CHAPTER 15

THRESHOLD MODELS

It can be shown (Wold, 1948) that any weakly stationary process $\{Y_t\}$ admits the Wold decomposition

$$Y_t = U_t + e_t + \psi_1 e_{t-1} + \psi_2 e_{t-2} + \cdots$$

where e_t equals the deviation of Y_t from the best linear predictor based on all past Y values, and $\{U_t\}$ is a purely deterministic stationary process, with e_t being uncorrelated with U_s, for any t and s. A purely deterministic process is a process that can be predicted to arbitrary accuracy; (that is, with arbitrarily small mean squared error) by some linear predictors of finitely many past lags of the process. A simple example of a purely deterministic process is $U_t \equiv \mu$, a constant. A more subtle example is the random cosine wave model introduced on page 18. In essence, $\{U_t\}$ represents the stochastic, stationary "trend" in the data. The prediction errors $\{e_t\}$ are a white noise sequence, and e_t represents the "new" component making up Y_t and hence is often called the innovation of the process. The Wold decomposition then states that any weakly stationary process is the sum of a (possibly infinite-order) MA process and a deterministic trend. Thus, we can compute the best linear predictor within the framework of MA(∞) processes that can further be approximated by finite-order ARMA processes. The Wold decomposition thus guarantees the versatility of the ARMA models in prediction with stationary processes.

However, except for convenience, there is no reason for restricting to linear predictors. If we allow nonlinear predictors and seek the best predictor of Y_t based on past values of Y that minimizes the mean squared prediction error, then the best predictor need no longer be the best linear predictor. The solution is simply the conditional mean of Y_t given all past Y values. The Wold decomposition makes it clear that the best one-step-ahead linear predictor is the best one-step-ahead predictor if and only if $\{e_t\}$ in the Wold decomposition satisfies the condition that the conditional mean of e_t given past e's is identically equal to 0. The $\{e_t\}$ satisfying the latter condition is called a sequence of martingale differences, so the condition will be referred to as the martingale difference condition. The martingale difference condition holds if, for example, $\{e_t\}$ is a sequence of independent, identically distributed random variables with zero mean. But it also holds if $\{e_t\}$ is some GARCH process. Nonetheless, when the martingale difference condition fails, nonlinear prediction will lead to a more accurate prediction. Hannan (1973) defines a linear process to be one where the best one-step-ahead linear predictor is the best one-step-ahead predictor.

The time series models discussed so far are essentially linear models in the sense that, after suitable instantaneous transformation, the one-step-ahead conditional mean is a linear function of the current and past values of the time series variable. If the errors are normally distributed, as is commonly assumed, a linear ARIMA model results in a normally distributed process. Linear time series methods have proved to be very useful in practice. However, linear, normal processes do suffer from some limitations. For example, a stationary normal process is completely characterized by its mean and autocovariance function; hence the process reversed in time has the same distribution as the original process. The latter property is known as time reversibility. Yet, many real processes appear to be time-irreversible. For example, the historical daily closing price of a stock generally rose gradually but, if it crashed, it did so precipitously, signifying a time-irreversible data mechanism. Moreover, the one-step-ahead conditional mean may be nonlinear rather than linear in the current and past values. For example, animal abundance processes may be nonlinear due to finite-resource constraints. Specifically, while moderately high abundance in one period is likely to be followed by higher abundance in the next period, extremely high abundance may lead to a population crash in the ensuing periods. Nonlinear time series models generally display rich dynamical structure. Indeed, May (1976) showed that a very simple nonlinear deterministic difference equation may admit chaotic solutions in the sense that its time series solutions are sensitive to the initial values, which may appear to be indistinguishable from a white noise sequence based on correlation analysis. Nonlinear time series analysis thus may provide more accurate predictions, which can be very substantial in certain parts of the state space, and shed novel insights on the underlying dynamics of the data. Nonlinear time series analysis was earnestly initiated around the late 1970s, prompted by the need for modeling the nonlinear dynamics shown by real data; see Tong (2007). Except for cases with well-developed theory accounting for the underlying mechanism of an observed time series, the nonlinear data mechanism is generally unknown. Thus, a fundamental problem of empirical nonlinear time series analysis concerns the choice of a general nonlinear class of models. Here, our goal is rather modest in that we introduce the threshold model, which is one of the most important classes of nonlinear time series models. For a systematic account of nonlinear time series analysis and chaos, see Tong (1990) and Chan and Tong (2001).

15.1 Graphically Exploring Nonlinearity

In ARIMA modeling, the innovation (error) process is often specified as independent and identically normally distributed. The normal error assumption implies that the stationary time series is also a normal process; that is, any finite set of time series observations are jointly normal. For example, the pair (Y_1, Y_2) has a bivariate normal distribution and so does any pair of Y's; the triple (Y_1, Y_2, Y_3) has a trivariate normal distribution and so does any triple of Y's, and so forth. When data are nonnormal, instantaneous transformation of the form $h(Y_t)$, for example, $h(Y_t) = \sqrt{Y_t}$, may be applied to the data in the hope that a normal ARIMA model can serve as a good approximation to the underlying data-generating mechanism. The normality assumption is mainly

adopted for convenience in statistical inference. In practice, an ARIMA model with nonnormal innovations may be entertained. Indeed, such processes have very rich and sometimes exotic dynamics; see Tong (1990). If the normal error assumption is maintained, then a nonlinear time series is generally not normally distributed. Nonlinearity may then be explored by checking whether or not a finite set of time series observations are jointly normal; for example, whether or not the two-dimensional distribution of pairs of Y's is normal. This can be checked by plotting the scatter diagram of Y_t against Y_{t-1} or Y_{t-2}, and so forth. For a bivariate normal distribution, the scatter diagram should resemble an elliptical data cloud with decreasing density from its center. Departure from such a pattern (for example, existence of a large hole in the data cloud) may signify that the data are nonnormal and the underlying process may be nonlinear.

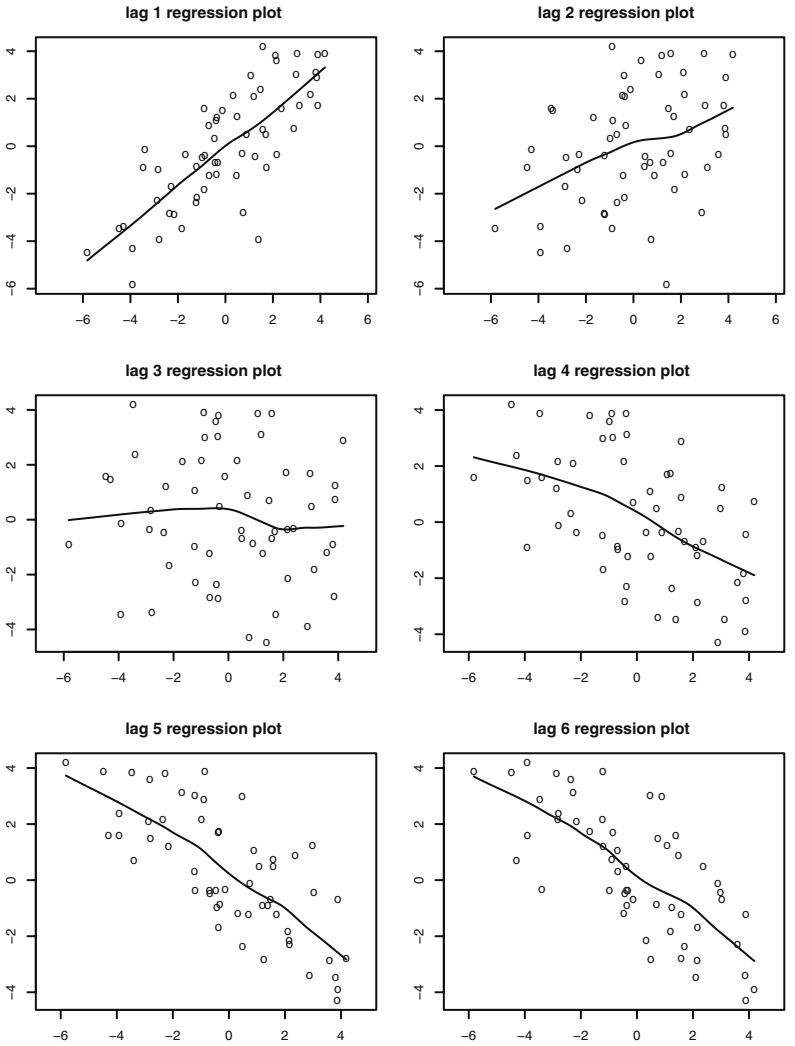
Exhibit 15.1 shows the scatter diagrams of Y_t versus its lag 1 to lag 6, where we simulated data from the ARIMA(2,1) model

$$Y_t = 1.6Y_{t-1} - 0.94Y_{t-2} + e_t - 0.64e_{t-1} \tag{15.1.1}$$

with the innovations being standard normal. Note that the data clouds in the scatter diagrams are roughly elliptically shaped.

To help us visualize the relationship between the response and its lags, we draw fitted nonparametric regression lines on each scatter diagram. For example, on the scatter diagram of Y_t against Y_{t-1}, a nonparametric estimate of the conditional mean function of Y_t given Y_{t-1}, also referred to as the lag 1 regression function, is superimposed. (Specifically, the lag 1 regression function equals $m_1(y) = E(Y_t|Y_{t-1}=y)$ as a function of y.) If the underlying process is linear and normal, the true lag 1 regression function must be linear and so we expect the nonparametric estimate of it to be close to a straight line. On the other hand, a curved lag 1 regression estimate may suggest that the underlying process is nonlinear. Similarly, one can explore the lag 2 regression function (that is, the conditional mean of Y_t given $Y_{t-2} = y$) as a function of y and higher-lag analogues. In the case of strong departure from linearity, the shape of these regression functions may provide some clue as to what nonlinear model may be appropriate for the data. Note that all lagged regression curves in Exhibit 15.1 are fairly straight, suggesting that the underlying process is linear, which indeed we know is the case.

Exhibit 15.1 Lagged Regression Plots for a Simulated ARMA(2,1) Process. Solid lines are fitted regression curves.

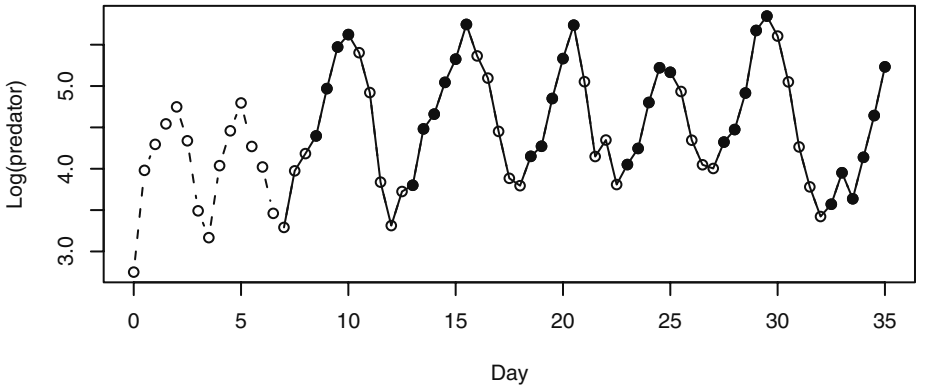



```
> win.graph(width=4.875, height=6.5,pointsize=8)
> set.seed(2534567); par(mfrow=c(3,2))
> y=arima.sim(n=61,model=list(ar=c(1.6,-0.94),ma=-0.64))
> lagplot(y)
```

We now illustrate the technique of a lagged regression plot with a real example. Exhibit 15.2 plots an experimental time series response as the number of individuals (*Didinium natsutum*, a protozoan) per ml measured every twelve hours over a period of 35 days; see Veilleux (1976) and Jost and Ellner (2000). The experiment studied the

population fluctuation of a prey-predator system; the prey is *Paramecium aurelia*, a unicellular ciliate protozon, whereas the predator species is *Didinium natsutum*. The initial part of the data appears to be nonstationary owing to transient effects. It can be seen that the increasing phase of the series is generally longer than that of the decreasing phase, suggesting that the time series is time-irreversible. Below, we shall omit the first 14 data points from the analysis; that is, only the (log-transformed) data corresponding to the solid curve in Exhibit 15.2 are used in subsequent analysis.

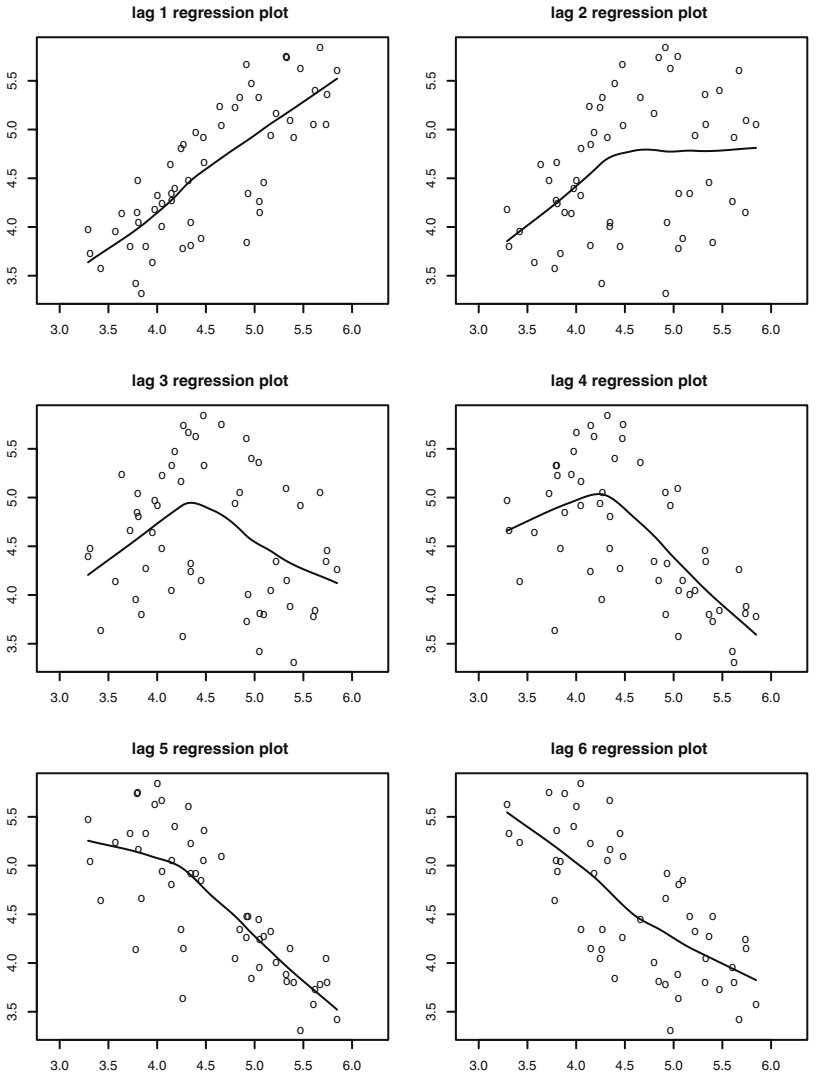
Exhibit 15.2 Logarithmically Transformed Number of Predators. The stationary part of the time series is displayed as a solid line. Solid circles indicate data in the lower regime of a fitted threshold autoregressive model.




```
> data(veilleux); predator=veilleux[,1]
> win.graph(width=4.875,height=2.5,pointsize=8)
> plot(log(predator),lty=2,type='o',xlab='Day',
    ylab='Log(predator)')
> predator.eq=window(predator,start=c(7,1))
> lines(log(predator.eq))
> index1=zlag(log(predator.eq),3)<=4.661
> points(y=log(predator.eq)[index1],(time(predator.eq))[index1],
    pch=19)
```

Exhibit 15.3 shows the lagged regression plots of the predator series. Notice that several scatter diagrams have a large hole in the center, hinting that the data need to be nonnormal. Also, the regression function estimates appear to be strongly nonlinear for lags 2 to 4, suggesting a nonlinear data mechanism; in fact, the histogram (not shown) suggests that the series is bimodal.

Exhibit 15.3 Lagged Regression Plots for the Predator Series




```
> win.graph(width=4.875,height=6.5,pointsize=8)
> data(predator.eq)
> lagplot(log(predator.eq)) # libraries mgcv and locfit required
```

We now elaborate on how the regression curves are estimated nonparametrically. Readers not interested in the technical details may skip to the next section. For concreteness, suppose we want to estimate the lag 1 regression function. (The extension to other lags is straightforward.) Nonparametric estimation of the lag 1 regression function gen-

erally makes use of the idea of estimating the conditional mean $m_1(y) = E(Y_t|Y_{t-1} = y)$ by averaging those Y's whose lag 1 values are close to y. Clearly, the averaging may be rendered more accurate by giving more weight to those Y's whose lag 1 value is closer to y. The weights are usually assigned systematically via some probability density function $k(y)$ and a bandwidth parameter $h > 0$. The data pair (Y_t, Y_{t-1}) is assigned the weight

$$w_t = \frac{1}{h} k\left(\frac{Y_{t-1} - y}{h}\right) \tag{15.1.2}$$

Hereafter we assume that $k(\cdot)$ is the standard normal probability density function. Note that then the right-hand side of Equation (15.1.2) is the normal probability density function with mean y and variance h^2. Finally, we define the Nadaraya-Watson estimator†

$$\hat{m}_1^{(0)}(y) = \frac{\sum_{t=2}^{n} w_t Y_t}{\sum_{t=2}^{n} w_t} \tag{15.1.3}$$

(The meaning of the superscript 0 will become clear later on.) Since the normal probability density function is negligible for values that differ from the mean by more than three standard deviations, the Nadaraya-Watson estimator essentially averages the Y_t whose Y_{t-1} is within $3h$ units from y, and the averaging is weighted with more weight to those observations whose lag 1 values are closer to y. The use of the Nadaraya-Watson estimator of the lag 1 regression function requires us to specify the bandwidth. There are several methods, including cross-validation for determining h. However, for an exploratory analysis, we can always use some default bandwidth value and vary it a bit to get some feel of the shape of the lag 1 regression function.

A more efficient nonparametric estimator may be obtained by assuming that the underlying regression function can be well-approximated locally by a linear function; see Fan and Gijbels (1996). The local linear estimator of the lag 1 regression function at y equals $\hat{m}_1^{(1)}(y) = b_0$, which is obtained by minimizing the local weighted residual sum of squares:

$$\sum_{t=2}^{n} w_t (Y_t - b_0 - b_1 Y_{t-1})^2 \tag{15.1.4}$$

The reader may now guess that the superscript k in the notation $\hat{m}_1^{(k)}(y)$ refers to the degree of the local polynomial. Often, data are unevenly spaced, in which case a single bandwidth may not work well. Instead, a variable bandwidth tied to the density of the data may be more efficient. A simple scheme is the nearest-neighbor scheme that varies the window width so that it covers a fixed fraction of data nearest to the center of the window. We set the fraction to be 70% for all our reported lagged regression plots.

† See Nadaraya (1964) and Watson (1964).

It is important to remember that the local polynomial approach assumes that the true lag 1 regression function is a smooth function. If the true lag 1 regression function is discontinuous, then the local polynomial approach may yield misleading estimates. However, a sharp turn in the estimated regression function may serve as a warning that the smoothness condition may not hold for the true lag 1 regression function.

15.2 Tests for Nonlinearity

Several tests have been proposed for assessing the need for nonlinear modeling in time series analysis. Some of these tests, such as those studied by Keenan (1985), Tsay (1986), and Luukkonen et al. (1988), can be interpreted as Lagrange multiplier tests for specific nonlinear alternatives.

Keenan (1985) derived a test for nonlinearity analogous to Tukey's one degree of freedom for nonadditivity test (see Tukey, 1949). Keenan's test is motivated by approximating a nonlinear stationary time series by a second-order Volterra expansion (Wiener, 1958)

$$Y_t = \mu + \sum_{\mu=-\infty}^{\infty} \theta_\mu \varepsilon_{t-\mu} + \sum_{\nu=-\infty}^{\infty} \sum_{\mu=-\infty}^{\infty} \theta_{\mu\nu} \varepsilon_{t-\mu} \varepsilon_{t-\nu} \tag{15.2.1}$$

where $\{\varepsilon_t, -\infty < t < \infty\}$ is a sequence of independent and identically distributed zero-mean random variables. The process $\{Y_t\}$ is linear if the double sum on the right-hand side of (15.2.1) vanishes. Thus, we can test the linearity of the time series by testing whether or not the double sum vanishes. In practice, the infinite series expansion has to be truncated to a finite sum. Let $Y_1,\ldots,Y_n$ denote the observations. Keenan's test can be implemented as follows:

(i) Regress Y_t on $Y_{t-1},\ldots,Y_{t-m}$, including an intercept term, where m is some prespecified positive integer; calculate the fitted values $\{\hat{Y}_t\}$ and the residuals $\{\hat{e}_t\}$, for $t = m+1,\ldots,n$; and set $RSS = \Sigma\hat{e}_t^2$, the residual sum of squares.

(ii) Regress $\hat{Y}_t^2$ on $Y_{t-1},\ldots,Y_{t-m}$, including an intercept term, and calculate the residuals $\{\hat{\xi}_t\}$ for $t = m+1,\ldots, n$.

(iii) Regress $\hat{e}_t$ on the residuals $\hat{\xi}_t$ without an intercept for $t = m+1,\ldots, n$, and Keenan's test statistic, denoted by $\hat{F}$, is obtained by multiplying $(n-2m-2)/(n-m-1)$ to the F-statistic for testing that the last regression function is identically zero. Specifically, let

$$\eta = \eta_0 \sqrt{\sum_{t=m+1}^{n} \hat{\xi}_t^2} \tag{15.2.2}$$

where η_0 is the regression coefficient. Form the test statistic

$$\hat{F} = \frac{\eta^2(n-2m-2)}{RSS - \eta^2} \tag{15.2.3}$$

Under the null hypothesis of linearity, the test statistic $\hat{F}$ is approximately distributed as an F-distribution with degrees of freedom 1 and $n - 2m - 2$.

Keenan's test can be derived heuristically as follows. Consider the following model.

$$Y_t = \theta_0 + \phi_1 Y_{t-1} + \ldots + \phi_m Y_{t-m} + \exp\left\{\eta\left(\sum_{j=1}^{m} \phi_j Y_{t-j}\right)^2\right\} + \varepsilon_t \tag{15.2.4}$$

where $\{\varepsilon_t\}$ are independent and normally distributed with zero mean and finite variance. If $\eta = 0$, the exponential term becomes 1 and can be absorbed into the intercept term so that the preceding model becomes an AR(m) model. On the other hand, for nonzero η, the preceding model is nonlinear. Using the expansion $\exp(x) \approx 1 + x$, which holds for x of small magnitude, it can be seen that, for small η, Y_t follows approximately a quadratic AR model:

$$Y_t = \theta_0 + 1 + \phi_1 Y_{t-1} + \ldots + \phi_m Y_{t-m} + \eta\left(\sum_{j=1}^{m} \phi_j Y_{t-j}\right)^2 + \varepsilon_t \tag{15.2.5}$$

This is a restricted linear model in that the last covariate is the square of the linear term $\phi_1 Y_{t-1} + \cdots + \phi_m Y_{t-m}$, which is replaced by the fitted values $\hat{Y}_t$ under the null hypothesis. Keenan's test is equivalent to testing $\eta = 0$ in the multiple regression model (with the constant 1 being absorbed into θ_0):

$$Y_t = \theta_0 + \phi_1 Y_{t-1} + \ldots + \phi_m Y_{t-m} + \eta \hat{Y}_t^2 + \varepsilon_t \tag{15.2.6}$$

which can be carried out in the manner described in the beginning of this section. Note that the fitted values are only available for $n \geq t \geq m + 1$. Keenan's test is the same as the F-test for testing whether or not $\eta = 0$. A more formal approach is facilitated by the Lagrange multiplier test; see Tong (1990).

Keenan's test is both conceptually and computationally simple and only has one degree of freedom, which makes the test very useful for small samples. However, Keenan's test is powerful only for detecting nonlinearity in the form of the square of the approximating linear conditional mean function. Tsay (1986) extended Keenan's approach by considering more general nonlinear alternatives. A more general alternative to nonlinearity may be formulated by replacing the term

$$\exp\left\{\eta\left(\sum_{j=1}^{m} \phi_j Y_{t-j}\right)^2\right\} \tag{15.2.7}$$

by

$$\left.\begin{aligned} &\exp(\delta_{1,1} Y_{t-1}^2 + \delta_{1,2} Y_{t-1} Y_{t-2} + \cdots + \delta_{1,m} Y_{t-1} Y_{t-m} \\ &\quad + \delta_{2,2} Y_{t-2}^2 + \delta_{2,3} Y_{t-2} Y_{t-3} + \cdots + \delta_{2,m} Y_{t-2} Y_{t-m} + \cdots \\ &\quad + \delta_{m-1,m-1} Y_{t-m+1}^2 + \delta_{m-1,m} Y_{t-m+1} Y_{t-m} + \delta_{m,m} Y_{t-m}^2) + \varepsilon_t \end{aligned}\right\} \tag{15.2.8}$$

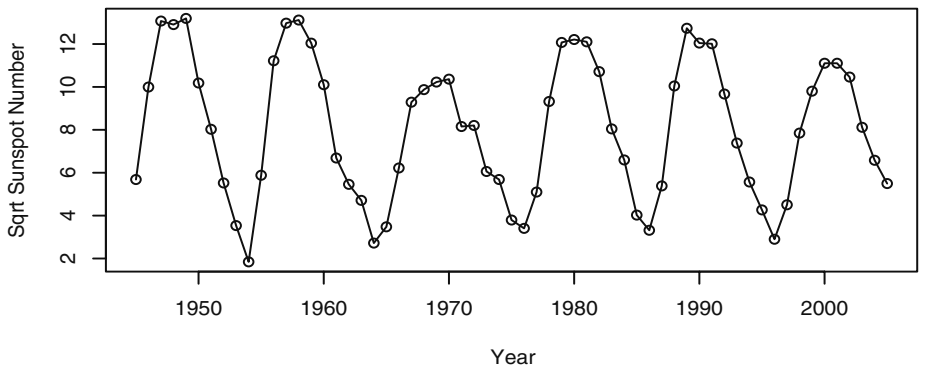
Using the approximation $\exp(x) \approx 1 + x$, we see that the nonlinear model is approximately a quadratic AR model. But the coefficients of the quadratic terms are now unconstrained. Tsay's test is equivalent to considering the following quadratic regression model:

$$\left.\begin{aligned} Y_t = {} & \theta_0 + \phi_1 Y_{t-1} + \cdots + \phi_m Y_{t-m} \\ & + \delta_{1,1} Y_{t-1}^2 + \delta_{1,2} Y_{t-1} Y_{t-2} + \cdots + \delta_{1,m} Y_{t-1} Y_{t-m} \\ & + \delta_{2,2} Y_{t-2}^2 + \delta_{2,3} Y_{t-2} Y_{t-3} + \cdots + \delta_{2,m} Y_{t-2} Y_{t-m} + \cdots \\ & + \delta_{m-1,m-1} Y_{t-m+1}^2 + \delta_{m-1,m} Y_{t-m+1} Y_{t-m} + \delta_{m,m} Y_{t-m}^2 + \varepsilon_t \end{aligned}\right\} \quad (15.2.9)$$

and testing whether or not all the $m(m+1)/2$ coefficients $\delta_{i,j}$ are zero. Again, this can be carried out by an F-test that all $\delta_{i,j}$'s are zero in the preceding equation. For a rigorous derivation of Tsay's test as a Lagrange multiplier test, see Tong (1990).

We now illustrate these tests with two real datasets. In the first application, we use the annual American (relative) sunspot numbers collected from 1945 to 2007. The annual (relative) sunspot number is a weighted average of solar activities measured from a network of observatories. Historically, the daily sunspot number was computed as some weighted sum of the count of visible, distinct spots and that of clusters of spots on the solar surface. The sunspot number reflects the intensity of solar activity. Below, the sunspot data are square root transformed to make them more normally distributed; see Exhibit 15.4. The time series plot shows that the sunspot series tends to rise up more quickly than when it declines, suggesting that it is time-irreversible.

Exhibit 15.4 Annual American Relative Sunspot Numbers




```
> win.graph(width=4.875,height=2.5,pointsize=8)
> data(spots)
> plot(sqrt(spots),type='o',xlab='Year',
    ylab='Sqrt Sunspot Number')
```

To carry out the tests for nonlinearity, we have to specify m, the working autoregressive order. Under the null hypothesis that the process is linear, the order can be specified by using some information criterion, for example, the AIC. For the sunspot data, $m = 5$ based on the AIC. Both the Keenan test and the Tsay test reject linearity, with p-values being 0.0002 and 0.0009, respectively.

For the second example, we consider the predator series discussed in the preceding section. The working AR order is found to be 4. Both the Keenan test and the Tsay test reject linearity, with p-values being 0.00001 and 0.03, respectively, which is consistent with the inference drawn from the lagged regression plots reported earlier.

There are some other tests, such as the BDS test developed by Brock, Deckert and Seheinkman (1996), based on concepts that arise in the theory of chaos, and the neural-network test, proposed by White (1989) for testing "neglected nonlinearity." For a recent review of tests for nonlinearity, see Tong (1990) and Granger and Teräsvirta (1993). We shall introduce one more test later.

15.3 Polynomial Models Are Generally Explosive

In nonlinear regression analysis, polynomial regression models of higher degrees are sometimes employed, even though they are deemed not useful for extrapolation because of their quick blowup to infinity. For this reason, polynomial regression models are of limited practical use. Based on the same reasoning, polynomial time series models may be expected to do poorly in prediction. Indeed, polynomial time series models of degree higher than 1 and with Gaussian errors are invariably explosive. To see this, consider the following simple quadratic AR(1) model.

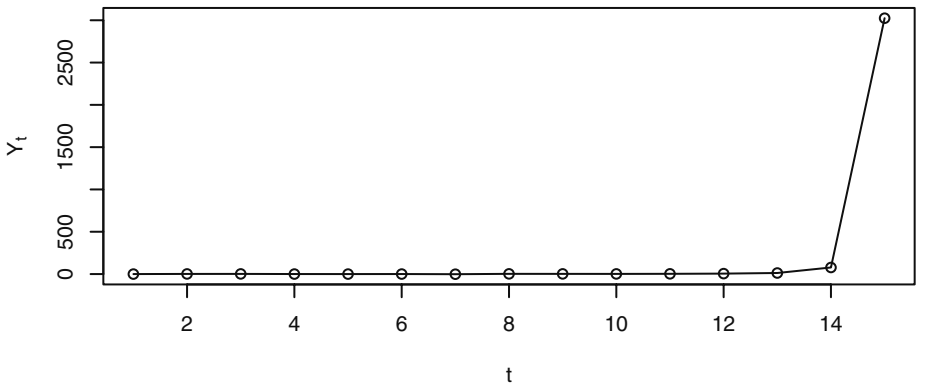
$$Y_t = \phi Y_{t-1}^2 + e_t \tag{15.3.1}$$

where $\{e_t\}$ are independent and identically distributed standard normal random variables. Let $\phi > 0$ and let c be a large number that is greater than $3/\phi$. If $Y_1 > c$ (which may happen with positive probability due to the normality of the errors), then $Y_2 > 3Y_1 + e_2$ and hence $Y_2 > 2c$ with some nonzero probability. With careful probability analysis, it can be shown that, with positive probability, the quadratic AR(1) process satisfies the inequality $Y_t > 2^t c$ for $t = 1, 2, 3, \ldots$ and hence blows up to $+\infty$. Indeed, the quadratic AR(1) process, with normal errors, goes to infinity with probability 1.

As an example, Exhibit 15.5 displays a realization from a quadratic AR(1) model with $\phi = 0.5$ and standard normal errors that takes off to infinity at $t = 15$.

Note that the quadratic AR(1) process becomes explosive only when the process takes some value of sufficiently large magnitude. If the coefficient ϕ is small, it may take much longer for the quadratic AR(1) process to take off to infinity. Normal errors can take arbitrarily large values, although rather rarely, but when this happens, the process becomes explosive. Thus, any noise distribution that is unbounded will guarantee the explosiveness of the quadratic AR(1) model. Chan and Tong (1994) further showed that this explosive behavior is true for any polynomial autoregressive process of degree higher than 1 and of any finite order when the noise distribution is unbounded.

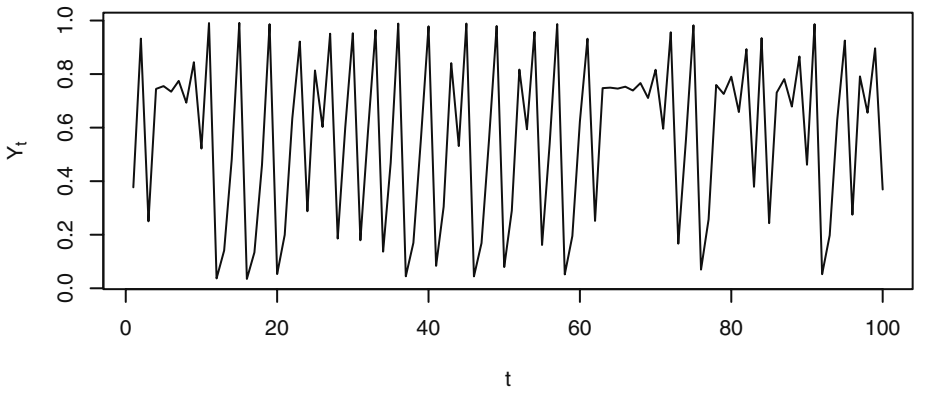
Exhibit 15.5 A Simulated Quadratic AR(1) Process with $\phi = 0.5$




```
> set.seed(1234567)
> plot(y=qar.sim(n=15,phi1=.5,sigma=1),x=1:15,type='o',
    ylab=expression(Y[t]),xlab='t')
```

It is interesting to note that, for bounded errors, a polynomial autoregressive model may admit a stationary distribution that could be useful for modeling nonlinear time series data; see Chan and Tong (1994). For example, Exhibit 15.6 displays the time series solution of a deterministic logistic map, namely $Y_t = 3.97Y_{t-1}(1 - Y_{t-1})$, $t = 2$, 3,… with the initial value $Y_1 = 0.377$. Its corresponding sample ACF is shown in Exhibit 15.7, which, except for the mildly significant lag 4, resembles that of white noise. Note that, for a sufficiently large initial value, the solution of the logistic map will explode to infinity.

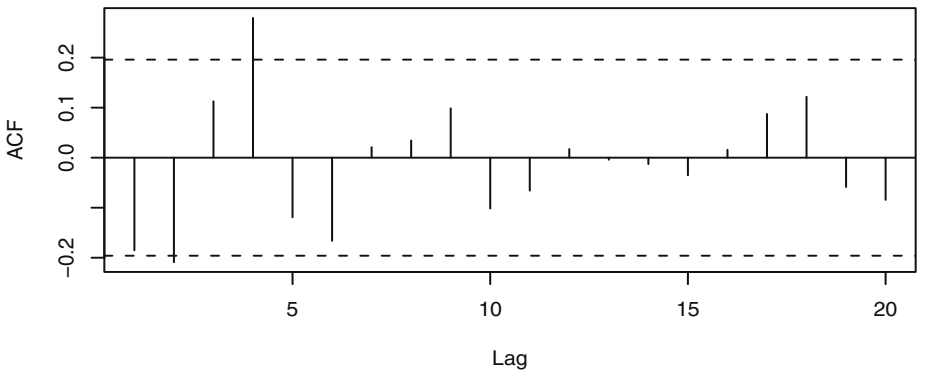
Exhibit 15.6 The Trajectory of the Logistic Map with Parameter 3.97 and Initial Value $Y_1 = 0.377$




```
> y=qar.sim(n=100,const=0.0,phi0=3.97,phi1=-3.97,sigma=0,
    init=.377)
```

```
> plot(x=1:100,y=y,type='l',ylab=expression(Y[t]),xlab='t')
```

Exhibit 15.7 Sample ACF of the Logistic Time Series




```
> acf(y)
```

However, the bound on the noise distribution necessary for the existence of a stationary polynomial autoregressive model varies with the model parameters and the initial value, which greatly complicates the modeling task. Henceforth, we shall not pursue the use of polynomial models in time series analysis.

15.4 First-Order Threshold Autoregressive Models

The discussion in the preceding section provides an important insight that for a nonlinear time series model to be stationary, it must be either linear or approaching linearity in the "tail." From this perspective, piecewise linear models, more widely known as threshold models, constitute the simplest class of nonlinear model. Indeed, the usefulness of threshold models in nonlinear time series analysis was well-documented by the seminal work of Tong (1978, 1983, 1990) and Tong and Lim (1980), resulting in an extensive literature of ongoing theoretical innovations and applications in various fields.

The specification of a threshold model requires specifying the number of linear submodels and the mechanism dictating which of them is operational. Consequently, there exist many variants of the threshold model. Here, we focus on the two-regime self-exciting threshold autoregressive (SETAR) model introduced by Tong, for which the switching between the two linear submodels depends solely on the position of the threshold variable. For the SETAR model (simply referred to as the TAR model below), the threshold variable is a certain lagged value of the process itself; hence the adjective self-exciting. (More generally, the threshold variable may be some vector covariate process or even some latent process, but this extension will not be pursued here.) To fix ideas, consider the following first-order TAR model:

$$Y_t = \begin{cases} \phi_{1,0} + \phi_{1,1} Y_{t-1} + \sigma_1 e_t, & \text{if } Y_{t-1} \le r \\ \phi_{2,0} + \phi_{2,1} Y_{t-1} + \sigma_2 e_t, & \text{if } Y_{t-1} > r \end{cases} \tag{15.4.1}$$

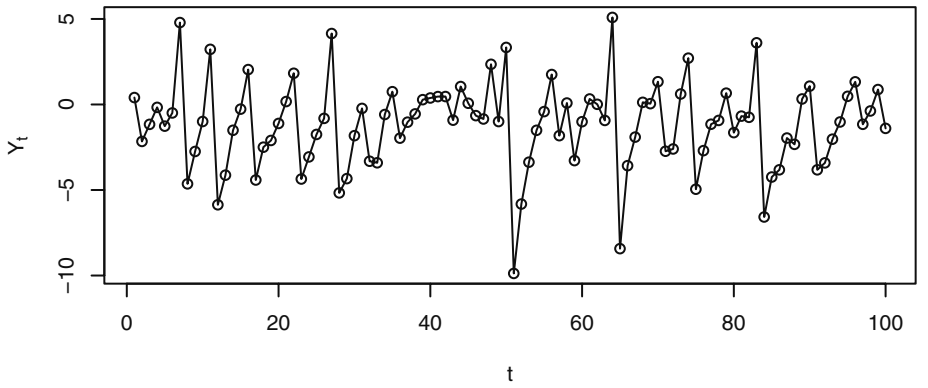
where the ϕ's are autoregressive parameters, σ's are noise standard deviations, r is the threshold parameter, and $\{e_t\}$ is a sequence of independent and identically distributed random variables with zero mean and unit variance. Thus, if the lag 1 value of Y_t is not greater than the threshold, the conditional distribution of Y_t is the same as that of an AR(1) process with intercept $\phi_{1,0}$, autoregressive coefficient $\phi_{1,1}$, and error variance σ_1^2, in which case we may say that the first AR(1) submodel is operational. On the other hand, when the lag 1 value of Y_t exceeds the threshold r, the second AR(1) process with parameters $(\phi_{2,0}, \phi_{2,1}, \sigma_2^2)$ is operational. Thus, the process switches between two linear mechanisms dependent on the position of the lag 1 value of the process. When the lag 1 value does not exceed the threshold, we say that the process is in the lower (first) regime, and otherwise it is in the upper regime. Note that the error variance need not be identical for the two regimes, so that the TAR model can account for some conditional heteroscedasticity in the data.

As a concrete example, we simulate some data from the following first-order TAR model:

$$Y_t = \begin{cases} 0.5 Y_{t-1} + e_t, & \text{if } Y_{t-1} \le -1 \\ -1.8 Y_{t-1} + 2e_t, & \text{if } Y_{t-1} > -1 \end{cases} \tag{15.4.2}$$

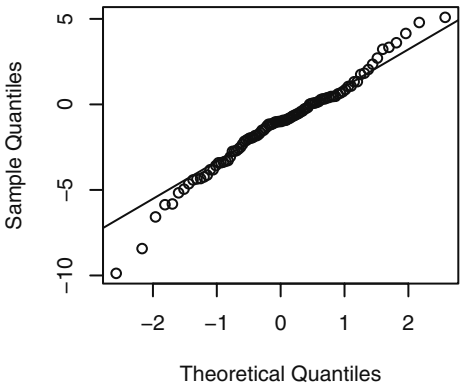
Exhibit 15.8 shows the time series plot of the simulated data of size $n = 100$. A notable feature of the plot is that the time series is somewhat cyclical, with asymmetrical cycles where the series tends to drop rather sharply but rises relatively slowly. This asymmetry means that the probabilistic structure of the process will be different if we reverse the direction of time. One way to see this is to make a transparency of the time series plot and flip the transparency over to see the time series plot with time reversed. In this case, the simulated data will rise sharply and drop slowly with time reversed. Recall that this phenomenon is known as time irreversibility. For a stationary Gaussian ARMA process, the probabilistic structure is determined by its first and second moments, which are invariant with respect to time reversal, hence the process must be time-reversible. Many real time series, for example the predator series and the relative sunspot series, appear to be time-irreversible, suggesting that the underlying process is nonlinear. Exhibit 15.9 shows the QQ normal score plot for the simulated data. It shows that the distribution of simulated data has a thicker tail than a normal distribution, despite the fact that the errors are normally distributed.

Exhibit 15.8 A Simulated First-Order TAR Process




```
> set.seed(1234579)
> y=tar.sim(n=100,Phi1=c(0,0.5),Phi2=c(0,-1.8),p=1,d=1,sigma1=1,
    thd=-1,sigma2=2)$y
> plot(y=y,x=1:100,type='o',xlab='t',ylab=expression(Y[t]))
```

Exhibit 15.9 QQ Normal Plot for the Simulated TAR Process




```
> win.graph(width=2.5,height=2.5,pointsize=8)
> qqnorm(y); qqline(y)
```

The autoregressive coefficient of the submodel in the upper regime equals −1.8, yet the simulated data appear to be stationary, which may be unexpected from a linear perspective, as an AR(1) model cannot be stationary if the autoregressive coefficient exceeds 1 in magnitude. This puzzle may be better understood by considering the case of no noise terms in either regime; that is, $\sigma_1 = \sigma_2 = 0$. The deterministic process thus defined is referred to as the *skeleton* of the TAR model. We show below that, for any ini-

tial value, the skeleton is eventually a bounded process; the stability of the skeleton underlies the stationarity of the TAR model. Readers not interested in the detailed analysis verifying the ultimate boundedness of the skeleton may skip to the next paragraph. Let the initial value y_1 be some large number, say 10, a value falling in the upper regime. So, the next value is $y_2 = (-1.8)\times 10 = -18$, which is in the lower regime. Therefore, the third value equals $y_3 = 0.5\times(-18) = -9$. As the third value is in the lower regime, the fourth value equals $y_4 = 0.5\times(-9) = -4.5$, which remains in the lower regime, so that the fifth value equals $y_5 = 0.5\times(-4.5) = -2.25$. It is clear that once the data remain in the lower regime, they will be halved in the next iterate and this process continues until some future iterate crosses the threshold −1, which occurs for $y_7 = -0.5625$. Now the second linear submodel is operational, so that $y_8 = (-1.8)\times(-0.5625) = 1.0125$ and $y_9 = (-1.8)\times 1.0125 = -1.8225$, which is again in the lower regime. In conclusion, if some iterate is in the lower regime, the next iterate is obtained by halving the previous iterate until some future iterate exceeds −1. On the other hand, if some iterate exceeds 1, the next iterate must be less than −1 and hence in the lower regime. By routine analysis, it can be checked that the process is eventually trapped between −1 and 1.8 and hence is a bounded process.

A bounded skeleton is stable in some sense. Chan and Tong (1985), showed that under some mild conditions, a TAR model is asymptotically stationary if its skeleton is stable. In fact, stability of the skeleton together with some regularity conditions imply the stronger property of *ergodicity*; namely, the process admits a stationary distribution and for any function $h(Y_t)$ having a finite stationary first moment (which holds if h is a bounded function),

$$\frac{1}{n}\sum_{t=1}^{n} h(Y_t) \tag{15.4.3}$$

converges to the stationary mean of $h(Y_t)$, computed according to the stationary distribution. See Cline and Pu (2001) for a recent survey on the linkage between stability and ergodicity and counterexamples when this linkage may fail to hold.

The stability analysis of the skeleton can be much simplified by the fact that the ergodicity of a TAR model can be inferred from the stability of an associated skeleton defined by a difference equation obtained by modifying the equation defining the TAR model by suppressing the noise terms and the intercepts (that is, zero errors and zero intercepts) and setting the threshold to 0. For the simulated example, the associated skeleton is then defined by the following difference equation:

$$Y_t = \begin{cases} 0.5Y_{t-1}, & \text{if } Y_{t-1} \le 0 \\ -1.8Y_{t-1}, & \text{if } Y_{t-1} > 0 \end{cases} \tag{15.4.4}$$

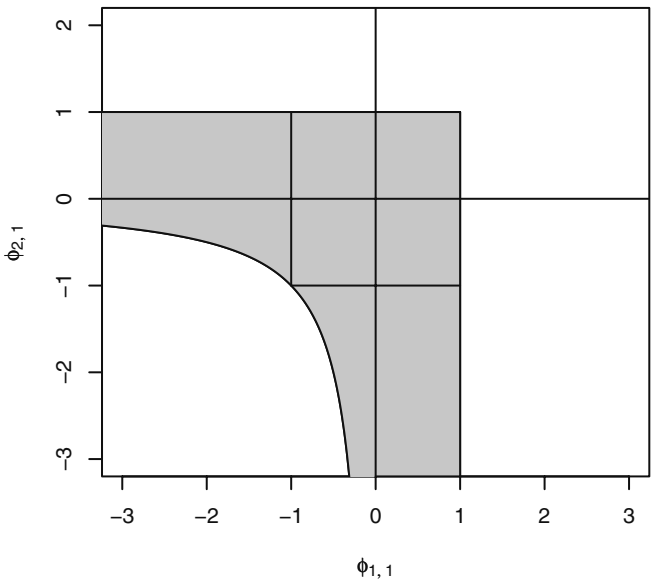
Now, the solution to the skeleton above can be readily obtained: Given a positive value for y_1, $y_t = (-1.8)\times 0.5^{t-2}\times y_1$, for all $t \ge 2$. For negative y_1, $y_t = 0.5^{t-1}\times y_1$. In both cases, $y_t \to 0$, as $t \to \infty$. The origin is said to be an *equilibrium point* as $y_t \equiv 0$, for all t, if $y_1 = 0$. The origin is then said to be a *globally exponentially stable limit point*, as the skeleton approaches it exponentially fast for any nonzero initial value. It can be shown (Chan and

Tong, 1985) that the origin is a globally exponentially stable limit point for the skeleton if the parameters satisfy the constraints

$$\phi_{1,1} < 1, \phi_{2,1} < 1, \phi_{1,1}\phi_{2,1} < 1 \tag{15.4.5}$$

in which case the first-order TAR model is ergodic and hence stationary. Exhibit 15.10 shows the region of stationarity shaded in gray. Note that the region of stationarity is substantially larger than the region defined by the linear time series inspired constraints $|\phi_{1,1}| < 1$, $|\phi_{2,1}| < 1$, corresponding to the region bounded by the inner square in Exhibit 15.10. For parameters lying strictly outside the region defined by the constraints (Equations (15.4.5)), the skeleton is unstable and the TAR model is nonstationary. For example, if $\phi_{2,1}>1$, then the skeleton will escape to positive infinity for all sufficiently large initial values. On the boundary of the parametric region defined by (15.4.5), the intercept terms of the TAR model are pivotal in determining the stability of the skeleton and the stationarity of the TAR models; see Chan et al. (1985). In practice, we can check if the skeleton is stable numerically by using several different initial values. A stable skeleton gives us more confidence in assuming that the model is stationary.

Exhibit 15.10 Stationarity Region for the First-Order TAR Model (Shaded)



15.5 Threshold Models

The first-order (self-exciting) threshold autoregressive model can be readily extended to higher order and with a general integer delay:

$$Y_t = \begin{cases} \phi_{1,0} + \phi_{1,1} Y_{t-1} + \cdots + \phi_{1,p_1} Y_{t-p_1} + \sigma_1 e_t, & \text{if } Y_{t-d} \le r \\ \phi_{2,0} + \phi_{2,1} Y_{t-1} + \cdots + \phi_{2,p_2} Y_{t-p_2} + \sigma_2 e_t, & \text{if } Y_{t-d} > r \end{cases} \tag{15.5.1}$$

Note that the autoregressive orders p_1 and p_2 of the two submodels need not be identical, and the delay parameter d may be larger than the maximum autoregressive orders. However, by including zero coefficients if necessary, we may and shall henceforth assume that $p_1 = p_2 = p$ and $1 \le d \le p$, which simplifies the notation. The TAR model defined by Equation (15.5.1) is denoted as the TAR$(2;p_1,p_2)$ model with delay d.

Again, the stability of the associated skeleton, obtained by setting the threshold to zero and suppressing the noise terms and the intercepts, implies that the TAR model is ergodic and stationary. However, the stability of the associated skeleton is now much more complex in the higher-order case so much so that the necessary and sufficient parametric conditions for the stationarity of the TAR model are still unknown. Nonetheless, there exist some simple sufficient conditions for the stationarity of a TAR model. For example, the TAR model is ergodic and hence asymptotically stationary if $|\phi_{1,1}| + \cdots + |\phi_{1,p}| < 1$ and $|\phi_{2,1}| + \cdots + |\phi_{2,p}| < 1$; see Chan and Tong (1985).

So far, we have considered the case of two regimes defined by the partition $-\infty < r < \infty$ of the real line, so that the first (second) submodel is operational if Y_{t-d} lies in the first (second) interval. The extension to the case of m regimes is straightforward and effected by partitioning the real line into $-\infty < r_1 < r_2 < \cdots < r_{m-1} < \infty$, and the position of Y_{t-d} relative to these thresholds determines which linear submodel is operational. We shall not pursue this topic further but shall restrict our discussion to the case of two regimes.

15.6 Testing for Threshold Nonlinearity

While Keenan's test and Tsay's test for nonlinearity are designed for detecting quadratic nonlinearity, they may not be sensitive to threshold nonlinearity. Here, we discuss a likelihood ratio test with the threshold model as the specific alternative. The null hypothesis is an AR(p) model versus the alternative hypothesis of a two-regime TAR model of order p and with constant noise variance, that is; $\sigma_1 = \sigma_2 = \sigma$. With these assumptions, the general model can be rewritten as

$$\begin{aligned} Y_t = {} & \phi_{1,0} + \phi_{1,1} Y_{t-1} + \cdots + \phi_{1,p} Y_{t-p} \\ & + \{\phi_{2,0} + \phi_{2,1} Y_{t-1} + \cdots + \phi_{2,p} Y_{t-p}\} I(Y_{t-d} > r) + \sigma e_t \end{aligned} \tag{15.6.1}$$

where the notation $I(\cdot)$ is an indicator variable that equals 1 if and only if the enclosed expression is true. Moreover, in this formulation, the coefficient $\phi_{2,0}$ represents the change in the intercept in the upper regime relative to that of the lower regime, and similarly interpreted are $\phi_{2,1},\ldots,\phi_{2,p}$. The null hypothesis states that $\phi_{2,0} = \phi_{2,1} = \cdots = \phi_{2,p} = 0$. While the delay may be theoretically larger than the autoregressive order, this is seldom the case in practice. Hence, it is assumed that $d \le p$ throughout this section, and

under this assumption and assuming the validity of linearity, the large-sample distribution of the test does not depend on d.

In practice, the test is carried out with fixed p and d. The likelihood ratio test statistic can be shown to be equivalent to

$$T_n = (n-p)\log\left\{\frac{\hat{\sigma}^2(H_0)}{\hat{\sigma}^2(H_1)}\right\} \tag{15.6.2}$$

where $n - p$ is the effective sample size, $\hat{\sigma}^2(H_0)$ is the maximum likelihood estimator of the noise variance from the linear AR(p) fit and $\hat{\sigma}^2(H_1)$ from the TAR fit with the threshold searched over some finite interval. See the next section for a detailed discussion on estimating a TAR model. Under the null hypothesis that $\phi_{2,0} = \phi_{2,1} = \cdots = \phi_{2,p} = 0$, the (nuisance) parameter r is absent. Hence, the sampling distribution of the likelihood ratio test under H_0 is no longer approximately χ^2 with p degrees of freedom. Instead, it has a nonstandard sampling distribution; see Chan (1991) and Tong (1990). Chan (1991) derived an approximation method for computing the p-values of the test that is highly accurate for small p-values. The test depends on the interval over which the threshold parameter is searched. Typically, the interval is defined to be from the $a \times 100$th percentile to the $b \times 100$th percentile of $\{Y_t\}$, say from the 25th percentile to the 75th percentile. The choice of a and b must ensure that there are adequate data falling into each of the two regimes for fitting the linear submodels.

The reader may wonder why the search of the threshold is restricted to some finite interval. Intuitively, such a restriction is desirable, as we want enough data to estimate the parameters for the two regimes under the alternative hypothesis. A deeper reason is mathematical in nature. This restriction is necessary because if the true model is linear, the threshold parameter is undefined, in which case an unrestricted search may result in the threshold estimator being close to the minimum or maximum data values, making the large-sample approximation ineffective.

We illustrate the likelihood ratio test for threshold nonlinearity using the (square-root-transformed) relative sunspot data and the (log-transformed) predator data. Recall that both Keenan's test and Tsay's test suggested that these data are nonlinear. Setting $p = 5$, $a = 0.25$, and $b = 0.75$ for the sunspot data, we tried the likelihood ratio test for threshold nonlinearity with different delays from 1 to 5, resulting in the test statistics being 46.9, 111.3, 99.1, 85.0, and 45.1, respectively.† Repeating the test with $a = 0.1$ and $b = 0.9$ yields identical results for this case. All the tests above have p-values less than 0.000, suggesting that the data-generating mechanism is highly nonlinear. Notice that the test statistic attains the largest value when $d = 2$; hence we may tentatively estimate

† The R code to carry out these calculations is as follows:

```
> pvaluem=NULL
> for (d in 1:5) { res=tlrt(sqrt(spots),p=5,d=d,a=0.25,b=0.75)
> pvaluem= cbind( pvaluem, c(d,res$test.statistic,res$p.value)) }
> rownames(pvaluem)=c('d','test statistic','p-value')
> round(pvaluem,3)
```

the delay to be 2. But delay 3 is very competitive.

Next, consider the predator series, with $p = 4$, $a = 0.25$, $b = 0.75$, and $1 \le d \le 4$. The test statistics and their p-values, enclosed in parentheses, are found to equal 19.3 (0.026), 28.0 (0.001), 32.0 (0.000), and 16.2 (0.073), respectively. Thus, there is some evidence that the predator series is nonlinear, with the delay likely to be 2 or 3. Note that the test is not significant for $d = 4$ at the 5% significance level.†

15.7 Estimation of a TAR Model

Because the stationary distribution of a TAR model does not have a closed-form solution, estimation is often carried out conditional on the max(p,d) initial values, where p is the order of the process and d the delay parameter. Moreover, the noise series is often assumed to be normally distributed, and we will make this assumption throughout this section. The normal error assumption implies that the response is conditionally normal, but see Samia, Chan and Stenseth (2007) for some recent work on the nonnormal case. If the threshold parameter r and the delay parameter d are known, then the data cases can be split into two parts according to whether or not $Y_{t-d} \le r$. Let there be n_1 data cases in the lower regime. With the data in the lower regime, we can regress Y_t on its lags 1 to p to find the estimates of $\hat{\phi}_{1,0}, \hat{\phi}_{1,1}, \ldots, \hat{\phi}_{1,p}$ and the maximum likelihood noise variance estimate $\hat{\sigma}_1^2$; that is, the sum of squared residuals divided by n_1. The number n_1 and the parameter estimates for the lower regime generally depend on r and d; we sometimes write the more explicit notation, for example $n_1(r,d)$, below for clarity. Similarly, using the data, say n_2 of them, falling in the upper regime, we can obtain the parameter estimates $\hat{\phi}_{2,0}, \hat{\phi}_{2,1}, \ldots, \hat{\phi}_{2,p}$ and $\hat{\sigma}_2^2$. Clearly, $n_1 + n_2 = n - p$, where n is the sample size. Substituting these estimates into the log-likelihood function yields the so-called *profile log-likelihood function* of (r,d):

$$l(r,d) = -\frac{n-p}{2}\{1 + \log(2\pi)\} - \frac{n_1(r,d)}{2}\log((\hat{\sigma}_1(r,d))^2) - \frac{n_2(r,d)}{2}\log((\hat{\sigma}_2(r,d))^2) \tag{15.7.1}$$

The estimates of r and d can be obtained by maximizing the profile likelihood function above. The optimization need only be searched with r over the observed Y's and integer d between 1 and p. This is because, for fixed d, the function above is constant between two consecutive observations.

However, without some restrictions on the threshold parameter, the (conditional) maximum likelihood method discussed above will not work. For example, if the lower regime contains only one data case, the noise variance $\hat{\sigma}_1^2 = 0$ so that the conditional log-likelihood function equals ∞, in which case the conditional maximum likelihood estimator is clearly inconsistent. This problem may be circumvented by restricting the

† The R code for this calculation is similar to that shown on the previous page. The details may be found in the R code scripts for Chapter 15 available on the textbook Website.

search of the threshold to be between two predetermined percentiles of Y; for example, between the tenth and ninetieth percentiles.

Another approach to handle the aforementioned difficulty is to estimate the parameters using the conditional least squares (CLS) approach. The CLS approach estimates the parameters by minimizing the predictive sum of squared errors, or equivalently conditional maximum likelihood estimation for the case of homoscedastic (constant-variance) Gaussian errors; that is, $\sigma_1 = \sigma_2 = \sigma$ so that maximizing the log-likelihood function is equivalent to minimizing the conditional residual sum of squares:

$$\begin{aligned} L(r, d) = \sum_{t=p+1}^{n} \{ & (Y_t - \phi_{1,0} - \phi_{1,1} Y_{t-1} - \cdots - \phi_{1,p} Y_{t-p})^2 I(Y_{t-d} \le r) \\ & + (Y_t - \phi_{2,0} - \phi_{2,1} Y_{t-1} - \cdots - \phi_{2,p} Y_{t-p})^2 I(Y_{t-d} > r) \} \end{aligned} \tag{15.7.2}$$

where $I(Y_{t-d} \le r)$ equals 1 if $Y_{t-d} \le r$ and 0 otherwise; the expression $I(Y_{t-d} > r)$ is similarly defined. Again, the optimization need only be done with r searched over the observed Y's and d an integer between 1 and p. The conditional least squares approach has the advantage that the threshold parameter can be searched without any constraints. Under mild conditions, including stationarity and that the true conditional mean function is a discontinuous function, Chan (1993) showed that the CLS method is consistent; that is, the estimator approaches the true value with increasing sample size. As the delay is an integer, the consistency property implies that the delay estimator is eventually equal to the true value with very large sample size. Furthermore, the sampling error of the threshold estimator is of the order $1/n$, whereas the sampling error of the other parameters is of order $1/\sqrt{n}$. The faster convergence of the threshold parameter and the delay parameter to their true values implies that in assessing the uncertainty of the autoregressive parameter estimates, the threshold and the delay may be treated as if they were known. Consequently, the autoregressive parameter estimators from the two regimes are approximately independent of each other, and their sampling distributions are approximately the same as those from the ordinary least squares regression with data from the corresponding true regimes. These large-sample distribution results can be lifted to the case of the conditional maximum likelihood estimator provided the true parameter satisfies the regularity conditions alluded to before. Finally, we note that the preceding large-sample properties of the estimator are radically different if the true conditional mean function is continuous; see Chan and Tsay (1998).

In practice, the AR orders in the two regimes need not be identical or known. Thus, an efficient estimation procedure that also estimates the orders is essential. Recall that for linear ARMA models, the AR orders can be estimated by minimizing the AIC. For fixed r and d, the TAR model is essentially fitting two AR models of orders p_1 and p_2, respectively, so that the AIC becomes

$$\text{AIC}(p_1, p_2, r, d) = -2l(r, d) + 2(p_1 + p_2 + 2) \tag{15.7.3}$$

where the number of parameters, excluding r, d, σ_1, and σ_2, equals $p_1 + p_2 + 2$. Now, the minimum AIC (MAIC) estimation method estimates the parameters by minimizing the AIC subject to the constraint that the threshold parameter be searched over some inter-

val that guarantees any regimes have adequate data for estimation. Adding 2 to the minimum AIC so found is defined as the nominal AIC of the estimated threshold model, based on the naive idea of counting the threshold parameter as one additional parameter. Since the threshold parameter generally adds much flexibility to the model, it is likely to add more than one degree of freedom to the model. An asymptotic argument suggests that it may be equivalent to adding three degrees of freedom to the model; see Tong (1990, p. 248).

We illustrate the estimation methods with the predator series. In the estimation, the maximum order is set to be $p = 4$ and $1 \le d \le 4$. This maximum order is the AR order determined by AIC, which is likely to be not smaller than the order of the true TAR model. Alternatively, the order may be determined by cross-validation, which is computer-intensive; see Cheng and Tong (1992). Using the MAIC method with the search of threshold roughly between the tenth and ninetieth percentiles, the table in Exhibit 15.11 displays the nominal AIC value of the estimated TAR model for $1 \le d \le 4$. The nominal AIC is smallest when $d = 3$, so we estimate the delay to be 3. The table in Exhibit 15.12 summarizes the corresponding model fit.

Exhibit 15.11 Nominal AIC of the TAR Models Fitted to the Log(predator) Series for $1 \le d \le 4$

d	AIC	$\hat{r}$	$\hat{p}_1$	$\hat{p}_2$
1	19.04	4.15	2	1
2	12.15	4.048	1	4
3	10.92	4.661	1	4
4	18.42	5.096	3	4

```
> AICM=NULL
> for(d in 1:4)
   {predator.tar=tar(y=log(predator.eq),p1=4,p2=4,d=d,a=.1,b=.9)
> AICM=rbind(AICM,
   c(d,predator.tar$AIC,signif(predator.tar$thd,4),
   predator.tar$p1,predator.tar$p2))}
> colnames(AICM)=c('d','nominal AIC','r','p1','p2')
> rownames(AICM)=NULL
> AICM
```

Although the maximum autoregressive order is 4, the MAIC method selects order 1 for the lower regime and order 4 for the upper regime. The submodel in each regime is estimated by ordinary least squares (OLS) using the data falling in that regime. Hence a less biased estimator of the noise variance may be estimated by the within-regime residual sum of squared errors normalized by the effective sample size which equals the number of data in that regime minus the number of autoregressive parameters (including the intercept) of the corresponding submodel. The "unbiased" noise variance $\tilde{\sigma}_i^2$ of the ith regime relates to its maximum likelihood counterpart by the formula

$$\tilde{\sigma}_i^2 = \frac{n_i}{n_i - p_i - 1}\hat{\sigma}_i^2, \tag{15.7.4}$$

where p_i is the autoregressive order of the ith submodel. Moreover, $(n_i - p_i - 1)\tilde{\sigma}_i^2/\sigma_i^2$ is approximately distributed as χ^2 with $n_i - p_i - 1$ degrees of freedom. For each regime, the t-statistics and corresponding p-values reported in Exhibit 15.12 are identical with the computer output for the case of fitting an autoregressive model with the data falling in that regime. Notice that the coefficients of lags 2 and 3 in the upper regime are not significant, while that of lag 4 is mildly significant at the 5% significance level. Hence, the model for the upper regime may be approximated by a first-order autoregressive model. We shall return to this point later.

Exhibit 15.12 Fitted TAR(2;1,4) Model for the Predator Data: MAIC Method

	Estimate	Std. Error	*t*-statistic	*p*-value
$\hat{d}$	3			
$\hat{r}$	4.661			
Lower Regime (n_1 = 30)				
$\hat{\phi}_{1,0}$	0.262	0.316	0.831	0.41
$\hat{\phi}_{1,1}$	1.02	0.0704	14.4	0.00
$\tilde{\sigma}_1^2$	0.0548			
Upper Regime (n_2 = 23)				
$\hat{\phi}_{2,0}$	4.20	1.28	3.27	0.00
$\hat{\phi}_{2,1}$	0.708	0.202	3.50	0.00
$\hat{\phi}_{2,2}$	–0.301	0.312	–0.965	0.35
$\hat{\phi}_{2,3}$	0.279	0.406	0.686	0.50
$\hat{\phi}_{2,4}$	–0.611	0.273	–2.24	0.04
$\tilde{\sigma}_2^2$	0.0560			

```
> predator.tar.1=tar(y=log(predator.eq),p1=4,p2=4,d=3,a=.1,b=.9,
    print=T)
> tar(y=log(predator.eq),p1=1,p2=4,d=3,a=.1,b=.9,print=T,
    method='CLS') # re-do the estimation using the CLS method
> tar(y=log(predator.eq),p1=4,p2=4,d=3,a=.1,b=.9,print=T,
    method='CLS') # the CLS method does not estimate the AR orders
```

The threshold estimate is 4.661, roughly the 57th percentile. In general, a threshold estimate that is too close to the minimum or the maximum observation may be unreliable due to small sample size in one of the regimes, which, fortunately, is not the case

here. Exhibit 15.12 does not report the standard error of the threshold estimate because its sampling distribution is nonstandard and rather complex. Similarly, the discreteness of the delay estimator renders its standard error useless. However, a parametric bootstrap may be employed to draw inferences on the threshold and the delay parameters. An alternative is to adopt the Bayesian approach of Geweke and Terui (1993). In contrast, the fitted AR(4) model has the coefficient estimates of lags 1 to 4 equal to 0.943 (0.136), −0.171 (0.188), −0.1621 (0.186), and −0.238 (0.136), respectively, with their standard errors enclosed in parentheses; the noise variance is estimated to be 0.0852, which is substantially larger than the noise variances of the TAR(2;1,4) model. Notice that the AR(4) coefficient estimate is close to being nonsignificant, and the AR(2) and AR(3) coefficient estimates are not significant.

An interesting question concerns the interpretation of the two regimes. One way to explore the nature of the regimes is to identify which data value falls in which regime in the time series plot of the observed process. In the time series plot in Exhibit 15.2 on page 387, data falling in the lower regime (that is, those whose lag 3 values are less than 4.661) are drawn as solid circles, whereas those in the upper regime are displayed as open circles. The plot reveals that the estimated lower regime corresponds to the increasing phase of the predator cycles and the upper regime corresponds to the decreasing phase of the predator cycles. A biological interpretation is the following. When the predator number was low one and a half days earlier, the prey species would have been able to increase in the intervening period so that the predator species would begin to thrive. On the other hand, when the predator numbered more than $106 \approx \exp(4.661)$ one and a half days earlier, the prey species crashed in the intervening period so that the predator species would begin to crash. The increasing phase (lower regime) of the predator population tends to be associated with a robust growth of the prey series that may be less affected by other environmental conditions. On the other hand, during the decreasing phase (upper regime), the predator species would be more susceptible to environmental conditions, as they were already weakened by having less food around. This may explain why the lower regime has a slightly smaller noise variance than the upper regime; hence the slight conditional heteroscedasticity. The difference of the noise variance in the two regimes is unlikely to be significant, although the conditional heteroscedasticity is more apparent in the TAR(2;1,1) model to be discussed below. In general, the regimes defined by the relative position of the lag d values of the response are proxies for some underlying latent process that effects the switching between the linear submodels. With more substantive knowledge of the switching mechanism, the threshold mechanism may, however, be explicitly modeled.

While the interpretation of the regimes above is based on the time series plot, it may be confirmed by examining the fitted submodels. The fitted model of the lower regime implies that on the logarithmic scale

$$Y_t = 0.262 + 1.02Y_{t-1} + 0.234e_t \tag{15.7.5}$$

The lag 1 coefficient is essentially equal to 1 and suggests that the predator species had a (median) growth rate of $(\exp(0.262) - 1)100\% \approx 30\%$ every half day, although the intercept is not significant at the 5% level. This submodel is explosive because $Y_t \to \infty$ as $t \to \infty$ if left unchecked.

Interpretation of the fitted model of the upper regime is less straightforward because it is an order 4 model. However, it was suggested earlier that it may be approximated by an AR(1) model. Taking up this suggestion, we reestimated the TAR model with the maximum order being 1 for both regimes.† The threshold estimate is unchanged. The lower regime gains one data case, with less of an initial data requirement, but the autoregressive coefficients are almost unchanged. The fitted model of the upper regime becomes

$$Y_t = 0.517 + 0.807 Y_{t-1} + 0.989 e_t \tag{15.7.6}$$

which is a stationary submodel. The growth rate on the logarithmic scale equals

$$Y_t - Y_{t-1} = 0.517 - 0.193 Y_{t-1} + 0.989 e_t \tag{15.7.7}$$

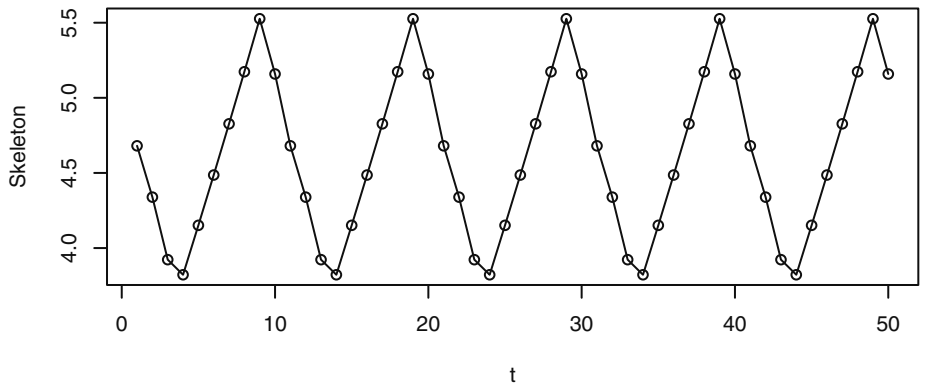
which has a negative median since $Y_{t-1} > 4.661$ on the upper regime. Notice that the conditional heteroscedasticity is more apparent now than the fitted TAR(2;1,4) model. The (nominal) AIC of the TAR(2;1,1) model with $d = 3$ equals 14.78, which is, however, not directly comparable with 10.92 of the TAR(2;1,4) model because of the difference in sample size. Models with different sample sizes may be compared by their nominal AIC per observation. In this case, the normalized AIC increases from 0.206 = 10.92/53 to 0.274 = 14.78/54 when the order is decreased from 4 to 1, suggesting that the TAR(2;1,4) model is preferable to the TAR(2;1,1) model.

Another way to assess a nonlinear model is to examine the long-term (asymptotic) behavior of its skeleton. Recall that the skeleton of a model is obtained by suppressing the noise term from the model; that is, replacing the noise term by 0. The skeleton may diverge to infinity, or it may converge to a limit point, a limit cycle, or a strange attractor; see Chan and Tong (2001) for definitions and further discussion. The skeleton of a stationary ARMA model always converges to some limit point. On the other hand, the skeleton of a stationary nonlinear model may display the full complexity of dynamics alluded to earlier. The skeleton of the fitted TAR(2;1,4) model appears to converge to a limit cycle of period 10, as shown in Exhibit 15.13. The limit cycle is symmetric in the sense that its increase phase and decrease phase are of the same length. The apparent long-run stability of the skeleton suggests that the fitted TAR(2;1,4) model with $d = 3$ is stationary. In general, with the noise term in the model, the dynamic behavior of the model may be studied by simulating some series from the stochastic model. Exhibit 15.14 shows a typical realization from the fitted TAR(2;1,4) model.

---

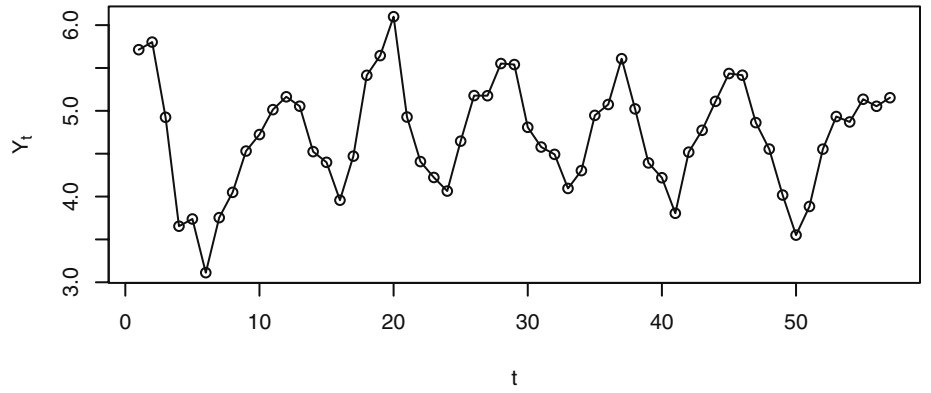
† `predator.tar.2=tar(log(predator.eq),p1=1,p2=1,d=3,a=.1, b=.9, print=T)`

Exhibit 15.13 Skeleton of the TAR(2;1,4) Model for the Predator Series




```
> tar.skeleton(predator.tar.1)
```

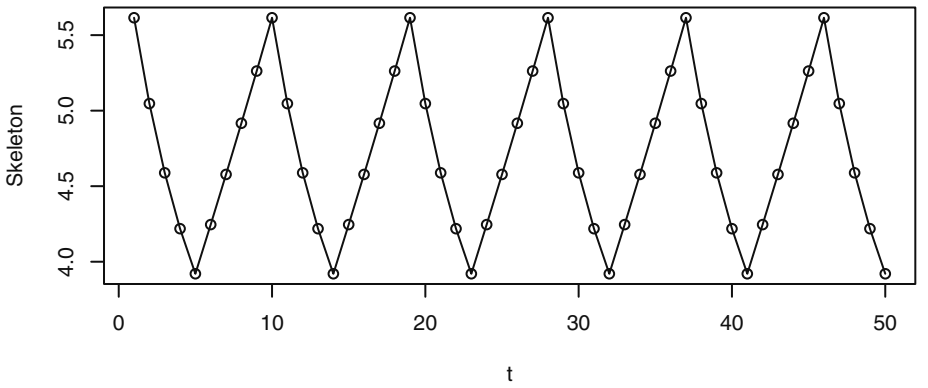
Exhibit 15.14 Simulated TAR(2;1,4) Series




```
> set.seed(356813)
> plot(y=tar.sim(n=57,object=predator.tar.1)$y,x=1:57,
    ylab=expression(Y[t]),xlab=expression(t),type='o')
```

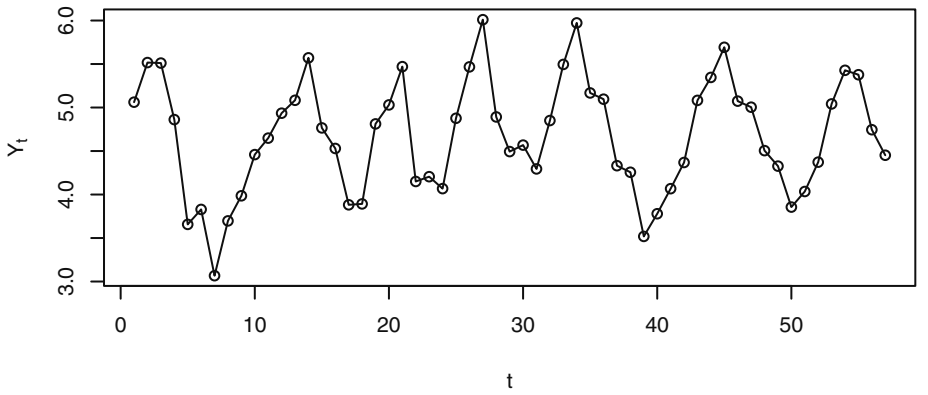
The limit cycle of the skeleton of the fitted TAR(2;1,1) model with $d = 3$ is asymmetric, with the increase phase of length 5 and the decrease phase of length 4; see Exhibit 15.15. A realization of the fitted TAR(2;1,1) model is shown in Exhibit 15.16.

Exhibit 15.15 Skeleton of the First-Order TAR Model for the Predator Series




```
> predator.tar.2=tar(log(predator.eq),p1=1,p2=1,d=3,a=.1,b=.9,
    print=T)
> tar.skeleton(predator.tar.2)
```

Exhibit 15.16 Simulation of the Fitted TAR(2;1,1) Model




```
> set.seed(356813)
> plot(y=tar.sim(n=57,object=predator.tar.2)$y,x=1:57,
    ylab=expression(Y[t]),xlab=expression(t),type='o')
```

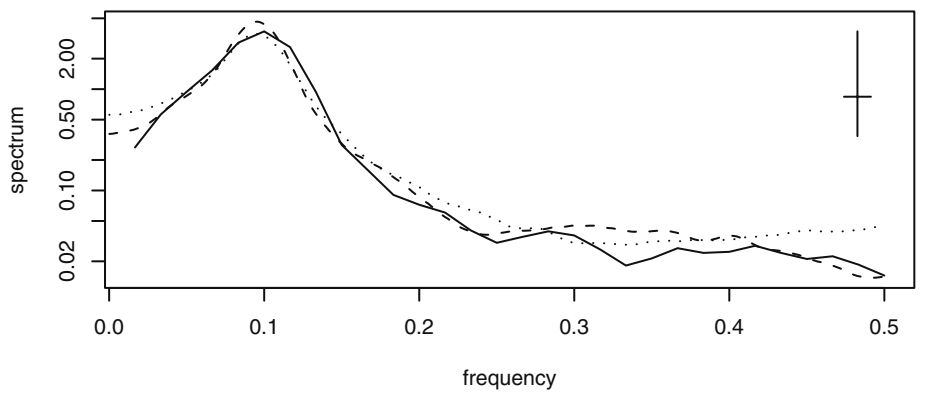
For the predator data, excluding the two initial transient cycles and the last incomplete cycle, the table in Exhibit 15.17 lists the length of the successive increasing and decreasing phases. Observe that the mean length of the increasing phases is 5.4 and that of the decreasing phases is 4.6.

Exhibit 15.17 Length of the Increasing and Decreasing Phases of the Predator Series

Phase	
Increasing	**Decreasing**
6	4
7	5
5	4
4	5
5	5

There is some evidence of asymmetry with a longer increase phase than the decrease phase. Based on the cycle length analysis, the TAR(2;1,1) model appears to pick up the asymmetric cycle property better than the TAR(2;1,4) model, but the latter model gets the cycle length better matched to the observed average cycle length. A more rigorous comparison between the cyclical behavior of a fitted model and that of the data can be done by comparing the spectral density of the data with that of a long realization from the fitted model. Exhibit 15.18 plots the spectrum of the data using a modified Daniell window with a (3,3) span. Also plotted is the spectrum of the fitted TAR(2;1,4) model (dashed line) and that of the fitted TAR(2;1,1) model (dotted line), both of which are based on a simulated realization of size 10,000, a modified Daniell window with a (200,200) span, and 10% tapering. It can be seen that the spectrum of the TAR(2;1,4) model follows that of the predator series quite closely and is slightly better than the simplified TAR(2;1,1) model.

Exhibit 15.18 Spectra of Log(predator) Series, Dashed Line for TAR(2;1,1), Dotted Line for TAR(2;1,4)




```
> set.seed(2357125)
> yy.1.4=tar.sim(predator.tar.1,n=10000)$y
> yy.1=tar.sim(predator.tar.2,n=10000)$y
> spec.1.4=spec(yy.1.4,taper=.1, span=c(200,200),plot=F)
```

```
> spec.1=spec(yy.1,taper=.1, span=c(200,200),plot=F)
> spec.predator=spec(log(predator.eq),taper=.1,
    span=c(3,3),plot=F)
> spec.predator=spec(log(predator.eq),taper=.1,span=c(3,3),
    ylim=range(c(spec.1.4$spec,spec.1$spec,spec.predator$spec)))
> lines(y=spec.1.4$spec,x=spec.1.4$freq,lty=2)
> lines(y=spec.1$spec,x=spec.1$freq,lty=3)
```

We note that the conditional least squares method with the predator data yields the same threshold estimate for $d = 3$ and hence also the other parameter estimates, although this need not always be the case. Finally, a couple of clarifying remarks on the predator series analysis are in order. As the experimental prey series is also available, a bivariate time series analysis may be studied. But it is not pursued here since nonlinear time series analysis with multiple time series is not a well-charted area. Moreover, real biological data are often observational, and abundance data of the prey population are often much noisier than those of the predator population because the predator population tends to be fewer in number than the prey population. Furthermore, predators may switch from their favorite prey food to other available prey species when the former becomes scarce, rendering a more complex prey-predator system. For example, in a good year, hares may be seen hopping around in every corner in the neighborhood, whereas it is rare to spot a lynx, their predator! Thus, biological analysis often focuses on the abundance data of the predator population. Nonetheless, univariate time series analysis of the abundance of the predator species may shed valuable biological insights on the prey-predator interaction; see Stenseth et al. (1998, 1999) for some relevant discussion on a panel of Canadian lynx series. For the lynx data, a TAR(2;2,2) model with delay equal to 2 is the prototypical model, with delay 2 lending some nice biological interpretations. We note that, for the predator series, delay 2 is very competitive; see Exhibit 15.11, and hence may be preferred on biological grounds. In one exercise, we ask the reader to fit a TAR model for the predator series with delay set to 2 and interpret the findings by making use of the framework studied in Stenseth et al. (1998, 1999).

15.8 Model Diagnostics

In Section 15.7, we introduced some model diagnostic techniques; for example, skeleton analysis and simulation. Here, we discuss some formal statistical approaches to model diagnostics via residual analysis. The raw residuals are defined as subtracting the fitted value from the data, where the tth fitted value is the estimated conditional mean of Y_t given past values of Y's; that is, the residuals $\hat{\varepsilon}_t$ are given by

$$\begin{aligned}\hat{\varepsilon}_t = Y_t &- \{\hat{\phi}_{1,0} + \hat{\phi}_{1,1} Y_{t-1} + \cdots + \hat{\phi}_{1,p} Y_{t-p}\} I(Y_{t-\hat{d}} \le \hat{r}) \\ &- \{\hat{\phi}_{2,0} + \hat{\phi}_{2,1} Y_{t-1} + \cdots + \hat{\phi}_{2,p} Y_{t-p}\} I(Y_{t-\hat{d}} > \hat{r})\end{aligned} \tag{15.8.1}$$

These are the same as the raw residuals from the fitted submodels. The standardized residuals are obtained by normalizing the raw residuals by their appropriate standard deviations:

$$\hat{e}_t = \frac{\hat{\varepsilon}_t}{\hat{\sigma}_1 I(Y_{t-\hat{d}} \le \hat{r}) + \hat{\sigma}_2 I(Y_{t-\hat{d}} > \hat{r})} \tag{15.8.2}$$

that is, raw residuals from the lower (upper) regime are normalized by the noise standard deviation estimate of the lower (upper) regime. As in the linear case, the time series plot of the standardized residuals should look random, as they should be approximately independent and identically distributed if the TAR model is the true data mechanism; that is, if the TAR model is correctly specified. As before, we look for the presence of outliers and any systematic pattern in such a plot, in which case it may provide a clue for specifying a more appropriate model. The independence assumption of the standardized errors can be checked by examining the sample ACF of the standardized residuals. Nonconstant variance may be checked by examining the sample ACF of the squared standardized residuals or that of the absolute standardized residuals.

Here, we consider the generalization of the portmanteau test based on some overall measure of the magnitude of the residual autocorrelations. The reader may want to review the discussion in Section 12.5 on page 301, where we explain that even if the model is correctly specified, the residuals are generally dependent and so are their sample autocorrelations. Unlike the case of linear ARIMA models, the dependence of the residuals necessitates the employment of a (complex) quadratic form of the residual autocorrelations:

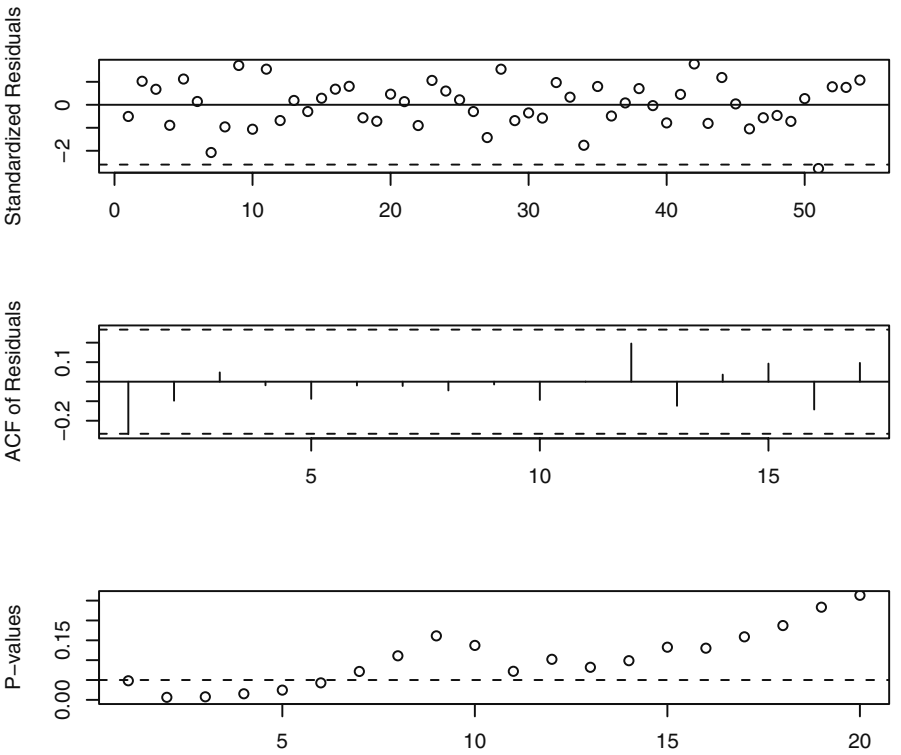
$$B_m = n_{\text{eff}} \sum_{i=1}^{m} \sum_{j=1}^{m} q_{i,j} \hat{\rho}_i \hat{\rho}_j \tag{15.8.3}$$

where $n_{\text{eff}} = n - \max(p_1, p_2, d)$ is the effective sample size, $\hat{\rho}_i$ the ith-lag sample autocorrelation of the standardized residuals, and $q_{i,j}$ some model-dependent constants given in Appendix L on page 421. If the true model is a TAR model, $\hat{\rho}_i$ are likely close to zero and so is B_m, but B_m tends to be large if the model specification is incorrect. The quadratic form is designed so that B_m is approximately distributed as χ^2 with m degrees of freedom. Mathematical theory predicts that the χ^2 distribution approximation is generally more accurate with larger sample size and relatively small m as compared with the sample size.

In practice, the p-value of B_m may be plotted against m over a range of m values to provide a more comprehensive assessment of the independence assumption on the standardized errors. The bottom figure of Exhibit 15.19 reports the portmanteau test of the TAR(2;1,1) model fitted to the predator series discussed earlier for $1 \le m \le 12$. The top figure there is the time series plot of the standardized residuals. Except for a possible outlier, the plot shows no particular pattern. The middle figure is the ACF plot of the standardized residuals. The confidence band is based on the simple $1.96/\sqrt{n}$ rule and should be regarded as a rough guide on the significance of the residual ACF. It suggests that the lag 1 residual autocorrelation is significant. The more rigorous portmanteau tests are all significant for $m \le 6$, suggesting a lack of fit for the TAR(2;1,1) model. Similar diagnostics for the TAR(2;1,4) model are shown in Exhibit 15.20. Now, the only potential problem is a possible outlier. However, the fitted model changed little upon deleting the last four data points, including the potential outlier; hence we conclude that

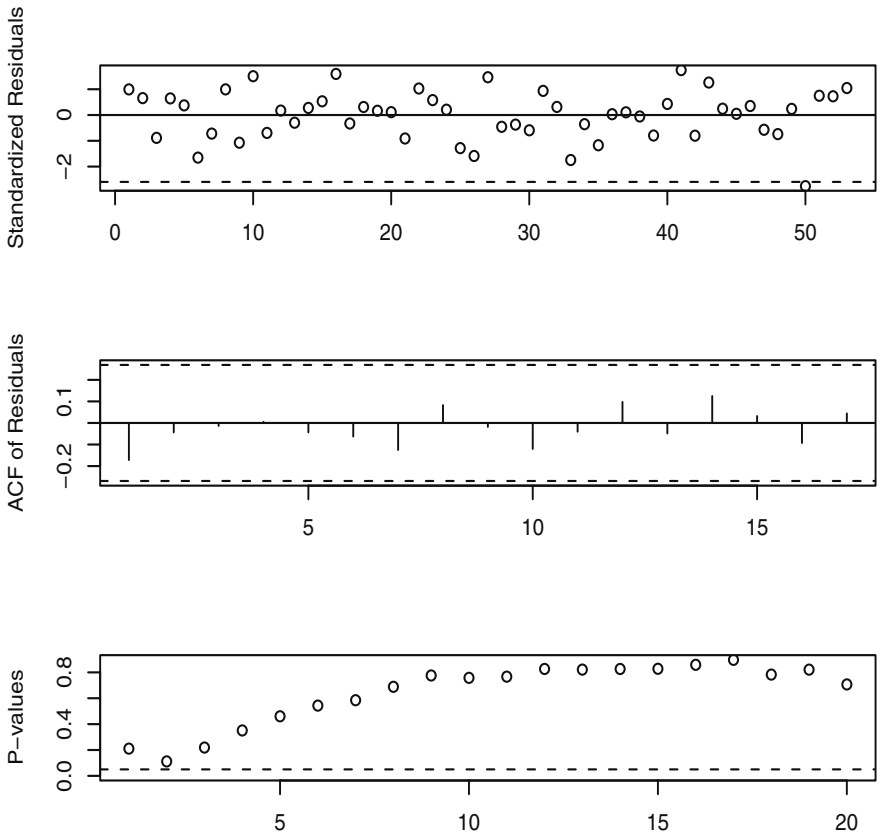
the fitted TAR(2;1,4) model is fairly robust. Exhibit 15.21 displays the QQ normal score plot of the standardized residuals, which is apparently straight and hence the errors appear to be normally distributed. In summary, the fitted TAR(2;1,4) model provides a good fit to the predator series.

Exhibit 15.19 Model Diagnostics of the First-Order TAR Model: Predator Series



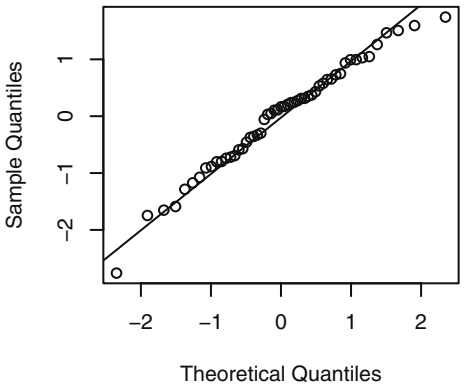

```
> win.graph(width=4.875,height=4.5)
> tsdiag(predator.tar.2,gof.lag=20)
```

Exhibit 15.20 Model Diagnostics for the TAR(2;1,4) Model: Predator Series




```
> tsdiag(predator.tar.1,gof.lag=20)
```

Exhibit 15.21 QQ Normal Plot of the Standardized Residuals




```
> win.graph(width=2.5,height=2.5,pointsize=8)
> qqnorm(predator.tar.1$std.res); qqline(predator.tar.1$std.res)
```

15.9 Prediction

In this section, we consider the problem of predicting future values from a TAR process. In practice, prediction is based on an estimated TAR model. But, as in the case of ARIMA models, the uncertainty due to parameter estimation is generally small compared with the natural variation of the underlying process. So, we shall proceed below as if the fitted model were the true model. The uncertainty of a future value, say $Y_{t+\ell}$, is completely characterized by its conditional probability distribution given the current and past data Y_t, Y_{t-1},…, referred to as the ℓ-step-ahead predictive distribution below. For ARIMA models with normal errors, all predictive distributions are normal, which greatly simplifies the computation of a predictive interval, as it suffices to find the mean and variance of the predictive distribution. However, for nonlinear models, the predictive distributions are generally nonnormal and often intractable. Hence, a prediction interval may have to be computed by brute force via simulation. The simulation approach may be best explained in the context of a first-order nonlinear autoregressive model:

$$Y_{t+1} = h(Y_t, e_{t+1}) \tag{15.9.1}$$

Given $Y_t = y_t$, $Y_{t-1} = y_{t-1}$,…, we have $Y_{t+1} = h(y_t, e_{t+1})$ so a realization of Y_{t+1} from the one-step-ahead predictive distribution can be obtained by drawing e_{t+1} from the error distribution and computing $h(y_t, e_{t+1})$. Repeating this procedure independently B times, say 1000 times, we get a random sample of B values from the one-step-ahead predictive distribution. The one-step-ahead predictive mean may be estimated by the sample mean of these B values. However, it is important to inspect the shape of the one-step-ahead predictive distribution in order to decide how best to summarize the predictive information. For example, if the predictive distribution is multimodal or very

skewed, the one-step-ahead predictive mean need not be an appropriate point predictor. A generally useful approach is to construct a 95% prediction interval for Y_{t+1}; for example, the interval defined by the 2.5th percentile to the 97.5th percentile of the simulated B values.

The simulation approach can be readily extended to finding the ℓ-step-ahead predictive distribution for any integer $\ell \geq 2$ by iterating the nonlinear autoregression.

$$\left.\begin{aligned} Y_{t+1} &= h(Y_t, e_{t+1}) \\ Y_{t+2} &= h(Y_{t+1}, e_{t+2}) \\ &\vdots \\ Y_{t+\ell} &= h(Y_{t+\ell-1}, e_{t+\ell}), \end{aligned}\right\} \tag{15.9.2}$$

where $Y_t = y_t$ and $\{e_{t+1},\ldots,e_{t+\ell}\}$ is a random sample of ℓ values drawn from the error distribution. This procedure may be repeated B times to yield a random sample from the ℓ-step-ahead predictive distribution, with which we can compute prediction intervals of $Y_{t+\ell}$ or any other predictive summary statistic.

Indeed, the ℓ-tuple $(Y_{t+1},\ldots,Y_{t+\ell})$ is a realization from the joint predictive distribution of the first ℓ-step-ahead predictions. So, the procedure above actually yields a random sample of B vectors from the joint predictive distribution of the first ℓ-step-ahead predictions.

Henceforth in this section, we focus on the prediction problem when the true model is a TAR model. Fortunately, the simulation approach is not needed for computing the one-step-ahead predictive distribution in the case of a TAR model. To see this, consider the simple case of a first-order TAR model. In this case, Y_{t+1-d} is known, so that the regime for Y_{t+1} is known. If $Y_{t+1-d} \leq r$, then Y_{t+1} follows the AR(1) model

$$Y_{t+1} = \phi_{1,0} + \phi_{1,1} Y_t + \sigma_1 e_{t+1} \tag{15.9.3}$$

Because $Y_t = y_t$ is fixed, the conditional distribution of Y_{t+1} is normal with mean equal to $\phi_{1,0} + \phi_{1,1} y_t$ and variance σ_1^2. Similarly, if $Y_t > r$, Y_{t+1} follows the AR(1) model of the upper regime so that, conditionally, it is normal with mean $\phi_{2,0} + \phi_{2,1} y_t$ and variance σ_2^2. A similar argument shows that, for any TAR model, the one-step-ahead predictive distribution is normal. The predictive mean is, however, a piecewise linear function, and the predictive standard deviation is piecewise constant.

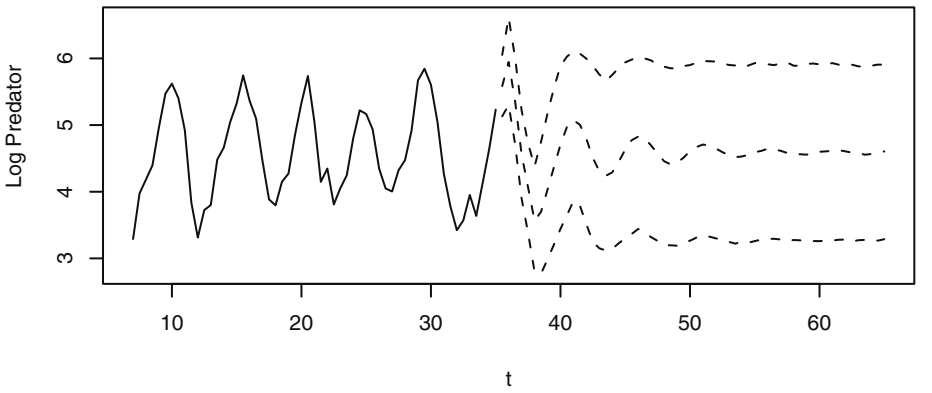
Similarly, it can be shown that if $\ell \leq d$, then the ℓ-step-ahead predictive distribution of a TAR model is also normal. But if $\ell > d$, the ℓ-step-ahead predictive distribution is no longer normal. The problem can be illustrated in the simple case of a first-order TAR model with $d = 1$ and $\ell = 2$. While Y_{t+1} follows a *fixed* linear model determined by the observed value of Y_t, Y_{t+2} may be in the lower or upper regime, depending on the random value of Y_{t+1}. Suppose that $y_t \leq r$. Now, Y_{t+1} falls in the lower regime if $Y_{t+1} = \sigma_1 e_{t+1} + \phi_{1,0} + \phi_{1,1} y_t \leq r$, which happens with probability $p_t = Pr(\sigma_1 e_{t+1} + \phi_{1,0} + \phi_{1,1} y_t \leq r)$ and in which case

$$\begin{aligned} Y_{t+2} &= \sigma_1 e_{t+2} + \phi_{1,0} + \phi_{1,1} Y_{t+1} \\ &= \sigma_1 e_{t+2} + \phi_{1,1}\sigma_1 e_{t+1} + \phi_{1,1}\phi_{1,0} + \phi_{1,1}^2 y_t + \phi_{1,0} \end{aligned} \tag{15.9.4}$$

which is a normal distribution with mean equal to $\phi_{1,1}\phi_{1,0} + \phi_{1,1}^2 y_t + \phi_{1,0}$ and variance $\sigma_1^2 + \phi_{1,1}^2\sigma_1^2$. On the other hand, with probability $1 - p_t$, Y_{t+1} falls in the upper regime, in which case the conditional distribution of Y_{t+2} is normal but with mean $\phi_{2,1}(\phi_{1,0} + \phi_{1,1}y_t) + \phi_{2,0}$ and variance $\sigma_2^2 + \phi_{2,1}^2\sigma_1^2$. Therefore, the conditional distribution of Y_{t+2} is a mixture of two normal distributions. Note that the mixture probability p_t depends on y_t. In particular, the higher-step-ahead predictive distributions are nonnormal for a TAR model if $\ell > d$, and so we have to resort to simulation to find the predictive distributions.

As an example, we compute the prediction intervals for the logarithmically transformed predator data based on the fitted TAR(2;1,4) model with $d = 3$; see Exhibit 15.22, where the middle dashed line is the median of the predictive distribution and the other dashed lines are the 2.5th and 97.5th percentiles of the predictive distribution.

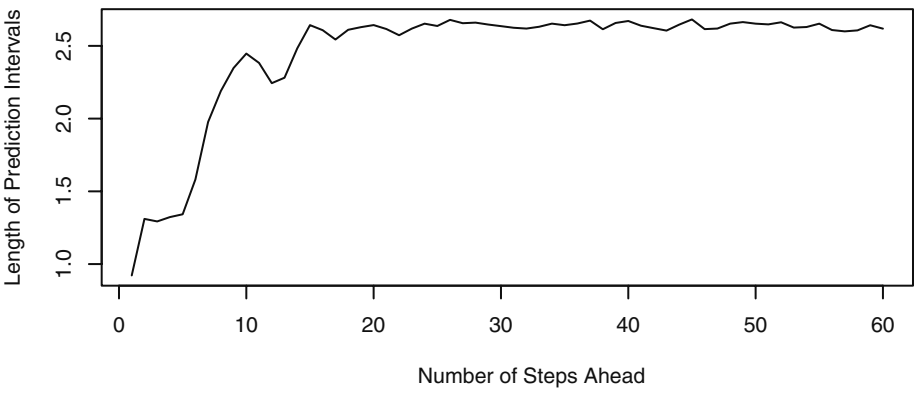
Exhibit 15.22 Prediction of the Predator Series




```
> set.seed(2357125)
> win.graph(width=4.875,height=2.5,pointsize=8)
> pred.predator=predict(predator.tar.1,n.ahead=60,n.sim=10000)
> yy=ts(c(log(predator.eq),pred.predator$fit),frequency=2,
    start=start(predator.eq))
> plot(yy,type='n',ylim=range(c(yy,pred.predator$pred.interval)),
    ylab='Log Predator',xlab=expression(t))
> lines(log(predator.eq))
> lines(window(yy, start=end(predator.eq)+c(0,1)),lty=2)
> lines(ts(pred.predator$pred.interval[2,],
    start=end(predator.eq)+c(0,1),freq=2),lty=2)
> lines(ts(pred.predator$pred.interval[1,],
    start=end(predator.eq)+c(0,1),freq=2),lty=2)
```

The simulation size here is 10,000. In practice, a smaller size such as 1000 may be adequate. The median of the predictive distribution can serve as a point predictor. Notice that the predictive medians display the cyclical pattern of the predator data initially and then approach the long-run median with increasing number of steps ahead. Similarly, the predictive intervals approach the interval defined by the 2.5th and 97.5th percentiles of the stationary distribution of the fitted TAR model. However, a new feature is that prediction need not be less certain with increasing number of steps ahead, as the length of the prediction intervals does not increase monotonically with increasing number of steps ahead; see Exhibit 15.23. This is radically different from the case of ARIMA models, for which the prediction variance always increases with the number of prediction steps ahead.

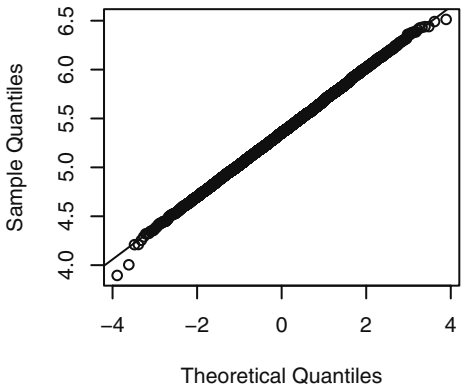
Exhibit 15.23 Width of the 95% Prediction Intervals Against Lead Time




```
> plot(ts(apply(pred.predator$pred.interval,2,
    function(x){x[2]-x[1]})),
    ylab='Length of Prediction Intervals',
    xlab='Number of Steps Ahead')
```

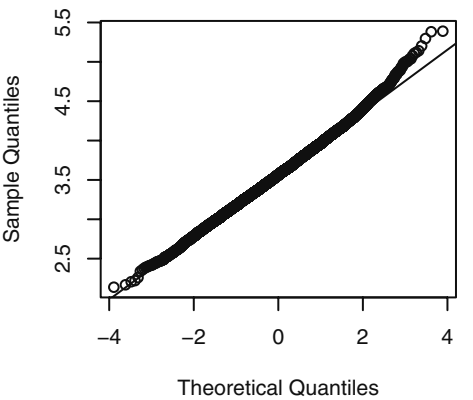
Recall that, for the TAR model, the prediction distribution is normal if and only if the number of steps ahead $\ell \leq d$. Exhibit 15.24 shows the QQ normal score plot of the three-step-ahead predictive distribution, which is fairly straight. On the other hand, the QQ normal score plot of the six-step-ahead predictive distribution (Exhibit 15.25) is consistent with nonnormality.

Exhibit 15.24 QQ Normal Plot of the Three-Step-Ahead Predictive Distribution




```
> win.graph(width=2.5,height=2.5,pointsize=8)
> qqnorm(pred.predator$pred.matrix[,3])
> qqline(pred.predator$pred.matrix[,3])
```

Exhibit 15.25 QQ Normal Plot of the Six-Step-Ahead Predictive Distribution




```
> qqnorm(pred.predator$pred.matrix[,6])
> qqline(pred.predator$pred.matrix[,6])
```

15.10 Summary

In this chapter, we have introduced an important nonlinear times serie model—the threshold model. We have shown how to test for nonlinearity and, in particular, for threshold nonlinearity. We then proceeded to consider the estimation of the unknown parameters in these models using both the minimum AIC (MAIC) criterion and the conditional least squares approach. As with all models, we learned how to criticize them through various model diagnostics, including an extended portmanteau test. Finally, we demonstrated how to form predictions from threshold models, including the calculation and display of prediction intervals. Several substantial examples were used to illustrate the methods and techniques discussed.

EXERCISES

15.1 Fit a TAR model for the predator series with delay set to 2, and interpret the findings by making use of the framework studied in Stenseth et al. (1998, 1999). (You may first want to check whether or not their framework is approximately valid for the TAR model.) Also, compare the fitted model with the TAR(2;1,4) model with delay 3 reported in the text. (The data file is named veilleux.)

15.2 Fit a TAR model to the square-root-transformed relative sunspot data, and examine its goodness of fit. Interpret the fitted TAR model. (The data file is named spots.)

15.3 Predict the annual relative sunspot numbers for ten years using the fitted model obtained in Exercise 15.2. Draw the prediction intervals and the predicted medians. (The data file is named spots.)

15.4 Examine the long-run behavior of the skeleton of the fitted model for the relative sunspot data. Is the fitted model likely to be stationary? Explain your answer.

15.5 Simulate a series of size 1000 from the TAR model fitted to the relative sunspot data. Compute the spectrum of the simulated realization and compare it with the spectrum of the data. Does the fitted model capture the correlation structure of the data?

15.6 Draw the lagged regression plots for the square-root-transformed hare data. Is there any evidence that the hare data are nonlinear? (The data file is named hare.)

15.7 Carry out formal tests (Keenan's test, Tsay's test, and threshold likelihood ratio test) for nonlinearity for the hare data. Is the hare abundance process nonlinear? Explain your answer. (The data file is named hare.)

15.8 Assuming that the hare data are nonlinear, fit a TAR model to the hare data and examine the goodness of fit. (The data file is named hare.)

15.9 This exercise assumes that the reader is familiar with Markov chain theory. Consider a simple TAR model that is piecewise constant:

$$Y_t = \begin{cases} \phi_{1,0} + \sigma_1 e_t, & \text{if } Y_{t-1} \le r \\ \phi_{2,0} + \sigma_2 e_t, & \text{if } Y_{t-1} > r \end{cases}$$

where $\{e_t\}$ are independent standard normal random variables. Let $R_t = 1$ if $Y_t \le r$ and 2 otherwise, which is a Markov chain.

(a) Find the transition probability matrix of R_t and its stationary distribution.

(b) Derive the stationary distribution of $\{Y_t\}$.

(c) Find the lag 1 autocovariance of the TAR process.

Appendix L: The Generalized Portmanteau Test for TAR

The basis of the portmanteau test is the result that, if the TAR model is correctly specified, $\hat{\rho}_1, \hat{\rho}_2, \ldots, \hat{\rho}_m$ are approximately jointly normally distributed with zero mean and covariances $Cov(\hat{\rho}_i, \hat{\rho}_j) = q_{ij}$, where Q is an $m \times m$ matrix whose (i,j) element equals q_{ij} and whose formula is given below; See Chan (2008) for a proof of this result. It can be shown that $Q = I - UV^{-1}U^T$ where I is an $m \times m$ identity matrix,

$$U = E\left\{ \begin{bmatrix} e_{t-1} \\ e_{t-2} \\ \vdots \\ e_{t-m} \end{bmatrix} [I_t, Y_{t-1}I_t, \ldots, Y_{t-p}I_t, (1-I_t), Y_{t-1}(1-I_t), \ldots, Y_{t-p_2}(1-I_t)] \right\}$$

where $I_t = I(Y_{t-d} \le r)$, the expectation of a matrix is taken elementwise, and

$$V = E\left\{ \begin{bmatrix} I_t \\ Y_{t-1}I_t \\ \vdots \\ Y_{t-p_1}I_t \\ (1-I_t) \\ Y_{t-1}(1-I_t) \\ \vdots \\ Y_{t-p_2}(1-I_t) \end{bmatrix} [I_t, Y_{t-1}I_t, \ldots, Y_{t-p}I_t, (1-I_t), Y_{t-1}(1-I_t), \ldots, Y_{t-p_2}(1-I_t)] \right\}$$

These expectations can be approximated by sample averages computed with the true errors replaced by the standardized residuals and the unknown parameters by their estimates. For example, $E\{e_{t-1}I(Y_{t-d} \le r)\}$ can be approximated by

$$\frac{1}{n}\sum_{t=1}^{n} \hat{e}_{t-1} I(Y_{t-\hat{d}} \le \hat{r})$$

where the initial standardized residuals $\hat{e}_t = 0$ for $t \le \max(p_1, p_2, \hat{d})$.