Time Series Analysis: Computer Exercises

Jussi Klemelä

September 4, 2014

1 Reading Data

Let us study the time series of S&P 500 stock index prices. Data can be read to R with commaneds

```
file<-"http://cc.oulu.fi/~jklemela/marketrisk/sp500.csv"
data<-read.csv(file=file)
sp500<-data[,7]                  # we take the adjusted closing price
sp500<-sp500[length(sp500):1]    # time series starts with the oldest obs

time<-data[,1]
time<-time[length(time):1]
```

Data is also available from page

```
http://finance.yahoo.com/q/hp?s=^GSPC+Historical+Prices
```

2 Exercises

1. Plotting and removing trend.

   (a) Plot the time series of prices $\mathrm{SP500}_t$.

   (b) Plot the time series of logarithmic prices $\log(\mathrm{SP500}_t)$.

   (c) Make the transformation

   $$Y_t = \mathrm{SP500}_t - \mathrm{SP500}_{t-1}$$

   and plot Y_t.

(d) Make the transformation

$$Y_t = \log \frac{\text{SP500}_t}{\text{SP500}_{t-1}} = \log(\text{SP500}_t) - \log(\text{SP500}_{t-1})$$

and plot Y_t.

(e) Make the transformation

$$Y_t = \frac{\text{SP500}_t - \text{SP500}_{t-1}}{\text{SP500}_{t-1}} = \frac{\text{SP500}_t}{\text{SP500}_{t-1}} - 1$$

and plot Y_t.

2. Simulating time series

(a) Simulate AR(1) time series

$$X_t = aX_{t-1} + \epsilon_t, \qquad t = 1, \ldots, T,$$

miss $T = 1000$, $X_0 = 0$ and $\epsilon_t \sim N(0, 1)$ are i.i.d.
Try values $0 < a < 1$, $-1 < a < 0$, $a = 1$, $a = -1$, $a = 1.1$ ja $a = -1.1$.
Choose the distribution of ϵ_t to be the t-distribution with degrees of freedom n: $\epsilon_t \sim t(n)$ are i.i.d. with $n = 1, 2, 3, \ldots$.

(b) Simulate MA(1) time series

$$X_t = \epsilon_t + a\epsilon_{t-1}, \qquad t = 1, \ldots, T,$$

where $T = 1000$ and $\epsilon_t \sim N(0, 1)$ are i.i.d.
Try diffrent values of a.

(c) Simulate GARCH(1,1) time series

$$X_t = \sigma_t \epsilon_t, \qquad t = 1, \ldots, T,$$

where

$$\sigma_t^2 = a_0 + a_1 \sigma_{t-1}^2 + bX_{t-1}^2,$$

$a_0, a_1, b \geq 0$, $T = 1000$, $X_0 = 0$ and $\epsilon_t \sim N(0, 1)$ are i.i.d.
Try different values with $a_1 + b < 1$ and $a_1 + b \geq 1$.

3. Autocorrelation

(a) Let the returns be

$$R_t = \frac{\mathrm{SP500}_t - \mathrm{SP500}_{t-1}}{\mathrm{SP500}_{t-1}} = \frac{\mathrm{SP500}_t}{\mathrm{SP500}_{t-1}} - 1.$$

Calculate the autocorrelation function $\rho(k)$, $1 \le k \le 30$.

(b) Calculate the autocorrelation function $\rho(k)$, $1 \le k \le 1000$, for the time series of absolute returns $|R_t|$.

4. Fitting ARMA model

(a) Estimate the parameters of ARMA(p, q) model for the time series

$$R_t = \frac{\mathrm{SP500}_t - \mathrm{SP500}_{t-1}}{\mathrm{SP500}_{t-1}} = \frac{\mathrm{SP500}_t}{\mathrm{SP500}_{t-1}} - 1,$$

when $p = 1$ and $q = 1$.

You can appply package “tseries”, which is made available with the command “library(tseries)”. The package contains functions “arma” and “arima0” for fitting of ARMA-models.

(b) Estimate parameters of ARMA(p, q) model for the times series of squared returns R_t^2, when $p = 20$ ja $q = 0$.

5. Diagnostics and model selection

(a) Let

$$R_t = \frac{\mathrm{SP500}_t - \mathrm{SP500}_{t-1}}{\mathrm{SP500}_{t-1}}.$$

Fit AR(p) model to returns, absolute returns $|R_t|$ or to the squared returns R_t^2. Plot the time series of residuals and the correlogram of the residuals. Try different values of p.

(b) Find the value of p which minimizes the AIC criterion.

6. GARCH-model

(a) Let

$$X_t = \frac{\mathrm{SP500}_t - \mathrm{SP500}_{t-1}}{\mathrm{SP500}_{t-1}}.$$

Fit GARCH$(1, 1)$ and GARCH$(1, 3)$ models to the returns.

(b) Calculate the estimates of the conditional variances

$$\hat{\sigma}_t^2 = \hat{c}_0 + \hat{b} X_{t-1}^2 + \hat{a} \hat{\sigma}_{t-1}^2$$

and plot the time series of the estimates $\hat{\sigma}_t$.

(c) Calculate the residuals

$$\hat{\epsilon}_t = X_t/\hat{\sigma}_t$$

and plot the time series.

3 Answers to Exercises

3.1 Exercise 1

Plotting and transforming the time series.

```
# a) plot time series
plot(sp500,type="l")

# include time
times<-matrix(0,length(sp500),1)
delta<-1/251.4; beg<-1950
for (i in 1:length(times)) times[i]<-beg+(i-1)*delta
plot(times,sp500,type="l")

# use time series object
as.character(time[1])
as.character(time[length(time)])
#[1] 1950-01-03
#[1] 2014-08-28

start<-c(1950,03)
end<-c(2014,8)
sp500.ts<-ts(sp500,start=start,end=end,frequency=250)
plot(sp500.ts)

# b) plot log(sp500)
plot(log(sp500),type="l")

# c) plot differences
y<-sp500[2:length(sp500)]-sp500[1:(length(sp500)-1)]
plot(y,type="l")

# d) plot log-returns
logr<-log(sp500[2:length(sp500)]/sp500[1:(length(sp500)-1)])
plot(logr,type="l")

# e) plot the net returns
r<-y/sp500[1:(length(sp500)-1)]
plot(r,type="l")
```

3.2 Exercise 2

Simulate AR(1) model.

```
a<-0.5
x0<-0
T<-1001
seed<-2
set.seed(seed)

# normal distribution
e<-rnorm(T)
x<-matrix(x0,T,1)
for (t in 2:T) x[t]<-a*x[t-1]+e[t]
plot(x,type="l")

# t-distribution
df<-10
kh<-sqrt(df/(df-2))
e<-rt(T,df=df)/kh
x<-matrix(x0,T,1)
for (t in 2:T) x[t]<-a*x[t-1]+e[t]
plot(x,type="l")
```

Simualate MA(1) model

```
# MA(1)-malli
a1<-0.8
T<-1001
seed<-2
set.seed(seed)
e<-rnorm(T)
x<-matrix(0,T-1,1)
for (t in 2:T) x[t]<-e[t]+a1*e[t-1]
plot(x,type="l")

# MA(2)-malli
a1<--1
a2<--1
T<-1001
seed<-2
set.seed(seed)
```

```
e<-rnorm(T)
x<-matrix(0,T-2,1)
for (t in 3:T) x[t]<-e[t]+a1*e[t-1]+a2*e[t-2]
plot(x,type="l")
```

Simulate GARCH(1,1) model.

```
T<-1000
seed<-3
set.seed(seed)
e<-rnorm(T)
# df<-10; e<-rt(T,df=df)

a0<-1
a1<-0.6
b1<-0.1

x<-matrix(0,T,1)
sigma2<-matrix(0,T,1)
for (t in 2:T){
    sigma2[t]<-a0+a1*sigma2[t-1]+b1*x[t-1]^2
    x[t]<-sqrt(sigma2[t])*e[t]
}
plot(x,type="l")
```

3.3 Exercise 3

Correlation function of returns and absolute returns.

```
# plot returns
y<-sp500[2:length(sp500)]-sp500[1:(length(sp500)-1)]
r<-y/sp500[1:(length(sp500)-1)]
plot(r,type="l")

n<-length(r)
k<-300
cors<-matrix(0,k,1)
for (i in 1:k){
   x<-r[1:(n-i)]
   y<-r[(1+i):n]
   cors[i]<-cor(x,y)
```

```
}
a<-n^(-1/2)*1.96  # qnorm(0.975)
plot(cors,type="l")
segments(0,a,k,a)
segments(0,-a,k,-a)

sum(abs(cors)<=a)/k
cors

# acf ja pacf ########################

acf(r)

pacf(r)

# autocorrelation of absolute returns #################

n<-length(r)
k<-1000
cors<-matrix(0,k,1)
for (i in 1:k){
   x<-abs(r[1:(n-i)])
   y<-abs(r[(1+i):n])
   cors[i]<-cor(x,y)
}
plot(cors,type="l")
a<-n^(-1/2)*1.96  # qnorm(0.975)
segments(0,a,k,a)
segments(0,-a,k,-a)

sum(abs(cors)<=a)/k

# acf ja pacf ########################

acf(abs(r),lag.max=1000)

pacf(abs(r),lag.max=100)
```

3.4 Exercise 4

Fit ARMA(p, q) model to S&P 500 returns.

```
# plot returns
y<-sp500[2:length(sp500)]-sp500[1:(length(sp500)-1)]
r<-y/sp500[1:(length(sp500)-1)]
plot(r,type="l")

library(tseries)

# estimate ARMA(1,1) model  #####################

order<-c(1,1)
ar<-arma(r,order=order)
ar
summary(ar)

#################################
Call:
arma(x = r, order = order)

Coefficient(s):
       ar1          ma1   intercept
-0.4651911   0.5037691   0.0004926

# summary ########################

Call:
arma(x = r, order = order)

Model:
ARMA(1,1)

Residuals:
       Min         1Q     Median         3Q        Max
-0.2030770 -0.0044682  0.0001331  0.0046357  0.1143614

Coefficient(s):
            Estimate  Std. Error  t value Pr(>|t|)
ar1       -0.4651911   0.0825093   -5.638 1.72e-08 ***
ma1        0.5037691   0.0805997    6.250 4.10e-10 ***
intercept  0.0004926   0.0001187    4.148 3.35e-05 ***
---
Signif. codes:  0 *** 0.001 ** 0.01 * 0.05 . 0.1   1
```

```
Fit:
sigma^2 estimated as 9.45e-05,  Conditional Sum-of-Squares = 1.52,  AIC = -103161.1


######  arima0-function  ##############################

order<-c(1,0,1)
ari<-arima0(r,order=order)
ari

Call:
arima0(x = r, order = order)

Coefficients:
          ar1     ma1  intercept
      -0.4892  0.5273      3e-04
s.e.   0.0072  0.1103      1e-04

sigma^2 estimated as 9.449e-05:  log likelihood = 51584.44,  aic = -103160.9

#### AR(p)-model for squared returns #################

order<-c(10,0)
ar<-arma(r^2,order=order)
ar
summary(ar)

Call:
arma(x = r^2, order = order)

Model:
ARMA(10,0)

Residuals:
       Min         1Q     Median         3Q        Max
-2.754e-03 -6.229e-05 -4.120e-05  1.113e-07  4.143e-02

Coefficient(s):
            Estimate  Std. Error  t value Pr(>|t|)
ar1        9.103e-02   7.894e-03   11.532  < 2e-16 ***
```

```
ar2        1.973e-01   7.911e-03   24.943  < 2e-16 ***
ar3        2.068e-02   8.058e-03    2.567   0.0103 *
ar4        5.542e-03   8.059e-03    0.688   0.4917
ar5        1.480e-01   8.055e-03   18.370  < 2e-16 ***
ar6        3.558e-02   8.055e-03    4.417 1.00e-05 ***
ar7       -3.907e-03   8.059e-03   -0.485   0.6278
ar8        3.567e-02   8.058e-03    4.427 9.55e-06 ***
ar9        6.257e-02   7.911e-03    7.910 2.66e-15 ***
ar10       1.216e-02   7.894e-03    1.541   0.1233
intercept  3.752e-05   3.655e-06   10.264  < 2e-16 ***
---
Signif. codes:  0 *** 0.001 ** 0.01 * 0.05 . 0.1   1

Fit:
sigma^2 estimated as 1.829e-07,  Conditional Sum-of-Squares = 0,  AIC = -203400.5

#################
order<-c(5,0,0)
ari<-arima0(r^2,order=order)
ari

Call:
arima0(x = r^2, order = order)

Coefficients:
         ar1     ar2     ar3     ar4     ar5  intercept
      0.1034  0.2036  0.0314  0.0263  0.1596      1e-04
s.e.  0.0078  0.0078  0.0080  0.0078  0.0078      0e+00

sigma^2 estimated as 1.843e-07:  log likelihood = 101647.5,  aic = -203281

ari$aic

[1] -203281

order<-c(5,0,0)
arima0(abs(r),order=order)
```

3.5 Exercise 5

Fit AR(p) model to absolute returns.

```
# plot returns
y<-sp500[2:length(sp500)]-sp500[1:(length(sp500)-1)]
r<-y/sp500[1:(length(sp500)-1)]
plot(r,type="l")

plot(abs(r),type="l")

library(tseries)

# fit AR-model  ->  arma

p<-20
order<-c(p,0,0)
ari<-arma(abs(r),order=order)
ari
summary(ari)

Call:
arma(x = abs(r), order = order)

Coefficient(s):
      ar1        ar2        ar3        ar4        ar5        ar6        ar7
 0.060794   0.100058   0.064395   0.055392   0.109019   0.048397   0.044250
      ar8        ar9       ar10       ar11       ar12       ar13       ar14
 0.036990   0.038166   0.030609   0.052837   0.027897   0.015074   0.002718
     ar15       ar16       ar17       ar18       ar19       ar20  intercept
 0.020151   0.021892   0.020400   0.036885   0.015948   0.022011   0.001161


Call:
arma(x = abs(r), order = order)

Model:
ARMA(20,0)

Residuals:
      Min         1Q    Median         3Q        Max
-0.040189 -0.003684 -0.001267  0.002467  0.190536

Coefficient(s):
           Estimate  Std. Error  t value Pr(>|t|)
```

```
ar1       0.0607937   0.0078922    7.703 1.33e-14 ***
ar2       0.1000583   0.0079068   12.655  < 2e-16 ***
ar3       0.0643954   0.0079410    8.109 4.44e-16 ***
ar4       0.0553917   0.0079557    6.963 3.34e-12 ***
ar5       0.1090188   0.0079659   13.686  < 2e-16 ***
ar6       0.0483974   0.0080105    6.042 1.52e-09 ***
ar7       0.0442496   0.0080195    5.518 3.43e-08 ***
ar8       0.0369903   0.0080264    4.609 4.05e-06 ***
ar9       0.0381662   0.0080284    4.754 2.00e-06 ***
ar10      0.0306089   0.0080231    3.815 0.000136 ***
ar11      0.0528370   0.0080231    6.586 4.53e-11 ***
ar12      0.0278970   0.0080282    3.475 0.000511 ***
ar13      0.0150737   0.0080260    1.878 0.060366 .
ar14      0.0027177   0.0080180    0.339 0.734647
ar15      0.0201508   0.0080088    2.516 0.011867 *
ar16      0.0218917   0.0079643    2.749 0.005983 **
ar17      0.0204001   0.0079543    2.565 0.010328 *
ar18      0.0368855   0.0079398    4.646 3.39e-06 ***
ar19      0.0159482   0.0079061    2.017 0.043674 *
ar20      0.0220106   0.0078922    2.789 0.005289 **
intercept 0.0011609   0.0001012   11.467  < 2e-16 ***
---
Signif. codes:  0 *** 0.001 ** 0.01 * 0.05 . 0.1   1

Fit:
sigma^2 estimated as 4.114e-05,  Conditional Sum-of-Squares = 0.66,  AIC = -116470.

# residuals ################################

resi<-residuals(ari)
plot(resi)

acf(resi,na.action=na.pass)

# AIC  ->   arima0 ###############################
# Warning: calculation takes a long time

p<-30
aics<-matrix(0,p,1)
for (i in 1:p){
  order<-c(i,0,0)
```

```
  aics[i]<-arima0(r,order=order)$aic
}
plot(aics)

aics<-c(
-103145.1,
-103174.5,
-103172.5,
-103171.3,
-103173.5,
-103172.1,
-103177.2,
-103176.7,
-103176.1,
-103176.6,
-103178.5,
-103189.1,
-103187.1,
-103185.1,
-103185.1,
-103200.1,
-103199.0,
-103204.3,
-103202.4,
-103201.3,
-103207.2,
-103205.2,
-103203.3,
-103202.2,
-103202.4,
-103207.7,
-103211.9,
-103210.0,
-103217.6,
-103216.5
)
plot(aics,type="b")

p<-30
aics<-matrix(0,p,1)
for (i in 1:p){
```

```
  order<-c(i,0,0)
  aics[i]<-arima0(abs(r),order=order)$aic
}
plot(aics)
aics

aics<-c(
-113887.8,
-114749.5,
-115170.9,
-115444.7,
-115876.5,
-116013.9,
-116117.5,
-116188.3,
-116253.2,
-116295.5,
-116371.9,
-116398.2,
-116409.6,
-116411.0,
-116424.4,
-116436.7,
-116445.6,
-116469.6,
-116472.4,
-116478.0,
-116482.8,
-116481.0,
-116488.0,
-116494.3,
-116494.0,
-116492.2,
-116522.7,
-116523.7,
-116532.8
)
plot(aics)

 # Sum of squared prediction errors ####################
```

```
p<-10
sspe<-matrix(0,p,1)
t0<-10000
T<-length(r)
for (i in 1:p){
  order<-c(i,0)
  errs<-matrix(0,T-t0,1)
  for (t in t0:(T-1)){
     x<-r[1:t]
     coef<-arma(x,order=order)$coef
     rhat<-sum(as.vector(coef[1:i])*r[t:(t-i+1)])
     errs[t-t0+1]<-r[t+1]-rhat
  }
  sspe[i]<-sqrt(sum(errs^2))
}
plot(sspe)

sspe<-c(
0.9058649,
0.9058457,
0.9060230,
0.9061900,
0.9063012,
0.9066239,
0.9064767,
0.9065603,
0.9068165,
0.9068918
)
plot(sspe,type="b")
```

3.6 Exercise 6

Fit GARCH(1,1) model to SP500 returns, plot $\hat{sigma}_t$ and residuals.

```
# plot returns
y<-sp500[2:length(sp500)]-sp500[1:(length(sp500)-1)]
r<-y/sp500[1:(length(sp500)-1)]
plot(r,type="l")

# Estimate GARCH(1,1) ######################################
```

```
library(tseries)

x<-100*r
order<-c(1,1)
ga<-garch(x,order=order)
summary(ga)

Call:
garch(x = x, order = order)

Model:
GARCH(1,1)

Residuals:
      Min        1Q    Median        3Q       Max
-10.65981  -0.52453   0.06144   0.64030   6.44386

Coefficient(s):
    Estimate  Std. Error  t value Pr(>|t|)
a0 0.0074141   0.0006603    11.23   <2e-16 ***
a1 0.0775141   0.0016973    45.67   <2e-16 ***
b1 0.9164638   0.0022025   416.10   <2e-16 ***
---
Signif. codes:  0 *** 0.001 ** 0.01 * 0.05 . 0.1   1

Diagnostic Tests:
Jarque Bera Test

data:  Residuals
X-squared = 9478.852, df = 2, p-value < 2.2e-16


Box-Ljung test

data:  Squared.Residuals
X-squared = 7.7792, df = 1, p-value = 0.005285

# Estimate GARCH(1,3) ###################################

# 'order[2]' corresponds to the ARCH part and 'order[1] to the GARCH part
```

```
# GARCH(p,q): p on ARCH-part ja q on GARCH-part

x<-100*r
order<-c(3,1)
ga<-garch(x,order=order)
summary(ga)

Call:
garch(x = x, order = order)

Model:
GARCH(3,1)

Residuals:
      Min        1Q    Median        3Q       Max
-10.60827  -0.52684   0.06149   0.63883   6.52770

Coefficient(s):
    Estimate  Std. Error  t value Pr(>|t|)
a0 9.410e-03   9.158e-04   10.275  < 2e-16 ***
a1 1.055e-01   3.925e-03   26.881  < 2e-16 ***
b1 6.520e-01   5.869e-02   11.110  < 2e-16 ***
b2 4.893e-13   8.861e-02    0.000        1
b3 2.350e-01   5.634e-02    4.171 3.03e-05 ***
---
Signif. codes:  0 *** 0.001 ** 0.01 * 0.05 . 0.1   1

Diagnostic Tests:
Jarque Bera Test

data:  Residuals
X-squared = 8724.943, df = 2, p-value < 2.2e-16


Box-Ljung test

data:  Squared.Residuals
X-squared = 1.0553, df = 1, p-value = 0.3043


# Fan & Yao, 1974-1999 s. 174: #################################
```

```
# a0  0.015
# a1  0.112
# b1  0.492
# b2 -0.034
# b3  0.420


# hat(sigma)_t ######################################

a0<-0.0074141
a1<-0.0775141
b1<-0.9164638

sig<-sd(x)                 # [1] 0.9729996
sig<-sqrt(a0/(1-a1-b1))    # [1] 1.109571
T<-length(x)

hatsigma2<-matrix(sig^2,T,1)
for (i in 2:T) hatsigma2[i]<-a0+a1*x[i-1]^2+b1*hatsigma2[i-1]

plot(sqrt(hatsigma2),type="l")

# residuals #######################################

hateps<-matrix(0,T,1)
for (i in 1:T) hateps[i]<-x[i]/sqrt(hatsigma2[i])
plot(hateps,type="l")

# Akaike's information criterion ##########################

AIC(ga)

# [1] 38653.25

aics<-matrix(0,4,1)
for (i in 1:4){
    order<-c(1,i)
    ga<-garch(r,order=order)
    aics[i]<-AIC(ga)
}
plot(aics)
```

```
aics
          [,1]
[1,] -106356.3
[2,] -106292.0
[3,] -106136.5
[4,] -105892.5
```