6 Chapter 6: ARIMA Models

We have already discussed the importance of the class of ARMA models for representing stationary series. A generalization of this class, which incorporates a wide range of non-stationary series, is provided by the ARIMA processes, i.e. processes which, after differencing finitely many times, reduce to ARMA processes.

Definition If d is a non-negative integer, then $\{X_t\}$ is said to be an ARIMA(p, d, q) process if $Y_t = (1-B)^d X_t$ is a causal ARMA(p, q) process.

The definition means that $\{X_t\}$ satisfies a difference equation of the form

$$\phi^*(B)X_t \equiv \phi(B)(1-B)^d X_t = \theta(B)\varepsilon_t, \qquad \{\varepsilon_t\} \sim WN\left(0, \sigma^2\right),$$

where $\phi(\cdot)$ and $\theta(\cdot)$ are polynomials of degree p and q respectively and $\phi(\lambda) \neq 0$ for $|\lambda| \leq 1$. The polynomial $\phi^*(\lambda)$ has a zero of order d at $\lambda = 1$. The process is stationary if $d = 0$, in which case it reduces to an ARMA(p, q) process.

Notice that if $d \geq 1$ we can add an arbitrary trend of degree $(d-1)$ to $\{X_t\}$ without violating the difference ARIMA equation. (Exercise. Show that this is true.)

ARIMA models are therefore useful for representing data with trend. However, ARIMA processes can also be appropriate for modelling series with no trend.

Notice that except when $d = 0$, the mean of $\{X_t\}$ is not determined by the ARIMA equation and it can in particular be zero.

Since for $d \geq 1$, the ARIMA equation determines the second order properties of $(1-B)^d\{X_t\}$ but not those of $\{X_t\}$, estimation of ϕ, θ and σ^2 will be based on the observed differences $(1-B)^d\{X_t\}$.

We apply the same estimation and model selection methods to ARIMA models as to ARMA models, i.e. LS, MLE, AIC etc. The estimation methods for ARMA models that have been discussed in the previous lecture are meaningful if it is plausible to assume that the data are in fact a realization of a weakly stationary random process. (**Remark.** In case of ARIMA, we need to check that $(1-B)^d\{X_t\}$ is a weakly stationary process.) If the data display characteristics suggesting non-stationarity (trend and seasonality), then it may be necessary to make a transformation so as to produce a new series which is more compatible with the assumption of stationarity.

Deviations from stationarity may be suggested by the graph of the series itself or by the sample autocorrelation function or both.

For example, as we discussed earlier, various variance stabilizing transformations such as log or square root, are used whenever the series has a standard deviation changing with time. (See the Box-Cox procedure on a more systematic account of variance-stabilizing transformations.)

Trend and seasonality are usually detected by inspecting the graph of the (possibly transformed) series. However they are also characterized by sample autocorrelation functions which are slowly decaying and nearly periodic respectively. The elimination of trend and seasonality may be performed using two methods:

- "classical decomposition" of the series into a trend component, a seasonal component, and a random residual component;

- differencing. (Typically differencing d times is incorporated into an ARIMA model through the parameter d.)

After the elimination of trend and seasonality it is still possible that the sample autocorrelation function may appear to be that of a non-stationary or nearly non-stationary process, in which case further differencing may be carried out. This non-stationarity is due to the fact that some roots of the polynomial $\phi^*(z)$ may be located outside but very close to the unit circle. So check that the roots of $\phi^*(z)$ are located comfortably outside of the unit circle.

One may apply the unit roots tests, for example, the most popular in econometrics Dickey-Fuller test. Consider for simplicity an AR(1) model

$$X_t = \phi_1 X_{t-1} + \varepsilon_t.$$

Is $\phi_1 = 1$? Subtract X_{t-1} from both sides to get

$$\begin{aligned} X_t - X_{t-1} &= (\phi_1 - 1)X_{t-1} + \varepsilon_t, \\ \nabla X_t &= \gamma X_{t-1} + \varepsilon_t. \end{aligned} \tag{6.1}$$

Our test of hypothesis is

$H_0 : \gamma = 0,$ X_t has a unit root; p - value is large

$H_1 : \gamma \neq 0,$ X_t is stationary; p - value is small.

The Dickey-Fuller test statistic is constructed as a t-statistic. That is $\hat{\gamma}/\sqrt{\mathrm{Var}\,(\hat{\gamma})}$. The critical values come from a set of tables prepared by Dickey and Fuller.

Similar Dickey-Fuller procedure may be applied to AR(p) models.

Example with the BJsales data.

```
> library(tseries)
> adf.test(BJsales)

        Augmented Dickey-Fuller Test

data:  BJsales Dickey-Fuller = -2.1109, Lag order = 5,

p-value =0.5302 alternative hypothesis: stationary

#### DF test on single differenced data ####
> adf.test(diff(BJsales))

        Augmented Dickey-Fuller Test

data:  diff(BJsales) Dickey-Fuller = -3.3485, Lag order = 5,
```

```
p-value = 0.06585 alternative hypothesis: stationary

#### DF test on twice differenced data ####
> adf.test(diff(diff(BJsales)))

        Augmented Dickey-Fuller Test

data:  diff(diff(BJsales)) Dickey-Fuller = -6.562, Lag order = 5,
p-value = 0.01 alternative hypothesis: stationary

Warning message: p-value smaller than printed p-value in:
adf.test(diff(diff(BJsales)))
```

Notice that the Dickey-Fuller (DF) test suggests an existence of a unit root for the **BJsales** data and even for the **single differenced BJsales** data yielding p-values of 0.53 and 0.066 respectively. Only **twice differenced BJsales data** pass the DF test with the p-value of 0.01. Thus, we were right when we differenced **BJsales data** twice!

Remark. Notice that we already fitted ARIMA models to the **BJsales** data! In fact, we twice differenced the data and then have selected an ARMA(1,1) model for the twice differenced data, which corresponds to an ARIMA(1,2,1) for the original **BJsales** data. Compare the R outputs:

```
###### ARIMA(1,2,1) for the original data#######
> arima121<-arima0(BJsales, order=c(1, 2, 1))
> arima121

Call: arima0(x = BJsales, order = c(1, 2, 1))

Coefficients:
         ar1      ma1
      0.0533  -0.7805
s.e.  0.0821   0.2564

sigma^2 estimated as 1.863:  log likelihood = -256.48,
aic = 518.97

###### ARMA(1,1) for the twice differenced data#######
> arma11<-arima0(BJ2, order=c(1, 0, 1))
> arma11
Call: arima0(x = BJ2, order = c(1, 0, 1))

Coefficients:
         ar1      ma1  intercept
      0.0532  -0.7805     0.0024
s.e.  0.0821   0.2560     0.0267
```

```
sigma^2 estimated as 1.863:  log likelihood = -256.48,
aic = 520.96
```

Notice that we obtained *almost* identical results. The slight difference is due to numerical approximations in optimization.

Hence, now we can update our model selection table from the Lecture 3.

Table 6.1: Model fits to **BJ sales** data using the ML method.

Twice differenced **BJsales**					Original **BJsales**				
Order	loglik	AIC	σ^2	Converged	Order	loglik	AIC	σ^2	Converged
(1,0,0)	-268.98	543.96	2.22	T	(1,2,0)	-268.98	541.96	2.22	T
(0,0,1)	-256.56	519.13	1.87	T	(0,2,1)	-256.57	517.14	1.87	T
(2,0,0)	-263.15	534.29	2.05	T	(2,2,0)	-263.15	532.3	2.05	T
(0,0,2)	-256.68	521.36	1.24	T (Warn)	(0,2,2)	-256.69	519.37	1.24	T (Warn)
(1,0,1)	-256.48	520.96	1.86	T	(1,2,1)	-256.48	518.97	1.86	T
(2,0,1)	-256.14	522.29	1.85	T(Warn)	(2,2,1)	-256.11	520.22	1.85	T(Warn)
(1,0,2)	-256.52	523.04	1.87	T	(1,2,2)	-256.51	521.03	1.86	T
(3,0,0)	-258.38	526.76	1.92	T	(3,2,0)	-258.38	524.76	1.92	T
(3,0,1)	-254.61	521.22	1.80	T(Warn)	(3,2,1)	-254.64	519.28	1.80	T
(3,0,2)	-255.24	524.48	1.80	T	(3,2,2)	-255.45	522.90	1.82	T
(3,0,3)	-254.12	524.23	1.78	T(Warn)	(3,2,3)	-255.1	524.20	1.80	T(Warn)
(1,0,3)	-255.99	523.98	1.19	T(Warn)	(1,2,3)	-256.00	522.00	1.19	F
(2,0,3)	-254.22	522.43	1.78	T	(2,2,3)	-254.41	520.82	1.78	T(Warn)

The best model in terms of AIC is again ARIMA(0,2,1) that corresponds to MA(1) for twice differenced data, and ARIMA(1,2,1) that corresponds to ARMA(1,1) for twice differenced data. Overall, the result and therefore the conclusions are just the same as in Lecture 3.

Also notice that we do not need to re-run the residual diagnostics for the ARIMA(1,2,1) model for **BJsales** data since the ARIMA(1,2,1) residuals are **the same** as the ARMA(1,1) residuals for the **twice differenced BJsales** data.

Now let us forecast two steps ahead from the ARIMA(1,2,1) model.

```
> arima121<-arima0(BJsales, order=c(1, 2, 1))

> predict(arima121, n.ahead = 2)
$pred Time Series: Start = 151 End = 152
Frequency = 1 [1] 263.0114 263.3128
$se Time Series: Start = 151 End = 152
Frequency = 1 [1] 1.365041 2.209501
```

However, if we predict from the ARMA(1,1) model for the the twice differenced **BJsales** data, the resulting forecast will be also for the twice differenced data. Hence, we would be forced to undifference data manually, which is undesirable. Thus, we will always forecast from the final ARIMA model for the original data rather than from ARMA models for the differenced data even though theoretically those models are equivalent.