file** (which results from clicking the *Knit* button) **not a .Rmd file**
- You'll need the following R packages: `ggfortify`, `astsa`, `forecast`.

Exercice 0: basic R for time series

We'll work with the "AirPassengers" dataset (available in basic R distribution).

0. Load the R packages

```
knitr::opts_chunk$set(fig.height=5, fig.width=7)
library(ggfortify, quietly = TRUE) # for nice plots
```

```
## Warning: package 'ggplot2' was built under R version 3.3.2
```

```
library(astsa, quietly = TRUE) # for some of the data
```

```
## Warning: package 'astsa' was built under R version 3.3.2
```

```r
library(forecast, quietly = TRUE) # time series R package
```

```
##
## Attaching package: 'zoo'

## The following objects are masked from 'package:base':
##
##     as.Date, as.Date.numeric

## This is forecast 7.3

##
## Attaching package: 'forecast'

## The following object is masked from 'package:astsa':
##
##     gas

## The following object is masked from 'package:ggfortify':
##
##     gglagplot
```

1. Load the data, get a description

```r
data(AirPassengers) # load the data
?AirPassengers # get a description
print(AirPassengers)
```

```
##      Jan Feb Mar Apr May Jun Jul Aug Sep Oct Nov Dec
## 1949 112 118 132 129 121 135 148 148 136 119 104 118
## 1950 115 126 141 135 125 149 170 170 158 133 114 140
## 1951 145 150 178 163 172 178 199 199 184 162 146 166
## 1952 171 180 193 181 183 218 230 242 209 191 172 194
## 1953 196 196 236 235 229 243 264 272 237 211 180 201
## 1954 204 188 235 227 234 264 302 293 259 229 203 229
## 1955 242 233 267 269 270 315 364 347 312 274 237 278
## 1956 284 277 317 313 318 374 413 405 355 306 271 306
## 1957 315 301 356 348 355 422 465 467 404 347 305 336
## 1958 340 318 362 348 363 435 491 505 404 359 310 337
## 1959 360 342 406 396 420 472 548 559 463 407 362 405
## 1960 417 391 419 461 472 535 622 606 508 461 390 432
```

2. The `ts` type
   - Check the type of object
   - Read the help on `ts` object
   - Methods associated to `ts` object

```r
class(AirPassengers) # check the type of this data
```

```
## [1] "ts"
```

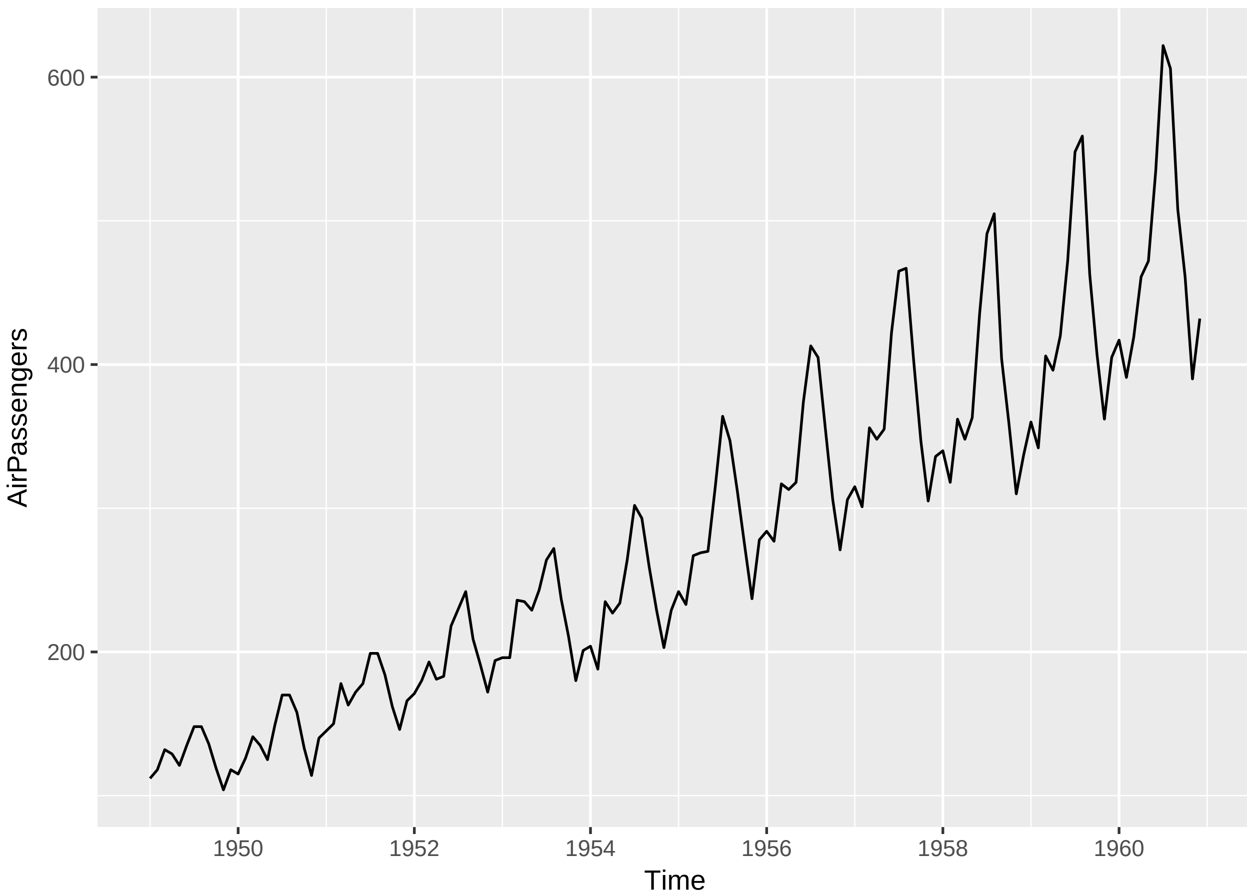
```r
?ts # get help to learn what is a `ts` object
methods(class = `ts`)
```

```
##  [1] [              [<-            aggregate      as.data.frame as.Date
##  [6] as.list        as.zoo         as.zooreg      autoplot      cbind
## [11] coerce         coredata       coredata<-     cycle         diff
## [16] diffinv        forecast       fortify        index         initialize
## [21] is.regular     kernapply      lines          Math          Math2
```

```
## [26] monthplot     na.approx     na.fill       na.omit       na.StructTS
## [31] na.trim       Ops           plot          print         rollapply
## [36] rollmax       rollmean      rollmedian    rollsum       show
## [41] slotsFromS3   subset        t             time          window
## [46] window<-      xblocks
## see '?methods' for accessing help and source code
```

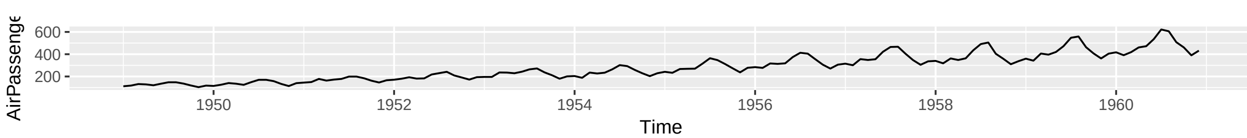
3. Plot the series

```r
autoplot(AirPassengers)
```



If you want to change the size of the graphic

```r
autoplot(AirPassengers)
```



4. What do these commands do ?

```r
start(AirPassengers)
```

```
## [1] 1949    1
```

```r
end(AirPassengers)
```

```
## [1] 1960   12
```

```r
frequency(AirPassengers)
```

```
## [1] 12
```

```r
deltat(AirPassengers)
```

```
## [1] 0.08333333
```

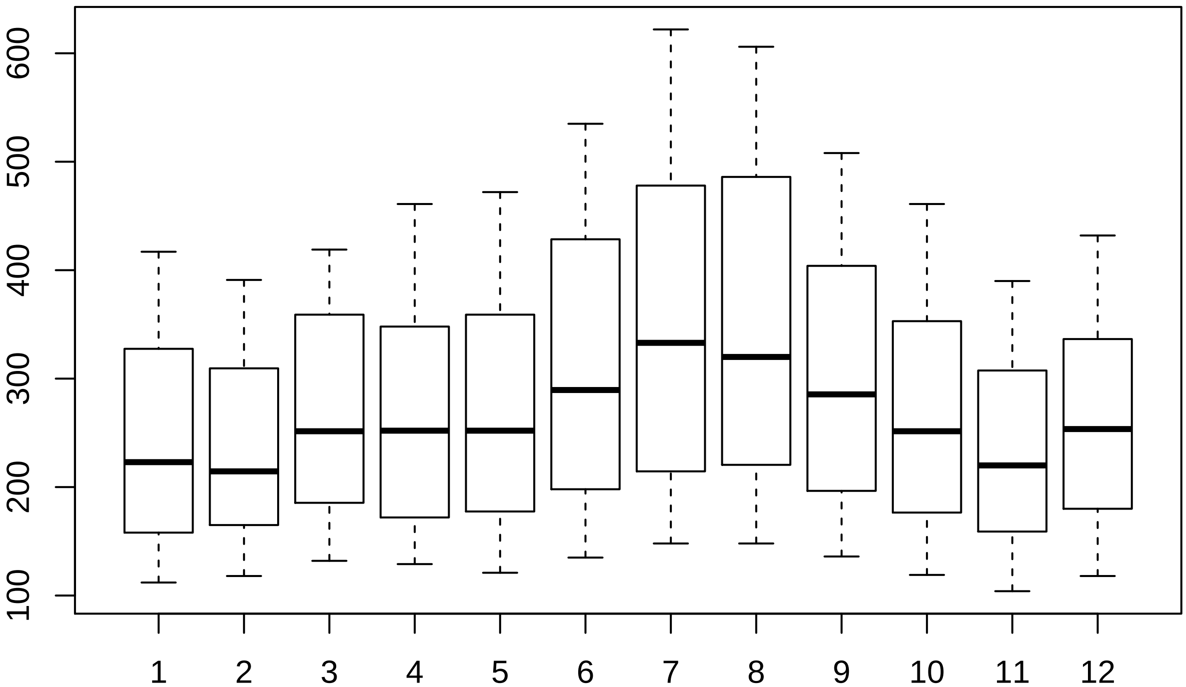
```r
summary(AirPassengers)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   104.0   180.0   265.5   280.3   360.5   622.0
```

```r
cycle(AirPassengers)
```

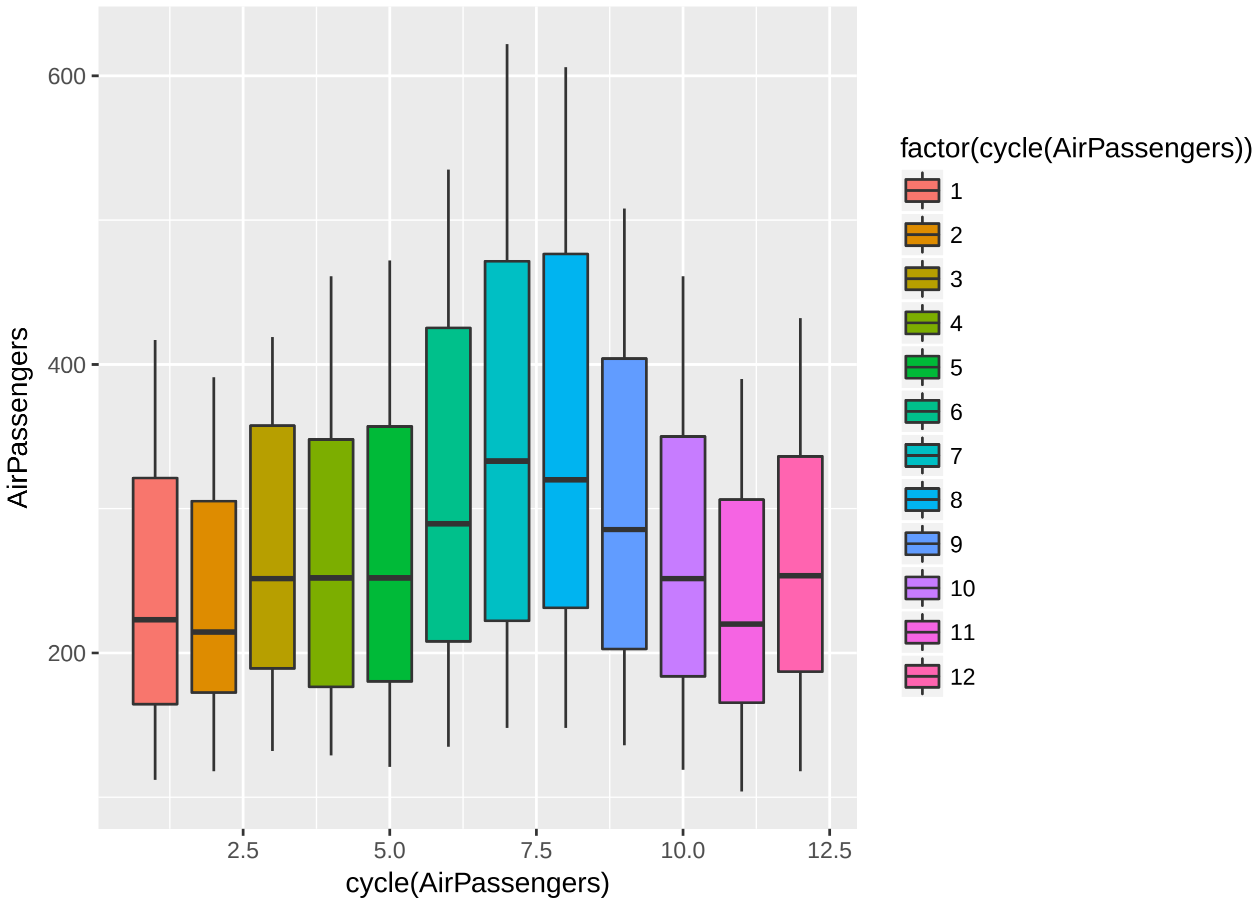
```
##      Jan Feb Mar Apr May Jun Jul Aug Sep Oct Nov Dec
## 1949   1   2   3   4   5   6   7   8   9  10  11  12
## 1950   1   2   3   4   5   6   7   8   9  10  11  12
## 1951   1   2   3   4   5   6   7   8   9  10  11  12
## 1952   1   2   3   4   5   6   7   8   9  10  11  12
## 1953   1   2   3   4   5   6   7   8   9  10  11  12
## 1954   1   2   3   4   5   6   7   8   9  10  11  12
## 1955   1   2   3   4   5   6   7   8   9  10  11  12
## 1956   1   2   3   4   5   6   7   8   9  10  11  12
## 1957   1   2   3   4   5   6   7   8   9  10  11  12
## 1958   1   2   3   4   5   6   7   8   9  10  11  12
## 1959   1   2   3   4   5   6   7   8   9  10  11  12
## 1960   1   2   3   4   5   6   7   8   9  10  11  12
```

```r
boxplot(AirPassengers~cycle(AirPassengers))
```




```r
ggplot(data=AirPassengers, aes(cycle(AirPassengers),AirPassengers)) + geom_boxplot(aes(fill = factor(cy
```

```
## Don't know how to automatically pick scale for object of type ts. Defaulting to continuous.
## Don't know how to automatically pick scale for object of type ts. Defaulting to continuous.
```



Your answers have to be written under the chuck For exemple The first line gives the date of the first observations. etc

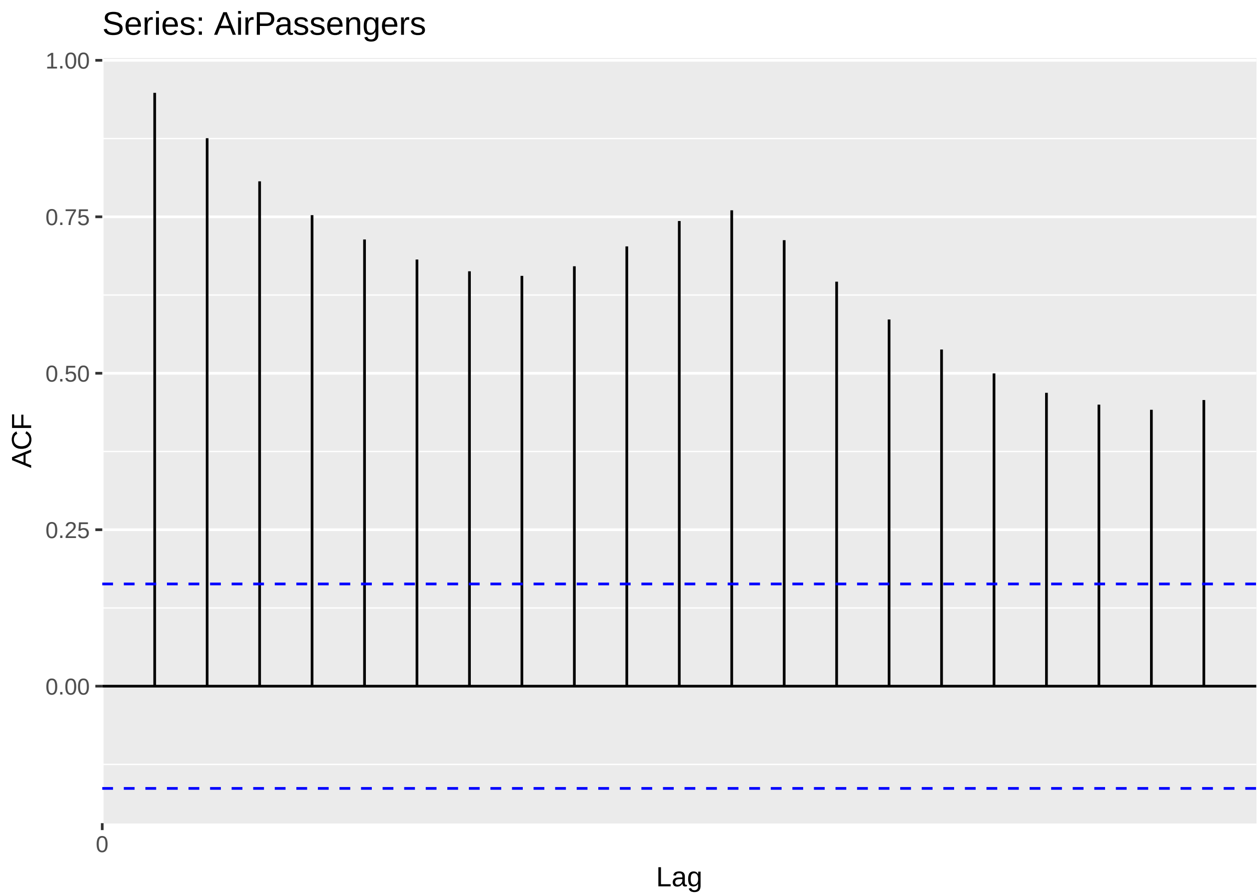
Ici doivent apparaître vos réponses aux questions : sous les chucks. La première commande donne la date de première observation de la série. etc

5. Get the ACF and PACF of the series.

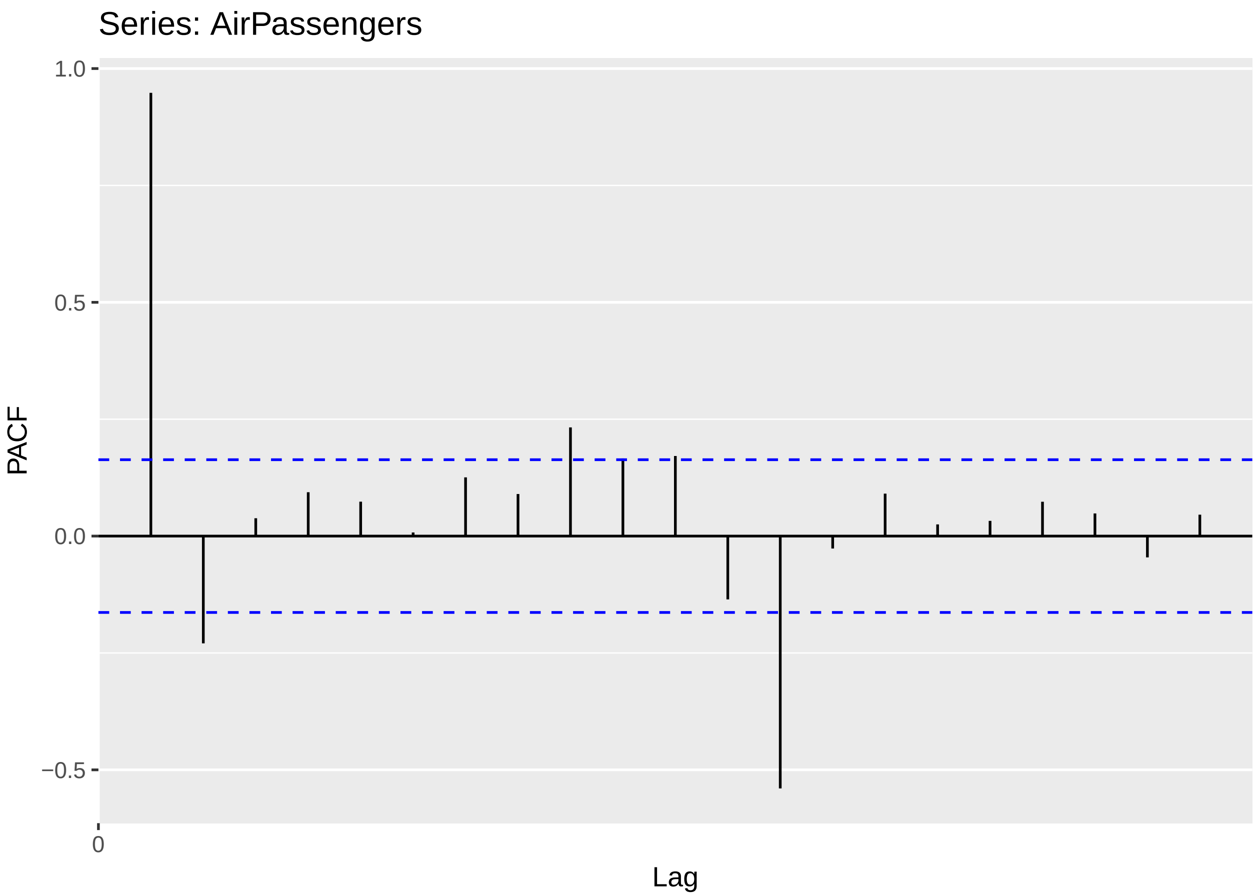
```
acf = acf(AirPassengers,plot=FALSE)

pacf = pacf(AirPassengers,  plot =FALSE)

autoplot(acf)
```




```
autoplot(pacf)
```



6. Creating a `ts` object. Now suppose that you have a raw series `as.numeric(AirPassengers)` and the information of the first time it had been measured and the frequency (`start = c(1949,1)`, `frequency=12`). Can you recreate a `ts` object ?

```
attributes(AirPassengers)
```

```
## $tsp
## [1] 1949.000 1960.917   12.000
##
## $class
## [1] "ts"
```

```
raw_data = as.numeric(AirPassengers)
byhand_airpassengers = ts(raw_data, frequency=12, start=c(1949, 1))
attributes(byhand_airpassengers)
```

```
## $tsp
## [1] 1949.000 1960.917   12.000
##
## $class
## [1] "ts"
```

Exercice 1: Simulation

1. Simulate 1200 realizations of a Gaussian white noise $WN(0, 1)$. Plot the series, denoted by BB, its ACF and PACF

2. Choose and add to BB a polynomial trend T and a seasonality S with amplitudes reasonable compared to the noise variance. Create a time series ST with the `ts` function, assuming that its observation is monthly, and begins on January 1900.
3. Plot the series BB,S,T,TS and compare with the results of `slt` function. Comment
4. What is represented by the `monthplot` function ? What information can you gather from this plot ?

Exercice 2: Times series, ACF, PACF

1. The following code simulates trajectories (of size $n = 500$) of 4 time series T_1, T_2, T_3, T_4. Can you deduce from their ACF and PACF which models have been used ?
2. Now change n to 200 are the conclusions still so clear ?

Exercice 3: AR(2) processes and their ACFs

(from Brockwell and Davis p91 - example 3.2.4)

We consider a stationary AR(2) model

$$X_t = \phi_1 X_{t-1} + \phi_2 X_{t-2} + \omega_t$$

where ω is a Gaussian white noise with variance 1. We denote by ξ_1 and ξ_2 the roots of $1 - \phi_1 z - \phi_2 z^2$. In this case the autocorrelation function verifies

$$\rho(h) = \phi_1 \rho(h-1) + \phi_2 \rho(h-2)$$

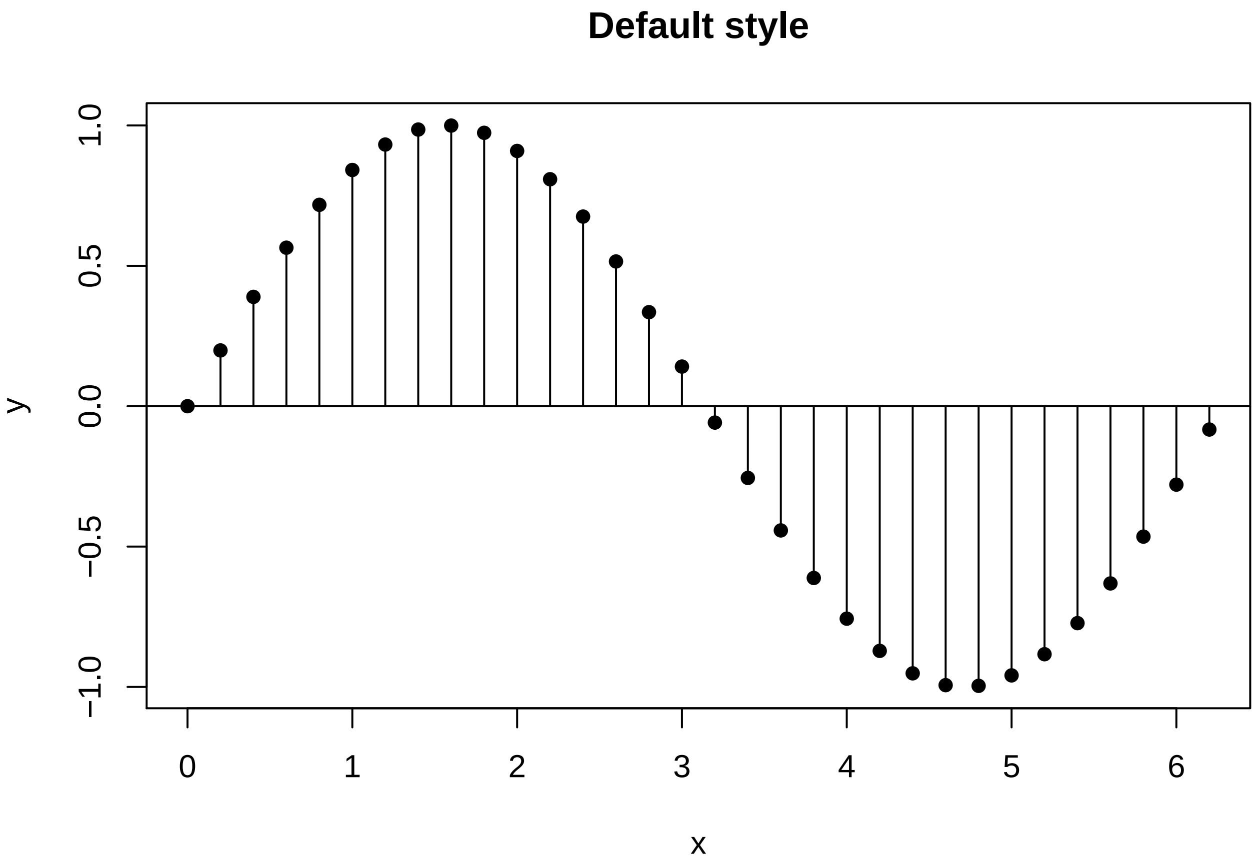
with initial conditions $\rho(0) = 1$ and $\rho(1) = \phi_1/(1-\phi_2)$ and is then given by

$$\rho(h) = \frac{(\xi_2^2 - 1)\xi_1^{1-h} - (\xi_1^2 - 1)\xi_2^{1-h}}{(\xi_1 \xi_2 + 1)(\xi_2 - \xi_1)}$$

when $\xi_1 \neq \xi_2$. We illustrate in this exercise the different behaviors of the ACF ρ.

Here is a code to generate a stem plot that you'll need for plotting the true ACFs

```
stem <- function(x,y,pch=16,linecol=1,clinecol=1,...){
if (missing(y)){
    y = x
    x = 1:length(x) }
    plot(x,y,pch=pch,...)
    for (i in 1:length(x)){
       lines(c(x[i],x[i]), c(0,y[i]),col=linecol)
    }
    lines(c(x[1]-2,x[length(x)]+2), c(0,0),col=clinecol)
}
#An example
x <- seq(0, 2*pi, by = 0.2)
stem(x,sin(x), main = 'Default style')
```



1. Write

- a function `roots2ar` that calculates the coefficients ϕ_1 and ϕ_2 from the roots ξ_1 and ξ_2.
- a function `true_acf` that calculates the acf at point h from ξ_1 and ξ_2

2. Choose coefficients ϕ_1 and ϕ_2 such that the process is non-causal.
   - Try to simulate $n = 50$ observations from the `arima.sim` function. Comment the comment.
   - Simulate $n = 50$ observations from the `filter` function. You need to simulated $n+20$ observations and keep only the last 50 to overcome the initialization problem.
   - Represent the series and its sample acf. What happens ?
3. A causal AR(2) with complex roots
   - Choose two pairs of complex conjugate roots one with moduli near 1 but such that the process is causal the other with moduli farther to 1 .
   - In each case, compute ϕ_1 and ϕ_2 and simulate 500 observations of the process and draw a plot.
   - Plot the true and sample autocorrelation functions and compare the plots.
4. Choose two different pairs of real roots, one pair close to 1 and the second farther.
   - Make plots of simulations, and autocorrelation functions as in question 3.
   - Compare the autocorrelation plots of questions 3 and 4. What is the main qualitative difference?

Exercice 4: Prediction for the `lh` data

We now work with the `lh` dataset.

1. Visualize the data, their ACF and PACF.
2. We choose an AR(1) model
$$X_t = \phi X_{t-1} + \omega_t$$
for this series (cf Lab2 for a justification).
    - Estimate the mean.
    - Propose an estimation of ϕ and σ^2 based of the results of the `acf` and `pacf` functions (you'll need the option `type="correlation"` in the `acf` function).
    - Compare your results with `arima(lh, order = c(1, 0, 0))`. Why, in your opinion, is there a (slight) difference ?

- Using the prediction equations, the fact that, for an AR(1), $\gamma(h) = \phi\gamma(h-1)$, and your estimations of $\gamma(0)$ and ϕ
    - compute the matrix Γ_n (use the `toeplitz` function) and its inverse (function `solve`)
    - and the vectors $\gamma_n^{(m)} = \big(\gamma(m), \dots, \gamma(m+n-1)\big)^\top$ for $m = 1$ to 12
    - Propose a prediction for the next 12 values of the series (don't forget to subtract the mean)
    - Build prediction intervals.
    - Compare your results with the `predict` method applied to the `arima` function.

Exercice 5 : prediction in a MA model via the innovations algorithm

We consider in this exercise the `varve` dataset available in the `astsa` package.

1. Represent the series, its ACF and PACF. Give some argument for considering the log transformation to stationarize the series.
2. Plot the new series, its ACF and PACF and give arguments for the non-stationarity. Argue why a differentiation should stationarize the log of varve.
3. Differentiate the log of varve, plot the new series, its ACF and PACF and explain the choice of an MA(1) model for this new series.
4. Estimations in the MA(1) $X_t = \omega_t + \theta\omega_{t-1}$ model.
    - Estimate θ (you can use the `polyroot` function). Choose the solution that leads to an invertible MA.
    - and $\sigma^2 = \text{Var}(\omega_t)$ (don't forget the option `type = "covariance"` of the `acf` function)
    - Compare your estimations with the result of `arima(lh, order = c(0, 0, 1))`
5. The innovation algorithm for a MA(1) model
    - Code the innovations algorithm and deduce a prediction and a prediction interval for the next value.
    - Compare your results with the `predict` method applied to the `arima` function.
    - Admit that $X^n_{n+m} = \theta_{n+m-1,1}X_n$ where the $\theta_{n+m-1,1}$ are determined by the same algorithm. Deduce predictions and their errors for the next 10 values of the varve series itself.
    - Compare with the result of the `predict` method applied to the `arima` function.