Module 6 - Parameter Estimation

MATH1318 Time Series Analysis

Prepared by: Dr. Haydar Demirhan based on the textbook by Cryer and Chan, Time Series Analysis with R, Springer, 2008.

Introduction

After ensuring the series is stationary and specification of orders of autoregressive and moving average elements of ARMA models, the next step is to estimate parameters of the specified tentative models.

For this aim, we will focus on

- the method of moments,
- least squares estimation,
- maximum likelihood and unconditional least squares, and
- bootstrap approach to ARIMA models.

In this module, we will use `arima()` (http://stat.ethz.ch/R-manual/R-devel/library/stats/html/arima.html) and `ar()` (http://stat.ethz.ch/R-manual/R-devel/library/stats/html/ar.html) functions. However, `arima()` and `ar()` functions only give parameters estimates with a specified method.

To obtain significance tests for each parameter, we will use `coeftest()` function from `lmtest` (https://cran.r-project.org/web/packages/lmtest/) package with `arima()` function.

The Method of Moments

The method of moments is the easiest one to apply. The method consists of equating sample moments to corresponding theoretical moments and solving the resulting equations to obtain estimates of any unknown parameters.

Autoregressive Models

Let's consider the AR(1) model. For this model we know that $\rho_1 = \phi$. In the method of moments, ρ_1 is equated to r_1. Thus, we can estimate ϕ by $\hat{\phi} = r_1$.

Consider the AR(2) model. Here because we have two parameters ϕ_1 and ϕ_2 in the model, we need to use two sample moments for estimation. For this model, we know that

$$\rho_1 = \phi_1 + \rho_1\phi_2$$

and

$$\rho_2 = \rho_1\phi_1 + \phi_2.$$

We will equate ρ_1 to r_1 and ρ_2 to r_2 for estimation. Thus, we have

$$r_1 = \phi_1 + r_1\phi_2$$

and

$$r_2 = r_1\phi_1 + \phi_2.$$

Solving this system of equations, we obtain

$$\hat{\phi}_1 = \frac{r_1(1 - r_2)}{1 - r_1^2}$$

and

$$\hat{\phi}_2 = \frac{r_2 - r_1^2}{1 - r_1^2}.$$

For the general AR(p) models, we proceed similarly and solve a system of equations with p sample moments $r_1, r_2, \ldots, r_p$ and p parameters $\phi_1, \phi_2, \ldots, \phi_p$:

$$\begin{array}{rrrcl}
\phi_1 + & r_1\phi_2 + & r_2\phi_3 & + \cdots + r_{p-1}\phi_p & = r_1 \\
r_1\phi_1 + & \phi_2 + & r_1\phi_3 & + \cdots + r_{p-2}\phi_p & = r_2 \\
 & & & \vdots & \\
r_{p-1}\phi_1 + & r_{p-2}\phi_2 + & r_{p-3}\phi_3 & + \cdots + \quad \phi_p & = r_p
\end{array}$$

Because we use Yule-Walker equations to build up the system of equations to solve, the estimates obtained in this way are also called **Yule-Walker estimates**.

Moving Average Models

Let's consider MA(1) model. For this model, we know that

$$\rho_1 = -\frac{\theta}{1+\theta^2}.$$

Equating ρ_1 to r_1, we get the following two real roots if $|r_1| < 0.5$

$$-\frac{1}{2r_1} \pm \sqrt{\frac{1}{4r_1^2} - 1}.$$

Here, the product of the two solutions is always equal to 1; therefore, only one of the solutions satisfies the invertibility condition $|\theta| < 1$. After further algebraic manipulation, we see that the invertible solution can be written as

$$\hat{\theta} = \frac{-1 + \sqrt{1 - 4r_1^2}}{2r_1}.$$

If $r_1 = \pm 0.5$, unique, real solutions exist, namely ± 1, but neither is invertible. If $|r_1| > 0.5$ (which is certainly possible even though $|\rho_1| < 0.5$), no real solutions exist, and so the method of moments fails to yield an estimator of θ. Of course, if $|r_1| > 0.5$, the specification of an MA(1) model would be in considerable doubt.

As seen here, the method of moments quickly gets complicated for MA(q) models.

ARMA models

Consider ARMA(1,1) model. We have the following for this model for $k \geq 1$.

$$\rho_k = \frac{(1-\theta\phi)(\phi-\theta)}{1-2\theta\phi+\theta^2}\phi^{k-1}.$$

Because $\rho_2/\rho_1 = \phi$, we write $\hat{\phi} = r_2/r_1$. Then we use the following to solve for the estimate of θ, namely $\hat{\theta}$:

$$r_1 = \frac{(1-\theta\hat{\phi})(\hat{\phi}-\theta)}{1-2\theta\hat{\phi}+\theta^2}.$$

Note again that a quadratic equation must be solved and only the invertible solution, if any, retained.

Estimates of the Noise Variance

The final parameter to be estimated is the noise variance, σ_e^2. In all cases, we can first estimate the process variance, $\gamma_0 = Var(Y_t)$, by the sample variance

$$s^2 = \frac{1}{n-1}\sum_{t=1}^{n}(Y_t - \bar{Y})^2$$

and use known relationships among γ_0, σ_e^2 , and the θ's and ϕ's to estimate of σ_e^2.

For the AR(p) models,

$$\hat{\sigma}_e^2 = (1 - \hat{\phi}_1 r_1 - \hat{\phi}_2 r_2 - \cdots - \hat{\phi}_p r_p)s^2.$$

For the MA(q) models,

$$\hat{\sigma}_e^2 = \frac{s^2}{1 + \hat{\theta}_1^2 + \hat{\theta}_2^2 + \cdots + \hat{\theta}_q^2}.$$

For the ARMA(1,1) models,

$$\hat{\sigma}_e^2 = \frac{1 - \hat{\phi}^2}{1 - 2\hat{\phi}\hat{\theta} + \hat{\theta}^2}s^2.$$

Numerical Examples

The following table summarizes the method of moment parameter estimates for various simulated time series.

	True Parameters			Method-of-Moments Estimates			
Model	θ	ϕ_1	ϕ_2	θ	ϕ_1	ϕ_2	n
MA(1)	−0.9			−0.554			120
MA(1)	0.9			0.719			120
MA(1)	−0.9			NA[†]			60
MA(1)	0.5			−0.314			60
AR(1)		0.9			0.831		60
AR(1)		0.4			0.470		60
AR(2)		1.5	−0.75		1.472	−0.767	120

[†] No method-of-moments estimate exists since $r_1 = 0.544$ for this simulation.

While the estimates for all the autoregressive models are fairly good, the estimates for the moving average models are not acceptable. This is due to the inefficiency of the method of moments estimators for models containing moving average terms.

Consider the Canadian hare abundance series. Note that we have found that the square root transformation is appropriate for this series. So, we will proceed with the square root transformation in parameter estimation. First, we will fit an AR(2) model for this series with the method of moments. The first two sample autocorrelations are $r_1 = 0.736$ and $r_2 = 0.304$. Then, we find

$$\hat{\phi}_1 = \frac{r_1(1-r_2)}{1-r_1^2} = \frac{0.736(1-0.304)}{1-0.736^2} = 1.1178$$

and

$$\hat{\phi}_1 = \frac{r_2 - r_1^2}{1 - r_1^2} = \frac{0.304 - 0.736^2}{1 - 0.736^2} = -0.519.$$

The sample mean and variance for this series are 5.82 and 5.88, respectively. The the noise variance is estimated as

$$\begin{aligned}\hat{\sigma}_e^2 &= (1 - \hat{\phi}_1 r_1 - \hat{\phi}_2 r_2)s^2 \\ &= [1 - (1.1178 \cdot 0.736) - (-0.519)0.304]5.88 \\ &= 1.97.\end{aligned}$$

The prediction model is then

$$\sqrt{Y_t} = 2.335 + 1.1178\sqrt{Y_{t-1}} - 0.519\sqrt{Y_{t-2}} + e_t.$$

Let's consider the oil prices series. We have specified an MA(1) model for the first difference of natural logarithms of this series. For this series $r_1 = 0.21$, so the method of moments estimate of θ is

$$\hat{\theta} = \frac{-1 + \sqrt{1 - 4 \cdot 0.212^2}}{2 \cdot 0.212} = -0.222.$$

Hence the prediction model is

$$\log Y_t = \log(Y_{t-1}) + 0.004e_t + 0.222e_{t-1}.$$

with estimated noise variance of

$$\hat{\sigma}_e^2 = \frac{s^2}{1 + \hat{\theta}^2} = \frac{0.0072}{1 + (-0.222)^2} = 0.00686.$$

Least Squares Estimation

Although the method of moments approach is easy to apply for simple models, it is mostly inefficient. So, we have other more efficient (with smaller estimation variance) approaches for the parameter estimation of time series. The least squares approach is the first one we will consider.

For the AR(1) model, we have the following model which can also be considered as a regression model.

$$Y_t - \mu = \phi(Y_{t-1} - \mu) + e_t.$$

Here if we treat Y_{t-1} as the predictor variable and Y_t as the response variable, then we minimize the following for the least squares estimation:

$$S_c(\phi, \mu) = \sum_{t=2}^{n} [(Y_t - \mu) - \phi(Y_{t-1} - \mu)]^2.$$

This is usually called the **conditional sum of squares function**. We differentiate the conditional sum of squares function according to μ and ϕ and then solve the resulting system of equations with two Equations and two unknowns. As a result, we get the following least squares estimators:

$$\hat{\mu} \approx \frac{1}{1-\phi}(\bar{Y} - \phi\bar{Y}) = \bar{Y}$$

and

$$\hat{\phi} = \frac{\sum_{t=2}^{n}(Y_t - \bar{Y})(Y_{t-1} - \bar{Y})}{\sum_{t=2}^{n}(Y_{t-1} - \bar{Y})^2}$$

For the general AR(p) process, we obtain the following for μ

$$\hat{\mu} = \bar{Y}$$

The least squares estimates of $\phi 1, \phi_2, \ldots, \phi_p$ parameters are obtained by solving the system of sample Yule-Walker equations.

For the moving average models, it is hard to write the model in a regression model form. Even we write the model in a regression model format, it requires the use of numerical algorithms to find parameter estimates with least squares. These difficulties are also the case for ARMA models.

Maximum Likelihood and Unconditional Least Squares

In the maximum likelihood (ML) estimation we use all of the information contained in the sample than just the first and second moments. Another advantage is that many large-sample results are known under very general conditions. One disadvantage is that we must for the first time work specifically with the joint probability density function of the process.

Maximum Likelihood Estimation

The likelihood function L is obtained over the joint pdf of the sample in all kinds of data. For ARMA models L will be a function of ϕ's, θ's, μ, and σ_e^2 given the time series $Y_1, Y_2, \ldots, Y_n$.

> The maximum likelihood estimators are then defined as those values of the parameters for which the data actually observed are *most likely*, that is, the values that maximize the likelihood function.

The most common assumption of ML estimation is that the white noise terms are independent, normally distributed random variables with zero means and common standard deviation σ_e.

For the AR(1) model, we have the following likelihood function

$$L(\phi, \mu, \sigma_e) = (2\pi\sigma_e^2)^{-n/2}\sqrt{1-\phi^2}\exp[-0.5\sigma_e^{-2}S(\phi, \mu)]$$

where

$$S(\phi, \mu) = \sum_{t=2}^{n}[(Y_t - \mu) - \phi(Y_[t-1] - \mu)]^2 + (1-\phi^2)(Y_t - \mu).$$

The function $S(\phi, \mu)$ is called the **unconditional sum-of-squares function**. For the ease of computations, we go on with the logarithm of L, namely

$$\ell(\phi, \mu\sigma_e^2) = -\frac{n}{2}\log(2\pi) - \frac{n}{2}\log(\sigma_e^2) + \frac{1}{2}\log(1-\phi^2) - \frac{1}{2\sigma_e^2}S(\phi, \mu).$$

For given values of ϕ and μ, we obtain

$$\hat{\sigma}_e^2 = \frac{S(\hat{\phi}, \hat{\mu})}{n}$$

for the ML estimates of ϕ and μ get a system of nonlinear equations resulting from the partial derivation of S with respect to ϕ and μ. So, we need to use numerical algorithms to find ML estimated of ϕ and μ.

Illustrations of Parameter Estimation

Consider the simulated MA(1) series with $\theta = -0.9$. The method of moments estimate of θ was found -0.554. In contrast, the maximum likelihood estimate is -0.915, the unconditional sum-of-squares estimate is -0.923, and the conditional least squares estimate is -0.879. The maximum likelihood estimate is closest to the true value used in the simulation. The estimate of the standard error is about 0.04. So none of the maximum likelihood, conditional sum-of-squares, or unconditional sum-of-squares estimates are significantly far from the true value of -0.9.

For the MA(1) simulation with $\theta = 0.9$, the method of moments estimate was 0.719. The conditional sum of squares estimate is 0.958, the unconditional sum-of-squares estimate is 0.983, and the ML estimate is 1.000 which is a little disconcerting since it corresponds to a noninvertible model.

For the MA(1) simulation with $\theta = -0.9$, the method of moments estimate was -0.719. The conditional and unconditional sum of squares estimates are -0.979 and -0.961, respectively.

For the AR(1) models, we have the following results:

Parameter ϕ	Method-of-Moments Estimate	Conditional SS Estimate	Unconditional SS Estimate	Maximum Likelihood Estimate	n
0.9	0.831	0.857	0.911	0.892	60
0.4	0.470	0.473	0.473	0.465	60

```
data(ar1.s)
data(ar1.2.s)

# Method of moments estimate of the AR coefficient
# for simulated AR(1) model
ar(ar1.s,order.max=1,AIC=F,method='yw')
```

```
##
## Call:
## ar(x = ar1.s, order.max = 1, method = "yw", AIC = F)
##
## Coefficients:
##      1
## 0.8314
##
## Order selected 1  sigma^2 estimated as  1.382
```

```
# Least squares estimate of the AR coefficient
# for simulated AR(1) model
ar(ar1.s,order.max=1,AIC=F,method='ols')
```

```
##
## Call:
## ar(x = ar1.s, order.max = 1, method = "ols", AIC = F)
##
## Coefficients:
##     1
## 0.857
##
## Intercept: 0.02499 (0.1308)
##
## Order selected 1  sigma^2 estimated as  1.008
```

```
# Least squares estimate of the AR
# coefficient with significance tests
# for simulated AR(1) model
coeftest(arima(ar1.s,order=c(1,0,0),method='CSS'))
```

```
##
## z test of coefficients:
##
##           Estimate Std. Error z value Pr(>|z|)
## ar1       0.857041   0.062783 13.6508  < 2e-16 ***
## intercept 2.161228   0.908597  2.3786  0.01738 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
# Maximum likelihood estimate of the AR coefficient
# for simulated AR(1) model
ar(ar1.s,order.max=1,AIC=F,method='mle')
```

```
##
## Call:
## ar(x = ar1.s, order.max = 1, method = "mle", AIC = F)
##
## Coefficients:
##      1
## 0.8924
##
## Order selected 1  sigma^2 estimated as  1.041
```

```
# Maximum likelihood estimate of the AR
# coefficient with significance tests
# for simulated AR(1) model
coeftest(arima(ar1.s,order=c(1,0,0),method='ML'))
```

```
##
## z test of coefficients:
##
##           Estimate Std. Error z value Pr(>|z|)
## ar1       0.892436   0.059822 14.9181   <2e-16 ***
## intercept 1.263068   1.139865  1.1081   0.2678
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
# ------- The second simulated AR(1) model -------

# Method of moments estimate of the AR coefficient
# for simulated AR(1) model
ar(ar1.2.s,order.max=1,AIC=F,method='yw')
```

```
##
## Call:
## ar(x = ar1.2.s, order.max = 1, method = "yw", AIC = F)
##
## Coefficients:
##      1
## 0.4699
##
## Order selected 1  sigma^2 estimated as  0.9198
```

```
# Least squares estimate of the AR coefficient
# for simulated AR(1) model
ar(ar1.2.s,order.max=1,AIC=F,method='ols')
```

```
##
## Call:
## ar(x = ar1.2.s, order.max = 1, method = "ols", AIC = F)
##
## Coefficients:
##      1
## 0.4731
##
## Intercept: -0.006084 (0.1237)
##
## Order selected 1  sigma^2 estimated as  0.9024
```

```
# Least squares estimate of the AR
# coefficient with significance tests
# for simulated AR(1) model
coeftest(arima(ar1.2.s,order=c(1,0,0),method='CSS'))
```

```
##
## z test of coefficients:
##
##            Estimate Std. Error z value  Pr(>|z|)
## ar1         0.47310    0.11423  4.1416 3.449e-05 ***
## intercept -0.12051    0.23281 -0.5176    0.6047
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
# Maximum likelihood estimate of the AR coefficient
# for simulated AR(1) model
ar(ar1.2.s,order.max=1,AIC=F,method='mle')
```

```
##
## Call:
## ar(x = ar1.2.s, order.max = 1, method = "mle", AIC = F)
##
## Coefficients:
##      1
## 0.4654
##
## Order selected 1  sigma^2 estimated as  0.8875
```

```
# Maximum likelihood estimate of the AR
# coefficient with significance tests
# for simulated AR(1) model
coeftest(arima(ar1.2.s,order=c(1,0,0),method='ML'))
```

```
##
## z test of coefficients:
##
##            Estimate Std. Error z value  Pr(>|z|)
## ar1         0.46534    0.11274  4.1275 3.668e-05 ***
## intercept -0.11785    0.22428 -0.5255    0.5993
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

All four methods estimate reasonably well for AR(1) models and ML estimates are the closest ones to the true parameter values with an estimated standard error of 0.07 or 0.11.

For the simulated AR(2) models, we have the following results:

Parameters	Method-of-Moments Estimates	Conditional SS Estimates	Unconditional SS Estimates	Maximum Likelihood Estimate	*n*
$\phi_1 = 1.5$	1.472	1.5137	1.5183	1.5061	120
$\phi_2 = -0.75$	−0.767	−0.8050	−0.8093	−0.7965	120

```
data(ar2.s)

# Method of moments estimates of the AR coefficients
# for simulated AR(2) model
ar(ar2.s,order.max=2,AIC=F,method='yw')
```

```
##
## Call:
## ar(x = ar2.s, order.max = 2, method = "yw", AIC = F)
##
## Coefficients:
##       1        2
##  1.4694  -0.7646
##
## Order selected 2  sigma^2 estimated as  1.051
```

```
# Least squares estimates of the AR coefficients
# for simulated AR(2) model
ar(ar2.s,order.max=2,AIC=F,method='ols')
```

```
##
## Call:
## ar(x = ar2.s, order.max = 2, method = "ols", AIC = F)
##
## Coefficients:
##       1        2
##  1.5137  -0.8050
##
## Intercept: 0.02043 (0.08594)
##
## Order selected 2  sigma^2 estimated as  0.8713
```

```
# Least squares estimates of the AR
# coefficients with significance tests
# for simulated AR(2) model
coeftest(arima(ar2.s,order=c(2,0,0),method='CSS'))
```

```
##
## z test of coefficients:
##
##            Estimate Std. Error  z value Pr(>|z|)
## ar1        1.513715   0.055030  27.5070   <2e-16 ***
## ar2       -0.804991   0.054887 -14.6664   <2e-16 ***
## intercept  0.263679   0.292695   0.9009   0.3677
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
# Maximum likelihood estimates of the AR coefficients
# for simulated AR(2) model
ar(ar2.s,order.max=2,AIC=F,method='mle')
```

```
##
## Call:
## ar(x = ar2.s, order.max = 2, method = "mle", AIC = F)
##
## Coefficients:
##       1        2
##  1.5061  -0.7964
##
## Order selected 2  sigma^2 estimated as  0.862
```

```
# Maximum likelihood estimates of the AR
# coefficients with significance tests
# for simulated AR(2) model
coeftest(arima(ar2.s,order=c(2,0,0),method='ML'))
```

```
##
## z test of coefficients:
##
##             Estimate Std. Error  z value Pr(>|z|)
## ar1         1.506146   0.053739  28.0269   <2e-16 ***
## ar2        -0.796453   0.053322 -14.9367   <2e-16 ***
## intercept  0.237908   0.292719   0.8128   0.4164
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Again, considering the size of the standard errors, all four methods estimate reasonably well for AR(2) models.

For the simulated ARMA(1,1) model, we have the following results:

Parameters	Method-of-Moments Estimates	Conditional SS Estimates	Unconditional SS Estimates	Maximum Likelihood Estimate	*n*
$\phi = 0.6$	0.637	0.5586	0.5691	0.5647	100
$\theta = -0.3$	−0.2066	−0.3669	−0.3618	−0.3557	100

```
data(arma11.s)
# Least squares estimates of the coefficients for
# simulated ARMA(1,1) model
arima(arma11.s, order=c(1,0,1),method='CSS')
```

```
##
## Call:
## arima(x = arma11.s, order = c(1, 0, 1), method = "CSS")
##
## Coefficients:
##          ar1     ma1  intercept
##       0.5586  0.3669     0.3928
## s.e.  0.1219  0.1564     0.3380
##
## sigma^2 estimated as 1.199:  part log likelihood = -150.98
```

```
# Least squares estimates of the coefficients
# with significance tests for simulated ARMA(!,1) model
coeftest(arima(arma11.s,order=c(1,0,1),method='CSS'))
```

```
##
## z test of coefficients:
##
##            Estimate Std. Error z value  Pr(>|z|)
## ar1         0.55858    0.12192  4.5814 4.618e-06 ***
## ma1         0.36688    0.15645  2.3451   0.01902 *
## intercept   0.39277    0.33804  1.1619   0.24527
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
# Maximum likelihood estimates of the coefficients for
# simulated ARMA(1,1) model
arima(arma11.s, order=c(1,0,1),method='ML')
```

```
##
## Call:
## arima(x = arma11.s, order = c(1, 0, 1), method = "ML")
##
## Coefficients:
##          ar1     ma1  intercept
##       0.5647  0.3557     0.3216
## s.e.  0.1205  0.1585     0.3358
##
## sigma^2 estimated as 1.197:  log likelihood = -151.33,  aic = 308.65
```

```
# Maximum likelihood estimates of the coefficients
# with significance tests for simulated ARMA(!,1) model
coeftest(arima(arma11.s,order=c(1,0,1),method='ML'))
```

```
##
## z test of coefficients:
##
##           Estimate Std. Error z value  Pr(>|z|)
## ar1        0.56475    0.12046  4.6882 2.756e-06 ***
## ma1        0.35570    0.15847  2.2446   0.02479 *
## intercept  0.32162    0.33577  0.9579   0.33813
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

We have specified an AR(1) model for the industrial chemical property time series. Parameter estimates for this model are shown below:

Parameter	Method-of-Moments Estimate	Conditional SS Estimate	Unconditional SS Estimate	Maximum Likelihood Estimate	n
ϕ	0.5282	0.5549	0.5890	0.5703	35

```
data(color)

# Method of moments estimates
ar(color,order.max=1,AIC=F,method='yw')
```

```
##
## Call:
## ar(x = color, order.max = 1, method = "yw", AIC = F)
##
## Coefficients:
##      1
## 0.5282
##
## Order selected 1  sigma^2 estimated as  27.56
```

```
# Least squares estimates wit significance tests
coeftest(arima(color,order=c(1,0,1),method='CSS'))
```

```
##
## z test of coefficients:
##
##           Estimate Std. Error z value  Pr(>|z|)
## ar1        0.68286    0.17000  4.0168 5.899e-05 ***
## ma1       -0.22288    0.24651 -0.9042    0.3659
## intercept 75.39455    2.07262 36.3765 < 2.2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
# Maximum likelihood estimates wit significance tests
coeftest(arima(color,order=c(1,0,1),method='ML'))
```

```
##
## z test of coefficients:
##
##           Estimate Std. Error z value  Pr(>|z|)
## ar1        0.67227    0.21455  3.1333  0.001728 **
## ma1       -0.14700    0.27416 -0.5362  0.591842
## intercept 74.17257    2.13617 34.7223 < 2.2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

The standard error of the estimates is about

$$\sqrt{\widehat{Var}(\hat{\phi})} \approx \sqrt{\frac{1-0.57^2}{35}} \approx 0.14$$

so all of the estimates are comparable.

Parameter estimates of an AR(3) model the Canadian hare abundance series are shown below:

Coefficients:	ar1	ar2	ar3	Intercept†
	1.0519	–0.2292	–0.3931	5.6923
s.e.	0.1877	0.2942	0.1915	0.3371

sigma^2 estimated as 1.066: log-likelihood = -46.54, AIC = 101.08

```
data(hare)
coeftest(arima(sqrt(hare),order=c(3,0,0),method='ML'))
```

```
##
## z test of coefficients:
##
##            Estimate Std. Error z value  Pr(>|z|)
## ar1         1.05190    0.18767  5.6051 2.081e-08 ***
## ar2        -0.22925    0.29419 -0.7793    0.4358
## ar3        -0.39304    0.19148 -2.0527    0.0401 *
## intercept  5.69227    0.33709 16.8866 < 2.2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

The estimates of the lag 1 and lag 3 autoregressive coefficients are significantly different from zero, as is the intercept term, but the lag 2 autoregressive parameter estimate is not significant. Then the prediction model is written as

$$\sqrt{Y_t} - 5.6923 = 1.0519(\sqrt{Y_{t-1}} - 5.6923) - 0.2292(\sqrt{Y_{t-2}} - 5.6923) - 0.3930(\sqrt{Y_{t-3}} - 5.6923) + e_t$$

or

$$\sqrt{Y_t} = 3.25 + 1.0519\sqrt{Y_{t-1}} - 0.2292\sqrt{Y_{t-2}} - 0.3930\sqrt{Y_{t-3}} + e_t\,.$$

where Y_t is the hare abundance in year t in original terms.

We have specified a MA(1) model for n the differences of the logs of the oil prices. Estimates of θ are given below:

Coefficients:	ar1	ar2	ar3	Intercept†
	1.0519	–0.2292	–0.3931	5.6923
s.e.	0.1877	0.2942	0.1915	0.3371

sigma^2 estimated as 1.066: log-likelihood = -46.54, AIC = 101.08

```
data(oil.price)

# Least squares estimates wit significance tests
coeftest(arima(log(oil.price),order=c(0,1,1),method='CSS'))
```

```
##
## z test of coefficients:
##
##     Estimate Std. Error z value  Pr(>|z|)
## ma1 0.273113   0.068106  4.0101 6.069e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
# Maximum likelihood estimates wit significance tests
coeftest(arima(log(oil.price),order=c(0,1,1),method='ML'))
```

```
##
## z test of coefficients:
##
##     Estimate Std. Error z value Pr(>|z|)
## ma1 0.295600   0.069347  4.2626 2.02e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

The method of moments estimate differs quite a bit from the others which are nearly equal given their standard errors of about 0.07.

Bootstrapping ARIMA Models

We focused on some approximate normal distribution results for the estimation of parameters of ARMA models. These approximations are accurate for large samples. However, the general theory provides no practical guidance on how large the sample size should be for the normal approximation to be reliable. Bootstrap methods (Efron and Tibshirani, 1993; Davison and Hinkley, 2003) provide an alternative approach to assessing the uncertainty of an estimator and may be more accurate for small samples.

We will focus on the parametric bootstrap that generates the bootstrap time series $Y_1^*, Y_2^*, \dots, Y_n^*$ by simulation from the fitted ARIMA(p,d,q) model. If the errors are assumed to be normally distributed, the errors may be drawn randomly and with replacement from $N(0, \sigma_e^2$. For the case of unknown error distribution, the errors can be drawn randomly and with replacement from the residuals of the fitted model. For each bootstrap series, let γ^* be the estimator computed based on the bootstrap time series data using the method of full maximum likelihood estimation assuming stationarity. The bootstrap is replicated B times.From the B bootstrap parameter estimates, we obtain an empirical distribution and use it to calibrate the uncertainty in $\hat{\gamma}$. Suppose we are interested in estimating some function of γ, say $h(\gamma)$. Using the percentile method, a 95% bootstrap confidence interval for $h(\gamma)$ can be obtained as the interval from the 2.5 percentile to the 97.5 percentile of the bootstrap distribution of $h(\gamma^*)$.

Let us consider the hare series to illustrate the bootstrap method. We apply the bootstrap method - I by conditioning on the initial three observations and assuming normal errors. Those for the method - II are obtained using the same method except that the errors are drawn from the residuals. For the methods III and IV, the confidence intervals are based on the stationary bootstrap with a normal error distribution for the method - III and the empirical residual distribution for the method - IV. The theoretical one shows 95% confidence intervals based on the large-sample distribution results for the estimators.

Method	ar1	ar2	ar3	intercept	noise var.
I	(0.593, 1.269)	(−0.655, 0.237)	(−0.666, −0.018)	(5.115, 6.394)	(0.551, 1.546)
II	(0.612, 1.296)	(−0.702, 0.243)	(−0.669, −0.026)	(5.004, 6.324)	(0.510, 1.510)
III	(0.699, 1.369)	(−0.746, 0.195)	(−0.666, −0.021)	(5.056, 6.379)	(0.499, 1.515)
IV	(0.674, 1.389)	(−0.769, 0.194)	(−0.665, −0.002)	(4.995, 6.312)	(0.477, 1.530)
Theoretical	(0.684, 1.42)	(−0.8058, 0.3474)	(−0.7684, −0.01776)	(5.032, 6.353)	(0.536, 1.597)

The following code chunks are used to produce results in this table:

```
set.seed(12345)
res=arima(sqrt(hare),order=c(3,0,0),include.mean=T)
# Method I
coefm.cond.norm=arima.boot(res,cond.boot=T,is.normal=T,
                           B=1000,init=sqrt(hare))
signif(apply(coefm.cond.norm,2,
             function(x){quantile(x,c(.025,.975),na.rm=T)}),3)
```

```
##          ar1    ar2     ar3 intercept noise var
## 2.5%  0.593 -0.667 -0.6740      5.12     0.548
## 97.5% 1.280  0.244 -0.0135      6.38     1.540
```

```
# signif() function rounds to 3 decimal places

# Method II
# Apply bootstrapping
coefm.cond.replace=arima.boot(res,cond.boot=T,is.normal=F,
                              B=1000,init=sqrt(hare))
#Find the intervals from the Bootstrap output
signif(apply(coefm.cond.replace,2,
             function(x){quantile(x,c(.025,.975),na.rm=T)}),3)
```

```
##          ar1    ar2     ar3 intercept noise var
## 2.5%  0.611 -0.700 -0.6720      4.98     0.516
## 97.5% 1.300  0.241 -0.0417      6.32     1.500
```

```
# signif() function rounds to 3 decimal places

# Method III
# Apply bootstrapping
coefm.norm=arima.boot(res,cond.boot=F,is.normal=T,
                      ntrans=100,B=1000,init=sqrt(hare))
#Find the intervals from the Bootstrap output
signif(apply(coefm.norm,2,
             function(x){quantile(x,c(.025,.975),na.rm=T)}),3)
```

```
##          ar1    ar2     ar3 intercept noise var
## 2.5%   0.687 -0.747 -0.6600      4.99     0.508
## 97.5% 1.380  0.192 -0.0168      6.33     1.500
```

```
# signif() function rounds to 3 decimal places

# Method IV
# Apply bootstrapping
coefm.replace=arima.boot(res,cond.boot=F,is.normal=F,ntrans=100,
                         B=1000,init=sqrt(hare))
#Find the intervals from the Bootstrap output
signif(apply(coefm.replace,2,
             function(x){quantile(x,c(.025,.975),na.rm=T)}),3)
```

```
##         ar1    ar2     ar3 intercept noise var
## 2.5%   0.70 -0.715 -0.6620      4.98      0.47
## 97.5% 1.36  0.183 -0.0187      6.30      1.50
```

```
# signif() function rounds to 3 decimal places
```

In particular, the bootstrap time series for the first bootstrap method is generated recursively using the equation

$$Y_t^* - \hat{\phi}_1 Y_{t-1}^* - \hat{\phi}_2 Y_{t-2}^* - \hat{\phi}_3 Y_{t-3}^* = \hat{\theta}_0 + e^*$$

for $t = 4, 5, \dots, 31$, where the e_t^* are chosen independently from $N(0, \sigma_e^2$, $Y_1^* = Y_1, Y_2^* = Y_2, Y_3^* = Y_3$; and the parameters are set to be the estimates from the AR(3) model fitted to the (square root transformed) hare data with $\hat{\theta}_0 = \hat{\mu}(1 - \hat{\phi}_1 - \hat{\phi}_2 - \hat{\phi}_3)$. All results are based on about 1000 bootstrap replications, but full maximum likelihood estimation fails for 6.3%, 6.3%, 3.8%, and 4.8% of 1000 cases for the four bootstrap methods I, II, III, and IV, respectively.

All four methods yield similar bootstrap confidence intervals, although, as expected, the conditional bootstrap approach generally yields slightly narrower confidence intervals. The bootstrap confidence intervals are generally slightly wider than their theoretical counterparts that are derived from the large-sample results. Overall, we can draw the inference that the ϕ_2 coefficient estimate is insignificant, whereas both the ϕ_1 and ϕ_3 coefficient estimates are significant at the 5% significance level.

Summary

In this module, we focused on the estimation of the parameters of ARIMA models. We considered

- the method of moments,
- least squares, and
- maximum likelihood

Approaches for the estimation of ARMA parameters. The estimators were illustrated both with simulated and actual time series data.

We also considered Bootstrapping with ARIMA models, which is an important approach in practice.

References

Efron, B. and Tibshirani, R. J. (1993). An Introduction to the Bootstrap. New York: Chapman and Hall.

Davison, A. C. and Hinkley, D. V. (2003). Bootstrap Methods and Their Application, 2nd ed. New York: Cambridge University Press.

West, M. (1995). Time series decomposition and analysis in a study of oxygen isotope records (http://ftp.stat.duke.edu/WorkingPapers/95-18.pdf). Working paper. ISDS Discussion Paper, 95-18, Duke University.