STAT 626: Outline Lecture 219
ARCH-GARCH Models (§8.1)

1. **WN $\Rightarrow$ ARMA, ARIMA, SARIMA, ARCH/GARCH, ...**

2. **Taking Care of Time-Varying Variances:** σ_t^2

3. **Time Series Decomposition:** $x_t = \mu_t + \sigma_t \varepsilon_t,$ **Var**$(\sigma_t \varepsilon_t) = \sigma_t^2$.

4. **How to Model Time-Varying Variances?**
   Recall that Squared Residuals r_t^2 **are Reasonable "Estimates" of** σ_t^2**:**

$$r_t^2 \approx \sigma_t^2.$$

5. **Often** r_t^2**'s appear more correlated than** r_t**'s (Granger, 1970's).**

6. **AutoRegressive Conditionally Heteroscedastic (ARCH) Models:**(Engle, 1982)

$$r_t = \sigma_t \varepsilon_t,$$

$$\sigma_t^2 = \alpha_0 + \alpha_1 r_{t-1}^2.$$

   AR Models for Squared Residuals r_t^2.

   This point of view is helpful in using the ACF and PACF of the series r_t^2 to identify the orders of the ARCH(p) models.

7. **Generalized ARCH (GARCH) Models**

   ARMA Models for Squared Residuals r_t^2.

Unit-Root Test and Random Walks

8. **Random Walk vs AR(1):** $x_t = \phi x_{t-1} + w_t$,

$$H_0 : \phi = 1 \quad \text{vs} \quad H_1 : |\phi| < 1.$$

9. **Unit-Root Tests: DF, ADF, PP**.

10. **Why Unit-Root Test is Important in Economics and Finance?**

L. Bachelier Dissertation (1900).

Random Walk Hypothesis,

Efficient Market Hypothesis:

The weak form: All information about market prices is already reflected in the current stock price.

The strong form: All publicly available information about a company is already reflected in its stock price.

I. A Random Walk Down Wall Street, by Burton G. Malkiel

II. A Non-Random Walk Down Wall Street, by Andrew W. Lo & A. Craig MacKinlay

Books By Peter Bernstein:

III. Capital Ideas: The Improbable Origins of Modern Wall Street, (Free Press), 1991.

IV. Against the Gods: The Remarkable Story of Risk, (John Wiley & Son), 1996, Story of (Random Walk) Brownian Motion and how it Entered the World of Finance.

Chapter 8

Additional Topics *

In this chapter, we present special topics in the time domain. The sections may be read in any order. Each topic depends on a basic knowledge of ARMA models, forecasting and estimation, which is the material covered in Chapter 4 and Chapter 5.

8.1 GARCH Models

Various problems such as option pricing in finance have motivated the study of the *volatility*, or variability, of a time series. ARMA models were used to model the conditional mean (μ_t) of a process when the conditional variance (σ_t^2) was constant. For example, in the AR(1) model $x_t = \phi_0 + \phi_1 x_{t-1} + w_t$ we have

$$\mu_t = E(x_t \mid x_{t-1}, x_{t-2}, \dots) = \phi_0 + \phi_1 x_{t-1}$$
$$\sigma_t^2 = \text{var}(x_t \mid x_{t-1}, x_{t-2}, \dots) = \text{var}(w_t) = \sigma_w^2 .$$

In many problems, however, the assumption of a constant conditional variance is violated. Models such as the *autoregressive conditionally heteroscedastic* or ARCH model, first introduced by Engle (1982), were developed to model changes in volatility. These models were later extended to generalized ARCH, or GARCH models by Bollerslev (1986).

In these problems, we are concerned with modeling the return or growth rate of a series. Recall if x_t is the value of an asset at time t, then the return or relative gain, r_t, of the asset at time t is

$$r_t = \frac{x_t - x_{t-1}}{x_{t-1}} \approx \nabla \log(x_t) . \tag{8.1}$$

Either value, $\nabla \log(x_t)$ or $(x_t - x_{t-1})/x_{t-1}$, will be called the *return* and will be denoted by r_t.[1]

Typically, for financial series, the return r_t, has a constant conditional mean (typically $\mu_t = 0$ for assets), but does not have a constant conditional variance, and highly volatile periods tend to be clustered together. In addition, the autocorrelation

[1] Although it is a misnomer, $\nabla \log x_t$ is often called the *log-return*; but the returns are NOT being logged.

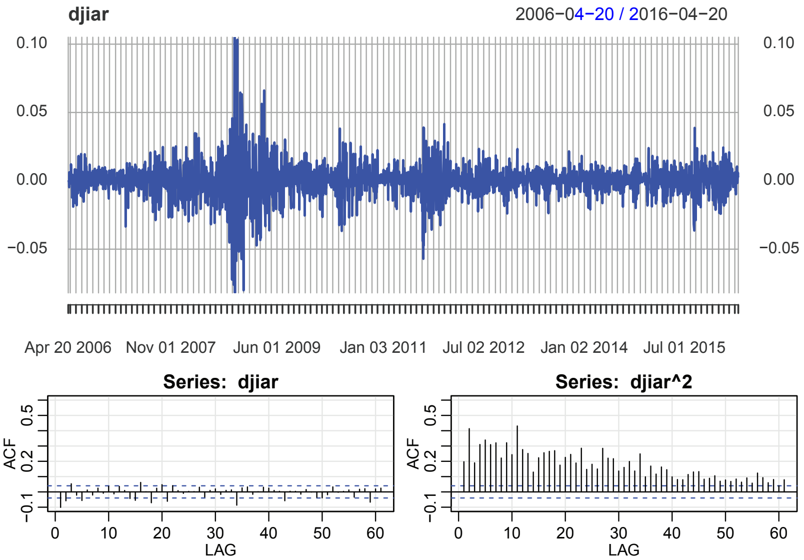

Figure 8.1 *DJIA daily closing returns and the sample ACF of the returns and of the squared returns.*

structure of r_t is that of white noise, while the returns are dependent. This can often be seen by looking at the sample ACF of the squared-returns (or some power transformation of the returns). For example, Figure 8.1 shows the daily returns of the Dow Jones Industrial Average (DJIA) that we saw in Chapter 1. In this case, as is typical, the return r_t is fairly constant (with $\mu_t = 0$) and nearly white noise, but there are short-term bursts of high volatility and the squared returns are autocorrelated.

The simplest ARCH model, the ARCH(1), models the returns as

$$r_t = \sigma_t \epsilon_t \tag{8.2}$$

$$\sigma_t^2 = \alpha_0 + \alpha_1 r_{t-1}^2, \tag{8.3}$$

where ϵ_t is standard Gaussian white noise, $\epsilon_t \sim \text{iid } \mathrm{N}(0,1)$. The normal assumption may be relaxed; we will discuss this later. As with ARMA models, we must impose some constraints on the model parameters to obtain desirable properties. An obvious constraint is that $\alpha_0, \alpha_1 \geq 0$ because σ_t^2 is a variance.

It is possible to write the ARCH(1) model as a non-Gaussian AR(1) model in the square of the returns r_t^2. First, rewrite (8.2)–(8.3) as

$$\begin{aligned} r_t^2 &= \sigma_t^2 \epsilon_t^2 \\ \alpha_0 + \alpha_1 r_{t-1}^2 &= \sigma_t^2, \end{aligned}$$

by squaring (8.2). Now subtract the two equations to obtain

$$r_t^2 - (\alpha_0 + \alpha_1 r_{t-1}^2) = \sigma_t^2 \epsilon_t^2 - \sigma_t^2\,,$$

and rearrange it as

$$r_t^2 = \alpha_0 + \alpha_1 r_{t-1}^2 + v_t, \tag{8.4}$$

where $v_t = \sigma_t^2(\epsilon_t^2 - 1)$. Because ϵ_t^2 is the square of a $\mathrm{N}(0,1)$ random variable, $\epsilon_t^2 - 1$ is a shifted (to have mean-zero), χ_1^2 random variable. In this case, v_t is non-normal white noise (see Section D.3 for details).

Thus, if $0 \le \alpha_1 < 1$, r_t^2 is a non-normal AR(1). This means that the ACF of the squared process is

$$\rho_{r^2}(h) = a_1^h \quad \text{for } h \ge 0\,.$$

In addition, it is shown in Section D.3 that, unconditionally, r_t is white noise with mean 0 and variance

$$\mathrm{var}(r_t) = \frac{\alpha_0}{1-\alpha_1}\,,$$

but conditionally,

$$r_t \bigm| r_{t-1} \sim \mathrm{N}(0, \alpha_0 + \alpha_1 r_{t-1}^2). \tag{8.5}$$

Hence, the model characterizes what we see in Figure 8.1:

- The returns are white noise.
- The conditional variance of a return depends on the previous return.
- The squared returns are autocorrelated.

Estimation of the parameters α_0 and α_1 of the ARCH(1) model is typically accomplished by conditional MLE based on the normal density specified in (8.5). This leads to weighted conditional least squares, which finds the values of α_0 and α_1 that minimize

$$S(\alpha_0, \alpha_1) = \tfrac{1}{2}\sum_{t=2}^{n} \ln(\alpha_0 + \alpha_1 r_{t-1}^2) + \tfrac{1}{2}\sum_{t=2}^{n}\left(\frac{r_t^2}{\alpha_0 + \alpha_1 r_{t-1}^2}\right), \tag{8.6}$$

using numerical methods, as described in Section 4.3.

The ARCH(1) model can be extended to the general ARCH(p) model in an obvious way. That is, (8.2), $r_t = \sigma_t \epsilon_t$, is retained, but (8.3) is extended to

$$\sigma_t^2 = \alpha_0 + \alpha_1 r_{t-1}^2 + \cdots + \alpha_p r_{t-p}^2. \tag{8.7}$$

Estimation for ARCH(p) also follows in an obvious way from the discussion of estimation for ARCH(1) models.

It is also possible to combine a regression or an ARMA model for the conditional mean, say

$$r_t = \mu_t + \sigma_t \epsilon_t\,, \tag{8.8}$$

where, for example, a simple AR-ARCH model would have

$$\mu_t = \phi_0 + \phi_1 r_{t-1}.$$

Of course the model can be generalized to have various types of behavior for μ_t.

To fit ARMA-ARCH models, simply follow these two steps:

1. First, look at the P/ACF of the *returns*, r_t, and identify an ARMA structure, if any. There is typically either no autocorrelation or very small autocorrelation and often a low order AR or MA will suffice if needed. Estimate μ_t in order to center the returns if necessary.
2. Look at the P/ACF of the *centered squared returns*, $(r_t - \hat{\mu}_t)^2$, and decide on an ARCH model. If the P/ACF indicate an AR structure (i.e., ACF tails off, PACF cuts off), then fit an ARCH. If the P/ACF indicate an ARMA structure (i.e., both tail off), use the approach discussed after the next example.

Example 8.1. Analysis of U.S. GNP

In Example 5.6, we fit an AR(1) model to the U.S. GNP series and we concluded that the residuals appeared to behave like a white noise process. Hence, we would propose that $\mu_t = \phi_0 + \phi_1 r_{t-1}$ where r_t is the quarterly growth rate in U.S. GNP.

It has been suggested that the GNP series has ARCH errors, and in this example, we will investigate this claim. If the GNP noise term is ARCH, the squares of the residuals from the fit should behave like a non-Gaussian AR(1) process, as pointed out in (8.4). Figure 8.2 shows the ACF and PACF of the squared residuals and it appears that there may be some dependence, albeit small, left in the residuals. The figure was generated in R as follows.

```
res = resid( sarima(diff(log(gnp)), 1,0,0, details=FALSE)$fit )
acf2(res^2, 20)
```

We used the R package `fGarch` to fit an AR(1)-ARCH(1) model to the U.S. GNP returns with the following results. A partial output is shown; we note that `garch(1,0)` specifies an ARCH(1) in the code below (details later).

```
library(fGarch)
gnpr = diff(log(gnp))
summary( garchFit(~arma(1,0) + garch(1,0), data = gnpr) )
           Estimate   Std.Error  t.value  Pr(>|t|) <- 2-sided !!!
   mu         0.005      0.001     5.867    0.000
   ar1        0.367      0.075     4.878    0.000
   omega      0.000      0.000     8.135    0.000 <- these parameters
   alpha1     0.194      0.096     2.035    0.042 <- can't be negative

   Standardised Residuals Tests:  Statistic p-Value
    Jarque-Bera Test   R    Chi^2      9.118   0.010
    Shapiro-Wilk Test  R    W          0.984   0.014
    Ljung-Box Test     R    Q(20)     23.414   0.269
    Ljung-Box Test     R^2  Q(20)     37.743   0.010
```

Note that the given p-values are two-sided, so they should be halved when considering the ARCH parameters. In this example, we obtain $\hat{\phi}_0 = .005$ (called `mu` in the output) and $\hat{\phi}_1 = .367$ (called `ar1`) for the AR(1) parameter estimates; in Example 5.6 the values were .005 and .347, respectively. The ARCH(1) parameter

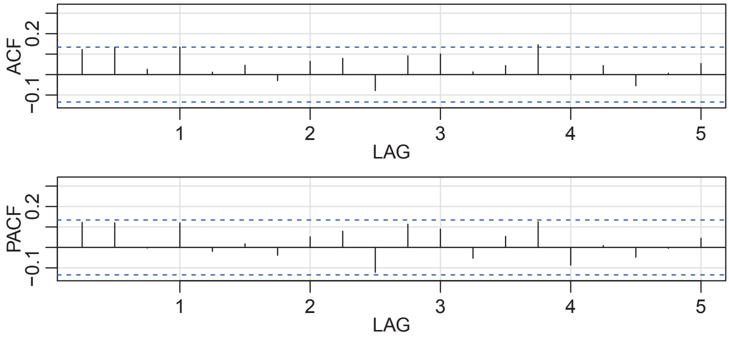

Figure 8.2 *ACF and PACF of the squares of the residuals from the AR(1) fit on U.S. GNP.*

estimates are $\hat{\alpha}_0 = 0$ (called `omega`) for the constant and $\hat{\alpha}_1 = .194$, which is significant with a p-value of about .02. There are a number of tests that are performed on the residuals [`R`] or the squared residuals [`R^2`]. For example, the Jarque–Bera statistic tests the residuals of the fit for normality based on the observed skewness and kurtosis, and it appears that the residuals have some non-normal skewness and kurtosis. The Shapiro–Wilk statistic tests the residuals of the fit for normality based on the empirical order statistics. The other tests, primarily based on the Q-statistic, are used on the residuals and their squares. ◊

The analysis of Example 8.1 had a few problems. First, it appears that the residuals are not normal (which was the assumption for the ϵ_t, and there may be some autocorrelation left in the squared residuals; see Problem 8.2). To address this kind of problem, the ARCH model was extended to generalized ARCH or GARCH. For example, a GARCH$(1,1)$ model retains (8.8), $r_t = \mu_t + \sigma_t \epsilon_t$, but extends (8.3) as follows:

$$\sigma_t^2 = \alpha_0 + \alpha_1 r_{t-1}^2 + \beta_1 \sigma_{t-1}^2. \tag{8.9}$$

Under the condition that $\alpha_1 + \beta_1 < 1$, using similar manipulations as in (8.4), the GARCH$(1,1)$ model, (8.2) and (8.9), admits a non-Gaussian ARMA$(1,1)$ model for the squared process

$$r_t^2 = \alpha_0 + (\alpha_1 + \beta_1) r_{t-1}^2 + v_t - \beta_1 v_{t-1}, \tag{8.10}$$

where we have set $\mu_t = 0$ for ease, and where v_t is as defined in (8.4). Representation (8.10) follows by writing (8.2) as

$$\begin{aligned} r_t^2 - \sigma_t^2 &= \sigma_t^2(\epsilon_t^2 - 1) \\ \beta_1(r_{t-1}^2 - \sigma_{t-1}^2) &= \beta_1 \sigma_{t-1}^2(\epsilon_{t-1}^2 - 1), \end{aligned}$$

subtracting the second equation from the first, and using the fact that, from (8.9), $\sigma_t^2 - \beta_1 \sigma_{t-1}^2 = \alpha_0 + \alpha_1 r_{t-1}^2$, on the left-hand side of the result. The GARCH(p,q)

model retains (8.8) and extends (8.9) to

$$\sigma_t^2 = \alpha_0 + \sum_{j=1}^{p} \alpha_j r_{t-j}^2 + \sum_{j=1}^{q} \beta_j \sigma_{t-j}^2. \tag{8.11}$$

Estimation of the model parameters is similar to the estimation of ARCH parameters. We explore these concepts in the following example.

Example 8.2. GARCH Analysis of the DJIA Returns

As previously mentioned, the daily returns of the DJIA shown in Figure 8.1 exhibit classic GARCH features. In addition, there is some low level autocorrelation in the series itself, and to include this behavior, we used the R `fGarch` package to fit an AR(1)-GARCH$(1,1)$ model to the series using t-errors (rather than normal):

```
library(xts)
djiar = diff(log(djia$Close))[-1]
acf2(djiar)     #  exhibits some autocorrelation - see Figure 8.1
u = resid( sarima(djiar, 1,0,0, details=FALSE)$fit )
acf2(u^2)       #  oozes autocorrelation - see Figure 8.1
library(fGarch)
summary(djia.g <- garchFit(~arma(1,0)+garch(1,1), data=djiar,
           cond.dist="std"))
            Estimate   Std.Error  t.value    Pr(>|t|)
   mu      8.585e-04   1.470e-04    5.842   5.16e-09
   ar1    -5.531e-02   2.023e-02   -2.735   0.006239
   omega   1.610e-06   4.459e-07    3.611   0.000305
   alpha1  1.244e-01   1.660e-02    7.497   6.55e-14
   beta1   8.700e-01   1.526e-02   57.022   < 2e-16
   shape   5.979e+00   7.917e-01    7.552   4.31e-14
   ---
   Standardised Residuals Tests:
                                 Statistic    p-Value
    Ljung-Box Test     R    Q(10)  16.81507  0.0785575
    Ljung-Box Test     R^2  Q(10)  15.39137  0.1184312
plot(djia.g, which=3) # similar to Figure 8.3
```

The `shape` parameter is the degrees of freedom for the t error distribution, which is estimated to be about 6. Also notice that $\hat{\alpha}_1 + \hat{\beta}_1$ is close to 1; this is often the case. To explore the GARCH predictions of volatility, we calculated and plotted part of the data surrounding the financial crises of 2008 along with the one-step-ahead predictions of the corresponding volatility, σ_t^2 as a solid line in Figure 8.3. ◊

Another model that we mention briefly is the *asymmetric power ARCH* model. The model retains (8.2), $r_t = \sigma_t \epsilon_t$, but the conditional variance is modeled as

$$\sigma_t^\delta = \alpha_0 + \sum_{j=1}^{p} \alpha_j (|r_{t-j}| - \gamma_j r_{t-j})^\delta + \sum_{j=1}^{q} \beta_j \sigma_{t-j}^\delta . \tag{8.12}$$

Note that the model is GARCH when $\delta = 2$ and $\gamma_j = 0$, for $j \in \{1, \dots, p\}$.

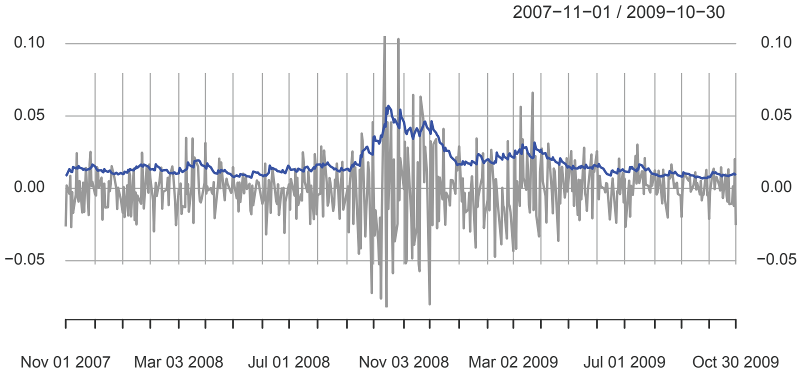

Figure 8.3 *GARCH one-step-ahead predictions of the DJIA volatility, σ_t, superimposed on part of the data including the financial crisis of 2008.*

The parameters γ_j ($|\gamma_j| \leq 1$) are the *leverage* parameters, which are a measure of asymmetry, and $\delta > 0$ is the parameter for the power term. A positive [negative] value of γ_j's means that past negative [positive] shocks have a deeper impact on current conditional volatility than past positive [negative] shocks. This model couples the flexibility of a varying exponent with the asymmetry coefficient to take the *leverage effect* into account. Further, to guarantee that $\sigma_t > 0$, we assume that $\alpha_0 > 0$, $\alpha_j \geq 0$ with at least one $\alpha_j > 0$, and $\beta_j \geq 0$.

We continue the analysis of the DJIA returns in the following example.

Example 8.3. APARCH Analysis of the DJIA Returns

The R package `fGarch` was used to fit an AR-APARCH model to the DJIA returns discussed in Example 8.2. As in the previous example, we include an AR(1) in the model to account for the conditional mean. In this case, we may think of the model as $r_t = \mu_t + y_t$ where μ_t is an AR(1), and y_t is APARCH noise with conditional variance modeled as (8.12) with t-errors. A partial output of the analysis is given below. We do not include displays, but we show how to obtain them. The predicted volatility is, of course, different than the values shown in Figure 8.3, but appear similar when graphed.

```
lapply( c("xts", "fGarch"), library, char=TRUE) # load 2 packages
djiar = diff(log(djia$Close))[-1]
summary(djia.ap <- garchFit(~arma(1,0)+aparch(1,1), data=djiar,
           cond.dist="std"))
plot(djia.ap)   # to see all plot options (none shown)
            Estimate  Std. Error  t value   Pr(>|t|)
  mu       5.234e-04   1.525e-04    3.432  0.000598
  ar1     -4.818e-02   1.934e-02   -2.491  0.012727
  omega    1.798e-04   3.443e-05    5.222  1.77e-07
  alpha1   9.809e-02   1.030e-02    9.525   < 2e-16
  gamma1   1.000e+00   1.045e-02   95.731   < 2e-16
```

```
beta1   8.945e-01   1.049e-02   85.280   < 2e-16
delta   1.070e+00   1.350e-01    7.928  2.22e-15
shape   7.286e+00   1.123e+00    6.489  8.61e-11
---
Standardised Residuals Tests:
                                Statistic p-Value
 Ljung-Box Test     R    Q(10)  15.71403  0.108116
 Ljung-Box Test     R^2  Q(10)  16.87473  0.077182
```

◊

In most applications, the distribution of the noise, ϵ_t in (8.2), is rarely normal. The R package `fGarch` allows for various distributions to be fit to the data; see the help file for information. Some drawbacks of GARCH and related models are as follows. (i) The GARCH model assumes positive and negative returns have the same effect because volatility depends on squared returns; the asymmetric models help alleviate this problem. (ii) These models are often restrictive because of the tight constraints on the model parameters. (iii) The likelihood is flat unless n is very large. (iv) The models tend to overpredict volatility because they respond slowly to large isolated returns.

Various extensions to the original model have been proposed to overcome some of the shortcomings we have just mentioned. For example, we have already discussed the fact that `fGarch` allows for asymmetric return dynamics. In the case of persistence in volatility, the integrated GARCH (IGARCH) model may be used. Recall (8.10) where we showed the GARCH$(1,1)$ model can be written as

$$r_t^2 = \alpha_0 + (\alpha_1 + \beta_1) r_{t-1}^2 + v_t - \beta_1 v_{t-1}$$

and r_t^2 is stationary if $\alpha_1 + \beta_1 < 1$. The IGARCH model sets $\alpha_1 + \beta_1 = 1$, in which case the IGARCH$(1,1)$ model is

$$r_t = \sigma_t \epsilon_t \quad \text{and} \quad \sigma_t^2 = \alpha_0 + (1 - \beta_1) r_{t-1}^2 + \beta_1 \sigma_{t-1}^2.$$

There are many different extensions to the basic ARCH model that were developed to handle the various situations noticed in practice. Interested readers might find the general discussions in Bollerslev et al. (1994) and Shephard (1996) worthwhile reading. Two excellent texts on financial time series analysis are Chan (2002) and Tsay (2005).

8.2 Unit Root Testing

The use of the first difference $\nabla x_t = (1 - B) x_t$ can sometimes be too severe a modification in the sense that an integrated model might represent an overdifferencing of the original process. For example, in Example 5.8 we fit an ARIMA(1,1,1) model to the logged varve series. The idea of differencing the series was first made in Example 4.27 because the series appeared to take long 100+ year walks in positive and negative directions.

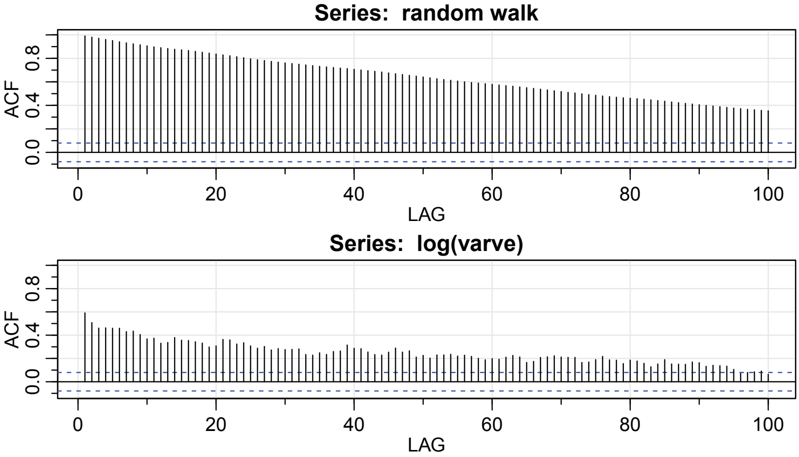

Figure 8.4 *Sample ACFs a random walk and of the log transformed varve series.*

Figure 8.4 compares the sample ACF of a generated random walk with that of the logged varve series. Although in both cases the sample correlations decrease linearly and remain significant for many lags, the sample ACF of the random walk has much larger values. (Recall that there is no ACF in terms of lag only for a random walk. But that doesn't stop us from computing one.)

```
layout(1:2)
acf1(cumsum(rnorm(634)), 100, main="Series:  random walk")
acf1(log(varve), 100, ylim=c(-.1,1))
```

Consider the normal AR(1) process,

$$x_t = \phi x_{t-1} + w_t. \tag{8.13}$$

A unit root test provides a way to test whether (8.13) is a random walk (the null case) as opposed to a causal process (the alternative). That is, it provides a procedure for testing

$$H_0: \phi = 1 \quad \text{versus} \quad H_1: |\phi| < 1.$$

To see if the null hypothesis is reasonable, an obvious test statistic would be to consider $(\widehat{\phi} - 1)$, appropriately normalized, in the hope to develop a t-test, where $\widehat{\phi}$ is one of the optimal estimators discussed in Section 4.3. Note that the distribution in Property 4.29 does not work in this case; if it did, under the null hypothesis, $\widehat{\phi} \mathrel{\dot\sim} N(1, 0)$, which is nonsense. The theory of Section 4.3 does not work in the null case because the process is not stationary under the null hypothesis.

However, the test statistic

$$T = n(\widehat{\phi} - 1)$$

can be used, and it is known as the unit root or Dickey–Fuller (DF) statistic, although the actual DF test statistic is normalized a little differently. In this case, the distribution of the test statistic does not have a closed form and quantiles of the distribution must be computed by numerical approximation or by simulation. The R package tseries provides this test along with more general tests that we mention briefly.

Toward a more general model, we note that the DF test was established by noting that if $x_t = \phi x_{t-1} + w_t$, then

$$\nabla x_t = (\phi - 1)x_{t-1} + w_t = \gamma x_{t-1} + w_t \,,$$

and one could test $H_0\colon \gamma = 0$ by regressing ∇x_t on x_{t-1} and obtaining the regression coefficient estimate $\widehat{\gamma}$. Then, the statistic $n\widehat{\gamma}$ was formed and its large sample distribution derived.

The test was extended to accommodate AR(p) models, $x_t = \sum_{j=1}^{p} \phi_j x_{t-j} + w_t$, in a similar way. For example, write an AR(2) model

$$x_t = \phi_1 x_{t-1} + \phi_2 x_{t-2} + w_t \,,$$

as

$$x_t = (\phi_1 + \phi_2)x_{t-1} - \phi_2(x_{t-1} - x_{t-2}) + w_t \,,$$

and subtract x_{t-1} from both sides. This yields

$$\nabla x_t = \gamma x_{t-1} + \phi_2 \nabla x_{t-1} + w_t, \tag{8.14}$$

where $\gamma = \phi_1 + \phi_2 - 1$. To test the hypothesis that the process has a unit root at 1 (i.e., the AR polynoimial $\phi(z) = 1 - \phi_1 z - \phi_2 z^2 = 0$ when $z = 1$), we can test $H_0\colon \gamma = 0$ by estimating γ in the regression of ∇x_t on x_{t-1} and ∇x_{t-1} and forming a test statistic. For AR(p) model, one regresses ∇x_t on x_{t-1} and $\nabla x_{t-1} \ldots, \nabla x_{t-p+1}$, in a similar fashion to the AR(2) case.

This test leads to the so-called augmented Dickey–Fuller test (ADF). While the calculations for obtaining the large sample null distribution change, the basic ideas and machinery remain the same as in the simple case. The choice of p is crucial, and we will discuss some suggestions in the example. For ARMA(p, q) models, the ADF test can be used by assuming p is large enough to capture the essential correlation structure; recall ARMA(p, q) models are AR(∞) models. An alternative is the Phillips–Perron (PP) test, which differs from the ADF tests mainly in how it deals with serial correlation and heteroscedasticity in the errors.

Example 8.4. Testing Unit Roots in the Glacial Varve Series

In this example we use the R package tseries to test the null hypothesis that the log of the glacial varve series has a unit root, versus the alternate hypothesis that the process is stationary. We test the null hypothesis using the available DF, ADF, and PP tests; note that in each case, the general regression equation incorporates a constant and a linear trend. In the ADF test, the default number of AR components included in the model is $k \approx (n-1)^{\frac{1}{3}}$, which has theoretical justification on how k should grow compared to the sample size n. For the PP test, the default value is $k \approx .04n^{\frac{1}{4}}$.

```
library(tseries)
adf.test(log(varve), k=0)                 # DF test
  Dickey-Fuller = -12.8572, Lag order = 0, p-value < 0.01
   alternative hypothesis: stationary
adf.test(log(varve))                      # ADF test
  Dickey-Fuller = -3.5166, Lag order = 8, p-value = 0.04071
   alternative hypothesis: stationary
pp.test(log(varve))                       # PP test
   Dickey-Fuller Z(alpha) = -304.5376,
    Truncation lag parameter = 6, p-value < 0.01
    alternative hypothesis: stationary
```

In each test, we reject the null hypothesis that the logged varve series has a unit root. The conclusion of these tests supports the conclusion of Example 8.5 in Section 8.3, where it is postulated that the logged varve series is long memory. Fitting a long memory model to these data would be the natural progression of model fitting once the unit root test hypothesis is rejected. ◊

8.3 Long Memory and Fractional Differencing

The conventional ARMA(p, q) process is often referred to as a short-memory process because the coefficients in the representation

$$x_t = \sum_{j=0}^{\infty} \psi_j w_{t-j},$$

are dominated by exponential decay where $\sum_{j=0}^{\infty} |\psi_j| < \infty$ (e.g., recall Example 4.3). This result implies the ACF of the short memory process $\rho(h) \to 0$ exponentially fast as $h \to \infty$. When the sample ACF of a time series decays slowly, the advice given in Chapter 6 has been to difference the series until it seems stationary. Following this advice with the glacial varve series first presented in Example 4.27 leads to the first difference of the logarithms of the data, say $x_t =$`log(varve)`, being represented as a first-order moving average. In Example 5.8, further analysis of the residuals leads to fitting an ARIMA$(1, 1, 1)$ model, where the estimates of the parameters (and the standard errors) were $\widehat{\phi} = .23_{(.05)}$, $\widehat{\theta} = -.89_{(.03)}$, and $\widehat{\sigma}^2_w = .23$:

$$\nabla \hat{x}_t = .23 \nabla \hat{x}_{t-1} + \hat{w}_t - .89 \hat{w}_{t-1}.$$

What the fitted model is saying is that the series itself, x_t, is not stationary and has random walk behavior, and the only way to make it stationary is to difference it. In terms of the actual logged varve series, the fitted model is

$$\hat{x}_t = (1 + .23)\hat{x}_{t-1} - .23\hat{x}_{t-2} + \hat{w}_t - .89\hat{w}_{t-1}$$

and there is no causal representation for the data because the ψ-weights are not square summable (in fact, they do not even go to zero):

```
round(ARMAtoMA(ar=c(1.23,-.23), ma=c(1,-.89), 20), 3)
  [1] 2.230 1.623 1.483 1.451 1.444 1.442 1.442 1.442 1.442 1.442
 [11] 1.442 1.442 1.442 1.442 1.442 1.442 1.442 1.442 1.442 1.442
```

But the use of the first difference $\nabla x_t = (1 - B)x_t$ can be too severe of a transformation. For example, if x_t is a causal AR(1), say

$$x_t = .9x_{t-1} + w_t\,,$$

then shifting back one unit of time,

$$x_{t-1} = .9x_{t-2} + w_{t-1}\,.$$

Now subtract the two to get,

$$x_t - x_{t-1} = .9(x_{t-1} - x_{t-2}) + w_t - w_{t-1}\,,$$

or

$$\nabla x_t = .9\nabla x_{t-1} + w_t - w_{t-1}\,.$$

This means that ∇x_t is a problematic ARMA$(1,1)$ because the moving average part is non-invertible. Thus, by overdifferencing in this example, we have gone from x_t being a simple causal AR(1) to x_t being a non-invertible ARIMA$(1,1,1)$. This is precisely why we gave several warnings about the overuse of differencing in Chapter 4 and Chapter 5.

Long memory time series were considered in Hosking (1981) and Granger and Joyeux (1980) as intermediate compromises between the short memory ARMA type models and the fully integrated nonstationary processes in the Box–Jenkins class. The easiest way to generate a long memory series is to think of using the difference operator $(1 - B)^d$ for fractional values of d, say, $0 < d < .5$, so a basic long memory series gets generated as

$$(1 - B)^d x_t = w_t, \tag{8.15}$$

where w_t still denotes white noise with variance σ_w^2. The fractionally differenced series (8.15), for $|d| < .5$, is often called *fractional noise* (except when d is zero). Now, d becomes a parameter to be estimated along with σ_w^2. Differencing the original process, as in the Box–Jenkins approach, may be thought of as simply assigning a value of $d = 1$. This idea has been extended to the class of fractionally integrated ARMA, or ARFIMA models, where $-.5 < d < .5$; when d is negative, the term antipersistent is used. Long memory processes occur in hydrology (see Hurst, 1951, McLeod and Hipel, 1978) and in environmental series, such as the varve data we have previously analyzed, to mention a few examples. Long memory time series data tend to exhibit sample autocorrelations that are not necessarily large (as in the case of $d = 1$), but persist for a long time. Figure 8.4 shows the sample ACF, to lag 100, of the log-transformed varve series, which exhibits classic long memory behavior.

To investigate its properties, we can use the binomial expansion[2] $(d > -1)$ to write

$$w_t = (1-B)^d x_t = \sum_{j=0}^{\infty} \pi_j B^j x_t = \sum_{j=0}^{\infty} \pi_j x_{t-j} \tag{8.16}$$

where

$$\pi_j = \frac{\Gamma(j-d)}{\Gamma(j+1)\Gamma(-d)} \tag{8.17}$$

with $\Gamma(x+1) = x\Gamma(x)$ being the gamma function. Similarly $(d < 1)$, we can write

$$x_t = (1-B)^{-d} w_t = \sum_{j=0}^{\infty} \psi_j B^j w_t = \sum_{j=0}^{\infty} \psi_j w_{t-j} \tag{8.18}$$

where

$$\psi_j = \frac{\Gamma(j+d)}{\Gamma(j+1)\Gamma(d)}. \tag{8.19}$$

When $|d| < .5$, the processes (8.16) and (8.18) are well-defined stationary processes (see Brockwell and Davis, 2013, for details). In the case of fractional differencing, however, the coefficients satisfy $\sum \pi_j^2 < \infty$ and $\sum \psi_j^2 < \infty$ as opposed to the absolute summability of the coefficients in ARMA processes.

Using the representation (8.18)–(8.19), and after some nontrivial manipulations, it can be shown that the ACF of x_t is

$$\rho(h) = \frac{\Gamma(h+d)\Gamma(1-d)}{\Gamma(h-d+1)\Gamma(d)} \sim h^{2d-1} \tag{8.20}$$

for large h. From this we see that for $0 < d < .5$

$$\sum_{h=-\infty}^{\infty} |\rho(h)| = \infty$$

and hence the term *long memory*.

In order to examine a series such as the varve series for a possible long memory pattern, it is convenient to look at ways of estimating d. Using (8.17) it is easy to derive the recursions

$$\pi_{j+1}(d) = \frac{(j-d)\pi_j(d)}{(j+1)}, \tag{8.21}$$

for $j = 0, 1, \ldots$, with $\pi_0(d) = 1$. In the normal case, we may estimate d by minimizing the sum of squared errors

$$Q(d) = \sum w_t^2(d)\,.$$

The usual Gauss–Newton method, described in Section 4.3, leads to the expansion

$$w_t(d) \approx w_t(d_0) + w_t'(d_0)(d - d_0),$$

[2] The binomial expansion in this case is the Taylor series about $z = 0$ for functions of the form $(1-z)^d$

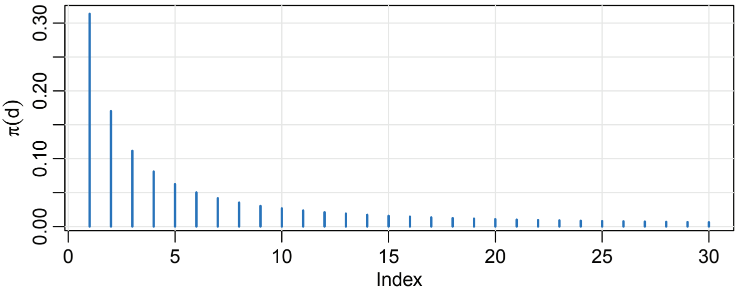

Figure 8.5 *Coefficients* $\pi_j(.373)$, $j = 1, 2, \ldots, 30$ *in the representation* (8.21).

where

$$w_t'(d_0) = \left.\frac{\partial w_t}{\partial d}\right|_{d=d_0}$$

and d_0 is an initial estimate (guess) at to the value of d. Setting up the usual regression leads to

$$d = d_0 - \frac{\sum_t w_t'(d_0) w_t(d_0)}{\sum_t w_t'(d_0)^2}. \tag{8.22}$$

The derivatives are computed recursively by differentiating (8.21) successively with respect to d: $\pi_{j+1}'(d) = [(j-d)\pi_j'(d) - \pi_j(d)]/(j+1)$, where $\pi_0'(d) = 0$. The errors are computed from an approximation to (8.16), namely,

$$w_t(d) = \sum_{j=0}^{t} \pi_j(d) x_{t-j}. \tag{8.23}$$

It is advisable to omit a number of initial terms from the computation and start the sum, (8.22), at some fairly large value of t to have a reasonable approximation.

Example 8.5. Long Memory Fitting of the Glacial Varve Series

We consider analyzing the glacial varve series discussed in Example 3.12 and Example 4.27. Figure 3.9 shows the original and log-transformed series (which we denote by x_t). In Example 5.8, we noted that x_t could be modeled as an ARIMA$(1,1,1)$ process. We fit the fractionally differenced model, (8.15), to the mean-adjusted series, $x_t - \bar{x}$. Applying the Gauss–Newton iterative procedure previously described leads to a final value of $d = .373$, which implies the set of coefficients $\pi_j(.373)$, as given in Figure 8.5 with $\pi_0(.373) = 1$.

```
d = 0.3727893
p = c(1)
for (k in 1:30){
  p[k+1] = (k-d)*p[k]/(k+1)
}
tsplot(1:30, p[-1], ylab=expression(pi(d)), lwd=2, xlab="Index",
           type="h", col="dodgerblue3")
```

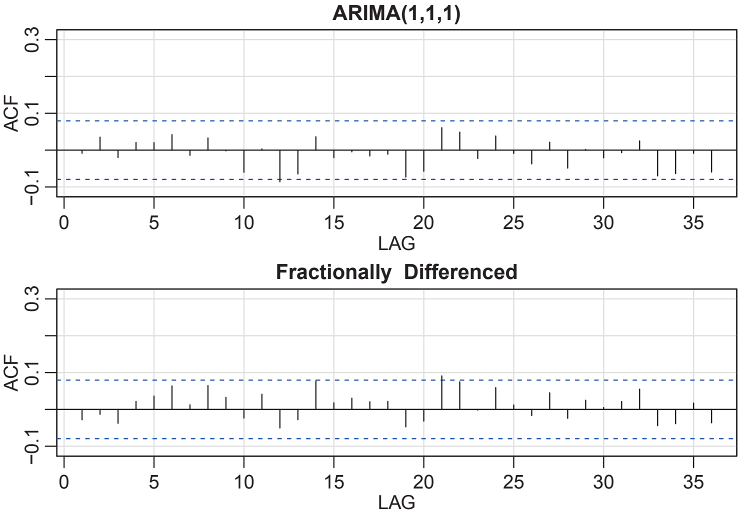

Figure 8.6 *ACF of residuals from the ARIMA*$(1, 1, 1)$ *fit to* x_t*, the logged varve series (top) and of the residuals from the long memory model fit,* $(1 - B)^d x_t = w_t$*, with* $d = .373$ *(bottom).*

We can compare roughly the performance of the fractional difference operator with the ARIMA model by examining the autocorrelation functions of the two residual series as shown in Figure 8.6. The ACFs of the two residual series are roughly comparable with the white noise model.

To perform this analysis in R, use the `arfima` package. Note that after the analysis, when the innovations (residuals) are pulled out of the results, they are in the form of a list and thus the need for double brackets (`[[ ]]`) below:

```
library(arfima)
summary(varve.fd <- arfima(log(varve), order = c(0,0,0)))
  Mode 1 Coefficients:
                Estimate Std. Error Th. Std. Err. z-value   Pr(>|z|)
 d.f          0.3727893  0.0273459     0.0309661 13.6324 < 2.22e-16
 Fitted mean 3.0814142  0.2646507            NA 11.6433 < 2.22e-16
 ---
 sigma^2 estimated as 0.229718;
 Log-likelihood = 466.028; AIC = -926.056; BIC = 969.944
# innovations (aka residuals)
innov = resid(varve.fd)[[1]]  # resid() produces a `list`
tsplot(innov)       # not shown
par(mfrow=2:1, cex.main=1)
acf1(resid(sarima(log(varve),1,1,1, details=FALSE)$fit),
           main="ARIMA(1,1,1)")
acf1(innov, main="Fractionally Differenced")
```

◊

Forecasting long memory processes is similar to forecasting ARIMA models.

That is, (8.16) and (8.21) can be used to obtain the truncated forecasts

$$x_{n+m}^n = -\sum_{j=1}^{n+m-1} \pi_j(\widehat{d})\, x_{n+m-j}^n, \tag{8.24}$$

for $m = 1, 2, \ldots$. Error bounds can be approximated by using

$$P_{n+m}^n = \widehat{\sigma}_w^2 \sum_{j=0}^{m-1} \psi_j^2(\widehat{d}) \tag{8.25}$$

where, as in (8.21),

$$\psi_j(\widehat{d}) = \frac{(j+\widehat{d})\psi_j(\widehat{d})}{(j+1)}, \tag{8.26}$$

with $\psi_0(\widehat{d}) = 1$.

No obvious short memory ARMA-type component can be seen in the ACF of the residuals from the fractionally differenced varve series shown in Figure 8.6. It is natural, however, that cases will exist in which substantial short memory-type components will also be present in data that exhibits long memory. Hence, it is natural to define the general ARFIMA(p,d,q), $-.5 < d < .5$ process as

$$\phi(B)\nabla^d(x_t - \mu) = \theta(B)w_t, \tag{8.27}$$

where $\phi(B)$ and $\theta(B)$ are as given in Chapter 4. Writing the model in the form

$$\phi(B)\pi_d(B)(x_t - \mu) = \theta(B)w_t \tag{8.28}$$

makes it clear how we go about estimating the parameters for the more general model. Forecasting for the ARFIMA(p,d,q) series can be easily done, noting that we may equate coefficients in

$$\phi(z)\psi(z) = (1-z)^{-d}\theta(z) \tag{8.29}$$

and

$$\theta(z)\pi(z) = (1-z)^d\phi(z) \tag{8.30}$$

to obtain the representations

$$x_t = \mu + \sum_{j=0}^{\infty} \psi_j w_{t-j}$$

and

$$w_t = \sum_{j=0}^{\infty} \pi_j (x_{t-j} - \mu).$$

We then can proceed as discussed in (8.24) and (8.25).

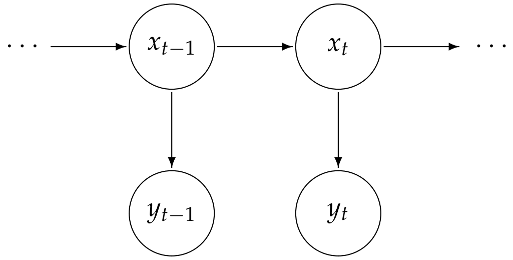

Figure 8.7 *Diagram of a state space model.*

8.4 State Space Models

A very general model that subsumes a whole class of special cases of interest in much the same way that linear regression does is the state space model that was introduced in Kalman (1960) and Kalman and Bucy (1961). The model arose in the space tracking setting, where the state equation defines the motion equations for the position or state of a spacecraft with location x_t and the data y_t reflect information that can be observed from a tracking device. Although it is typically applied to multivariate time series, we focus on the univariate case here.

In general, the state space model is characterized by two principles. First, there is a hidden or latent process x_t called the state process. The unobserved state process is assumed to be an AR(1),

$$x_t = \alpha + \phi x_{t-1} + w_t \,, \tag{8.31}$$

where $w_t \sim$ iid $\mathrm{N}(0, \sigma_w^2)$. In addition, we assume the initial state is $x_0 \sim \mathrm{N}(\mu_0, \sigma_0^2)$. The second condition is that the observations, y_t, are given by

$$y_t = Ax_t + v_t \,, \tag{8.32}$$

where A is a constant and the observation noise is $v_t \sim$ iid $\mathrm{N}(0, \sigma_v^2)$. In addition, x_0, $\{w_t\}$ and $\{v_t\}$ are uncorrelated. This means that the dependence among the observations is generated by states. The principles are displayed in Figure 8.7.

A primary aim of any analysis involving the state space model, (8.31)–(8.32), is to produce estimators for the underlying unobserved signal x_t, given the data $y_{1:s} = \{y_1, \dots, y_s\}$, to time s. When $s < t$, the problem is called *forecasting* or *prediction*. When $s = t$, the problem is called *filtering*, and when $s > t$, the problem is called *smoothing*. In addition to these estimates, we would also want to measure their precision. The solution to these problems is accomplished via the *Kalman filter* and *smoother*.

First, we present the Kalman filter, which gives the prediction and filtering equations. We use the following notation,

$$x_t^s = E(x_t \mid y_{1:s}) \quad \text{and} \quad P_t^s = E(x_t - x_t^s)^2 \,.$$