3 Chapter 3: Modelling and Forecasting with ARMA processes

3.1 Introduction

The process of fitting an ARMA model, as it was made explicit by Box and Jenkins, may be divided into three components,

- Identification
- Estimation
- Verification

which are iterated until a suitable model is identified.

3.2 Identification

This refers both to the initial preprocessing of the data to make the series stationary, and also to the identification of suitable orders p and q for the ARMA components of the model. The latter identification, however, is always preliminary, since there is plenty of scope to adjust p and q on the basis of the models fitted.

A time series analysis should always begin with a preliminary plot of the data, as an indication of gross features that should guide the analysis. For example, Fig. 3.1 shows a raw time series plot of some electroencephalogram (EEG) data (data of Professor Mike West, Duke University): one would certainly not want to analyze this as a stationary time series, even after differencing, without some initial preprocessing of the data! It might be reasonable to fit ARIMA models to portions of the series.

Another example of data that needs preprocessing is the Merck volume data. Merck & Co., Inc. (Merck) is a global pharmaceutical company that discovers, develops, manufactures and markets a range of products to improve human and animal health. The Company's operations consist of two segments: the Pharmaceutical segment and the Vaccines segment[1]. It was established in 1891 as the United States subsidiary of the German company now known as Merck KGaA. It is now one of the top 7 largest pharmaceutical companies in the world both by capital and revenue.

The main tools for initial preprocessing are differencing and variance stabilizing transformations (square root, log, inverse etc, for more details see literature on the Box-Cox variance

[1]The data are kindly provided by Saad Zaman.

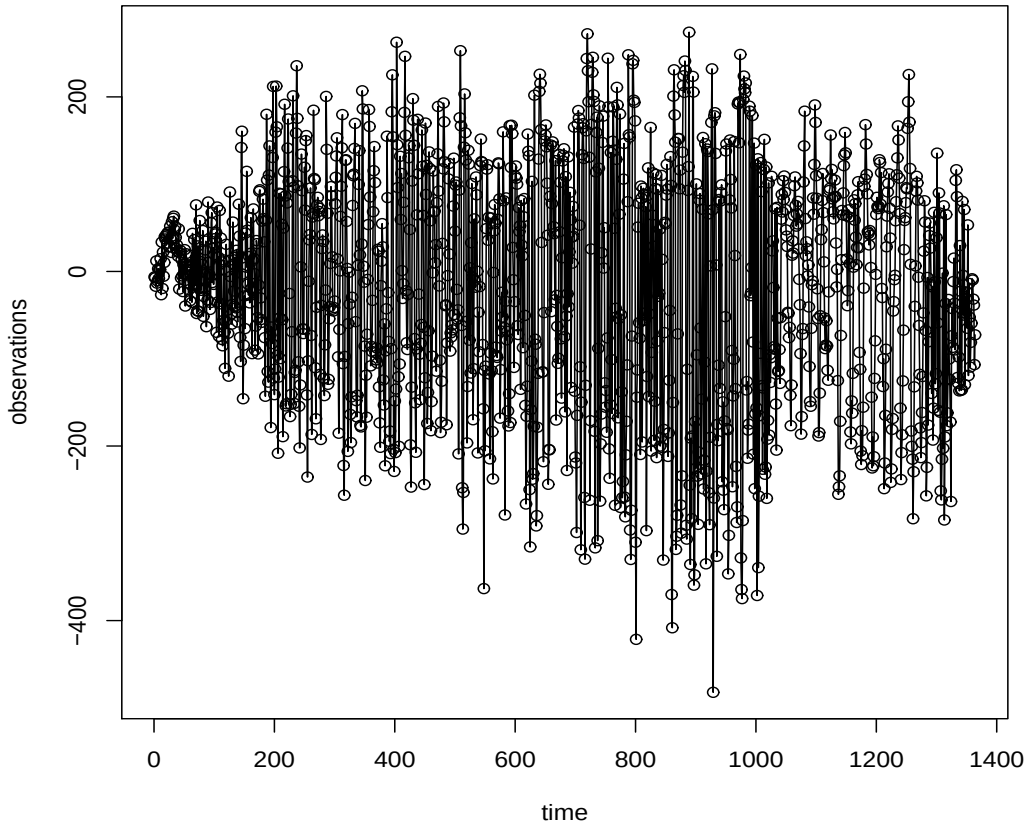

Figure 3.1: Plot of EEG recordings from a long EEG trace recorded in the ECT Lab at Duke, on a patient undergoing ECT therapy for clinical depression. The data are fluctuations in electrical potential at a point on the patient's scalp during seizure, one of several "channels" of recordings. They are measured in microvolts and represent measurement taken at time intervals of roughly one fortieth of a second.

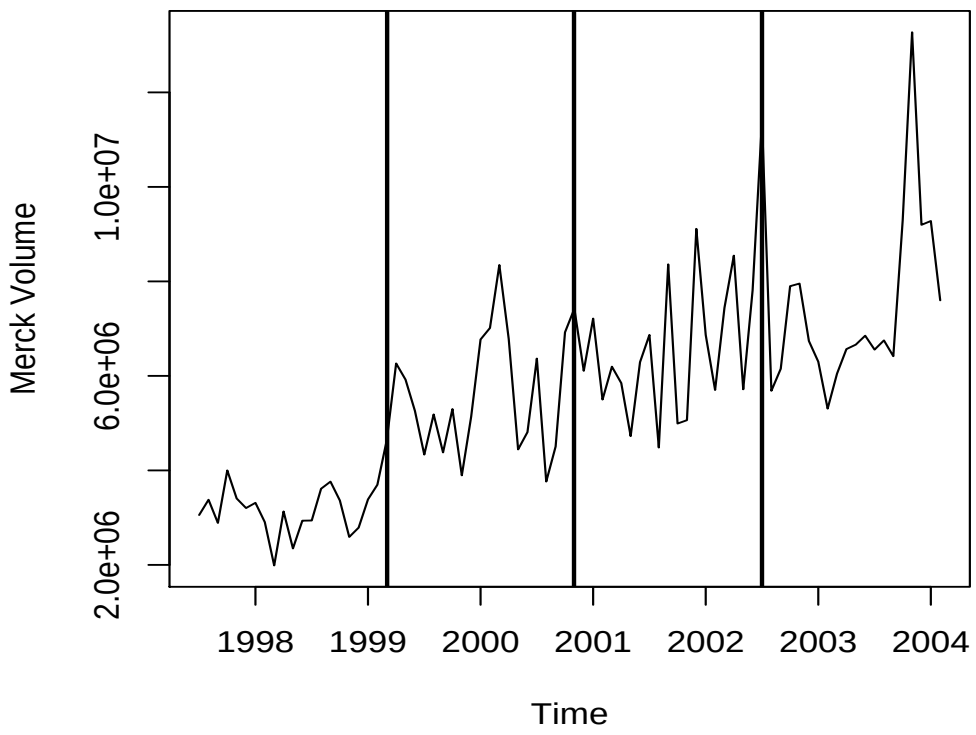

Figure 3.2: The Merck Volume data from July 1997 to Feb. 2004.

stabilizing transformations). Other methods, e.g. removal of deterministic components by linear regression, are perfectly acceptable, especially when there is some well-defined physical reason for the presence of this component, e.g. annual cycles in meteorological data.

As a guide to the amount of differencing (or other preprocessing) required, the main tool is the autocorrelation function (acf).

- With a stationary series, this should decay fairly rapidly to 0.
- If it fails to do so, then another layer of differencing is usually required.

In practice it is rare to go beyond $d = 2$: if the series fails to look stationary after two or at most three applications of differencing, there is probably some more fundamental reason that needs separate investigation.

Once the series is accepted as stationary, the next step is initial identification of p and q. The main tools used for this are the a.c.f. and p.a.c.f. (partial autocorrelation function) plots. In particular,

- An MA(q) series is identified from the property that all values of the a.c.f. after the q-th are negligible,
- An AR(p) series is identified from the property that all values of the p.a.c.f. after the p-th are negligible.

As a guide to what constitutes negligibility, it is worth noting that sample values of the a.c.f. and p.a.c.f. very approximately have standard deviation around $1/\sqrt{T}$ where T is the length of the series. Thus a rule of thumb for treating these values as negligible is based on two standard deviations, or $\pm 2/\sqrt{T}$. In S-Plus and R, lines at $\pm 2/\sqrt{T}$ are shown on the plot as an aid in this process.

3.3 Estimation

(a) **the Yule-Walker method**

The standard tool for autoregressive processes is to solve the Yule-Walker equations. They are derived from the model relationship

$$X_t = \sum_{j=1}^{p} \phi_j X_{t-j} + \varepsilon_t. \tag{3.1}$$

Taking the covariance of both sides with X_{t-k}, we deduce, for $k > 0$,

$$\gamma_k = \sum_{j=1}^{p} \phi_j \gamma_{|j-k|}. \tag{3.2}$$

If we consider equation (3.2) for $1 \leq k \leq p$, we get a system of p equations in p unknowns $\phi_1, \phi_2, \ldots, \phi_p$, which we can therefore solve in terms of $\gamma_1, \gamma_2, \ldots, \gamma_p$. In practice, of course, we substitute the sample estimates of the autocovariances $\hat{\gamma}_1, \hat{\gamma}_2, \ldots, \hat{\gamma}_p$ to obtain sample estimates $\hat{\phi}_1, \hat{\phi}_2, \ldots, \hat{\phi}_p$. The estimated variance is defined by

$$\hat{\sigma}^2 = \hat{\gamma}_0 - \hat{\phi}' \hat{\gamma}_p,$$

where $\hat{\phi}$ is a vector of estimated coefficients $\phi_1, \phi_2, \ldots, \phi_p$ ($\hat{\phi} = \hat{\phi}_1, \hat{\phi}_2, \ldots, \hat{\phi}_p$).

It is often the case that moment estimators, i.e. estimators like $\hat{\phi}$ that are obtained by equating the sample and theoretical moments, have much higher variances than estimators obtained by the alternative methods such as maximum likelihood. However, asymptotically the Yule-Walker estimators of the coefficients $\phi_1, \phi_2, \ldots, \phi_p$ of an AR(p) process have approximately the same distribution for large samples as the corresponding maximum likelihood estimators. In particular, for large samples

$$\hat{\phi} \sim N(\phi, n^{-1}\sigma^2 \Gamma_p^{-1}), \tag{3.3}$$

where Γ_p is the covariance matrix, $\Gamma_p = [\gamma_{i-j}]_{i,j=1}^p$.

If we replace σ^2 and Γ_p by their estimates $\hat{\sigma}^2$ and $\hat{\Gamma}_p$, we can use this result to find large-sample confidence regions for ϕ.

In practice we do not know the true order of the model generating the data. In fact, it will usually be the case that there is no *true* AR model, in which case our goal is to simply to find the model which represents the data optimally in some sense. Two useful techniques for selecting an appropriate AR model are given below.

The first guidance is based on considering the partial autocorrelation coefficients. In fact, if an AR(p) model is suitable to the data then all values of the p.a.c.f. after the p-th are negligible and will fall between the bounds $\pm 2/\sqrt{T}$. This suggests using as a preliminary estimator of the order p the smallest value m such that $|\hat{\phi}_{kk}| < 1.96 n^{-1/2}$ for all $k > m$.

One can also use the estimated residual variance $\hat{\sigma}_p^2$ as a guide to the selection of an order p. In particular, define an approximate log likelihood

$$-2\ln(L) = T \ln(\hat{\sigma}_p^2) \tag{3.4}$$

as a basis for a likelihood ratio test statistic.

A more systematic approach to model selection is based on minimization of Akaike Information Criterion (AIC)

$$AIC = -2\ln(L) + 2k, \tag{3.5}$$

where k is the number of unknown parameters in the model. The idea is to choose the model of the order which minimizes AIC. This is a widely used measure in time series analysis, which

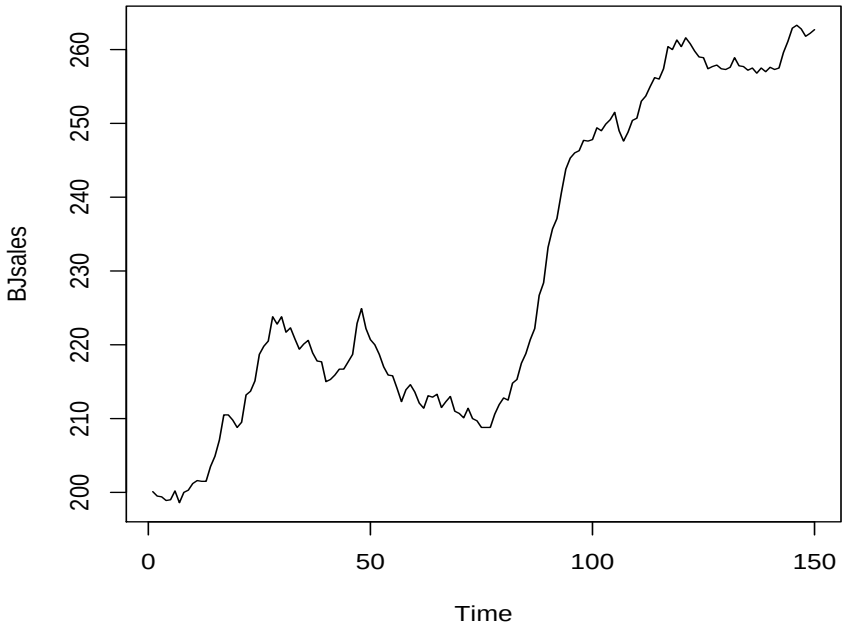

Figure 3.3: Sales of product C.

has the advantage of being very convenient and quick to apply, though as with any automatic procedure, it should not be used in a totally undiscriminating way.

If $\hat{\Gamma}_M$ is non-singular, we can solve the Yule-Walker system of equations of order m, $m = 1, 2, \ldots, M$; estimate an m-vector of unknown coefficients $\hat{\phi} = \hat{\phi}_1, \hat{\phi}_2, \ldots, \hat{\phi}_m$, and then apply AIC to find the optimal order m^* of the AR model, by comparing AIC(for $m = 1$), AIC(for $m = 2$),..., AIC(for $m = M$). To solve the Yule-Walker equations of increasing order it is possible to use the Levinson-Durbin algorithm defined in lecture 2.

Example. Sales of product C. The data are given in Box and Jenkins (1976).

In first, we plot data:

```
> ts.plot(BJsales)
```

We definitely need to difference data in order to obtain stationarity.

```
> BJ1<-diff(BJsales)
```

It is probably worth to difference one more time.

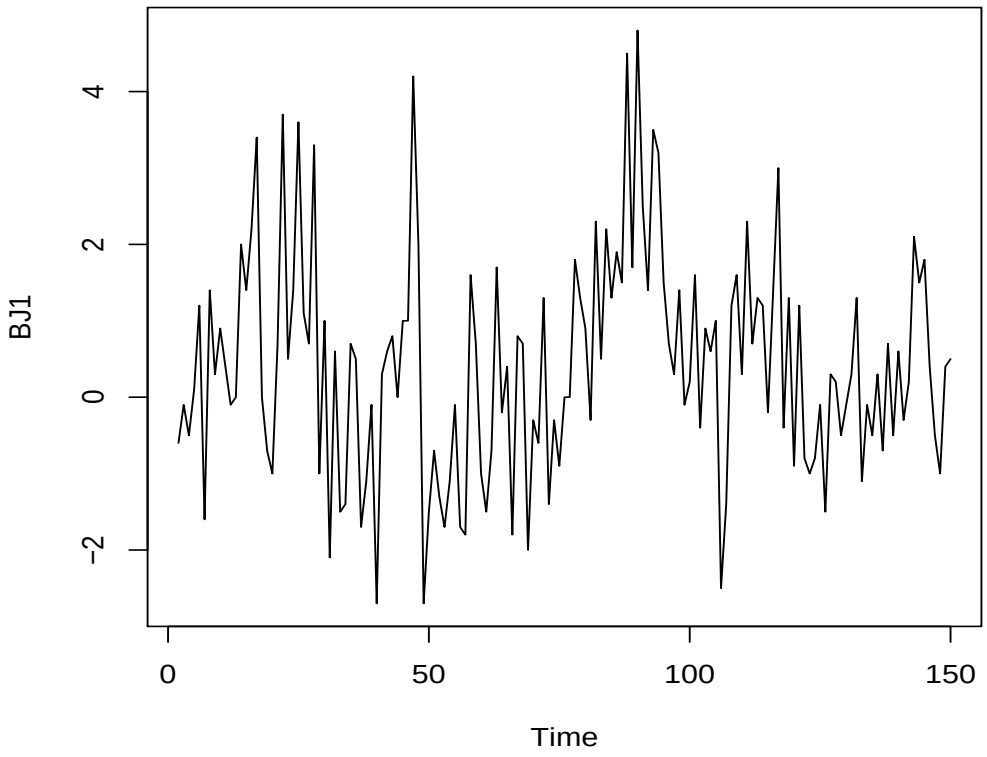

Figure 3.4: Differenced data of sales of product C.

```
> BJ2<-diff(BJ1)
```

Now check a.c.f. and p.a.c.f:

```
> acf(BJ2)
> pacf(BJ2)
```

Now we try to us an AR model for the process. The AR coefficients are estimated by the Yule-Walker method.

```
> yu<-ar.yw(BJ2, order.max=10)
> yu

Call: ar.yw.default(x = BJ2, order.max = 10)
```

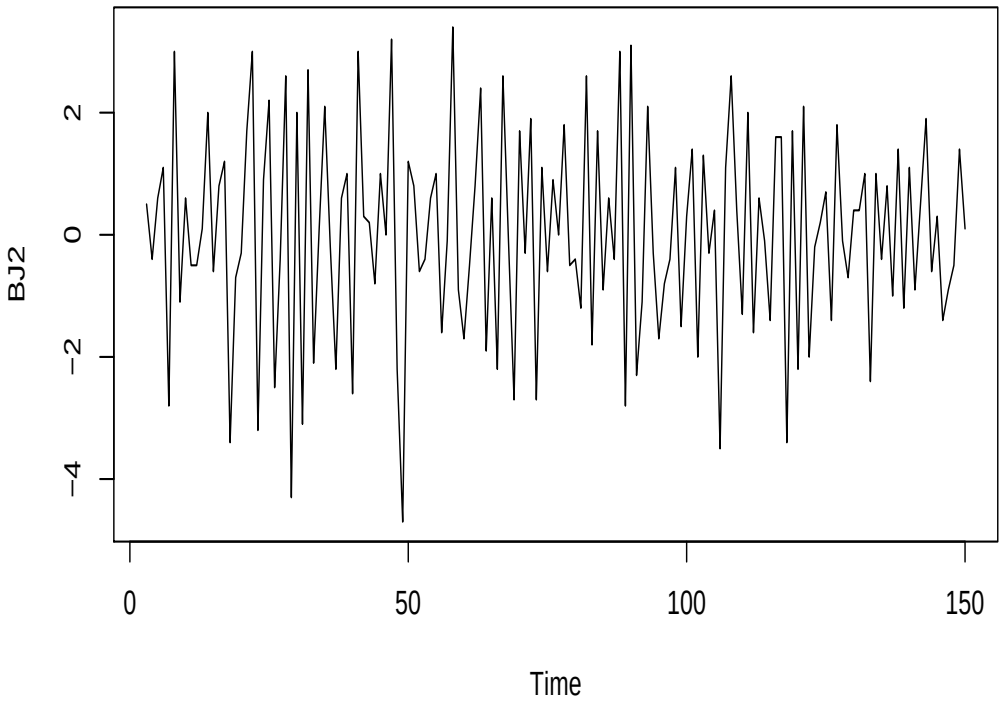

Figure 3.5: Twice differenced data of sales of product C.

```
Coefficients:
      1        2        3
-0.6758  -0.4277  -0.2500

Order selected 3  sigma^2 estimated as  1.972
> yu$aic
        0         1         2         3         4         5
53.224494 17.280068  7.552695  0.000000  1.004474  1.633639

    6         7         8         9        10
 3.522343 2.580833  4.251077  6.056177  2.019384
```

(c) **Maximum Likelihood Method**

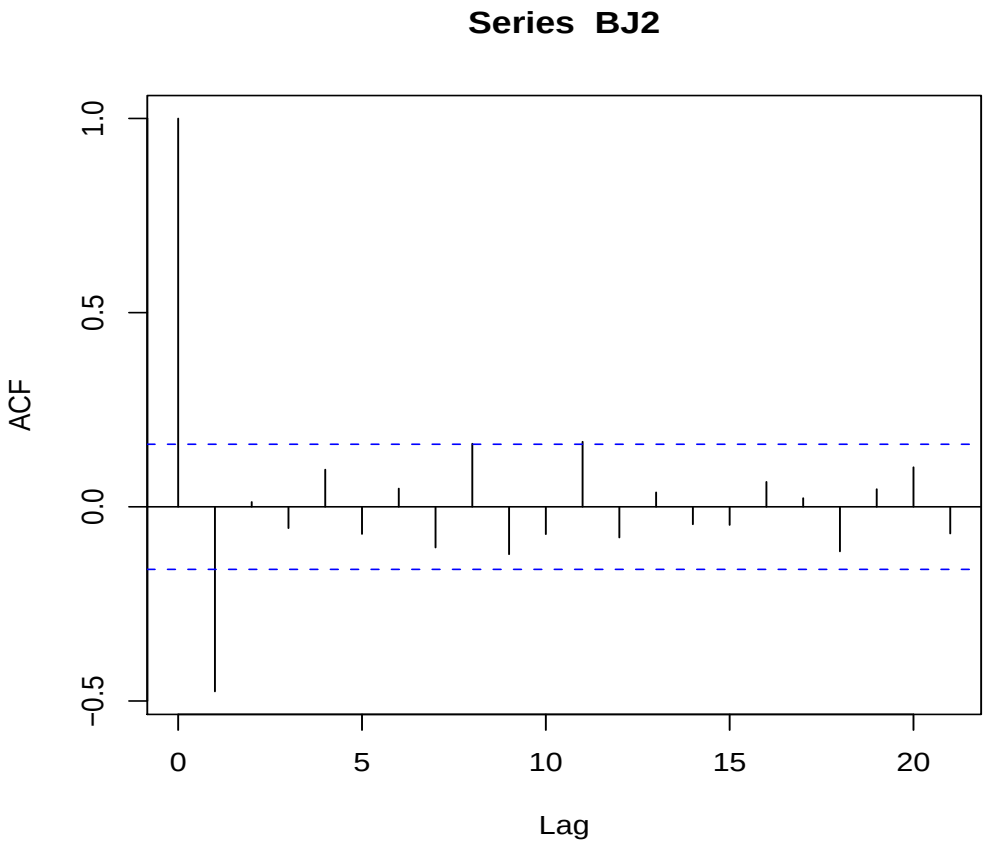

Figure 3.6: A.c.f. of sales of product C.

Now we turn to the general ARMA process. The idea here is based on numerical maximum likelihood estimation. However, most existing methods do not use exact maximum likelihood but various approximations thereto. The maximum likelihood estimator is defined as the value maximizing

$$f(X_1, \ldots, X_T; \psi)$$

over ψ, where $f(.|\psi)$ is the joint distribution of $\mathbf{X_T} = X_1, \ldots, X_T$. The likelihood function has the same functional form as the density but it is viewed as a function of the parameters for fixed data. For a given parameter ψ_0, the likelihood measures the plausibility of ψ_0 given observed values $\mathbf{X_T}$. The maximum likelihood estimate (MLE), $\hat{\psi}$, is the parameter value that maximizes the likelihood (the "most plausible" parameter value).

The ML estimators have the following properties:

The ML estimators are consistent. Let $\hat{\psi}_n$ be the MLE of ψ based on n observations. Then $\hat{\psi}_n \to \psi$ in probability as $n \to \infty$, that is, for any $\epsilon > 0$

$$\lim_{n\to\infty} P(|\hat{\psi}_n - \psi| \geq \epsilon) = 0.$$

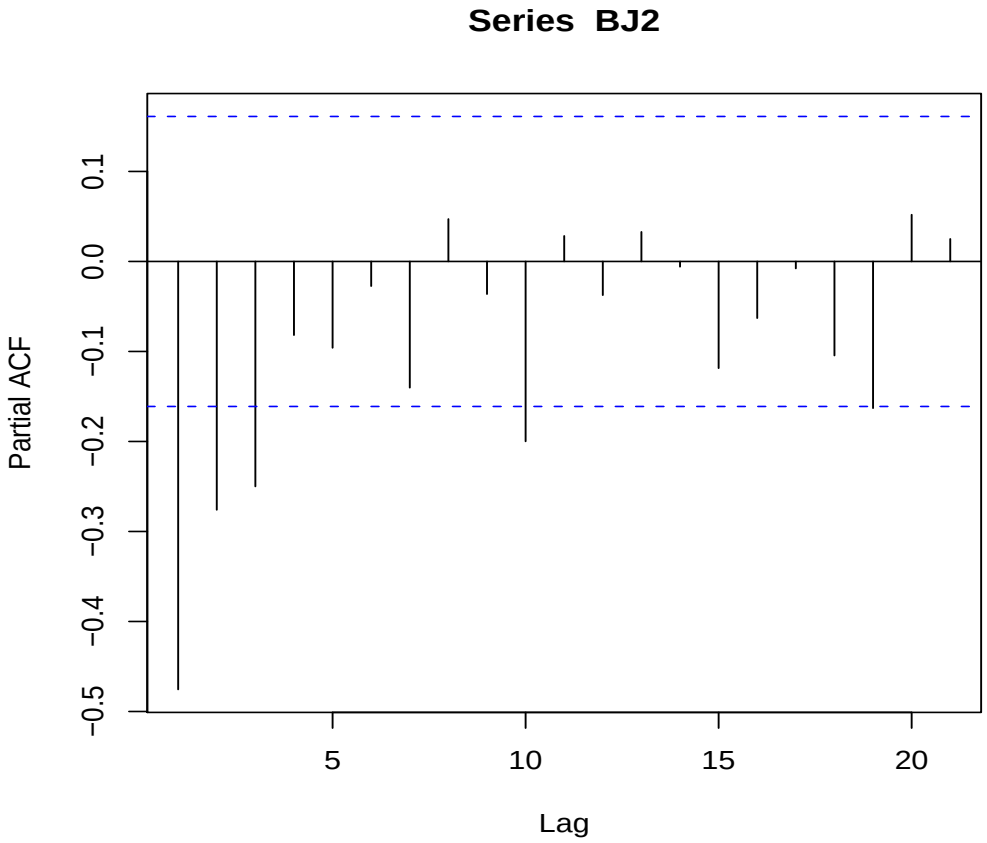

Figure 3.7: P.a.c.f. of sales of product C.

The quantity $\sqrt(n)(\hat{\psi}_n - \psi)$ is asymptotically normally distributed with mean 0 and variance given by the Cramer-Rao lower bound, i.e.

$$\sqrt(n)(\hat{\psi}_n - \psi) \to N(0, V(\psi)),$$

where $V(\psi)$ is the asymptotic covariance matrix.

If $\mathbf{X_T}$ is a Gaussian time series then the likelihood function is given by

$$\begin{aligned} & L(X_1, \ldots, X_T; \psi) \\ & = \frac{1}{(2\pi)^{t/2}} \det(\Gamma_T(\psi))^{-1/2} \exp(-\frac{1}{2}\mathbf{X_T}\Gamma_T^{-1}(\psi)\mathbf{X_T}), \end{aligned} \tag{3.6}$$

where $\Gamma_T = E\mathbf{X_T}\mathbf{X_T}$ is the $T \times T$ covariance matrix of $\mathbf{X_T}$.

The covariance matrix is a nonlinear function of the underlying parameters. Maximizing the likelihood function directly is therefore a highly nonlinear optimization problem.

For an ARMA(p, q) process with $\psi = (\phi_1, \ldots, \phi_p, \theta_1, \ldots, \theta_q)'$. The conditional "reduced" log-likelihood is given by

$$l(\psi) = \ln\left(\frac{S(\psi)}{n}\right) + \frac{1}{n}\sum_{j=1}^{n} \ln r_{j-1},$$

where $S(\psi) = \sum_{j=1}^{n} \frac{(X_j - \hat{X}_j)^2}{r_{j-1}^2}$, $E(X_j - \hat{X}_j)^2 = \sigma^2 r_{j-1}$ and $\hat{\sigma}^2 = S(\psi)/n$.

Minimization of $l(\psi)$ must be done numerically. Initial values for ϕ and θ can be obtained using the Yule-Walker, Burg's method and others.

Even if $\mathbf{X_t}$ is not Gaussian, it makes sense to regard $l(\psi)$ as a measure of the goodness of fit of the model with parameters ψ to the observed data, and still to choose the parameters ψ in such a way as to maximize L. We shall refer to the estimators $\hat{\psi}$ as "maximum likelihood" estimators even when $\mathbf{X_t}$ is not Gaussian.

```
> mle<-ar.mle(x, order.max=10)
> mle

Call: ar.mle(x = x, order.max = 10)

Coefficients:
      1        2        3
-0.6738  -0.4264  -0.2482

Order selected 3  sigma^2 estimated as  1.915

> mle$aic
        0         1         2         3    4         5
52.913967 17.198419  7.532484  0.000000 1.015224  1.6713

      6         7         8         9        10
3.579993  2.794593    4.433334  6.220627  2.165906
```

(d) **Least Squares Estimation**

The least squares estimates $\tilde{\phi}$ and $\tilde{\theta}$ of ϕ and θ are obtained by minimizing the function S rather than minimizing l, subject to the constraints that the model is causal and invertible. The least squares (LS) estimate of σ^2 is given by

$$\tilde{\sigma}^2 = \frac{S(\tilde{\phi}, \tilde{\theta})}{n - p - q}.$$

The sum $n^{-1} \sum_{j=1}^{n} \ln r_{j-1}$ is asymptotically negligible compared with $\ln(S(\psi)/n)$ if the model is invertible since $r_n \to 1$. Then minimization of S will be equivalent to minimization of l and the LSE and MLE will have similar asymptotic properties.

The LS estimates satisfy the recursive equations which enables to use the LS estimates for the on-line parameter estimation.

Consider an AR(p) model. Define vectors

$$\Phi_{t-1} = (y_{t-1}, y_{t-2}, \ldots, y_{t-p})', \qquad \tau_* = (\phi_1, \phi_2, \ldots, \phi_p)',$$

the model may be expressed in the form of a linear observation scheme

$$y_t = \Phi'_{t-1}\tau_* + \epsilon_t, \tag{3.7}$$

where ϵ is a white noise.

Then the vector of unknown parameters τ_* is estimated using the recursive LS method

$$\begin{aligned} \tau_{t+1} &= \tau_t + \gamma_t \Phi_t (y_{t+1} - \Phi'_t \tau_t), \\ \gamma_{t+1} &= \gamma_t - \gamma_t \Phi_{t+1} (1 + \Phi'_{t+1} \gamma_t \Phi_{t+1})^{-1} \Phi'_{t+1} \gamma_t \end{aligned} \tag{3.8}$$

For choosing a model, we may again define order selection criterions, and compare the values provided by different models.

```
> ols<-ar.ols(BJ2, order.max=10)
> ols

Call: ar.ols(x = BJ2, order.max = 10)

Coefficients:
      1        2        3
-0.6776  -0.4325  -0.2529

Intercept: -0.01311 (0.1160)

Order selected 3  sigma^2 estimated as  1.952


> ols$aic
        0         1         2         3       4        5
50.7106242 15.6582754  6.8298541  0.0000000 0.8483743 0.79704

  6         7          8         9        10
```

```
1.9880806  2.0365148  4.4785731  7.2961460  3.6758156
```

(c) **Order Selection**

Let the process $\mathbf{X_t}$ has a true density $f(X, \psi_0)$ and the ARMA-class has densities $f(x, \psi_1)$ for

$$\psi_1 = (\phi_1, \ldots, \phi_p, \theta_1, \ldots, \theta_q)'.$$

Then the Kullback-Leibler distance is

$$\begin{aligned} d(\psi_1|\psi_0) = & \int_{R^T} -2\ln\left(\frac{f(x,\psi_1)}{f(x,\psi_0)}\right) f(x,\psi_0)dx \\ \geq & -2\ln\int \frac{f(x,\psi_1)}{f(x,\psi_0)} f(x,\psi_0)dx \\ = & -2\ln\int f(x,\psi_1)dx = 0, \end{aligned} \tag{3.9}$$

where $d(\psi_1|\psi_0) = 0$ if and only if $f(x, \psi_0) = f(x, \psi_1)$ almost everywhere. To derive the first lower bound we use the Jensen inequality. The distance measure $d(\psi_1|\psi_0)$ can be approximated by AICC (Akaike Information Corrected Criterion)

$$\begin{aligned} & AICC(\psi_1) \\ & = -2lnL(\psi_1, S(\psi_1)/n) + 2(p+q+1)n/(n-p-q-2), \end{aligned} \tag{3.10}$$

or by the AIC (Akaike Information Criterion) statistic, defined as

$$\begin{aligned} & AIC(\psi_1) \\ & = -2lnL(\psi_1, S(\psi_1)/n) + 2(p+q+1) = ln\sigma^2 + 2(p+q)/n. \end{aligned} \tag{3.11}$$

Both AICC(ψ_1) and AIC(ψ_1) can be defined for arbitary σ^2 by replacing $S(\psi_1)/n$ by σ^2. AICC and AIC are minimized for any given ψ_1 by setting $\sigma^2 = S(\psi_1)/n$.

The BIC (Bayesian Information Criterion)

$$BIC(\psi_1) = ln\sigma^2 + (p+q)\ln n/n$$

Increasing the number in p and q reduces the value of $S(\psi_1)/n$. This comes at a cost of overparametrizing the model which is captured in the second term of AICC and AIC. It can be shown that AICC and AIC are inconsistent in the sense that they asymptotically pick up p and q too large. In contrast, the BIC is consistent. However, AICC and AIC are asymptotically efficient for AR(∞) processes while BIC is not. Efficiency is a desirable property defined in terms of the one-step mean-square prediction error achieved by the fitted model.

(d) **Examples of using MLE and LS for ARMA(p, q)**

You see from above that among the Yule-Walker method, the ML method and the LS method,

(a) the ML method provides the smallest error variance σ^2 of 1.915,

(b) the LS method provides a little bit worse error variance σ^2 of 1.952,

(c) the Yule-Walker method provides the worst σ^2 of 1.972.

Now let us fit various ARMA(p, q) models to the same data **BJ2** data, i.e. twice differenced **BJsales** data. We will use an R function **arima0** that allows to estimate parameters of AR(p), MA(q) and general ARMA(p, q) for $p, q \neq 0$.

Note that we cannot compare directly AIC from the R function **ar** and AIC from the R function **arima0** even if we use the same estimation method, e.g. ML or LS. The reason is that those R functions use different solvers and optimization routines. So we need to compare model fits from **ar** and **arima0** in terms of the error variance σ^2.

Example of fitting the same AR(3) model to twice differenced **BJsales** data using the same ML method with **ar.mle** and **arima0**.

```
> ar3mle<-ar.mle(BJ2, order.max=10)
> ar3mle

Call: ar.mle(x = BJ2, order.max = 10)

Coefficients:
      1        2        3
-0.6738  -0.4264  -0.2482

Order selected 3  sigma^2 estimated as  1.915

> ar3mle$aic
        0         1         2         3         4         5
52.913967 17.198419  7.532484  0.000000  1.015224  1.671305

3.579993  2.794593
        8         9        10
```

```
 4.433334  6.220627  2.165906


> arima0(BJ2, order = c(3, 0, 0))

Call: arima0(x = BJ2, order = c(3, 0, 0))

Coefficients:
          ar1      ar2      ar3  intercept
      -0.6738  -0.4264  -0.2482     0.0031
s.e.   0.0795   0.0902   0.0790     0.0488

sigma^2 estimated as 1.915:  log likelihood = -258.38,
aic = 526.76
```

As you see the AIC for the AR(3) model estimated by the ML method but with different R functions (**ar.mle** and **arima0**) are different, i.e. 0 and 526.76 respectively. This is because in **ar.mle** AIC is defined as a difference in AIC values between each model and the best-fitting model, and the the latter can have an AIC of $-\infty$.

Now, in first, we fit ARMA(1, 2) with 3 parameters using the ML method which is the default estimation method in **arima0**.

```
> arma12ML<-arima0(BJ2, order=c(1, 0, 2))
> arma12ML

Call: arima0(x = BJ2, order = c(1, 0, 2))

Coefficients:
          ar1      ma1      ma2  intercept
      -0.1223  -0.6097  -0.1209     0.0017
s.e.   0.0823   0.0418   0.0424     0.0277

sigma^2 estimated as 1.865:  log likelihood = -256.52,
aic = 523.04
```

Now we fit the same ARMA(1, 2) using the LS method with **arima0**.

```
> arma12LS<-arima0(BJ2, order=c(1, 0, 2), method="CSS")
> arma12LS

Call: arima0(x = BJ2, order = c(1, 0, 2), method = "CSS")

Coefficients:
        ar1     ma1     ma2  intercept
      0.836  -1.589  0.5823     0.0094
s.e.    NaN     NaN     NaN        NaN

sigma^2 estimated as 1.762:  part log likelihood = -251.92 Warning
message: NaNs produced in: sqrt(diag(x$var.coef))
```

No convergence unfortunately for the LS method and ARMA(1,2)!

Now we fit ARMA(2, 2) with 4 parameters using ML and LS.

```
##############ML method############################
> arma22ML<-arima0(BJ2, order=c(2, 0, 2))
> arma22ML

Call: arima0(x = BJ2, order = c(2, 0, 2))

Coefficients:
          ar1     ar2      ma1      ma2  intercept
      -0.6772  0.1257  -0.0463  -0.6288     0.0021
s.e.   0.0820  0.2294   0.1808   0.3328     0.0241

sigma^2 estimated as 1.850:  log likelihood = -255.99,
aic = 523.97

#################LS method#######################
> arma22LS<-arima0(BJ2, order=c(2, 0, 2), method="CSS")
> arma22LS
```

```
Call: arima0(x = BJ2, order = c(2, 0, 2), method = "CSS")

Coefficients:
          ar1     ar2      ma1      ma2  intercept
      -0.6898  0.1239  -0.0346  -0.6317     0.0028
s.e.   0.2772  0.1323   0.2637   0.1843     0.0246

sigma^2 estimated as 1.879:  part log likelihood = -256.67
```

You can fit other competing models and organize them in a table (see the Table 3.3) and then choose the best model in terms of AIC or loglikelihood or σ^2.

Table 3.1: Model fits to twice differerenced **BJ sales** data using the ML method.

Order	loglik	AIC	σ	Converged
(1,0,0)	-268.98	543.96	2.22	T
(0,0,1)	-256.56	519.13	1.87	T
(2,0,0)	-263.15	534.29	2.05	T
(0,0,2)	-256.68	521.36	1.24	T (Warn)
(1,0,1)	-256.48	520.96	1.86	T
(2,0,1)	-256.14	522.29	1.85	T(Warn)
(1,0,2)	-256.52	523.04	1.87	F
(3,0,0)	-258.38	526.76	1.92	T
(3,0,1)	-254.61	521.22	1.80	T(Warn)
(3,0,2)	-255.24	524.48	1.80	T
(3,0,3)	-254.12	524.23	1.78	T(Warn)
(1,0,3)	-255.99	523.98	1.19	F
(2,0,3)	-254.22	522.43	1.78	T

The potential candidates in terms of AIC are MA(1), ARMA(1,1). In terms of σ^2 the best candidates are ARMA(2,3), ARMA(3,2), ARMA(1,1) and MA(1). In terms of loglikelihood, the best models are ARMA(2,3), ARMA(3,2), ARMA(1,1) and MA(1).