The AUTOREG Procedure

Overview: AUTOREG Procedure

The AUTOREG procedure estimates and forecasts linear regression models for time series data when the errors are autocorrelated or heteroscedastic. The autoregressive error model is used to correct for autocorrelation, and the generalized autoregressive conditional heteroscedasticity (GARCH) model and its variants are used to model and correct for heteroscedasticity.

When time series data are used in regression analysis, often the error term is not independent through time. Instead, the errors are serially correlated (autocorrelated). If the error term is autocorrelated, the efficiency of ordinary least squares (OLS) parameter estimates is adversely
affected and standard error estimates are biased.

The autoregressive error model corrects for serial correlation. The AUTOREG procedure can fit autoregressive error models of any order and can fit subset autoregressive models. You can also specify stepwise autoregression to select the autoregressive error model automatically.

To diagnose autocorrelation, the AUTOREG procedure produces generalized Durbin-Watson (DW) statistics and their marginal probabilities. Exact p-values are reported for generalized DW tests to any specified order. For models with lagged dependent regressors, PROC AUTOREG performs the Durbin t test and the Durbin h test for first-order autocorrelation and reports their marginal significance levels.

Ordinary regression analysis assumes that the error variance is the same for all observations. When the error variance is not constant, the data are said to be heteroscedastic, and ordinary least squares estimates are inefficient. Heteroscedasticity also affects the accuracy of forecast confidence limits. More efficient use of the data and more accurate prediction error estimates can be made by models that take the heteroscedasticity into account.

To test for heteroscedasticity, the AUTOREG procedure uses the portmanteau Q test statistics (McLeod and Li; 1983), Engle’s Lagrange multiplier tests (Engle; 1982), tests from Lee and King (1993), and tests from Wong and Li (1995). Test statistics and significance p-values are reported for conditional heteroscedasticity at lags 1 through 12. The Bera-Jarque normality test statistic and its significance level are also reported to test for conditional nonnormality of residuals. The following tests for independence are also supported by the AUTOREG procedure for residual analysis and diagnostic checking: Brock-Dechert-Scheinkman (BDS) test, runs test, turning point test, and the rank version of the von Neumann ratio test.

The family of GARCH models provides a means of estimating and correcting for the changing variability of the data. The GARCH process assumes that the errors, although uncorrelated, are not independent, and it models the conditional error variance as a function of the past realizations of the series.

The AUTOREG procedure supports the following variations of the GARCH models:

- generalized ARCH (GARCH)
- integrated GARCH (IGARCH)
- exponential GARCH (EGARCH)
- quadratic GARCH (QGARCH)
- threshold GARCH (TGARCH)
- power GARCH (PGARCH)
- GARCH-in-mean (GARCH-M)

For GARCH-type models, the AUTOREG procedure produces the conditional prediction error variances in addition to parameter and covariance estimates.

The AUTOREG procedure can also analyze models that combine autoregressive errors and GARCH-type heteroscedasticity. PROC AUTOREG can output predictions of the conditional mean and variance for models with autocorrelated disturbances and changing conditional error variances over time.

Four estimation methods are supported for the autoregressive error model:

- Yule-Walker
- iterated Yule-Walker
- unconditional least squares
- exact maximum likelihood

The maximum likelihood method is used for GARCH models and for mixed AR-GARCH models.

The AUTOREG procedure produces forecasts and forecast confidence limits when future values of the independent variables are included in the input data set. PROC AUTOREG is a useful tool for forecasting because it uses the time series part of the model in addition to the systematic part in generating predicted values. The autoregressive error model takes into account recent departures from the trend in producing forecasts.

The AUTOREG procedure permits embedded missing values for the independent or dependent variables. The procedure should be used only for ordered and equally spaced time series data.
The AUTOREG Procedure

Getting Started: AUTOREG Procedure

Ordinary regression analysis is based on several statistical assumptions. One key assumption is that the errors are independent of each other. However, with time series data, the ordinary regression residuals usually are correlated over time. It is not desirable to use ordinary regression analysis for time series data since the assumptions on which the classical linear regression model is based will usually be violated.

Violation of the independent errors assumption has three important consequences for ordinary regression. First, statistical tests of the significance of the parameters and the confidence limits for the predicted values are not correct. Second, the estimates of the regression coefficients are not as efficient as they would be if the autocorrelation were taken into account. Third, since the ordinary regression residuals are not independent, they contain information that can be used to improve the prediction of future values.

The AUTOREG procedure solves this problem by augmenting the regression model with an autoregressive model for the random error, thereby accounting for the autocorrelation of the errors. Instead of the usual regression model, the following autoregressive error model is used:

\[ y_t = \beta' + \varepsilon_t \]

\[ \varepsilon_t = -\phi_2 \varepsilon_{t-2} - \ldots - \phi_m \varepsilon_{t-m} + \varepsilon_t \]

\[ \varepsilon_t \sim \mathcal{N}(0, \sigma^2) \]

The notation \( \varepsilon_t \sim \mathcal{N}(0, \sigma^2) \) indicates that each \( \varepsilon_t \) is normally and independently distributed with mean 0 and variance \( \sigma^2 \).

By simultaneously estimating the regression coefficients \( \beta \) and the autoregressive error model parameters \( \phi \), the AUTOREG procedure corrects the regression estimates for autocorrelation. Thus, this kind of regression analysis is often called \textit{autoregressive error correction} or \textit{serial correlation correction}.

Example of Autocorrelated Data

A simulated time series is used to introduce the AUTOREG procedure. The following statements generate a simulated time series \( Y \) with second-order autocorrelation:

```sas
/* Regression with Autocorrelated Errors */
data a;
  ul = 0; ull = 0;
  do time = -10 to 36;
    u = +1.3 * ul - .5 * ull + 2*rannor(12346);
    y = 10 + .5 * time + u;
    if time > 0 then output;
    ull = ul; ul = u;
  end;
run;
```

The series \( Y \) is a time trend plus a second-order autoregressive error. The model simulated is

\[ y_t = 10 + 0.5t + \varepsilon_t \]
\[ \varepsilon_t = 1.3 \varepsilon_{t-1} - 0.5 \varepsilon_{t-2} + \varepsilon_t \]
\[ \varepsilon_t \sim \mathcal{N}(0,4) \]
The following statements plot the simulated time series Y. A linear regression trend line is shown for reference.

```sas
title 'Autocorrelated Time Series';
proc sgplot data=a noautolegend;
  series x=time y=y / markers;
  reg x=time y=y/ lineattrs=(color=black);
run;
```

The plot of series Y and the regression line are shown in Figure 8.1.

**Figure 8.1 Autocorrelated Time Series**

![Autocorrelated Time Series](image)

Note that when the series is above (or below) the OLS regression trend line, it tends to remain above (below) the trend for several periods. This pattern is an example of positive autocorrelation.

Time series regression usually involves independent variables other than a time trend. However, the simple time trend model is convenient for illustrating regression with autocorrelated errors, and the series Y shown in Figure 8.1 is used in the following introductory examples.

**Ordinary Least Squares Regression**

To use the AUTOREG procedure, specify the input data set in the PROC AUTOREG statement and specify the regression model in a MODEL statement. Specify the model by first naming the dependent variable and then listing the regressors after an equal sign, as is done in other SAS regression procedures. The following statements regress Y on TIME by using ordinary least squares:

```sas
proc autoreg data=a;
  model y = time;
run;
```

The AUTOREG procedure output is shown in Figure 8.2.

**Figure 8.2 PROC AUTOREG Results for OLS Estimation**

<table>
<thead>
<tr>
<th>Dependent Variable</th>
<th>y</th>
</tr>
</thead>
</table>

![Ordinary Least Squares Estimates](table)

<table>
<thead>
<tr>
<th>Ordinary Least Squares Estimates</th>
</tr>
</thead>
<tbody>
<tr>
<td>SSE 214.953429</td>
</tr>
<tr>
<td>MSE 6.32216</td>
</tr>
</tbody>
</table>

The AUTOREG Procedure
The output first shows statistics for the model residuals. The model root mean square error (Root MSE) is 2.51, and the model $R^2$ is 0.82. Notice that two $R^2$ statistics are shown, one for the regression model (Reg Rsq) and one for the full model (Total Rsq) that includes the autoregressive error process, if any. In this case, an autoregressive error model is not used, so the two $R^2$ statistics are the same.

Other statistics shown are the sum of square errors (SSE), mean square error (MSE), mean absolute error (MAE), mean absolute percentage error (MAPE), error degrees of freedom (DFE, the number of observations minus the number of parameters), the information criteria SBC, HQC, AIC, and AICC, and the Durbin-Watson statistic. (Durbin-Watson statistics, MAE, MAPE, SBC, HQC, AIC, and AICC are discussed in the section Goodness-of-fit Measures and Information Criteria later in this chapter.)

The output then shows a table of regression coefficients, with standard errors and $t$ tests. The estimated model is

$$y_t = 8.2308 + 0.5021t + \varepsilon_t$$

$Est. Var(\varepsilon_t) = 6.32$

The OLS parameter estimates are reasonably close to the true values, but the estimated error variance, 6.32, is much larger than the true value, 4.

Autoregressive Error Model

The following statements regress $Y$ on $TIME$ with the errors assumed to follow a second-order autoregressive process. The order of the autoregressive model is specified by the NLAG=2 option. The Yule-Walker estimation method is used by default. The example uses the METHOD=ML option to specify the exact maximum likelihood method instead.

```
proc autoreg data=a;
  model y = time / nlag=2 method=ml;
run;
```

The first part of the results is shown in Figure 8.3. The initial OLS results are produced first, followed by estimates of the autocorrelations computed from the OLS residuals. The autocorrelations are also displayed graphically.

**Figure 8.3 Preliminary Estimate for AR(2) Error Model**

Autocorrelated Time Series
The maximum likelihood estimates are shown in Figure 8.4. Figure 8.4 also shows the preliminary Yule-Walker estimates used as starting values for the iterative computation of the maximum likelihood estimates.

**Figure 8.4 Maximum Likelihood Estimates of AR(2) Error Model**

<table>
<thead>
<tr>
<th>Lag</th>
<th>Covariance</th>
<th>Correlation</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>1.000000</td>
<td>1.000000</td>
</tr>
<tr>
<td>1</td>
<td>0.756451</td>
<td>-0.756451</td>
</tr>
<tr>
<td>2</td>
<td>0.338995</td>
<td>-0.338995</td>
</tr>
</tbody>
</table>

Preliminary MSE: 1.7943

The diagnostic statistics and parameter estimates tables in Figure 8.4 have the same form as in the OLS output, but the values shown are for the autoregressive error model. The MSE for the autoregressive model is 1.71, which is much smaller than the true value of 4. In small samples, the autoregressive error model tends to underestimate $\sigma^2$, while the OLS MSE overestimates $\sigma^2$.

Notice that the total $R^2$ statistic computed from the autoregressive model residuals is 0.954, reflecting the improved fit from the use of past residuals to help predict the next $Y$ value. The Reg Rsq value 0.728 is the $R^2$ statistic for a regression of transformed variables adjusted for the estimated autocorrelation. (This is not the $R^2$ for the estimated trend line. For details, see the section Goodness-of-fit Measures and Information Criteria later in this chapter.)
The parameter estimates table shows the ML estimates of the regression coefficients and includes two additional rows for the estimates of the autoregressive parameters, labeled AR(1) and AR(2).

The estimated model is

\[
\begin{align*}
y_t &= 7.88 + 0.5096t + \nu_t, \\
\nu_t &= 1.25\nu_{t-1} - 0.628\nu_{t-3} + \varepsilon_t, \\
\text{Est. Var}(\varepsilon_t) &= 1.71
\end{align*}
\]

Note that the signs of the autoregressive parameters shown in this equation for \(y_t\) are the reverse of the estimates shown in the AUTOREG procedure output. Figure 8.4 also shows the estimates of the regression coefficients with the standard errors recomputed on the assumption that the autoregressive parameter estimates equal the true values.

**Predicted Values and Residuals**

The AUTOREG procedure can produce two kinds of predicted values and corresponding residuals and confidence limits. The first kind of predicted value is obtained from only the structural part of the model, \(\psi_t\). This is an estimate of the unconditional mean of the response variable at time \(t\). For the time trend model, these predicted values trace the estimated trend. The second kind of predicted value includes both the structural part of the model and the predicted values of the autoregressive error process. The full model (conditional) predictions are used to forecast future values.

Use the OUTPUT statement to store predicted values and residuals in a SAS data set and to output other values such as confidence limits and variance estimates. The P= option specifies an output variable to contain the full model predicted values. The PM= option names an output variable for the predicted mean. The R= and RM= options specify output variables for the corresponding residuals, computed as the actual value minus the predicted value.

The following statements store both kinds of predicted values in the output data set. (The printed output is the same as previously shown in Figure 8.3 and Figure 8.4.)

```sas
proc autoreg data=a;
  model y = time / nlag=2 method=ml;
  output out=p p=yhat pm=trendhat;
run;
```

The following statements plot the predicted values from the regression trend line and from the full model together with the actual values:

```sas
title 'Predictions for Autocorrelation Model';
proc sgplot data=p;
  scatter x=time y=y / markerattrs=(color=blue);
  series x=time y=yhat / lineattrs=(color=blue);
  series x=time y=trendhat / lineattrs=(color=black);
run;
```

The plot of predicted values is shown in Figure 8.5.

**Figure 8.5 PROC AUTOREG Predictions**
In Figure 8.5 the straight line is the autocorrelation corrected regression line, traced out by the structural predicted values TRENDHAT. The jagged line traces the full model prediction values. The actual values are marked by asterisks. This plot graphically illustrates the improvement in fit provided by the autoregressive error process for highly autocorrelated data.

To produce forecasts for future periods, include observations for the forecast periods in the input data set. The forecast observations must provide values for the independent variables and have missing values for the response variable.

For the time trend model, the only regressor is time. The following statements add observations for time periods 37 through 46 to the data set A to produce an augmented data set B:

```sas
data b;
  y = .;
  do time = 37 to 46; output; end;
run;

data b;
  merge a b;
  by time;
run;
```

To produce the forecast, use the augmented data set as input to PROC AUTOREG, and specify the appropriate options in the OUTPUT statement. The following statements produce forecasts for the time trend with autoregressive error model. The output data set includes all the variables in the input data set, the forecast values (YHAT), the predicted trend (YTREND), and the upper (UCL) and lower (LCL) 95% confidence limits.

```sas
proc autoreg data=b;
  model y = time / nlag=2 method=ml;
  output out=p p=yhat pm=ytrend
       lcl=lcl ucl=ucl;
run;
```

The following statements plot the predicted values and confidence limits, and they also plot the trend line for reference. The actual observations are shown for periods 16 through 36, and a reference line is drawn at the start of the out-of-sample forecasts.
The plot is shown in Figure 8.6. Notice that the forecasts take into account the recent departures from the trend but converge back to the trend line for longer forecast horizons.

**Figure 8.6 PROC AUTOREG Forecasts**

Preceding section, it is assumed that the order of the autoregressive process is known. In practice, you need to test for the presence of autocorrelation.

The Durbin-Watson test is a widely used method of testing for autocorrelation. The first-order Durbin-Watson statistic is printed by default. This statistic can be used to test for first-order autocorrelation. Use the DWPROB option to print the significance level (p-values) for the Durbin-Watson tests. (Since the Durbin-Watson p-values are computationally expensive, they are not reported by default.)

You can use the DW= option to request higher-order Durbin-Watson statistics. Since the ordinary Durbin-Watson statistic tests only for first-order autocorrelation, the Durbin-Watson statistics for higher-order autocorrelation are called *generalized Durbin-Watson statistics*.

The following statements perform the Durbin-Watson test for autocorrelation in the OLS residuals for orders 1 through 4. The DWPROB option prints the marginal significance levels (p-values) for the Durbin-Watson statistics.

```sas
/*-- Durbin-Watson test for autocorrelation --*/
proc autoreg data=a;
  model y = time / dw=4 dwprob;
run;
```

The AUTOREG procedure output is shown in Figure 8.7. In this case, the first-order Durbin-Watson test is highly significant, with \( p < .0001 \) for the hypothesis of no first-order autocorrelation. Thus, autocorrelation correction is needed.
The AUTOREG Procedure

**Dependent Variable** y

### Ordinary Least Squares Estimates

<p>| | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>SSE</td>
<td>214.953429</td>
<td>DFE</td>
</tr>
<tr>
<td>MSE</td>
<td>6.32216</td>
<td>Root MSE</td>
</tr>
<tr>
<td>SBC</td>
<td>173.659101</td>
<td>AIC</td>
</tr>
<tr>
<td>MAE</td>
<td>2.0190356</td>
<td>AICC</td>
</tr>
<tr>
<td>MAPE</td>
<td>12.5270666</td>
<td>HQC</td>
</tr>
</tbody>
</table>

<p>| | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
</table>
| Regress R-Square | 0.8200  
| Total R-Square   | 0.8200  

### Durbin-Watson Statistics

<table>
<thead>
<tr>
<th>Order</th>
<th>DW</th>
<th>Pr &lt; DW</th>
<th>Pr &gt; DW</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0.4752</td>
<td>&lt;.0001</td>
<td>1.0000</td>
</tr>
<tr>
<td>2</td>
<td>1.2935</td>
<td>0.0137</td>
<td>0.9863</td>
</tr>
<tr>
<td>3</td>
<td>2.0694</td>
<td>0.6545</td>
<td>0.3455</td>
</tr>
<tr>
<td>4</td>
<td>2.5544</td>
<td>0.9818</td>
<td>0.0182</td>
</tr>
</tbody>
</table>

Note: Pr<DW is the p-value for testing positive autocorrelation, and Pr>DW is the p-value for testing negative autocorrelation.

Using the Durbin-Watson test, you can decide if autocorrelation correction is needed. However, generalized Durbin-Watson tests should not be used to decide on the autoregressive order. The higher-order tests assume the absence of lower-order autocorrelation. If the ordinary Durbin-Watson test indicates no first-order autocorrelation, you can use the second-order test to check for second-order autocorrelation. Once autocorrelation is detected, further tests at higher orders are not appropriate. In Figure 8.7, since the first-order Durbin-Watson test is significant, the order 2, 3, and 4 tests can be ignored.

When using Durbin-Watson tests to check for autocorrelation, you should specify an order at least as large as the order of any potential seasonality, since seasonality produces autocorrelation at the seasonal lag. For example, for quarterly data use DW=4, and for monthly data use DW=12.

### Lagged Dependent Variables

The Durbin-Watson tests are not valid when the lagged dependent variable is used in the regression model. In this case, the Durbin h test or Durbin t test can be used to test for first-order autocorrelation.

For the Durbin h test, specify the name of the lagged dependent variable in the LAGDEP= option. For the Durbin t test, specify the LAGDEP option without giving the name of the lagged dependent variable.

For example, the following statements add the variable YLAG to the data set A and regress Y on YLAG instead of TIME:

```sas
data b;
  set a;
  ylag = lag1( y );
run;

proc autoreg data=b;
  model y = ylag / lagdep=ylag;
run;
```
The results are shown in Figure 8.8. The Durbin h statistic 2.78 is significant with a p-value of 0.0027, indicating autocorrelation.

Figure 8.8 Durbin h Test with a Lagged Dependent Variable
Forecasting Autocorrelated Time Series

The AUTOREG Procedure

| Ordinary Least Squares Estimates |
|-------------------------------|------------------|
| SSE 97.711226                  | DFE 33           |
| MSE 2.96095                    | Root MSE 1.72074 |
| SBC 142.369787                 | AIC 139.259091  |
| MAE 1.29949385                 | AICC 139.634091 |
| MAPE 8.1922836                 | HQC 140.332903  |
| Regress R-Square 0.9109        |                  |
| Total R-Square 0.9109          |                  |

Miscellaneous Statistics

<table>
<thead>
<tr>
<th>Statistic</th>
<th>Value</th>
<th>Prob</th>
<th>Label</th>
</tr>
</thead>
<tbody>
<tr>
<td>Durbin h</td>
<td>2.7814</td>
<td>0.0027</td>
<td>Pr &gt; h</td>
</tr>
</tbody>
</table>

Parameter Estimates

<table>
<thead>
<tr>
<th>Variable</th>
<th>DF</th>
<th>Estimate</th>
<th>Standard Error</th>
<th>t Value</th>
<th>Approx Pr &gt;</th>
<th>t</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Intercept</td>
<td>1</td>
<td>1.5742</td>
<td>0.9300</td>
<td>1.69</td>
<td>0.0999</td>
<td></td>
<td></td>
</tr>
<tr>
<td>ylag</td>
<td>1</td>
<td>0.9376</td>
<td>0.0510</td>
<td>18.37</td>
<td>&lt;.0001</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

The AUTOREG Procedure

Stepwise Autoregression

Once you determine that autocorrelation correction is needed, you must select the order of the autoregressive error model to use. One way to select the order of the autoregressive error model is stepwise autoregression. The stepwise autoregression method initially fits a high-order model with many autoregressive lags and then sequentially removes autoregressive parameters until all remaining autoregressive parameters have significant t tests.

To use stepwise autoregression, specify the BACKSTEP option, and specify a large order with the NLAG= option. The following statements show the stepwise feature, using an initial order of 5:

```sas
/*-- stepwise autoregression --*/
proc autoreg data=a;
   model y = time / method=ml nlag=5 backstep;
run;
```

The results are shown in Figure 8.9.

Figure 8.9 Stepwise Autoregression
Forecasting Autocorrelated Time Series
The estimates of the autocorrelations are shown for 5 lags. The backward elimination of autoregressive terms report shows that the autoregressive parameters at lags 3, 4, and 5 were insignificant and eliminated, resulting in the second-order model shown previously in Figure 8.4. By default, retained autoregressive parameters must be significant at the 0.05 level, but you can control this with the SLSTAY= option.

The remainder of the output from this example is the same as that in Figure 8.3 and Figure 8.4, and it is not repeated here.

The stepwise autoregressive process is performed using the Yule-Walker method. The maximum likelihood estimates are produced after the order of the model is determined from the significance tests of the preliminary Yule-Walker estimates.

When using stepwise autoregression, it is a good idea to specify an NLAG= option value larger than the order of any potential seasonality, since seasonality produces autocorrelation at the seasonal lag. For example, for monthly data use NLAG=13, and for quarterly data use NLAG=5.

Subset and Factored Models

In the previous example, the BACKSTEP option dropped lags 3, 4, and 5, leaving a second-order model. However, in other cases a parameter at a longer lag may be kept while some smaller lags are dropped. For example, the stepwise autoregression method might drop lags 2, 3, and 5 but keep lags 1 and 4. This is called a subset model, since the number of estimated autoregressive parameters is lower than the order of the model.

Subset models are common for seasonal data and often correspond to factored autoregressive models. A factored model is the product of simpler autoregressive models. For example, the best model for seasonal monthly data may be the combination of a first-order model for recent effects with a 12th-order subset model for the seasonality, with a single parameter at lag 12. This results in a 13th-order subset model with nonzero parameters at lags 1, 12, and 13. See Chapter 7, The ARIMA Procedure, for further discussion of subset and factored autoregressive models.

You can specify subset models with the NLAG= option. List the lags to include in the autoregressive model within parentheses. The following statements show an example of specifying the subset model resulting from the combination of a first-order process for recent effects with a
fourth-order seasonal process:

```plaintext
/*-- specifying the lags --*/
proc autoreg data=a;
  model y = time / nlag=(1 4 5);
run;
```

The MODEL statement specifies the following fifth-order autoregressive error model:

\[ y_t = a + \beta t + \varepsilon_t \]

\[ \varepsilon_t = \phi_1 \varepsilon_{t-1} + \phi_2 \varepsilon_{t-4} - \phi_3 \varepsilon_{t-5} + \epsilon_t \]

---

Testing for Heteroscedasticity

One of the key assumptions of the ordinary regression model is that the errors have the same variance throughout the sample. This is also called the *homoscedasticity* model. If the error variance is not constant, the data are said to be *heteroscedastic*.

Since ordinary least squares regression assumes constant error variance, heteroscedasticity causes the OLS estimates to be inefficient. Models that take into account the changing variance can make more efficient use of the data. Also, heteroscedasticity can make the OLS forecast error variance inaccurate because the predicted forecast variance is based on the average variance instead of on the variability at the end of the series.

To illustrate heteroscedastic time series, the following statements create the simulated series \( Y \). The variable \( Y \) has an error variance that changes from 1 to 4 in the middle part of the series.

```plaintext
data a;
  do time = -10 to 120;
    s = 1 + (time >= 60 & time < 90);
    u = s*rannor(12346);
    y = 10 + .5 * time + u;
    if time > 0 then output;
  end;
run;
```

title 'Heteroscedastic Time Series';
proc sgplot data=a noautolegend;
  series x=time y=y / markers;
  reg x=time y=y / lineattrs=(color=black);
run;
```

The simulated series is plotted in Figure 8.10.

**Figure 8.10 Heteroscedastic and Autocorrelated Series**
To test for heteroscedasticity with PROC AUTOREG, specify the ARCHTEST option. The following statements regress Y on TIME and use the ARCHTEST option to test for heteroscedastic OLS residuals:

```
/*-- test for heteroscedastic OLS residuals --*/
proc autoreg data=a;
  model y = time / archtest;
  output out=r r=yresid;
run;
```

The PROC AUTOREG output is shown in Figure 8.11. The Q statistics test for changes in variance across time by using lag windows that range from 1 through 12. (See the section Testing for Nonlinear Dependence: Heteroscedasticity Tests for details.) The p-values for the test statistics strongly indicate heteroscedasticity, with $p < 0.0001$ for all lag windows.

The Lagrange multiplier (LM) tests also indicate heteroscedasticity. These tests can also help determine the order of the ARCH model that is appropriate for modeling the heteroscedasticity, assuming that the changing variance follows an autoregressive conditional heteroscedasticity model.

Figure 8.11 Heteroscedasticity Tests

### Heteroscedastic Time Series

<table>
<thead>
<tr>
<th>Dependent Variable</th>
<th>$y$</th>
</tr>
</thead>
</table>

<table>
<thead>
<tr>
<th>Ordinary Least Squares Estimates</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>SSE</td>
<td>223.645647</td>
</tr>
<tr>
<td>MSE</td>
<td>1.89530</td>
</tr>
<tr>
<td>Root MSE</td>
<td>1.37670</td>
</tr>
<tr>
<td>SBC</td>
<td>424.828766</td>
</tr>
<tr>
<td>AIC</td>
<td>419.253783</td>
</tr>
<tr>
<td>MAE</td>
<td>0.97683599</td>
</tr>
<tr>
<td>AICC</td>
<td>419.356347</td>
</tr>
<tr>
<td>MAPE</td>
<td>2.73888672</td>
</tr>
<tr>
<td>HQC</td>
<td>421.517809</td>
</tr>
<tr>
<td>Durbin-Watson</td>
<td>2.4444</td>
</tr>
<tr>
<td>Regress R-Square</td>
<td>0.9938</td>
</tr>
<tr>
<td>Total R-Square</td>
<td>0.9938</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Tests for ARCH Disturbances Based on OLS Residuals</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Order</td>
<td>Q</td>
</tr>
<tr>
<td>-------</td>
<td>----</td>
</tr>
<tr>
<td>1</td>
<td>19.4549</td>
</tr>
</tbody>
</table>
The tests of Lee and King (1993) and Wong and Li (1995) can also be applied to check the absence of ARCH effects. The following example shows that Wong and Li’s test is robust to detect the presence of ARCH effects with the existence of outliers.

```sas
/*-- data with outliers at observation 10 --*/
data b;
  do time = -10 to 120;
    s = 1 + (time >= 60 & time < 90);
    u = s * rannor(12346);
    y = 10 + .5 * time + u;
    if time = 10 then
      do; y = 200; end;
    if time > 0 then output;
  end;
run;
/*-- test for heteroscedastic OLS residuals --*/
proc autoreg data=b;
  model y = time / archtest=(qlm);
  model y = time / archtest=(lk, wl);
run;
```

As shown in Figure 8.12, the p-values of Q or LM statistics for all lag windows are above 90%, which fails to reject the null hypothesis of the absence of ARCH effects. Lee and King’s test, which rejects the null hypothesis for lags more than 8 at 10% significance level, works better. Wong and Li’s test works best, rejecting the null hypothesis and detecting the presence of ARCH effects for all lag windows.

**Figure 8.12 Heteroscedasticity Tests**

Heteroscedastic Time Series

<table>
<thead>
<tr>
<th>Order</th>
<th>Q</th>
<th>Pr &gt; Q</th>
<th>LM</th>
<th>Pr &gt; LM</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0.0076</td>
<td>0.9304</td>
<td>0.0073</td>
<td>0.9319</td>
</tr>
<tr>
<td>2</td>
<td>0.0150</td>
<td>0.9925</td>
<td>0.0143</td>
<td>0.9929</td>
</tr>
<tr>
<td>3</td>
<td>0.0229</td>
<td>0.9991</td>
<td>0.0217</td>
<td>0.9992</td>
</tr>
<tr>
<td>4</td>
<td>0.0308</td>
<td>0.9999</td>
<td>0.0290</td>
<td>0.9999</td>
</tr>
<tr>
<td>5</td>
<td>0.0367</td>
<td>1.0000</td>
<td>0.0345</td>
<td>1.0000</td>
</tr>
<tr>
<td>6</td>
<td>0.0442</td>
<td>1.0000</td>
<td>0.0413</td>
<td>1.0000</td>
</tr>
<tr>
<td>7</td>
<td>0.0522</td>
<td>1.0000</td>
<td>0.0485</td>
<td>1.0000</td>
</tr>
<tr>
<td>8</td>
<td>0.0612</td>
<td>1.0000</td>
<td>0.0565</td>
<td>1.0000</td>
</tr>
<tr>
<td>9</td>
<td>0.0701</td>
<td>1.0000</td>
<td>0.0643</td>
<td>1.0000</td>
</tr>
<tr>
<td>10</td>
<td>0.0701</td>
<td>1.0000</td>
<td>0.0742</td>
<td>1.0000</td>
</tr>
</tbody>
</table>
There are several approaches to dealing with heteroscedasticity. If the error variance at different times is known, weighted regression is a good method. If, as is usually the case, the error variance is unknown and must be estimated from the data, you can model the changing error variance.

The **generalized autoregressive conditional heteroscedasticity** (GARCH) model is one approach to modeling time series with heteroscedastic errors. The GARCH regression model with autoregressive errors is

\[
y_t = x_t' \beta + v_t
\]

\[
v_t = \phi_0 - \phi_1 v_{t-1} - \ldots - \phi_m v_{t-m}
\]

\[
e_t = \sqrt{h_t} e_t
\]

\[
h_t = \alpha_0 + \sum_{i=1}^{q} \alpha_i e_{t-i}^2 + \sum_{j=1}^{p} \beta_j h_{t-j}
\]

\[e_t \sim \text{IN}(0,1)\]

This model combines the \(m\)th-order autoregressive error model with the GARCH\((p,q)\) variance model. It is denoted as the AR\((m)\)-GARCH\((p,q)\) regression model.

The tests for the presence of ARCH effects (namely, Q and LM tests, tests from Lee and King (1993) and tests from Wong and Li (1995)) can help determine the order of the ARCH model appropriate for the data. For example, the Lagrange multiplier (LM) tests shown in Figure 8.11 are significant \((p < 0.0001)\) through order 12, which indicates that a very high-order ARCH model is needed to model the heteroscedasticity.

The basic ARCH\((q)\) model \((p = 0)\) is a *short memory* process in that only the most recent \(q\) squared residuals are used to estimate the changing variance. The GARCH model \((p > 0)\) allows *long memory* processes, which use all the past squared residuals to estimate the current variance. The LM tests in Figure 8.11 suggest the use of the GARCH model \((p > 0)\) instead of the ARCH model.

The GARCH\((p,q)\) model is specified with the GARCH\((P=p Q=q)\) option in the MODEL statement. The basic ARCH\((q)\) model is the same as the GARCH\((0,q)\) model and is specified with the GARCH\((Q=q)\) option.

The following statements fit an AR\((2)\)-GARCH\((1,1)\) model for the \(Y\) series that is regressed on \(TIME\). The GARCH\((P=1,Q=1)\) option...
specifies the $GARCH(1, 1)$ conditional variance model. The NLAG=2 option specifies the AR(2) error process. Only the maximum likelihood method is supported for GARCH models; therefore, the METHOD= option is not needed. The CEV= option in the OUTPUT statement stores the estimated conditional error variance at each time period in the variable VHAT in an output data set named OUT. The data set is the same as in the section Testing for Heteroscedasticity.

data c;
  ul=0; ull=0;
  do time = -10 to 120;
    s = 1 + (time >= 60 & time < 90);
    u = -1.3 * ul - .5 * ull + s * rannor(12346);
    y = 10 + .5 * time + u;
    if time > 0 then output;
    ull = ul; ul = u;
  end;
run;
title 'AR(2)-GARCH(1,1) model for the Y series regressed on TIME';
proc autoreg data=c;
  model y = time / nlag=2 garch=(q=1,p=1) maxit=50;
  output out=out cev=vhat;
run;
The results for the GARCH model are shown in Figure 8.13. (The preliminary estimates are not shown.)

Figure 8.13 AR(2)-GARCH(1,1) Model
AR(2)-GARCH(1,1) model for the Y series regressed on TIME

The AUTOREG Procedure

<table>
<thead>
<tr>
<th>GARCH Estimates</th>
</tr>
</thead>
<tbody>
<tr>
<td>SSE</td>
</tr>
<tr>
<td>MSE</td>
</tr>
<tr>
<td>Log Likelihood</td>
</tr>
<tr>
<td>SBC</td>
</tr>
<tr>
<td>MAE</td>
</tr>
<tr>
<td>MAPE</td>
</tr>
<tr>
<td>Normality Test</td>
</tr>
<tr>
<td>Pr &gt; ChiSq</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Parameter Estimates</th>
</tr>
</thead>
<tbody>
<tr>
<td>Variable</td>
</tr>
<tr>
<td>Intercept</td>
</tr>
<tr>
<td>time</td>
</tr>
<tr>
<td>AR1</td>
</tr>
<tr>
<td>AR2</td>
</tr>
<tr>
<td>ARCH0</td>
</tr>
<tr>
<td>ARCH1</td>
</tr>
<tr>
<td>GARCH1</td>
</tr>
</tbody>
</table>

The normality test is not significant ($p = 0.959$), which is consistent with the hypothesis that the residuals from the GARCH model, $\varepsilon_t / \sqrt{h_t}$, are normally distributed. The parameter estimates table includes rows for the GARCH parameters. ARCH0 represents the estimate for the parameter $\alpha_0$, ARCH1 represents $\alpha_1$, and GARCH1 represents $\gamma_1$.

The following statements transform the estimated conditional error variance series VHAT to the estimated standard deviation series SHAT. Then, they plot SHAT together with the true standard deviation S used to generate the simulated data.

data out;
  set out;
  shat = sqrt( vhat );
run;
title 'Predicted and Actual Standard Deviations';
proc sgplot data=out noautolegend;
  scatter x=time y=s;
In this example note that the form of heteroscedasticity used in generating the simulated series Y does not fit the GARCH model. The GARCH model assumes conditional heteroscedasticity, with homoscedastic unconditional error variance. That is, the GARCH model assumes that the changes in variance are a function of the realizations of preceding errors and that these changes represent temporary and random departures from a constant unconditional variance. The data-generating process used to simulate series Y, contrary to the GARCH model, has exogenous unconditional heteroscedasticity that is independent of past errors.

Nonetheless, as shown in Figure 8.14, the GARCH model does a reasonably good job of approximating the error variance in this example, and some improvement in the efficiency of the estimator of the regression parameters can be expected.

The GARCH model might perform better in cases where theory suggests that the data-generating process produces true autoregressive conditional heteroscedasticity. This is the case in some economic theories of asset returns, and GARCH-type models are often used for analysis of financial market data.

**GARCH Models**

The AUTOREG procedure supports several variations of GARCH models.

Using the TYPE= option along with the GARCH= option enables you to control the constraints placed on the estimated GARCH parameters. You can specify unconstrained, nonnegativity-constrained (default), stationarity-constrained, or integration-constrained models. The integration constraint produces the integrated GARCH (IGARCH) model.

You can also use the TYPE= option to specify the exponential form of the GARCH model, called the EGARCH model, or other types of GARCH models, namely the quadratic GARCH (QGARCH), threshold GARCH (TGARCH), and power GARCH (PGARCH) models. The MEAN= option along with the GARCH= option specifies the GARCH-in-mean (GARCH-M) model.

The following statements illustrate the use of the TYPE= option to fit an AR(2)-EGARCH(1,1) model to the series Y. (Output is not shown.)

```bash
/*-- AR(2)-EGARCH(1,1) model --*/
proc autoreg data=a;
   model y = time / nlag=2 garch=(p=1,q=1,type=exp);
run;
```

See the section GARCH Models for details.
The AUTOREG Procedure

Syntax: AUTOREG Procedure

The AUTOREG procedure is controlled by the following statements:

**PROC AUTOREG**

**BY** variables ;  
**CLASS** variables ;  
**MODEL** dependent = regressors / options ;  
**HETERO** variables / options ;  
**NLOPTIONS** options ;  
**RESTRICT** equation , ..., equation ;  
**TEST** equation , ..., equation / option ;  
**OUTPUT** OUT = SAS data set options ;

At least one MODEL statement must be specified. One OUTPUT statement can follow each MODEL statement. One HETERO statement can follow each MODEL statement.

## Functional Summary

### PROC AUTOREG Statement

### BY Statement

### CLASS Statement

### MODEL Statement

### HETERO Statement

### NLOPTIONS Statement

### RESTRICT Statement

### TEST Statement

### OUTPUT Statement

The statements and options used with the AUTOREG procedure are summarized in the following table.

### Table 8.1 AUTOREG Functional Summary

<table>
<thead>
<tr>
<th>Description</th>
<th>Statement</th>
<th>Option</th>
</tr>
</thead>
<tbody>
<tr>
<td>Data Set Options</td>
<td>AUTOREG</td>
<td>DATA=</td>
</tr>
<tr>
<td>Specify the input data set</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Write parameter estimates to an output data set</td>
<td>AUTOREG</td>
<td>OUTEST=</td>
</tr>
<tr>
<td>Include covariances in the OUTEST= data set</td>
<td>AUTOREG</td>
<td>COVOUT</td>
</tr>
<tr>
<td>Requests that the procedure produce graphics via the Output Delivery System</td>
<td>AUTOREG</td>
<td>PLOTS=</td>
</tr>
<tr>
<td>Write predictions, residuals, and confidence limits to an output data set</td>
<td>OUTPUT</td>
<td>OUT=</td>
</tr>
<tr>
<td>Declarining of Variables</td>
<td>BY</td>
<td></td>
</tr>
<tr>
<td>Specify BY-group processing</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Specify classification variables</td>
<td>CLASS</td>
<td></td>
</tr>
<tr>
<td>Printing Control Options</td>
<td>MODEL</td>
<td>ALL</td>
</tr>
<tr>
<td>Request all printing options</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Print transformed coefficients</td>
<td>MODEL</td>
<td>COEF</td>
</tr>
<tr>
<td>Print correlation matrix of the estimates</td>
<td>MODEL</td>
<td>CORRB</td>
</tr>
<tr>
<td>Print covariance matrix of the estimates</td>
<td>MODEL</td>
<td>COVB</td>
</tr>
<tr>
<td>Print DW statistics up to order j</td>
<td>MODEL</td>
<td>DW=j</td>
</tr>
<tr>
<td>Print marginal probability of the generalized Durbin-Watson test statistics for large sample sizes</td>
<td>MODEL</td>
<td>DWPROB</td>
</tr>
<tr>
<td>Print the p-values for the Durbin-Watson test be computed using a linearized</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
approximation of the design matrix
Print inverse of Toeplitz matrix
Print the Godfrey LM serial correlation test
Print details at each iteration step
Print the Durbin $t$ statistic
Print the Durbin $h$ statistic
Print the log-likelihood value of the regression model
Print the Jarque-Bera normality test
Print the tests for the absence of ARCH effects
Print BDS tests for independence
Print rank version of von Neumann ratio test for independence
Print runs test for independence
Print the turning point test for independence
Print the Lagrange multiplier test for independence
Print the Chow test
Print the predictive Chow test
Suppress printed output
Print partial autocorrelations
Print Ramsey’s RESET test
Print Phillips-Perron tests for stationarity or unit roots
Print Augmented Dickey-Fuller tests for stationarity or unit roots
Print ERS tests for stationarity or unit roots
Print Ng-Perron tests for stationarity or unit roots
Print tests of linear hypotheses
Specify the test statistics to use
Print the uncentered regression $R^2$

Options to Control the Optimization Process
Specify the optimization options

Model Estimation Options
Specify the order of autoregressive process
Center the dependent variable
Suppress the intercept parameter
Remove nonsignificant AR parameters
Specify significance level for BACKSTEP
Specify the convergence criterion
Specify the type of covariance matrix
Set the initial values of parameters used by the iterative optimization algorithm
Specify iterative Yule-Walker method
Specify maximum number of iterations
Specify the estimation method
Use only first sequence of nonmissing data
Specify the optimization technique
Imposes restrictions on the regression estimates
Estimate and test heteroscedasticity models

GARCH Related Options
Specify order of GARCH process
Specify type of GARCH model
Specify various forms of the GARCH-M model
Suppress GARCH intercept parameter
Specify the trust region method
Estimate the GARCH model for the conditional $t$ distribution
Estimate the start-up values for the conditional variance equation
Specify the functional form of the heteroscedasticity model
Specify that the heteroscedasticity model does not include the unit term
Impose constraints on the estimated parameters in the heteroscedasticity model
Output conditional error variance
Output conditional prediction error variance
Specify the flexible conditional variance form of the GARCH model

Output Control Options
Specify confidence limit size

NLOPTISee Chapter 6, Nonlinear Optimization Methods.
The following options can be used in the PROC AUTOREG statement:

**DATA=SAS-data-set**

specifies the input SAS data set. If the DATA= option is not specified, PROC AUTOREG uses the most recently created SAS data set.

**OUTEST=SAS-data-set**

writes the parameter estimates to an output data set. See the section **OUTEST= Data Set** later in this chapter for information on the contents of these data set.

**COVOUT**

writes the covariance matrix for the parameter estimates to the OUTEST= data set. This option is valid only if the OUTEST= option is specified.

**PLOTS<(global-plot-options)> < = (specific plot options)>**

requests that the AUTOREG procedure produce statistical graphics via the Output Delivery System, provided that the ODS GRAPHICS statement has been specified. For general information about ODS Graphics, see Chapter 21, *Statistical Graphics Using ODS (SAS/STAT 9.22 User's Guide)*. The global-plot-options apply to all relevant plots generated by the AUTOREG procedure. The global-plot-options supported by the AUTOREG procedure follow.

### Global Plot Options

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>ALPHACLML=</strong></td>
<td>Specify confidence limit size for structural predicted values</td>
</tr>
<tr>
<td><strong>ALPHACSM=</strong></td>
<td>Specify the significance level for the upper and lower bounds of the CUSUM and CUSUMSQ statistics</td>
</tr>
<tr>
<td><strong>BLUS=</strong></td>
<td>Specify the name of a variable to contain the values of the Theil’s BLUS residuals</td>
</tr>
<tr>
<td><strong>CEV=</strong></td>
<td>Output the value of the error variance $\sigma^2$</td>
</tr>
<tr>
<td><strong>CONSTANT=</strong></td>
<td>Output transformed intercept variable</td>
</tr>
<tr>
<td><strong>CUSUM=</strong></td>
<td>Specify the name of a variable to contain the CUSUM statistics</td>
</tr>
<tr>
<td><strong>CUSUMSQ=</strong></td>
<td>Specify the name of a variable to contain the CUSUMSQ statistics</td>
</tr>
<tr>
<td><strong>CUSUMUB=</strong></td>
<td>Specify the name of a variable to contain the upper confidence bound for the CUSUM statistic</td>
</tr>
<tr>
<td><strong>CUSUMULB=</strong></td>
<td>Specify the name of a variable to contain the lower confidence bound for the CUSUM statistic</td>
</tr>
<tr>
<td><strong>CUSUMSQUB=</strong></td>
<td>Specify the name of a variable to contain the upper confidence bound for the CUSUMSQ statistic</td>
</tr>
<tr>
<td><strong>CUSUMSQLB=</strong></td>
<td>Specify the name of a variable to contain the lower confidence bound for the CUSUMSQ statistic</td>
</tr>
<tr>
<td><strong>LCL=</strong></td>
<td>Output lower confidence limit</td>
</tr>
<tr>
<td><strong>LCLM=</strong></td>
<td>Output lower confidence limit for structural predicted values</td>
</tr>
<tr>
<td><strong>P=</strong></td>
<td>Output predicted values</td>
</tr>
<tr>
<td><strong>PM=</strong></td>
<td>Output predicted values of structural part</td>
</tr>
<tr>
<td><strong>R=</strong></td>
<td>Output residuals</td>
</tr>
<tr>
<td><strong>RM=</strong></td>
<td>Output residuals from structural predictions</td>
</tr>
<tr>
<td><strong>RECPEV=</strong></td>
<td>Specify the name of a variable to contain the part of the predictive error variance $(\psi)$</td>
</tr>
<tr>
<td><strong>RECREASE=</strong></td>
<td>Specify the name of a variable to contain recursive residuals</td>
</tr>
<tr>
<td><strong>RECVRES=</strong></td>
<td>Output transformed variables</td>
</tr>
<tr>
<td><strong>UCL=</strong></td>
<td>Output upper confidence limit</td>
</tr>
<tr>
<td><strong>UCLM=</strong></td>
<td>Output upper confidence limit for structural predicted values</td>
</tr>
</tbody>
</table>
suppresses the default plots. Only the plots specifically requested are produced.

UNPACKPANEL

breaks a graphic that is otherwise paneled into individual component plots.

**Specific Plot Options**

ALL

requests that all plots appropriate for the particular analysis be produced.

ACF

produces the autocorrelation function plot.

IACF

produces the inverse autocorrelation function plot of residuals.

PACF

produces the partial autocorrelation function plot of residuals.

FITPLOT

plots the predicted and actual values.

COOKSD

produces the Cook's D plot.

QQ

Q-Q plot of residuals.

RESIDUAL | RES

plots the residuals.

STUDENTRESIDUAL

plots the studentized residuals. For the models with the NLAG= or GARCH= options in the MODEL statement or with the HETERO statement, this option is replaced by the STANDARDRESIDUAL option.

STANDARDRESIDUAL

plots the standardized residuals.

WHITENOISE

plots the white noise probabilities.

RESIDUALHISTOGRAM | RESIDHISTOGRAM

plots the histogram of residuals.

NONE

suppresses all plots.

In addition, any of the following MODEL statement options can be specified in the PROC AUTOREG statement, which is equivalent to specifying the option for every MODEL statement: ALL, ARCHTEST, BACKSTEP, CENTER, COEF, CONVERGE=, CORRB, COVB, DW=, DWPROB, GINV, ITER, ITPRINT, MAXITER=, METHOD=, NOINT, NOMISS, NOPRINT, and PARTIAL.
BY variables;

A BY statement can be used with PROC AUTOREG to obtain separate analyses on observations in groups defined by the BY variables.

CLASS variables;

The CLASS statement names the classification variables to be used in the analysis. Classification variables can be either character or numeric.

In PROC AUTOREG, the CLASS statement enables you to output class variables to a data set that contains a copy of the original data. Class levels are determined from the formatted values of the CLASS variables. Thus, you can use formats to group values into levels. See the discussion of the FORMAT procedure in SAS Language Reference: Dictionary for details.

MODEL dependent = regressors / options;

The MODEL statement specifies the dependent variable and independent regressor variables for the regression model. If no independent variables are specified in the MODEL statement, only the mean is fitted. (This is a way to obtain autocorrelations of a series.)

Models can be given labels of up to eight characters. Model labels are used in the printed output to identify the results for different models. The model label is specified as follows:

    label: MODEL ...;

The following options can be used in the MODEL statement after a slash (/).

CENTER

centers the dependent variable by subtracting its mean and suppresses the intercept parameter from the model. This option is valid only when the model does not have regressors (explanatory variables).

NOINT

suppresses the intercept parameter.

Autoregressive Error Options

NLAG=number

NLAG=(number-list)

specifies the order of the autoregressive error process or the subset of autoregressive error lags to be fitted. Note that NLAG=3 is the same as NLAG=(1 2 3). If the NLAG= option is not specified, PROC AUTOREG does not fit an autoregressive model.
GARCH Estimation Options

DIST=value

specifies the distribution assumed for the error term in GARCH-type estimation. If no GARCH= option is specified, the option is ignored. If EGARCH is specified, the distribution is always the normal distribution. The values of the DIST= option are as follows:

T

specifies Student’s t distribution.

NORMAL

specifies the standard normal distribution. The default is DIST=NORMAL.

GARCH=(option-list)

specifies a GARCH-type conditional heteroscedasticity model. The GARCH= option in the MODEL statement specifies the family of ARCH models to be estimated. The GARCH(1,1) regression model is specified in the following statement:

```sas
model y = x1 x2 / garch=(q=1,p=1);
```

When you want to estimate the subset of ARCH terms, such as ARCH(1,3), you can write the SAS statement as follows:

```sas
model y = x1 x2 / garch=(q=(1 3));
```

With the TYPE= option, you can specify various GARCH models. The IGARCH(2,1) model without trend in variance is estimated as follows:

```sas
model y = / garch=(q=2,p=1,type=integ,noint);
```

The following options can be used in the GARCH=() option. The options are listed within parentheses and separated by commas.

Q=number

Q=(number-list)

specifies the order of the process or the subset of ARCH terms to be fitted.

P=number

P=(number-list)

specifies the order of the process or the subset of GARCH terms to be fitted. If only the P= option is specified, P= option is ignored and Q=1 is assumed.

TYPE=value

specifies the type of GARCH model. The values of the TYPE= option are as follows:

EXP | EGARCH

specifies the exponential GARCH or EGARCH model.

INTEGRATED | IGARCH

specifies the integrated GARCH or IGARCH model.

NELSON | NELSONCAO

specifies the Nelson-Cao inequality constraints.

NONNEG

specifies the GARCH model with nonnegativity constraints.

POWER | PGARCH

specifies the power GARCH or PGARCH model.
QUADR | QUADRATIC | QGARCH

specifies the quadratic GARCH or QGARCH model.

STATIONARY

constrains the sum of GARCH coefficients to be less than 1.

THRES | THRESHOLD | TGARCH

specifies the threshold GARCH or TGARCH model.

The default is TYPE=NELSON.

MEAN=value

specifies the functional form of the GARCH-M model. The values of the MEAN= option are as follows:

LINEAR

specifies the linear function:

\[ y_t = x'_t \beta + \delta h_t + \epsilon_t \]

LOG

specifies the log function:

\[ y_t = x'_t \beta + \delta \ln(h_t) + \epsilon_t \]

SQRT

specifies the square root function:

\[ y_t = x'_t \beta + \delta \sqrt{h_t} + \epsilon_t \]

NOINT

suppresses the intercept parameter in the conditional variance model. This option is valid only with the TYPE=INTEG option.

STARTUP=MSE | ESTIMATE

requests that the positive constant c for the start-up values of the GARCH conditional error variance process be estimated. By default or if STARTUP=MSE is specified, the value of the mean squared error is used as the default constant.

TR

uses the trust region method for GARCH estimation. This algorithm is numerically stable, though computation is expensive. The double quasi-Newton method is the default.

Printing Options

ALL

requests all printing options.

ARCHTEST

ARCHTEST=(option-list)

specifies tests for the absence of ARCH effects. The following options can be used in the ARCHTEST=( ) option. The options are listed within parentheses and separated by commas.

QLM | QLMARCH

requests the Q and Engle’s LM tests.

LK | LKARCH

requests Lee and King’s ARCH tests.
WL | WLARCH

requests Wong and Li’s ARCH tests.

ALL

requests all ARCH tests, namely Q and Engle’s LM tests, Lee and King’s tests, and Wong and Li’s tests.

If ARCHTEST is defined without additional suboptions, it requests the Q and Engle’s LM tests. That is, the statement

```plaintext
model return = x1 x2 / archtest;
```

is equivalent to the statement

```plaintext
model return = x1 x2 / archtest=(qlm);
```

The following statement requests Lee and King’s tests and Wong and Li’s tests:

```plaintext
model return = / archtest=(lk,wl);
```

BDS

BDS=(option-list)

specifies Brock-Dechert-Scheinkman (BDS) tests for independence. The following options can be used in the BDS=( ) option. The options are listed within parentheses and separated by commas.

M=number

specifies the maximum number of the embedding dimension. The BDS tests with embedding dimension from 2 to M are calculated. M must be an integer between 2 and 20. The default value of the M= suboption is 20.

D=number

specifies the parameter to determine the radius for BDS test. The BDS test sets up the radius as \( r = D \cdot \sigma \), where \( \sigma \) is the standard deviation of the time series to be tested. By default, D=1.5.

PVALUE=DIST | SIM

specifies the way to calculate the \( p \)-values. By default or if PVALUE=DIST is specified, the \( p \)-values are calculated according to the asymptotic distribution of BDS statistics (that is, the standard normal distribution). Otherwise, for samples of size less than 500, the \( p \)-values are obtained through Monte Carlo simulation.

Z=value

specifies the type of the time series (residuals) to be tested. The values of the Z= suboption are as follows:

Y

specifies the regressand. The default is Z=Y.

RO

specifies the OLS residuals.

R

specifies the residuals of the final model.

RM

specifies the structural residuals of the final model.

SR

specifies the standardized residuals of the final model, defined by residuals over the square root of the conditional variance.

If BDS is defined without additional suboptions, all suboptions are set as default values. That is, the statement

```plaintext
model return = x1 x2 / nlag=1 BDS;
```
is equivalent to the statement

\[
\text{model return} = x_1 \times x_2 / \text{nlag}=1 \ \text{BDS}=(M=20, \ D=1.5, \ \text{PVALUE}=\text{DIST}, \ Z=Y);
\]

To do the specification check of a GARCH(1,1) model, you can write the SAS statement as follows:

\[
\text{model return} = / \ \text{garch}=(p=1, q=1) \ \text{BDS}=(Z=\text{SR});
\]

**CHOW=(obs1 ... obsn)**

computes Chow tests to evaluate the stability of the regression coefficient. The Chow test is also called the analysis-of-variance test.

Each value \( \text{obs1} \) listed on the CHOW= option specifies a break point of the sample. The sample is divided into parts at the specified break point, with observations before \( \text{obs1} \) in the first part and \( \text{obs1} \) and later observations in the second part, and the fits of the model in the two parts are compared to whether both parts of the sample are consistent with the same model.

The break points \( \text{obs1} \) refer to observations within the time range of the dependent variable, ignoring missing values before the start of the dependent series. Thus, CHOW=10 specifies the 10th observation after the first nonmissing observation for the dependent variable. For example, if the dependent variable \( Y \) contains 10 missing values before the first observation with a nonmissing \( Y \) value, then CHOW=20 actually refers to the 30th observation in the data set.

When you specify the break point, you should note the number of presample missing values.

**COEF**

prints the transformation coefficients for the first \( p \) observations. These coefficients are formed from a scalar multiplied by the inverse of the Cholesky root of the Toeplitz matrix of autocovariances.

**CORRB**

prints the estimated correlations of the parameter estimates.

**COVB**

prints the estimated covariances of the parameter estimates.

**COVEST=OP | HESSIAN | QML**

specifies the type of covariance matrix for the GARCH or heteroscedasticity model. When COVEST=OP is specified, the outer product matrix is used to compute the covariance matrix of the parameter estimates. The COVEST=HESSIAN option produces the covariance matrix by using the Hessian matrix. The quasi-maximum likelihood estimates are computed with COVEST=QML. The default is COVEST=OP.

**DW=n**

prints Durbin-Watson statistics up to the order \( n \). The default is DW=1. When the LAGDEP option is specified, the Durbin-Watson statistic is not printed unless the DW= option is explicitly specified.

**DWPROB**

now produces \( p \)-values for the generalized Durbin-Watson test statistics for large sample sizes. Previously, the Durbin-Watson probabilities were calculated only for small sample sizes. The new method of calculating Durbin-Watson probabilities is based on the algorithm of Ansley, Kohn, and Shively (1992).

**GINV**

prints the inverse of the Toeplitz matrix of autocovariances for the Yule-Walker solution. See the section Computational Methods later in this chapter for details.

**GODFREY**

**GODFREY=r**

produces Godfrey’s general Lagrange multiplier test against ARMA errors.

**ITPRINT**

prints the objective function and parameter estimates at each iteration. The objective function is the full log likelihood function for the maximum likelihood method, while the error sum of squares is produced as the objective function of unconditional least squares. For the
ML method, the ITPRINT option prints the value of the full log likelihood function, not the concentrated likelihood.

**LAGDEP**

prints the Durbin $t$ statistic, which is used to detect residual autocorrelation in the presence of lagged dependent variables. See the section Generalized Durbin-Watson Tests later in this chapter for details.

**LAGDEP=**name

prints the Durbin $h$ statistic for testing the presence of first-order autocorrelation when regressors contain the lagged dependent variable whose name is specified as LAGDEP=**name. If the Durbin $h$ statistic cannot be computed, the asymptotically equivalent $t$ statistic is printed instead. See the section Generalized Durbin-Watson Tests for details.

When the regression model contains several lags of the dependent variable, specify the lagged dependent variable for the smallest lag in the LAGDEP= option. For example:

```xml
model y = x1 x2 ylag2 ylag3 / lagdep=ylag2;
```

**LOGLIKEL**

prints the log likelihood value of the regression model, assuming normally distributed errors.

**NOPRINT**

suppresses all printed output.

**NORMAL**

specifies the Jarque-Bera’s normality test statistic for regression residuals.

**PARTIAL**

prints partial autocorrelations.

**PCHOW=(obs1 ...obs_p)**

computes the predictive Chow test. The form of the PCHOW= option is the same as the CHOW= option; see the discussion of the CHOW= option earlier in this chapter.

**RESET**

produces Ramsey’s RESET test statistics. The RESET option tests the null model

$$y_t = x_t \beta + u_t$$

against the alternative

$$y_t = x_t \beta + \sum_{j=2}^{p} \phi_j \hat{y}_{t-j} + u_t$$

where $\hat{y}_t$ is the predicted value from the OLS estimation of the null model. The RESET option produces three RESET test statistics for $p = 2$, $3$, and $4$.

**RUNS**

**RUNS=(Z=value)**

specifies the runs test for independence. The Z= suboption specifies the type of the time series or residuals to be tested. The values of the Z= suboption are as follows:

Y
specifies the regressand. The default is $Z = Y$.

RO

specifies the OLS residuals.

R

specifies the residuals of the final model.

RM

specifies the structural residuals of the final model.

SR

specifies the standardized residuals of the final model, defined by residuals over the square root of the conditional variance.

\[
\text{STATIONARITY}=\left(\text{ADF}\right) \\
\text{STATIONARITY}=\left(\text{KPSS}\right) \\
\text{STATIONARITY}=\left(\text{PHILLIPS}\right) \\
\text{STATIONARITY}=\left(\text{ERS}\right) \\
\text{STATIONARITY}=\left(\text{NP}\right) \\
\text{STATIONARITY}=\left(\text{ERS}\left(\text{KERNEL}=\text{type}\right)\right) \\
\text{STATIONARITY}=\left(\text{NP}=\left(\text{value}\right)\right) \\
\text{STATIONARITY}=\left(\text{ADF}<\left(\text{value}\right), \text{ERS}<\left(\text{value}\right), \text{KPSS}<\left(\text{value}\right), \text{NP}<\left(\text{value}\right), \text{PHILLIPS}<\left(\text{value}\right)\right)
\]

specifies tests of stationarity or unit roots. The \text{STATIONARITY=} option provides Phillips-Perron, Phillips-Ouliaris, augmented Dickey-Fuller, Engle-Granger, KPSS, ERS, and NP tests.

The PHILLIPS or PHILLIPS= suboption of the STATIONARITY= option produces the Phillips-Perron unit root test when there are no regressors in the MODEL statement. When the model includes regressors, the PHILLIPS option produces the Phillips-Ouliaris cointegration test. The PHILLIPS option can be abbreviated as PP.

The PHILLIPS option performs the Phillips-Perron test for three null hypothesis cases: zero mean, single mean, and deterministic trend. For each case, the PHILLIPS option computes two test statistics, $Z_p$ and $\tilde{Z}$ (in the original paper they are referred to as $\hat{Z}_p$ and $\tilde{Z}_p$), and reports their $p$-values. These test statistics have the same limiting distributions as the corresponding Dickey-Fuller tests.

The three types of the Phillips-Perron unit root test reported by the PHILLIPS option are as follows:

**Zero mean**

computes the Phillips-Perron test statistic based on the zero mean autoregressive model:

\[
y_t = \rho y_{t-1} + u_t
\]

**Single mean**

computes the Phillips-Perron test statistic based on the autoregressive model with a constant term:

\[
y_t = \mu + \rho y_{t-1} + u_t
\]

**Trend**

computes the Phillips-Perron test statistic based on the autoregressive model with constant and time trend terms:

\[
y_t = \mu + \rho y_{t-1} + 6t + u_t
\]

You can specify several truncation points $I$ for weighted variance estimators by using the PHILLIPS=$(I_1, \ldots, I_n)$ specification. The statistic for each truncation point $I$ is computed as
\[ \sigma^2_T = \frac{1}{T} \sum_{t=1}^{T} \hat{\epsilon}_t^2 + \frac{2}{T} \sum_{s=1}^{l} w_{sl} \sum_{r=m+1}^{T} \hat{\epsilon}_r \hat{\epsilon}_{r-s} \]

where \( w_{sl} = 1 - s/(l + 1) \) and \( \hat{\epsilon}_t \) are OLS residuals. If you specify the PHILLIPS option without specifying truncation points, the default truncation point is \( \max(1, \sqrt{T}/5) \) where \( T \) is the number of observations.

The Phillips-Perron test can be used in general time series models since its limiting distribution is derived in the context of a class of weakly dependent and heterogeneously distributed data. The marginal probability for the Phillips-Perron test is computed assuming that error disturbances are normally distributed.

When there are regressors in the MODEL statement, the PHILLIPS option computes the Phillips-Ouliaris cointegration test statistic by using the least squares residuals. The normalized cointegrating vector is estimated using OLS regression. Therefore, the cointegrating vector estimates might vary with the regressand (normalized element) unless the regression R-square is 1.

The marginal probabilities for cointegration testing are not produced. You can refer to Phillips and Ouliaris (1990) tables Ia–Ic for the \( Z_d \) test and tables IIa–IIc for the \( Z_r \) test. The standard residual-based cointegration test can be obtained using the NOINT option in the MODEL statement, while the demeaned test is computed by including the intercept term. To obtain the demeaned and detrended test statistic in the Phillips-Ouliaris cointegration test. The asymptotic distribution is tabulated in tables IIa–IIc of Phillips and Ouliaris (1990) notation. We adopt the notation introduced in Hamilton. To distinguish from Student’s \( t \) distribution, these two statistics are named accordingly as \( \varphi (\rho) \) and \( \tau (\tau) \).

The ADF or ADF= suboption produces the augmented Dickey-Fuller unit root test (Dickey and Fuller; 1979). As in the Phillips-Perron test, three regression models can be specified for the null hypothesis for the augmented Dickey-Fuller test (zero mean, single mean, and trend). These models assume that the disturbances are distributed as white noise. The augmented Dickey-Fuller test can account for the serial correlation between the disturbances in some way. The model, with the time trend specification for example, is

\[ y_t = \mu + \rho y_{t-1} + \delta t + \gamma_1 y_{t-1} + \ldots + \gamma_p y_{t-p} + \epsilon_t \]

This formulation has the advantage that it can accommodate higher-order autoregressive processes in \( \mu \). The test statistic follows the same distribution as the Dickey-Fuller test statistic. For more information, see the section PROBDF Function for Dickey-Fuller Tests.

In the presence of regressors, the ADF option tests the cointegration relation between the dependent variable and the regressors. Following Engle and Granger (1987), a two-step estimation and testing procedure is carried out, in a fashion similar to the Phillips-Ouliaris test. The OLS residuals of the regression in the MODEL statement are used to compute the \( t \) statistic of the augmented Dickey-Fuller regression in a second step. Three cases arise based on which type of deterministic terms are included in the first step of regression. Only the constant term and linear trend cases are practically useful (Davidson and MacKinnon; 1993, page 721), and therefore are computed and reported. The test statistic, as shown in Phillips and Ouliaris (1990), follows the same distribution as the \( Z_r \) statistic in the Phillips-Ouliaris cointegration test. The asymptotic distribution is tabulated in tables IIa–IIc of Phillips and Ouliaris (1990), and the finite sample distribution is obtained in Table 2 and Table 3 in Engle and Yoo (1987) by Monte Carlo simulation.

The experimental ERS or ERS= suboption and NP or NP= suboption provide a class of efficient unit root tests, in the sense that they reduce the size distortion and improve the power compared with traditional unit root tests such as augmented Dickey-Fuller and Phillips-Perron tests. Two test statistics are provided by the ERS test: the point optimal test and the DF-GLS test, which are originally proposed in Elliott, Rothenberg, and Stock (1996). Four different tests, discussed in Ng and Perron (2001), are reported by NP test. These four tests include the two in the ERS test and two other tests, the modified PP test and the modified point optimal test, discussed in Ng and Perron (2001). The authors suggest using the modified AIC to select the optimal lag length in the augmented Dickey-Fuller type regression. The maximum lag length can be provided by using the NP= suboption. The default maximum lag length is 8.

The KPSS, KPSS=(KERNEL=TYPE), or KPSS=(KERNEL=TYPE TRUNCPOINTERMETHOD) specifications of the STATIONARITY= option produce the Kwiatkowski, Phillips, Schmidt, and Shin (1992) (KPSS) unit root test.

Unlike the null hypothesis of the Dickey-Fuller and Phillips-Perron tests, the null hypothesis of the KPSS states that the time series is stationary. As a result, it tends to reject a random walk more often. If the model does not have an intercept, the KPSS option performs the KPSS test for three null hypothesis cases: zero mean, single mean, and deterministic trend. Otherwise, it reports the single mean and deterministic trend only. It computes a test statistic and provides tabulated critical values (see Hobijn, Franses, and Ooms (2004) for the hypothesis that the random walk component of the time series is equal to zero in the following cases (for details, see Kwiatkowski, Phillips, Schmidt, and Shin (KPSS) Unit Root Test):

Zero mean

computes the KPSS test statistic based on the zero mean autoregressive model. The \( p \)-value reported is used from Hobijn, Franses, and Ooms (2004).
computes the KPSS test statistic based on the autoregressive model with a constant term. The $p$-value reported is used from Kwiatkowski et al. (1992).

$$y_t = \mu + u_t$$

Trend

computes the KPSS test statistic based on the autoregressive model with constant and time trend terms. The $p$-value reported is from Kwiatkowski et al. (1992).

$$y_t = \mu + \delta t + u_t$$

This test depends on the long-run variance of the series being defined as

$$\omega_j^2 = \frac{1}{T} \sum_{t=1}^{T} \hat{\sigma}^2_t + \frac{2}{T} \sum_{l=1}^{L} w_d T \sum_{t=1}^{T} \hat{\sigma}_t \hat{\sigma}_{t-l}$$

where $w_d$ is a kernel, $s$ is a maximum lag (truncation point), and $\hat{\sigma}_t$ are OLS residuals or original data series. You can specify two types of the kernel:

KERNEL=NW | BART

Newey-West (or Bartlett) kernel

$$w(s,l) = 1 - \frac{s}{l+1}$$

KERNEL=QS

Quadratic spectral kernel

$$w(s,l) = w(x) = \frac{\sin(5\pi x)}{12\pi x} \left( \frac{\sin(5\pi x)}{5\pi x} - \cos(5\pi x) \right)$$

You can set the truncation point $s$ by using three different methods:

SCHW=c

Schwert maximum lag formula

$$s = \max \left\{ 1, \lfloor \sqrt{c \left( \frac{T}{100} \right)^{1/4}} \rfloor \right\}$$

LAG=s

LAG=s manually defined number of lags.

AUTO

Automatic bandwidth selection (Hobijn, Franses, and Ooms; 2004) (for details, see Kwiatkowski, Phillips, Schmidt, and Shin (KPSS) Unit Root Test).

If STATIONARITY=KPSS is defined without additional parameters, the Newey-West kernel is used. For the Newey-West kernel the default is the Schwert truncation point method with $c = 4$. For the quadratic spectral kernel the default is AUTO.

The KPSS test can be used in general time series models since its limiting distribution is derived in the context of a class of weakly dependent and heterogeneously distributed data. The limiting probability for the KPSS test is computed assuming that error disturbances are normally distributed.

The asymptotic distribution of the test does not depend on the presence of regressors in the MODEL statement.

The marginal probabilities for the test are reported. They are copied from Kwiatkowski et al. (1992) and Hobijn, Franses, and Ooms (2004). When there is an intercept in the model, results for mean and trend tests statistics are provided.

Examples: To test for stationarity of regression residuals, using default KERNEL= NW and SCHW= 4, you can use the following code:

/*-- test for stationarity of regression residuals --*/
To test for stationarity of regression residuals, using quadratic spectral kernel and automatic bandwidth selection, you can use:

```latex
/*-- test for stationarity using quadratic 
spectral kernel and automatic bandwidth selection --*/
proc autoreg data=a;
  model y= / stationarity = (KPSS=(KERNEL=QS AUTO));
run;
```

**TP**

TP=(Z=value)

specifies the turning point test for independence. The Z= suboption specifies the type of the time series or residuals to be tested. The values of the Z= suboption are as follows:

- **Y**
  - specifies the regressand. The default is Z=Y.
- **RO**
  - specifies the OLS residuals.
- **R**
  - specifies the residuals of the final model.
- **RM**
  - specifies the structural residuals of the final model.
- **SR**
  - specifies the standardized residuals of the final model, defined by residuals over the square root of the conditional variance.

**URSQ**

prints the uncentered regression $R^2$. The uncentered regression $R^2$ is useful to compute Lagrange multiplier test statistics, since most LM test statistics are computed as $T \times URSQ$, where $T$ is the number of observations used in estimation.

**VNRRANK**

VNRRANK=(option-list)

specifies the rank version of the von Neumann ratio test for independence. The following options can be used in the VNRRANK=( ) option. The options are listed within parentheses and separated by commas.

- **PVALUE=**
  - specifies the way to calculate the $p$-value. By default or if PVALUE=DIST is specified, the $p$-value is calculated according to the asymptotic distribution of the statistic (that is, the standard normal distribution). Otherwise, for samples of size less than 100, the $p$-value is obtained though Monte Carlo simulation.

  - **DIST**
    - specifies the way to calculate the $p$-value. By default or if PVALUE=DIST is specified, the $p$-value is calculated according to the asymptotic distribution of the statistic (that is, the standard normal distribution). Otherwise, for samples of size less than 100, the $p$-value is obtained though Monte Carlo simulation.

  - **SIM**
    - specifies the way to calculate the $p$-value. By default or if PVALUE=DIST is specified, the $p$-value is calculated according to the asymptotic distribution of the statistic (that is, the standard normal distribution). Otherwise, for samples of size less than 100, the $p$-value is obtained though Monte Carlo simulation.

-Z=value

specifies the type of the time series or residuals to be tested. The values of the Z= suboption are as follows:

- **Y**
  - specifies the regressand. The default is Z=Y.
- **RO**
  - specifies the OLS residuals.
- **R**
  - specifies the residuals of the final model.
specifies the residuals of the final model.

RM

specifies the structural residuals of the final model.

SR

specifies the standardized residuals of the final model, defined by residuals over the square root of the conditional variance.

**Stepwise Selection Options**

**BACKSTEP**

removes insignificant autoregressive parameters. The parameters are removed in order of least significance. This backward elimination is done only once on the Yule-Walker estimates computed after the initial ordinary least squares estimation. The BACKSTEP option can be used with all estimation methods since the initial parameter values for other estimation methods are estimated using the Yule-Walker method.

**SLSTAY=value**

specifies the significance level criterion to be used by the BACKSTEP option. The default is SLSTAY=.05.

**Estimation Control Options**

**CONVERGE=value**

specifies the convergence criterion. If the maximum absolute value of the change in the autoregressive parameter estimates between iterations is less than this amount, then convergence is assumed. The default is CONVERGE=.001.

If the GARCH= option and/or the HETERO statement is specified, convergence is assumed when the absolute maximum gradient is smaller than the value specified by the CONVERGE= option or when the relative gradient is smaller than 1E-8. By default, CONVERGE=1E-5.

**INITIAL=(initial-values)**

**START=(initial-values)**

specifies initial values for some or all of the parameter estimates. The values specified are assigned to model parameters in the same order as the parameter estimates are printed in the AUTOREG procedure output. The order of values in the INITIAL= or START= option is as follows: the intercept, the regressor coefficients, the autoregressive parameters, the ARCH parameters, the GARCH parameters, the inverted degrees of freedom for Student's t distribution, the start-up value for conditional variance, and the heteroscedasticity model parameters specified by the HETERO statement.

The following is an example of specifying initial values for an AR(1)-GARCH(1, 1) model with regressors X1 and X2:

```plaintext
/*-- specifying initial values --*/
model y = w x / nlag=1 garch=(p=1,q=1)
    initial=(1 1 1 .5 .8 .1 .6);
```

The model specified by this MODEL statement is

\[ y_t = \beta_0 + \beta_1 w_t + \beta_2 x_{2t} + v_t \]
\[ v_t = \varphi v_{t-1} \]
\[ \varepsilon_t = \sqrt{h_t} \varepsilon_t \]
\[ h_t = \omega + \alpha_t \varepsilon_{t-1}^2 + \gamma h_{t-1} \]
\[ \varepsilon_t \sim \mathcal{N}(0, \sigma_e^2) \]

The initial values for the regression parameters, INTERCEPT (\( \beta_0 \)), X1 (\( \beta_1 \)), and X2 (\( \beta_2 \)), are specified as 1. The initial value of the AR(1) coefficient (\( \varphi \)) is specified as 0.5. The initial value of ARCH0 (\( \omega \)) is 0.8, the initial value of ARCH1 (\( \alpha_0 \)) is 0.1, and the initial value of GARCH1 (\( \gamma \)) is 0.6.

When you use the RESTRICT statement, the initial values specified by the INITIAL= option should satisfy the restrictions specified for the parameter estimates. If they do not, the initial values you specify are adjusted to satisfy the restrictions.
LDW

specifies that p-values for the Durbin-Watson test be computed using a linearized approximation of the design matrix when the model is nonlinear due to the presence of an autoregressive error process. (The Durbin-Watson tests of the OLS linear regression model residuals are not affected by the LDW option.) Refer to White (1992) for Durbin-Watson testing of nonlinear models.

\textbf{MAXITER=number}

sets the maximum number of iterations allowed. The default is MAXITER=50.

\textbf{METHOD=value}

requests the type of estimates to be computed. The values of the METHOD= option are as follows:

\textbf{METHOD=ML}

specifies maximum likelihood estimates.

\textbf{METHOD=ULS}

specifies unconditional least squares estimates.

\textbf{METHOD=YW}

specifies Yule-Walker estimates.

\textbf{METHOD=ITYW}

specifies iterative Yule-Walker estimates.

If the GARCH= or LAGDEP option is specified, the default is METHOD=ML. Otherwise, the default is METHOD=YW.

\textbf{NOMISS}

requests the estimation to the first contiguous sequence of data with no missing values. Otherwise, all complete observations are used.

\textbf{OPTMETHOD=QN | TR}

specifies the optimization technique when the GARCH or heteroscedasticity model is estimated. The OPTMETHOD=QN option specifies the quasi-Newton method. The OPTMETHOD=TR option specifies the trust region method. The default is OPTMETHOD=QN.
The HETERO statement specifies a model for the conditional variance $h_t$. The vector $\mathbf{z}_t$ is composed of the variables listed in the HETERO statement, $\eta$ is a parameter vector, and $l(\cdot)$ is a link function that depends on the value of the LINK= option. In the printed output, $HET0$ represents the estimate of sigma, while $HET1$- $HETn$ are the estimates of parameters in the $\eta$ vector.

The keyword XBETA can be used in the variables list to refer to the model predicted value $\mathbf{X}'\beta$. If XBETA is specified in the variables list, other variables in the HETERO statement will be ignored. In addition, XBETA cannot be specified in the GARCH process.

For heteroscedastic regression models without GARCH effects, the errors $\epsilon_t$ are assumed to be uncorrelated — the heteroscedasticity models specified by the HETERO statement cannot be combined with an autoregressive model for the errors. Thus, when a HETERO statement is used, the NLAG= option cannot be specified unless the GARCH= option is also specified.

You can specify the following options in the HETERO statement.

**LINK=value**

specifies the functional form of the heteroscedasticity model. By default, LINK=EXP. If you specify a GARCH model with the HETERO statement, the model is estimated using LINK=LINEAR only. For details, see the section Using the HETERO Statement with GARCH Models. Values of the LINK= option are as follows:

- **EXP**
  - specifies the exponential link function. The following model is estimated when you specify LINK=EXP:
    
    $h_t = \exp(\mathbf{z}'_t, \eta)$

- **SQUARE**
  - specifies the square link function. The following model is estimated when you specify LINK=SQUARE:
    
    $h_t = (1 + \mathbf{z}'_t, \eta)^2$

- **LINEAR**
  - specifies the linear function; that is, the HETERO statement variables predict the error variance linearly. The following model is estimated when you specify LINK=LINEAR:
    
    $h_t = (1 + \mathbf{z}'_t, \eta)$

**COEF=value**

imposes constraints on the estimated parameters $\eta$ of the heteroscedasticity model. The values of the COEF= option are as follows:

- **NONNEG**
  - specifies that the estimated heteroscedasticity parameters $\eta$ must be nonnegative. When the HETERO statement is used in conjunction with the GARCH= option, the default is COEF=NONNEG.

- **UNIT**
  - constrains all heteroscedasticity parameters $\eta$ to equal 1.

- **ZERO**
  - constrains all heteroscedasticity parameters $\eta$ to equal 0.

- **UNREST**
  - specifies unrestricted estimation of $\eta$. When the GARCH= option is not specified, the default is COEF=UNREST.

**STD=value**

imposes constraints on the estimated standard deviation $\sigma$ of the heteroscedasticity model. The values of the STD= option are as follows:

- **NONNEG**
  - specifies that the estimated standard deviation parameter $\sigma$ must be nonnegative.

- **UNIT**
  - constrains the estimated standard deviation parameter $\sigma$ to equal 1.
constrains the standard deviation parameter $\sigma$ to equal 1.

UNREST

specifies unrestricted estimation of $\sigma$. This is the default.

TEST=LM

produces a Lagrange multiplier test for heteroscedasticity. The null hypothesis is homoscedasticity; the alternative hypothesis is heteroscedasticity of the form specified by the HETERO statement. The power of the test depends on the variables specified in the HETERO statement.

The test may give different results depending on the functional form specified by the LINK= option. However, in many cases the test does not depend on the LINK= option. The test is invariant to the form of $h_\epsilon$ when $h_\epsilon(0) = 1$ and $h_\epsilon'(0) \neq 0$. (The condition $h_\epsilon(0) = 1$ is satisfied except when the NOCONST option is specified with LINK=SQUARE or LINK=LINEAR.)

NOCONST

specifies that the heteroscedasticity model does not include the unit term for the LINK=SQUARE and LINK=LINEAR options. For example, the following model is estimated when you specify the options LINK=SQUARE NOCONST:

$$h_\epsilon = (x', \eta)^2$$

The RESTRICT statement provides constrained estimation. The syntax of the RESTRICT statement is

RESTRICT equation , ..., equation ;

The RESTRICT statement places restrictions on the parameter estimates for covariates in the preceding MODEL statement. The AR, GARCH, and HETERO parameters are also supported in the RESTRICT statement. Any number of RESTRICT statements can follow a MODEL statement. Several restrictions can be specified in a single RESTRICT statement by separating the individual restrictions with commas.

Each restriction is written as a linear equation composed of constants and parameter names. Refer to model parameters by the name of the corresponding regressor variable. Each name used in the equation must be a regressor in the preceding MODEL statement. Use the keyword INTERCEPT to refer to the intercept parameter in the model. See the section OUTEST= Data Set for the names of these parameters.

The following is an example of a RESTRICT statement:

```plaintext
model y = a b c d;
restrict a=b=0, 2*d-c=0;
```

When restricting a linear combination of parameters to be 0, you can omit the equal sign. For example, the following RESTRICT statement is equivalent to the preceding example:
restrict a+b, 2*d-c;

The following RESTRICT statement constrains the parameters estimates for three regressors (X1, X2, and X3) to be equal:
restrict x1 = x2, x2 = x3;

The preceding restriction can be abbreviated as follows:
restrict x1 = x2 = x3;

The following example shows how to specify AR, GARCH, and HETERO parameters in the RESTRICT statement:

```
model y = a / nlag=2 garch=(p=2,q=3,mean=sqrt);
```

```
hetero c d;
```

```
restrict _A_1=0, _AH_2=0.2, _HET_2=1, _DELTA_=0.1;
```

Only simple linear combinations of parameters can be specified in RESTRICT statement expressions; complex expressions that involve parentheses, division, functions, or complex products are not allowed.

The AUTOREG Procedure supports a TEST statement for linear hypothesis tests. The syntax of the TEST statement is

```
TEST equation , ..., equation / option ;
```

The TEST statement tests hypotheses about the covariates in the model that are estimated by the preceding MODEL statement. The AR, GARCH, and HETERO parameters are also supported in the TEST statement. Each equation specifies a linear hypothesis to be tested. If more than one equation is specified, the equations are separated by commas.

Each test is written as a linear equation composed of constants and parameter names. Refer to parameters by the name of the corresponding regressor variable. Each name used in the equation must be a regressor in the preceding MODEL statement. Use the keyword INTERCEPT to refer to the intercept parameter in the model. See the section OUTEST= Data Set for the names of these parameters.

You can specify the following options in the TEST statement:

**TYPE=value**

specifies the test statistics to use. The default is TYPE=F. The following values for TYPE= option are available:

- **F**
  produces an F test. This option is supported for all models specified in MODEL statement.

- **WALD**
  produces a Wald test. This option is supported for all models specified in MODEL statement.

- **LM**
  produces a Lagrange multiplier test. This option is supported only when the GARCH= option is specified (for example, when there is a statement like `MODEL Y = C D I / GARCH=(Q=2)`).

- **LR**
  produces a likelihood ratio test. This option is supported only when the GARCH= option is specified (for example, when there is a statement like `MODEL Y = C D I / GARCH=(Q=2)`).

- **ALL**
  produces all tests applicable for a particular model. For non-GARCH-type models, only F and Wald tests are output. For all other models, all four tests (LR, LM, F, and Wald) are computed.
The following example of a TEST statement tests the hypothesis that the coefficients of two regressors A and B are equal:

```
model y = a b c d;
   test a = b;
```

To test separate null hypotheses, use separate TEST statements. To test a joint hypothesis, specify the component hypotheses on the same TEST statement, separated by commas.

For example, consider the following linear model:

\[ y_t = \beta_0 + \beta_1 x_{1t} + \beta_2 x_{2t} + \epsilon_t \]

The following statements test the two hypotheses \( H_0 : \beta_0 = 1 \) and \( H_0 : \beta_1 + \beta_2 = 0 \):

```
model y = x1 x2;
   test intercept = 1;
   test x1 + x2 = 0;
```

The following statements test the joint hypothesis \( H_0 : \beta_0 = 1 \) and \( \beta_1 + \beta_2 = 0 \):

```
model y = x1 x2;
   test intercept = 1, x1 + x2 = 0;
```

To illustrate the TYPE= option, consider the following examples.

```
model Y = C D I / garch=(q=2);
   test C + D = 1;
```

The preceding statements produce only one default test, the \( F \) test.

```
model Y = C D I / garch=(q=2);
   test C + D = 1 / type = LR;
```

The preceding statements produce one of four tests applicable for GARCH-type models, the likelihood ratio test.

```
model Y = C D I / nlag = 2;
   test C + D = 1 / type = LM;
```

The preceding statements produce the warning and do not output any test because the Lagrange multiplier test is not applicable for non-GARCH models.

```
model Y = C D I / nlag=2;
   test C + D = 1 / type = ALL;
```

The preceding statements produce all tests that are applicable for non-GARCH models (namely, the \( F \) and Wald tests). The TYPE= prefix is optional. Thus the test statement in the previous example could also have been written as:

```
test C + D = 1 / ALL;
```

The following example shows how to test AR, GARCH, and HETERO parameters:

```
model y = a b / nlag=2 garch=(p=2,q=3,mean=sqrt);
   hetero c d;
   test _A_1=0,_AH_2=0.2,_HET_2=1,_DELTA_=0.1;
```

The OUTPUT statement creates an output SAS data set as specified by the following options.

```
OUTPUT OUT=SAS-data-set keyword = options ...;
```

The OUTPUT statement creates an output SAS data set as specified by the following options.
names the output SAS data set containing the predicted and transformed values. If the OUT= option is not specified, the new data set is named according to the DATAn convention.

**ALPHACLI=** number

sets the confidence limit size for the estimates of future values of the response time series. The ALPHACLI= value must be between 0 and 1. The resulting confidence interval has 1-number confidence. The default is ALPHACLI=.05, corresponding to a 95% confidence interval.

**ALPHACLM=** number

sets the confidence limit size for the estimates of the structural or regression part of the model. The ALPHACLI= value must be between 0 and 1. The resulting confidence interval has 1-number confidence. The default is ALPHACLM=.05, corresponding to a 95% confidence interval.

**ALPHACSM=** .01 | .05 | .10

specifies the significance level for the upper and lower bounds of the CUSUM and CUSUMSQ statistics output by the CUSUMLB=, CUSUMUB=, CUSUMSQLB=, and CUSUMSQUB= options. The significance level specified by the ALPHACSM= option can be .01, .05, or .10. Other values are not supported.

The following options are of the form KEYWORD=name, where KEYWORD specifies the statistic to include in the output data set and name gives the name of the variable in the OUT= data set containing the statistic.

**BLUS=** variable

specifies the name of a variable to contain the values of the Theil’s BLUS residuals. Refer to Theil (1971) for more information on BLUS residuals.

**CEV=** variable

**HT=** variable

writes to the output data set the value of the error variance $\sigma^2$ from the heteroscedasticity model specified by the HETERO statement or the value of the conditional error variance $h_{t}$ by the GARCH= option in the MODEL statement.

**CPEV=** variable

writes the conditional prediction error variance to the output data set. The value of conditional prediction error variance is equal to that of the conditional error variance when there are no autoregressive parameters. For the exponential GARCH model, conditional prediction error variance cannot be calculated. See the section Predicted Values later in this chapter for details.

**CONSTANT=** variable

writes the transformed intercept to the output data set. The details of the transformation are described in Computational Methods later in this chapter.

**CUSUM=** variable

specifies the name of a variable to contain the CUSUM statistics.

**CUSUMSQ=** variable

specifies the name of a variable to contain the CUSUMSQ statistics.

**CUSUMUB=** variable

specifies the name of a variable to contain the upper confidence bound for the CUSUM statistic.

**CUSUMLB=** variable

specifies the name of a variable to contain the lower confidence bound for the CUSUM statistic.

**CUSUMSQUB=** variable

specifies the name of a variable to contain the upper confidence bound for the CUSUMSQ statistic.

**CUSUMSQLB=** variable
specifies the name of a variable to contain the lower confidence bound for the CUSUMSQ statistic.

**LCL=** *name *

writes the lower confidence limit for the predicted value (specified in the PREDICTED= option) to the output data set. The size of the confidence interval is set by the ALPHACL= option. See the section Predicted Values later in this chapter for details.

**LCLM=** *name *

writes the lower confidence limit for the structural predicted value (specified in the PREDICTEDM= option) to the output data set under the name given. The size of the confidence interval is set by the ALPHACL= option.

**PREDICTED=** *name *

**P=** *name *

writes the predicted values to the output data set. These values are formed from both the structural and autoregressive parts of the model. See the section Predicted Values later in this chapter for details.

**PREDICTEDM=** *name *

**PM=** *name *

writes the structural predicted values to the output data set. These values are formed from only the structural part of the model. See the section Predicted Values later in this chapter for details.

**RECPEV=** *variable *

specifies the name of a variable to contain the part of the predictive error variance (\( \sigma^2 \)) that is used to compute the recursive residuals.

**RECRS=** *variable *

specifies the name of a variable to contain recursive residuals. The recursive residuals are used to compute the CUSUM and CUSUMSQ statistics.

**RESIDUAL=** *name *

**R=** *name *

writes the residuals from the predicted values based on both the structural and time series parts of the model to the output data set.

**RESIDUALM=** *name *

**RM=** *name *

writes the residuals from the structural prediction to the output data set.

**TRANSFORM=** *variables *

transforms the specified variables from the input data set by the autoregressive model and writes the transformed variables to the output data set. The details of the transformation are described in Computational Methods later in this chapter. If you need to reproduce the data suitable for reestimation, you must also transform an intercept variable. To do this, transform a variable that is all 1s or use the CONSTANT= option.

**UCL=** *name *

writes the upper confidence limit for the predicted value (specified in the PREDICTED= option) to the output data set. The size of the confidence interval is set by the ALPHACL= option. See the section Predicted Values later in this chapter for details.

**UCLM=** *name *

writes the upper confidence limit for the structural predicted value (specified in the PREDICTEDM= option) to the output data set. The size of the confidence interval is set by the ALPHACL= option.
Details: AUTOREG Procedure

Missing Values

PROC AUTOREG skips any missing values at the beginning of the data set. If the NOMISS option is specified, the first contiguous set of data with no missing values is used; otherwise, all data with nonmissing values for the independent and dependent variables are used. Note, however, that the observations containing missing values are still needed to maintain the correct spacing in the time series. PROC AUTOREG can generate predicted values when the dependent variable is missing.

Autoregressive Error Model

The regression model with autocorrelated disturbances is as follows:

\[ y_t = \mathbf{x}_t' \beta + \varepsilon_t \]
\[ \varepsilon_t = \varepsilon_{t-1} - \phi_1 \varepsilon_{t-2} - \ldots - \phi_m \varepsilon_{t-m} \]
\[ \varepsilon_t \sim N(0, \sigma^2) \]

In these equations, \( y_t \) are the dependent values, \( \mathbf{x}_t \) is a column vector of regressor variables, \( \beta \) is a column vector of structural parameters, and \( \varepsilon_t \) is normally and independently distributed with a mean of 0 and a variance of \( \sigma^2 \). Note that in this parameterization, the signs of the autoregressive parameters are reversed from the parameterization documented in most of the literature.

PROC AUTOREG offers four estimation methods for the autoregressive error model. The default method, Yule-Walker (YW) estimation, is the fastest computationally. The Yule-Walker method used by PROC AUTOREG is described in Gallant and Goebel (1976). Harvey (1981) calls this method the two-step full transform method. The other methods are iterated YW, unconditional least squares (ULS), and maximum likelihood (ML). The ULS method is also referred to as nonlinear least squares (NLS) or exact least squares (ELS).

You can use all of the methods with data containing missing values, but you should use ML estimation if the missing values are plentiful. See the section Alternative Autocorrelation Correction Methods later in this chapter for further discussion of the advantages of different methods.

The Yule-Walker Method

Let \( \phi \) represent the vector of autoregressive parameters,
and let the variance matrix of the error vector $\mathbf{v} = (v_1, \ldots, v_T)'$ be $\Sigma$.

$$E(\mathbf{vv}') = \Sigma = \sigma^2 \mathbf{V}$$

If the vector of autoregressive parameters $\mathbf{\phi}$ is known, the matrix $\mathbf{V}$ can be computed from the autoregressive parameters. $\mathbf{\Sigma}$ is then $\sigma^2 \mathbf{V}$. Given $\mathbf{\Sigma}$, the efficient estimates of regression parameters $\hat{\beta}$ can be computed using generalized least squares (GLS). The GLS estimates then yield the unbiased estimate of the variance $\sigma^2$.

The Yule-Walker method alternates estimation of $\mathbf{\phi}$ using generalized least squares with estimation of $\mathbf{\Sigma}$ using the Yule-Walker equations applied to the sample autocorrelation function. The YW method starts by forming the OLS estimate of $\hat{\beta}$. Next, $\mathbf{\phi}$ is estimated from the sample autocorrelation function of the OLS residuals by using the Yule-Walker equations. Then $\mathbf{V}$ is estimated from the estimate of $\mathbf{\phi}$, and $\mathbf{\Sigma}$ is estimated from $\mathbf{V}$ and the OLS estimate of $\sigma^2$. The autocorrelation corrected estimates of the regression parameters $\hat{\beta}$ are then computed using GLS, using the estimated $\mathbf{\Sigma}$ matrix. These are the Yule-Walker estimates.

If the ITER option is specified, the Yule-Walker residuals are used to form a new sample autocorrelation function, the new autocorrelation function is used to form a new estimate of $\mathbf{\phi}$ and $\mathbf{V}$, and the GLS estimates are recomputed using the new variance matrix. This alternation of estimates continues until either the maximum change in the $\mathbf{\phi}$ estimate between iterations is less than the value specified by the CONVERGE= option or the maximum number of allowed iterations is reached. This produces the iterated Yule-Walker estimates. Iteration of the estimates may not yield much improvement.

The Yule-Walker equations, solved to obtain $\mathbf{\phi}$ and a preliminary estimate of $\sigma^2$, are

$$\mathbf{R} \mathbf{\phi} = -\mathbf{r}$$

Here $\mathbf{r} = (r_1, \ldots, r_m)'$, where $r_i$ is the lag $i$ sample autocorrelation. The matrix $\mathbf{R}$ is the Toeplitz matrix whose $i,j$th element is $r_{j-i}$. If you specify a subset model, then only the rows and columns of $\mathbf{R}$ and $\mathbf{r}$ corresponding to the subset of lags specified are used.

If the BACKSTEP option is specified, for purposes of significance testing, the matrix $|\mathbf{R} \mathbf{r}|$ is treated as a sum-of-squares-and-crossproducts matrix arising from a simple regression with $N-k$ observations, where $k$ is the number of estimated parameters.

### The Unconditional Least Squares and Maximum Likelihood Methods

Define the transformed error, $\mathbf{e}$, as

$$\mathbf{e} = \mathbf{I}^{-1} \mathbf{n}$$

where $\mathbf{n} = \mathbf{y} - \mathbf{X} \hat{\beta}$.

The unconditional sum of squares for the model, $S$, is

$$S = \mathbf{n}' \mathbf{V}^{-1} \mathbf{n} = \mathbf{e}' \mathbf{e}$$

The ULS estimates are computed by minimizing $S$ with respect to the parameters $\beta$ and $\phi$.

The full log likelihood function for the autoregressive error model is

$$l = -\frac{N}{2} \ln(2\pi) - \frac{N}{2} \ln(\sigma^2) - \frac{1}{2} \ln(|\mathbf{V}|) - \frac{S}{2\sigma^2}$$

where $|\mathbf{V}|$ denotes determinant of $\mathbf{V}$. For the ML method, the likelihood function is maximized by minimizing an equivalent sum-of-squares function.

Maximizing $l$ with respect to $\sigma^2$ (and concentrating $\sigma^2$ out of the likelihood) and dropping the constant term $-\frac{N}{2} \ln(2\pi) + 1 - \ln(N)$ produces the concentrated log likelihood function

$$l_c = -\frac{N}{2} \ln(S/|\mathbf{V}|^{1/N})$$

Rewriting the variable term within the logarithm gives
PROC AUTOREG computes the ML estimates by minimizing the objective function $S_{ML} = |L|^{1/N} e e' |L|^{1/N}$.

The maximum likelihood estimates may not exist for some data sets (Anderson and Mentz; 1980). This is the case for very regular data sets, such as an exact linear trend.

### Computational Methods

#### Sample Autocorrelation Function

The sample autocorrelation function is computed from the structural residuals or noise $n_i = y_i - X_i \hat{\theta}$, where $\hat{\theta}$ is the current estimate of $\theta$. The sample autocorrelation function is the sum of all available lagged products of $n_i$ of order $j$ divided by $\ell + j$, where $\ell$ is the number of such products.

If there are no missing values, then $\ell = N$, the number of observations. In this case, the Toeplitz matrix of autocorrelations, $\mathbf{R}$, is at least positive semidefinite. If there are missing values, these autocorrelation estimates of $r$ can yield an $\mathbf{R}$ matrix that is not positive semidefinite. If such estimates occur, a warning message is printed, and the estimates are tapered by exponentially declining weights until $\mathbf{R}$ is positive definite.

#### Data Transformation and the Kalman Filter

The calculation of $\mathbf{V}$ from $\mathbf{p}$ for the general AR$(m)$ model is complicated, and the size of $\mathbf{V}$ depends on the number of observations. Instead of actually calculating $\mathbf{V}$ and performing GLS in the usual way, in practice a Kalman filter algorithm is used to transform the data and compute the GLS results through a recursive process.

In all of the estimation methods, the original data are transformed by the inverse of the Cholesky root of $\mathbf{V}$ — that is, $\mathbf{V}^{-1/2}$ with $\mathbf{L}$ lower triangular. For an AR$(m)$ model, $\mathbf{L}^{-1}$ is a band diagonal matrix with $m$ anomalous rows at the beginning and the autoregressive parameters along the remaining rows. Thus, if there are no missing values, after the first $m$ observations the data are transformed as

$$z_t = x_t + \phi_1 x_{t-1} + \ldots + \phi_m x_{t-m}$$

The transformation is carried out using a Kalman filter, and the lower triangular matrix $\mathbf{L}$ is never directly computed. The Kalman filter algorithm, as it applies here, is described in Harvey and Phillips (1979) and Jones (1980). Although $\mathbf{L}$ is not computed explicitly, for ease of presentation the remaining discussion is in terms of $\mathbf{L}$. If there are missing values, then the submatrix of $\mathbf{L}$ consisting of the rows and columns with nonmissing values is used to generate the transformations.

#### Gauss-Newton Algorithms

The ULS and ML estimates employ a Gauss-Newton algorithm to minimize the sum of squares and maximize the log likelihood, respectively. The relevant optimization is performed simultaneously for both the regression and AR parameters. The OLS estimates of $\beta$ and the Yule-Walker estimates of $\phi$ are used as starting values for these methods.

The Gauss-Newton algorithm requires the derivatives of $e$ or $|L|^{1/N} e$ with respect to the parameters. The derivatives with respect to the parameter vector $\beta$ are

$$\frac{\partial e}{\partial \beta} = -L^{-1}X$$

$$\frac{\partial |L|^{1/N} e}{\partial \beta} = -|L|^{1/N} L^{-1}X$$

These derivatives are computed by the transformation described previously. The derivatives with respect to $\phi$ are computed by differentiating the Kalman filter recurrences and the equations for the initial conditions.

#### Variance Estimates and Standard Errors

For the Yule-Walker method, the estimate of the error variance, $\hat{\sigma}^2$, is the error sum of squares from the last application of GLS, divided by the error degrees of freedom (number of observations $N$ minus the number of free parameters).

The variance-covariance matrix for the components of $\beta$ is the matrix of derivatives of $e$ with respect to the parameters. For the ML method, $\mathbf{J}$ is the matrix of derivatives of $|\mathbf{L}|^{1/N} e$ divided by $|\mathbf{L}|^{1/N}$. The estimate of the variance-covariance matrix of $\beta$ assuming that $\phi$ is known is $\hat{\sigma}^2(XV^{-1}X)^{-1}$. 
Park and Mitchell (1980) investigated the small sample performance of the standard error estimates obtained from some of these methods. In particular, simulating an AR(1) model for the noise term, they found that the standard errors calculated using GLS with an estimated autoregressive parameter underestimated the true standard errors. These estimates of standard errors are the ones calculated by PROC AUTOREG with the Yule-Walker method.

The estimates of the standard errors calculated with the ULS or ML method take into account the joint estimation of the AR and the regression parameters and may give more accurate standard-error values than the YW method. At the same values of the autoregressive parameters, the ULS and ML standard errors will always be larger than those computed from Yule-Walker. However, simulations of the models used by Park and Mitchell (1980) suggest that the ULS and ML standard error estimates can also be underestimates. Caution is advised, especially when the estimated autocorrelation is high and the sample size is small.

High autocorrelation in the residuals is a symptom of lack of fit. An autoregressive error model should not be used as a nostrum for models that simply do not fit. It is often the case that time series variables tend to move as a random walk. This means that an AR(1) process with a parameter near one absorbs a great deal of the variation. See Example 8.3 later in this chapter, which fits a linear trend to a sine wave.

For ULS or ML estimation, the joint variance-covariance matrix of all the regression and autoregression parameters is computed. For the Yule-Walker method, the variance-covariance matrix is computed only for the regression parameters.

### Lagged Dependent Variables

The Yule-Walker estimation method is not directly appropriate for estimating models that include lagged dependent variables among the regressors. Therefore, the maximum likelihood method is the default when the LAGDEP or LAGDEP= option is specified in the MODEL statement. However, when lagged dependent variables are used, the maximum likelihood estimator is not exact maximum likelihood but is conditional on the first few values of the dependent variable.

### Alternative Autocorrelation Correction Methods

Autocorrelation correction in regression analysis has a long history, and various approaches have been suggested. Moreover, the same method may be referred to by different names.

Pioneering work in the field was done by Cochrane and Orcutt (1949). The Cochrane-Orcutt method refers to a more primitive version of the Yule-Walker method that drops the first observation. The Cochrane-Orcutt method is like the Yule-Walker method for first-order autoregression, except that the Yule-Walker method retains information from the first observation. The iterative Cochrane-Orcutt method is also in use.

The Yule-Walker method used by PROC AUTOREG is also known by other names. Harvey (1981) refers to the Yule-Walker method as the two-step full transform method. The Yule-Walker method can be considered as generalized least squares using the OLS residuals to estimate the covariances across observations, and Judge et al. (1985) use the term estimated generalized least squares (EGLS) for this method. For a first-order AR process, the Yule-Walker estimates are often termed Prais-Winsten estimates (Prais and Winsten; 1954). There are variations to these methods that use different estimators of the autocorrelations or the autoregressive parameters.

The unconditional least squares (ULS) method, which minimizes the error sum of squares for all observations, is referred to as the nonlinear least squares (NLS) method by Spitzer (1979).

The Hildreth-Lu method (Hildreth and Lu; 1960) uses nonlinear least squares to jointly estimate the parameters with an AR(1) model, but it omits the first transformed residual from the sum of squares. Thus, the Hildreth-Lu method is a more primitive version of the ULS method supported by PROC AUTOREG in the same way Cochrane-Orcutt is a more primitive version of Yule-Walker.

The maximum likelihood method is also widely cited in the literature. Although the maximum likelihood method is well defined, some early literature refers to estimators that are called maximum likelihood but are not full unconditional maximum likelihood estimates. The AUTOREG procedure produces full unconditional maximum likelihood estimates.

Harvey (1981) and Judge et al. (1985) summarize the literature on various estimators for the autoregressive error model. Although asymptotically efficient, the various methods have different small sample properties. Several Monte Carlo experiments have been conducted, although usually for the AR(1) model.

Harvey and McAvinchey (1978) found that for a one-variable model, when the independent variable is trending, methods similar to Cochrane-Orcutt are inefficient in estimating the structural parameter. This is not surprising since a pure trend model is well modeled by an autoregressive process with a parameter close to 1.

Harvey and McAvinchey (1978) also made the following conclusions:
The Yule-Walker method appears to be about as efficient as the maximum likelihood method. Although Spitzer (1979) recommended ML and NLS, the Yule-Walker method (labeled Prais-Winsten) did as well or better in estimating the structural parameter in Spitzer’s Monte Carlo study (table A2 in their article) when the autoregressive parameter was not too large. Maximum likelihood tends to do better when the autoregressive parameter is large.

For small samples, it is important to use a full transformation (Yule-Walker) rather than the Cochrane-Orcutt method, which loses the first observation. This was also demonstrated by Maeshiro (1976), Chipman (1979), and Park and Mitchell (1980).

For large samples (Harvey and McAvinche used 100), losing the first few observations does not make much difference.

Consider the series \( y_t \), which follows the GARCH process. The conditional distribution of the series \( Y \) for time \( t \) is written

\[
y_t | \Psi_{t-1} \sim N(0, h_t)
\]

where \( \Psi_{t-1} \) denotes all available information at time \( t - 1 \). The conditional variance \( h_t \) is

\[
h_t = \omega + \sum_{i=1}^{p} \alpha_i y_{t-i}^2 + \sum_{j=1}^{q} \gamma_j h_{t-j}
\]

where

\[
p \geq 0, q > 0 \]
\[
\omega > 0, \alpha_i \geq 0, \gamma_j \geq 0
\]

The GARCH\((p,q)\) model reduces to the ARCH\((q)\) process when \( p = 0 \). At least one of the ARCH parameters must be nonzero \((q > 0)\). The GARCH regression model can be written

\[
y_t = x'_t \beta + \epsilon_t
\]
\[
\epsilon_t = \sqrt{h_t} \epsilon_t
\]
\[
h_t = \omega + \sum_{i=1}^{p} \alpha_i \epsilon_{t-i}^2 + \sum_{j=1}^{q} \gamma_j h_{t-j}
\]

where \( \epsilon_t \sim N(0, 1) \).

In addition, you can consider the model with disturbances following an autoregressive process and with the GARCH errors. The AR\((m)\)-GARCH\((p,q)\) regression model is denoted

\[
y_t = x'_t \beta + \nu_t
\]
\[
\nu_t = \nu_t - \phi_1 \nu_{t-1} - \cdots - \phi_m \nu_{t-m}
\]
\[
\epsilon_t = \sqrt{h_t} \epsilon_t
\]
\[
h_t = \omega + \sum_{i=1}^{p} \alpha_i \epsilon_{t-i}^2 + \sum_{j=1}^{q} \gamma_j h_{t-j}
\]

**GARCH Estimation with Nelson-Cao Inequality Constraints**

The GARCH\((p,q)\) model is written in ARCH(\(\infty\)) form as
\[ h_t = \left( 1 - \sum_{j=1}^{p} \gamma_j B^j \right)^{-1} \left[ \omega + \sum_{i=1}^{q} \alpha_i e_{t-i}^2 \right] \]

\[ = \omega^* + \sum_{i=1}^{\infty} \phi_i e_{t-i}^2 \]

where \( B \) is a backshift operator. Therefore, \( h_t \geq 0 \) if \( \omega^* \geq 0 \) and \( \phi_i \geq 0 \), \( \forall i \). Assume that the roots of the following polynomial equation are inside the unit circle:

\[ \sum_{j=0}^{p} -\gamma_j Z^{p-j} \]

where \( \gamma_0 = -1 \) and \( Z \) is a complex scalar. \( -\sum_{j=0}^{p} \gamma_j Z^{p-j} \) and \( \sum_{i=1}^{q} \alpha_i Z^{q-i} \) do not share common factors. Under these conditions, \( |\omega^*| < \infty \), \( |\phi_i| < \infty \), and these coefficients of the ARCH(\( \infty \)) process are well defined.

Define \( n = \max(p, q) \). The coefficient \( \phi_i \) is written

\[ \phi_0 = \alpha_1 \]
\[ \phi_1 = \gamma_1 \phi_0 + \alpha_2 \]
\[ \vdots \]
\[ \phi_{n-1} = \gamma_1 \phi_{n-2} + \gamma_2 \phi_{n-3} + \cdots + \gamma_{n-1} \phi_0 + \alpha_n \]
\[ \phi_i = \gamma_1 \phi_{i-1} + \gamma_2 \phi_{i-2} + \cdots + \gamma_n \phi_{n-i} \text{ for } k \geq n \]

where \( \alpha_i = 0 \) for \( i > q \) and \( \gamma_j = 0 \) for \( j > p \).

Nelson and Cao (1992) proposed the finite inequality constraints for GARCH(1, q) and GARCH(2, q) cases. However, it is not straightforward to derive the finite inequality constraints for the general GARCH(p, q) model.

For the GARCH(1, q) model, the nonlinear inequality constraints are

\[ \omega \geq 0 \]
\[ \gamma_1 \geq 0 \]
\[ \phi_k \geq 0 \text{ for } k = 0, 1, \ldots, q-1 \]

For the GARCH(2, q) model, the nonlinear inequality constraints are

\[ \Delta_i \in R \text{ for } i = 1, 2 \]
\[ \omega \geq 0 \]
\[ \Delta_1 > 0 \]
\[ \sum_{j=0}^{q-1} \Delta_1^{-j} \alpha_{j+1} > 0 \]
\[ \phi_k \geq 0 \text{ for } k = 0, 1, \ldots, q \]

where \( \Delta_1 \) and \( \Delta_2 \) are the roots of \( (Z^2 - \gamma_1 Z - \gamma_2) \).

For the GARCH(p, q) model with \( p > 2 \) only \( \max(q - 1, p) + 1 \) nonlinear inequality constraints (\( \phi_k \geq 0 \) for \( k = 0 \) to \( \max(q - 1, p) \)) are imposed, together with the in-sample positivity constraints of the conditional variance \( h_t \).

**Using the HETERO Statement with GARCH Models**

The HETERO statement can be combined with the GARCH= option in the MODEL statement to include input variables in the GARCH conditional variance model. For example, the GARCH[1, 1] variance model with two dummy input variables D1 and D2 is

\[ \varepsilon_t = \sqrt{h_t} \varepsilon_t \]
\[ h_t = \omega + \alpha_t \varepsilon_{t-1}^2 + \gamma h_{t-1} + \eta_t D_t + \eta_2 D_2 \]

The following statements estimate this GARCH model:

```plaintext
proc autoreg data=one;
   model y = x z / garch=(p=1,q=1);
   hetero d1 d2;
run;
```

The parameters for the variables D1 and D2 can be constrained using the COEF= option. For example, the constraints \( \eta_1 = \eta_2 = 1 \) are imposed by the following statements:

```plaintext
proc autoreg data=one;
   model y = x z / garch=(p=1,q=1);
   hetero d1 d2 / coef=unit;
run;
```

**Limitations of GARCH and Heteroscedasticity Specifications**

When you specify both the GARCH= option and the HETERO statement, the GARCH=(TYPE=EXP) option is not valid. The COVEST= option is not applicable to the EGARCH model.

**IGARCH and Stationary GARCH Model**

The condition \( \sum_{t=1}^{\infty} \alpha_t + \sum_{j=1}^{\infty} \gamma_j < 1 \) implies that the GARCH process is weakly stationary since the mean, variance, and autocovariance are finite and constant over time. When the GARCH process is stationary, the unconditional variance of \( \varepsilon_t \) is computed as

\[
V(\varepsilon_t) = \frac{\omega}{(1 - \sum_{i=1}^{\infty} \alpha_i - \sum_{j=1}^{\infty} \gamma_j)}
\]

where \( \varepsilon_t = \sqrt{h_t} \) and \( h_t \) is the GARCH\((p,q)\) conditional variance.

Sometimes the multistep forecasts of the variance do not approach the unconditional variance when the model is integrated in variance; that is, \( \sum_{t=1}^{\infty} \alpha_t + \sum_{j=1}^{\infty} \gamma_j = 1 \).

The unconditional variance for the IGARCH model does not exist. However, it is interesting that the IGARCH model can be strongly stationary even though it is not weakly stationary. Refer to Nelson (1990) for details.

**EGARCH Model**

The EGARCH model was proposed by Nelson (1991). Nelson and Cao (1992) argue that the nonnegativity constraints in the linear GARCH model are too restrictive. The GARCH model imposes the nonnegative constraints on the parameters, \( \alpha \) and \( \gamma \), while there are no restrictions on these parameters in the EGARCH model. In the EGARCH model, the conditional variance, \( h_t \), is an asymmetric function of lagged disturbances \( \varepsilon_{t-i} \):

\[
\ln(h_t) = \omega + \sum_{k=1}^{q} \alpha_k |z_{t-k}| + \sum_{j=1}^{p} \gamma_j \ln(h_{t-j})
\]

where

\[
g(z_t) = \theta z_t + \gamma |z_t| - E[|z_t|]
\]

\[
z_t = \varepsilon_t / \sqrt{h_t}
\]

The coefficient of the second term in \( g(z_t) \) is set to be 1 (\( \gamma = 1 \)) in our formulation. Note that \( E[|z_t|] = (2/\pi)^{1/2} \) if \( z_t \sim N(0,1) \). The properties of the EGARCH model are summarized as follows:

1. The function \( g(z_t) \) is linear in \( z_t \) with slope coefficient \( \theta + 1 \) if \( z_t \) is positive while \( g(z_t) \) is linear in \( z_t \) with slope coefficient \( \theta - 1 \) if \( z_t \) is negative.

2. Suppose that \( \theta = 0 \). Large innovations increase the conditional variance if \( |z_t| - E[|z_t|] > 0 \) and decrease the conditional variance if \( |z_t| - E[|z_t|] < 0 \).

3. Suppose that \( \theta < 1 \). The innovation in variance, \( g(z_t) \), is positive if the innovations \( z_t \) are less than \( (2/\pi)^{1/2} / (\theta - 1) \). Therefore, the negative innovations in returns, \( \varepsilon_t \), cause the innovation to the conditional variance to be positive if \( \theta \) is much less than 1.
QGARCH, TGARCH, and PGARCH Models

As shown in many empirical studies, positive and negative innovations have different impacts on future volatility. There is a long list of variations of GARCH models that consider the asymmetricity. Three typical variations are the quadratic GARCH (QGARCH) model (Engle and Ng; 1993), the threshold GARCH (TGARCH) model (Glosten, Jaganathan, and Runkle; 1993; Zakoian; 1994), and the power GARCH (PGARCH) model (Ding, Granger, and Engle; 1993). For more details about the asymmetric GARCH models, see Engle and Ng (1993).

In the QGARCH model, the lagged errors’ centers are shifted from zero to some constant values:

\[ h_t = \omega + \sum_{i=1}^{q} \alpha_i (e_{t-i} - \psi_i)^2 + \sum_{j=1}^{p} \gamma_j h_{t-j} \]

In the TGARCH model, there is an extra slope coefficient for each lagged squared error,

\[ h_t = \omega + \sum_{i=1}^{q} (\alpha_i + 1_{e_{t-i} < 0}) e_{t-i}^2 + \sum_{j=1}^{p} \gamma_j h_{t-j} \]

where the indicator function \( 1_{e_{t-i} < 0} \) is one if \( e_{t-i} < 0 \); otherwise, zero.

The PGARCH model not only considers the asymmetric effect, but also provides another way to model the long memory property in the volatility,

\[ h^*_t = \omega + \sum_{i=1}^{q} \alpha_i (|e_{t-i}| - \psi_i e_{t-i})^{2\lambda} + \sum_{j=1}^{p} \gamma_j h^*_t \]

where \( \lambda > 0 \) and \( |\psi_i| \leq 1, i = 1, ..., q \).

Note that the implemented TGARCH model is also well known as GJR-GARCH (Glosten, Jaganathan, and Runkle; 1993), which is similar to the threshold GARCH model proposed by Zakoian (1994) but not exactly same. In Zakoian’s model, the conditional standard deviation is a linear function of the past values of the white noise. Zakoian’s version can be regarded as a special case of PGARCH model when \( \lambda = 1/2 \).

GARCH-in-Mean

The GARCH-M model has the added regressor that is the conditional standard deviation:

\[ y_t = \chi' \beta + \delta \sqrt{h_t} + e_t \]
\[ e_t = \sqrt{h_t} \epsilon_t \]

where \( h_t \) follows the ARCH or GARCH process.

Maximum Likelihood Estimation

The family of GARCH models are estimated using the maximum likelihood method. The log-likelihood function is computed from the product of all conditional densities of the prediction errors.

When \( \phi_t \) is assumed to have a standard normal distribution \( (\phi_t \sim N(0, 1) ) \), the log-likelihood function is given by

\[ l = \sum_{t=1}^{N} \left[ \ln(2\pi) - \ln(h_t) - \frac{\epsilon_t^2}{h_t} \right] \]

where \( \epsilon_t = y_t - \chi' \beta \) and \( h_t \) is the conditional variance. When the GARCH\((p, q)\)-M model is estimated, \( \epsilon_t = y_t - \chi' \beta - \delta \sqrt{h_t} \). When there are no regressors, the residuals \( \epsilon_t \) are denoted as \( y_t \) or \( y_t - \delta \sqrt{h_t} \).

If \( \epsilon_t \) has the standardized Student’s t distribution, the log-likelihood function for the conditional t distribution is

\[ \ell = \sum_{t=1}^{N} \left[ \ln \left( \Gamma \left( \frac{\nu + 1}{2} \right) \right) - \ln \left( \Gamma \left( \frac{\nu}{2} \right) \right) - \frac{1}{2} \ln((\nu - 2)h_t) \right. \\
- \frac{1}{2} (\nu + 1) \ln \left( 1 + \frac{\epsilon_t^2}{h_t(\nu - 2)} \right) \]
where $\Gamma(\cdot)$ is the gamma function and $\nu$ is the degree of freedom ($\nu > 2$). Under the conditional $t$ distribution, the additional parameter $1/\nu$ is estimated. The log-likelihood function for the conditional $t$ distribution converges to the log-likelihood function of the conditional normal GARCH model as $1/\nu \to 0$.

The likelihood function is maximized via either the dual quasi-Newton or the trust region algorithm. The default is the dual quasi-Newton algorithm. The starting values for the regression parameters $\beta$ are obtained from the OLS estimates. When there are autoregressive parameters in the model, the initial values are obtained from the Yule-Walker estimates. The starting value $1.0^{-9}$ is used for the GARCH process parameters.

The variance-covariance matrix is computed using the Hessian matrix. The dual quasi-Newton method approximates the Hessian matrix while the quasi-Newton method gets an approximation of the inverse of Hessian. The trust region method uses the Hessian matrix obtained using numerical differentiation. When there are active constraints, that is, $q(\theta) = 0$, the variance-covariance matrix is given by

$$V(\hat{\theta}) = H^{-1}[I - Q'(OH^{-1}Q')^{-1}QH^{-1}]$$

where $H = -\partial^2/\partial \theta \partial \theta'$ and $Q = \partial q(\theta)/\partial \theta'$. Therefore, the variance-covariance matrix without active constraints reduces to $V(\hat{\theta}) = H^{-1}$.

The AUTOREG Procedure

Goodness-of-fit Measures and Information Criteria

This section discusses various goodness-of-fit statistics produced by the AUTOREG procedure.

**Total R-Square Statistic**

The total R-Square statistic (Total Rsq) is computed as

$$R_{tot}^2 = 1 - \frac{SSE}{SST}$$

where SST is the sum of squares for the original response variable corrected for the mean and SSE is the final error sum of squares. The Total Rsq is a measure of how well the next value can be predicted using the structural part of the model and the past values of the residuals. If the NOINT option is specified, SST is the uncorrected sum of squares.

**Regression R-Square Statistic**

The regression R-Square statistic (Reg RSQ) is computed as

$$R_{reg}^2 = 1 - \frac{TSSE}{TSST}$$

where TSST is the total sum of squares of the transformed response variable corrected for the transformed intercept, and TSSE is the error sum of squares for this transformed regression problem. If the NOINT option is requested, no correction for the transformed intercept is made. The Reg RSQ is a measure of the fit of the structural part of the model after transforming for the autocorrelation and is the R-Square for the transformed regression.

The regression R-Square and the total R-Square should be the same when there is no autocorrelation correction (OLS regression).

**Mean Absolute Error and Mean Absolute Percentage Error**

The mean absolute error (MAE) is computed as

$$MAE = \frac{1}{T} \sum_{t=1}^{T} |\epsilon_t|$$

where $\epsilon_t$ are the estimated model residuals and $T$ is the number of observations.

The mean absolute percentage error (MAPE) is computed as
where $e_i$ are the estimated model residuals, $y_i$ are the original response variable observations, $\delta_{i,y_0} = 1$ if $y_t \neq 0$, $\delta_{i,y_0} |e_i|/y_i = 0$ if $y_t = 0$, and $T'$ is the number of nonzero original response variable observations.

**Calculation of Recursive Residuals and CUSUM Statistics**

The recursive residuals $w_t$ are computed as

$$w_t = \frac{e_t}{\sqrt{\sigma_w}}$$

$$e_t = y_t - \hat{x}_t^T \hat{\beta}^{(i)}$$

$$\hat{\beta}^{(i)} = \left[ \sum_{i=1}^{t-1} x_i x_i^T \right]^{-1} \left( \sum_{i=1}^{t-1} x_i y_i \right)$$

$$\sigma_w = \sqrt{\frac{\Sigma_{t=k+1}^{T-1} (w_j - \bar{w})^2}{(T-k-1)}}$$

$$\bar{w} = \frac{1}{T-k} \sum_{t=k+1}^{T} w_j$$

and $k$ is the number of regressors.

The CUSUM statistics can be used to test for misspecification of the model. The upper and lower critical values for CUSUM are

$$\pm a \sqrt{T-k+2} \left( \frac{(T-k)^{1/2}}{(T-k-1)^{1/2}} \right)$$

where $a = 1.143$ for a significance level 0.01, 0.948 for 0.05, and 0.850 for 0.10. These critical values are output by the CUSUMLB= and CUSUMUB= options for the significance level specified by the ALPHACSM= option.

The upper and lower critical values of CUSUMSQ are given by

$$\pm a + \frac{(t-k)}{T-k}$$

where the value of $a$ is obtained from the table by Durbin (1969) if the $t(T-k)-1 \leq 66$. Edgerton and Wells (1994) provided the method of obtaining the value of $a$ for large samples.

These critical values are output by the CUSUMSQLB= and CUSUMSQUB= options for the significance level specified by the ALPHACSM= option.
Information Criteria AIC, AICC, SBC, and HQC

Akaike's information criterion (AIC), the corrected Akaike's information criterion (AICC), Schwarz's Bayesian information criterion (SBC), and the Hannan-Quinn information criterion (HQC), are computed as follows:

\[
\begin{align*}
\text{AIC} &= -2\ln(L) + 2k \\
\text{AICC} &= \text{AIC} + \frac{k(k + 1)}{N - k - 1} \\
\text{SBC} &= -2\ln(L) + \ln(N)k \\
\text{HQC} &= -2\ln(L) + 2\ln(\ln(N))k
\end{align*}
\]

In these formulas, \(L\) is the value of the likelihood function evaluated at the parameter estimates, \(N\) is the number of observations, and \(k\) is the number of estimated parameters. Refer to Judge et al. (1985), Hurvich and Tsai (1989), Schwarz (1978) and Hannan and Quinn (1979) for additional details.

The modeling process consists of four stages: identification, specification, estimation, and diagnostic checking (Cromwell, Labys, and Terraza; 1994). The AUTOREG procedure supports tens of statistical tests for identification and diagnostic checking. Figure 8.15 illustrates how to incorporate these statistical tests into the modeling process. Figure 8.15 Statistical Tests in the AUTOREG procedure
Testing for Stationarity

Most of the theories of time series require stationarity; therefore, it is critical to determine whether a time series is stationary. Two nonstationary time series are fractionally integrated time series and autoregressive series with random coefficients. However, more often some time series are nonstationary due to an upward trend over time. The trend can be captured by either of the following two models.

The difference stationary process

\[(1-L)\gamma_t = \delta + \psi(L)e_t\]

where \(L\) is the lag operator, \(\psi(1) \neq 0\), and \(e_t\) is a white noise sequence with mean zero and variance \(\sigma^2\). Hamilton (1994) also refers to this model the unit root process.

The trend stationary process

\[y_t = \alpha + \delta t + \psi(L)e_t\]

When a process has a unit root, it is said to be integrated of order one or I(1). An I(1) process is stationary after differencing once. The trend stationary process and difference stationary process require different treatment to transform the process into stationary one for analysis. Therefore, it is important to distinguish the two processes. Bhargava (1986) nested the two processes into the following general model

\[y_t = \gamma_0 + \gamma_1 t + \alpha_0 (y_{t-1} - \gamma_0 - \gamma_1 (t-1)) + \psi(L)e_t\]

However, a difficulty is that the right-hand side is nonlinear in the parameters. Therefore, it is convenient to use a different parametrization

\[y_t = \beta_0 + \beta_1 t + \alpha y_{t-1} + \psi(L)e_t\]

The test of null hypothesis that \(\alpha = 1\) against the one-sided alternative of \(\alpha < 1\) is called a unit root test. Dickey-Fuller unit root tests are based on regression models similar to the previous model

\[y_t = \beta_0 + \beta_1 t + \alpha y_{t-1} + \epsilon_t\]

where \(\epsilon_t\) is assumed to be white noise. The \(t\) statistic of the coefficient \(\alpha\) does not follow the normal distribution asymptotically. Instead, its distribution can be derived using the functional central limit theorem. Three types of regression models including the preceding one are considered by the Dickey-Fuller test. The deterministic terms that are included in the other two types of regressions are either null or constant only.

An assumption in the Dickey-Fuller unit root test is that it requires the errors in the autoregressive model to be white noise, which is often not true. There are two popular ways to account for general serial correlation between the errors. One is the augmented Dickey-Fuller (ADF) test, which uses the lagged difference in the regression model. This was originally proposed by Dickey and Fuller (1979) and later studied by Said and Dickey (1984) and Phillips and Perron (1988). Another method is proposed by Phillips and Perron (1988); it is called Phillips-Perron (PP) test. The tests adopt the original Dickey-Fuller regression with intercept, but modify the test statistics to take account of the serial correlation and heteroscedasticity. It is called nonparametric because no specific form of the serial correlation of the errors is assumed.

A problem of the augmented Dickey-Fuller and Phillips-Perron unit root tests is that they are subject to size distortion and low power. It is reported in Schwert (1989) that the size distortion is significant when the series contains a large moving average (MA) parameter. DeJong et al. (1992) find that the ADF has power around one third and PP test has power less than 0.1 against the trend stationary alternative, in some common settings. Among some more recent unit root tests that improve upon the size distortion and the low power are the tests described by Elliott, Rothenberg, and Stock (1996) and Ng and Perron (2001). These tests involve a step of detrending before constructing the test statistics and are demonstrated to perform better than the traditional ADF and PP tests.

Most testing procedures specify the unit processes as the null hypothesis. Tests of the null hypothesis of stationarity have also been studied, among which Kwiatkowski et al. (1992) is very popular.

Economic theories often dictate that a group of economic time series are linked together by some long-run equilibrium relationship. Statistically, this phenomenon can be modeled by cointegration. When several nonstationary processes \(Z_t = (z_{1t}, \ldots, z_{kt})'\) are cointegrated, there exists a \((k \times 1)\) cointegrating vector \(c\) such that \(c'Z_t\) is stationary and \(c\) is a nonzero vector. One way to test the relationship of cointegration is the residual based cointegration test, which assumes the regression model

\[y_t = \beta_0 + \beta'Z_t + u_t\]

where \(y_t = z_{1t}, \ldots, z_{kt}\), and \(\beta = (\beta_0, \ldots, \beta_k)'\). The OLS residuals from the regression model are used to test for the null hypothesis of no

**Augmented Dickey-Fuller Unit Root and Engle-Granger Cointegration Testing**

Common unit root tests have the null hypothesis that there is an autoregressive unit root $H_0: \alpha = 1$, and the alternative is $H_a: |\alpha| < 1$ where $\alpha$ is the autoregressive coefficient of the time series

$$y_t = \alpha y_{t-1} + \epsilon_t$$

This is referred to as the zero mean model. The standard Dickey-Fuller (DF) test assumes that errors $\epsilon_t$ are white noise. There are two other types of regression models that include a constant or a time trend as follows:

$$y_t = \mu + \alpha y_{t-1} + \epsilon_t$$

$$y_t = \mu + \beta t + \alpha y_{t-1} + \epsilon_t$$

These two models are referred to as the constant mean model and the trend model, respectively. The constant mean model includes a constant mean $\mu$ of the time series. However, the interpretation of $\mu$ depends on the stationarity in the following sense: the mean in the stationary case when $\alpha < 1$ is the trend in the integrated case when $\alpha = 1$. Therefore, the null hypothesis should be the joint hypothesis that $\alpha = 1$ and $\mu = 0$. However for the unit root tests, the test statistics are concerned with the null hypothesis of $\alpha = 1$. The joint null hypothesis is not commonly used. This issue is address in Bhargava (1986) with a different nesting model.

There are two types of test statistics. The conventional ratio is

$$DF_t = \frac{\hat{\alpha} - 1}{\hat{\sigma}(\hat{\alpha})}$$

and the second test statistic, called $\rho$-test, is

$$T(\hat{\alpha} - 1)$$

For the zero mean model, the asymptotic distributions of the Dickey-Fuller test statistics are

$$T(\alpha - 1) \Rightarrow \left(\int_0^1 W(r)dw(r)\right)^{-1}$$

$$DF_t \Rightarrow \left(\int_0^1 W(r)dW(r)\right)^{-1/2}$$

For the constant mean model, the asymptotic distributions are

$$T(\alpha - 1) \Rightarrow \left(W(1)^2 - 1/2 - \int_0^1 W(r)dr\right) \left(\int_0^1 W(r)^2 dr - \left(\int_0^1 W(r)dr\right)^2\right)^{-1}$$

$$DF_t \Rightarrow \left(W(1)^2 - 1/2 - W(1)\int_0^1 W(r)dr\right) \left(\int_0^1 W(r)^2 dr - \left(\int_0^1 W(r)dr\right)^2\right)^{-1/2}$$

For the trend model, the asymptotic distributions are

$$T(\alpha - 1) \Rightarrow \left[W(r)dw + 12\left(\int_0^1 rW(r)dr - \frac{1}{2}\int_0^1 W(r)dr\right)\left(\int_0^1 W(r)dr - \frac{1}{2}W(1)\right) - W(1)\int_0^1 W(r)dr\right]D^{-1}$$

$$DF_t \Rightarrow \left[W(r)dw + 12\left(\int_0^1 rW(r)dr - \frac{1}{2}\int_0^1 W(r)dr\right)\left(\int_0^1 W(r)dr - \frac{1}{2}W(1)\right) - W(1)\int_0^1 W(r)dr\right]D^{1/2}$$

where
One problem of the Dickey-Fuller and similar tests that employ three types of regressions is the difficulty in the specification of the deterministic trends. Campbell and Perron (1991) claimed that "the proper handling of deterministic trends is a vital prerequisite for dealing with unit roots". However the "proper handling" is not obvious since the distribution theory of the relevant statistics about the deterministic trends is not available. Hayashi (2000) suggests to using the constant mean model when you think there is no trend, and using the trend model when you think otherwise. However no formal procedure is provided.

The null hypothesis of the Dickey-Fuller test is a random walk, possibly with drift. The differenced process is not serially correlated under the null of I(1). There is a great need for the generalization of this specification. The augmented Dickey-Fuller (ADF) test, originally proposed in Dickey and Fuller (1979), adjusts for the serial correlation in the time series by adding lagged first differences to the autoregressive model,

\[ \Delta y_t = \mu + \delta_t + \alpha y_{t-1} + \sum_{j=1}^{p} \alpha_j \Delta y_{t-j} + \epsilon_t \]

where the deterministic terms \( \delta_t \) and \( \mu \) can be absent for the models without drift or linear trend. As previously, there are two types of test statistics. One is the OLS \( t \) value

\[ \frac{\hat{\alpha} - 1}{sd(\hat{\alpha})} \]

and the other is given by

\[ \frac{T(\hat{\alpha} - 1)}{1 - \hat{\alpha}_1 - \ldots - \hat{\alpha}_p} \]

The asymptotic distributions of the test statistics are the same as those of the standard Dickey-Fuller test statistics.

Nonstationary multivariate time series can be tested for cointegration, which means that a linear combination of these time series is stationary. Formally, denote the series by \( x_t = (z_{t1}, \ldots, z_{tp})' \). The null hypothesis of cointegration is that there exists a vector \( \epsilon \) such that \( \epsilon' x_t \) is stationary. Residual-based cointegration tests were studied in Engle and Granger (1987) and Phillips and Ouliaris (1990). The latter are described in the next subsection. The first step regression is

\[ y_t = x_t' \beta + u_t \]

where \( y_t = (z_{t1}, \ldots, z_{tp})' \), and \( \beta = (\beta_2, \ldots, \beta_k)' \). This regression can also include an intercept or an intercept with a linear trend. The residuals are used to test for the existence of an autoregressive unit root. Engle and Granger (1987) proposed augmented Dickey-Fuller type regression without an intercept on the residuals to test the unit root. When the first step OLS does not include an intercept, the asymptotic distribution of the ADF test statistic \( D^F_t \) is given by

\[ D^F_t = \int_0^1 \frac{Q(r)(r)}{(r \hat{Q}^2)^{1/2}} dS \]

where \( W(r) \) is a \( k \) vector standard Brownian motion and

\[ W(r) = (W_1(r), W_2(r)) \]

is a partition such that \( W_1(r) \) is a scalar and \( W_2(r) \) is \( k - 1 \) dimensional. The asymptotic distributions of the test statistics in the other two cases have the same form as the preceding formula. If the first step regression includes an intercept, then \( W(r) \) is replaced by the demeaned Brownian motion \( \bar{W}(r) = W(r) - \int_0^r W(r) dW \). If the first step regression includes a time trend, then \( W(r) \) is replaced by the detrended Brownian motion.

The critical values of the asymptotic distributions are tabulated in Phillips and Ouliaris (1990) and MacKinnon (1991).
The residual based cointegration tests have a major shortcoming. Different choices of the dependent variable in the first step OLS might produce contradictory results. This can be explained theoretically. If the dependent variable is in the cointegration relationship, then the test is consistent against the alternative that there is cointegration. On the other hand, if the dependent variable is not in the cointegration system, the OLS residual $y_t - \mathbf{x}' \mathbf{\beta}$ do not converge to a stationary process. Changing the dependent variable is more likely to produce conflicting results in finite samples.

**Phillips-Perron Unit Root and Cointegration Testing**

Besides the ADF test, there is another popular unit root test that is valid under general serial correlation and heteroscedasticity, developed by Phillips (1997) and Phillips and Perron (1988). The tests are constructed using the AR(1) type regressions, unlike ADF tests, with corrected estimation of the long run variance of $\Delta y_t$. In the case without intercept, consider the driftless random walk process

$$y_t = y_{t-1} + u_t$$

where the disturbances might be serially correlated with possible heteroscedasticity. Phillips and Perron (1988) proposed the unit root test of the OLS regression model,

$$y_t = \rho y_{t-1} + u_t$$

Denote the OLS residual by $\hat{u}_t$. The asymptotic variance of $\sum_{i=1}^{T} \hat{a}_i^2$ can be estimated by using the truncation lag $l$.

$$\lambda = \sum_{j=0}^{l} \kappa_j [1 - j/(j+1)] \hat{\gamma}_j$$

where $\kappa_0 = 1$, $\kappa_j = 2$ for $j > 0$, and $\hat{\gamma}_j = \sum_{i=j+1}^{T} \hat{u}_i \hat{u}_{i-j}$. This is a consistent estimator suggested by Newey and West (1987).

The variance of $\hat{u}_t$ can be estimated by $\hat{\sigma}^2 = \frac{1}{T} \sum_{i=1}^{T} \hat{a}_i^2$. Let $\hat{\sigma}$ be the variance estimate of the OLS estimator $\hat{\rho}$. Then the Phillips-Perron $\hat{Z}_p$ test (zero mean case) is written

$$\hat{Z}_p - T(\rho - 1) - \frac{1}{2} \frac{T^2}{\hat{\sigma}^2} (\hat{\lambda} - \hat{\gamma}_0)/\hat{\sigma}^2$$

The $\hat{Z}_p$ statistic is just the ordinary Dickey-Fuller $\hat{Z}_a$ statistic with a correction term that accounts for the serial correlation. The correction term goes to zero asymptotically if there is no serial correlation.

Note that $\Phi(\hat{\rho} < 1) \approx 0.08$ as $T \to \infty$, which shows that the limiting distribution is skewed to the left.

Let $\tau_\rho$ be the $\tau$ statistic for $\hat{\rho}$. The Phillips-Perron $\hat{Z}_a$ (defined here as $\hat{Z}_\tau$) test is written

$$\hat{Z}_\tau = (\hat{\gamma}_0/\hat{\sigma}^{1/2}) (\hat{\lambda} - \hat{\gamma}_0)/(\hat{\sigma}^{1/2})$$

To incorporate a constant intercept, the regression model $y_t = \mu + \rho y_{t-1} + u_t$ is used (single mean case) and null hypothesis the series is a driftless random walk with nonzero unconditional mean. To incorporate a time trend, we used the regression model $y_t = \mu + \delta t + \rho y_{t-1} + u_t$ and under the null the series is a random walk with drift.

The limiting distributions of the test statistics for the zero mean case are

$$\hat{Z}_p = \frac{1}{2} \frac{[B(1)^2 - 1]}{\int_0^1 [B(s)]^2 ds}$$

$$\hat{Z}_\tau = \frac{1}{2} \frac{([B(1)]^2 - 1)}{\left\{ \int_0^1 [B(s)]^2 ds \right\}^{1/2}}$$

where $B(s)$ is a standard Brownian motion.

The limiting distributions of the test statistics for the intercept case are

$$\hat{Z}_p = \frac{1}{2} \frac{[B(1)^2 - 1] - B(1) \int_0^1 B(s) ds}{\int_0^1 [B(s)]^2 ds - \left\{ \int_0^1 B(s) ds \right\}^2}$$
Finally, the limiting distributions of the test statistics for the trend case are can be derived as

$$\hat{\mathcal{Z}}_T = \frac{\frac{1}{2} \{ [B(1)^2 - 1] - B(1) \int B(x)dx \}}{\left\{ \int B(x)^2dx - \left[ \int B(x)dx \right]^2 \right\}^{1/2}}$$

The finite sample performance of the PP test is not satisfactory (see Hayashi (2000)).

When several variables $z_t = (z_{1t}, \ldots, z_{kt})'$ are cointegrated, there exists a $(k \times 1)$ cointegrating vector $c$ such that $c' z_t$ is stationary and $c$ is a nonzero vector. The residual based cointegration test assumes the following regression model:

$$y_t = \bar{\beta}_t + \mathbf{x}_t c + u_t$$

where $y_t = z_{1t}, \mathbf{x}_t = (z_{2t}, \ldots, z_{kt})'$, and $\bar{\beta} = \left( \bar{\beta}_2, \ldots, \bar{\beta}_k \right)'$. You can estimate the consistent cointegrating vector by using OLS if all variables are difference stationary — that is, $I(1)$. The estimated cointegrating vector is $\hat{c} = (1, -\hat{\beta}_2, \ldots, -\hat{\beta}_k)'$. The Phillips-Ouliaris test is computed using the OLS residuals from the preceding regression model, and it uses the PP unit root tests $\hat{Z}_p$ and $\hat{Z}_c$ developed in Phillips (1997), although in Phillips and Ouliaris (1990) the asymptotic distributions of some other leading unit root tests are also derived. The null hypothesis is no cointegration.

You need to refer to the tables by Phillips and Ouliaris (1990) to obtain the $p$-value of the cointegration test. Before you apply the cointegration test, you may want to perform the unit root test for each variable (see the option STATIONARITY=(ADF)).

As in the Engle-Granger cointegration tests, the Phillips-Ouliaris test can give conflicting results for different choices of the regressand. There are other cointegration tests that are invariant to the order of the variables, including Johansen (1988), Johansen (1991), Stock and Watson (1988).

ERS and Ng-Perron Unit Root Tests

(Experimental)

As mentioned earlier, ADF and PP both suffer severe size distortion and low power. There is a class of newer tests that improves both size and power, sometimes called efficient unit root tests, among which Elliott, Rothenberg, and Stock (1996) and Ng and Perron (2001) are prominent.

Elliott, Rothenberg, and Stock (1996) consider the data generating process

$$y_t = \beta' z + u_t$$
$$u_t = \alpha u_{t-1} + v_t, t = 1, \ldots, T$$

where $\{z_t\}$ is either $\{t\}$ or $\{(1, t)\}$ and $\{v_t\}$ is an unobserved stationary zero-mean process with positive spectral density at zero frequency. The null hypothesis is $H_0: \alpha = 1$ and the alternative is $H_a: |\alpha| < 1$. The key idea of Elliott, Rothenberg, and Stock (1996) is to study the asymptotic power and asymptotic power envelope of some new tests. Asymptotic power is defined with a sequence of local alternatives. For a fixed alternative hypothesis, the power of a test usually goes to one when sample size goes to infinity; however, this does not say anything about the finite sample performance. On the other hand, when the data generating process under the alternative moves closer to the null as the sample size increases, the power does not necessarily converge to one. The local to unity alternatives in ERS are

$$\alpha = 1 + \frac{c}{T}$$

and the power against the local alternatives has a limit as $T$ goes to infinity, which is called asymptotic power. This value is strictly between 0 and 1. Asymptotic power indicates the adequacy of a test to distinguish small deviations from the null hypothesis.
Define

\[ y_t = (y_t, (1 - aL)y_2, \ldots, (1 - aL)y_T) \]

\[ z_t = (z_t, (1 - aL)z_2, \ldots, (1 - aL)z_T) \]

Let \( S(\alpha) \) be the sum of squared residuals from a least squares regression of \( y_t \) on \( z_t \). Then the point optimal test against the local alternative \( \alpha = 1 + \tilde{e}T \) has the form

\[ P^*_{GLS} = \frac{S(\hat{\alpha}) - \hat{\alpha}S(1)}{\hat{\sigma}^2} \]

where \( \hat{\sigma}^2 \) is an estimator for \( \sigma^2 = \sum_{k=-\infty}^{\infty} I_k \psi_k \). Note that the test rejects the null when \( P^* \) is small. The asymptotic power function for the point optimal test constructed with \( T^* \) under local alternatives with \( \alpha \) is denoted by \( \pi(c, \tilde{c}) \). Then the power envelope is \( \pi(c, \tilde{c}) \) because the test formed with \( T^* \) is the most powerful against the alternative \( \alpha = \tilde{c} \). In other words, the asymptotic function \( \pi(c, \tilde{c}) \) is always below the power envelope \( \pi(c, \tilde{c}) \) except that at one point \( \tilde{c} = \tilde{\alpha} \) they are tangent. Elliott, Rothenberg, and Stock (1996) show that choosing some specific values for \( \tilde{c} \) can cause the asymptotic power function \( \pi(c, \tilde{c}) \) of the point optimal test to be very close to the power envelope. The optimal \( \tilde{c} \) is \(-7 \) when \( \tilde{c} = 1 \) and \(-13.5 \) when \( \tilde{c} = (1, T) \). This choice of \( \tilde{c} \) corresponds to the tangent point where \( \pi = 0.5 \). This is also true for the DF-GLS test.

Elliott, Rothenberg, and Stock (1996) also propose the DF-GLS test, given by the \( t \) statistic for testing \( \psi_0 = 0 \) in the regression

\[ \Delta y_t^{(d)} = \psi_0 y_{t-1}^{(d)} + \sum_{j=1}^{p} \psi_j \Delta y_{t-j}^{(d)} + \delta_p \]

where \( y_t^{(d)} \) is obtained in a first step detrending

\[ y_t^{(d)} = y_t - \hat{\beta}_d z_t \]

and \( \hat{\beta}_d \) is least squares regression coefficient of \( y_t \) on \( z_t \). Regarding the lag length selection, Elliott, Rothenberg, and Stock (1996) favor the Schwartz Bayesian information criterion. The optimal selection of the lag length \( p \) and the estimation of \( \sigma_p^2 \) is further discussed in Ng and Perron (2001). The lag length is selected from the interval \( [0, p_{\text{max}}] \) for some fixed \( p_{\text{max}} \) by using the modified Akaikes information criterion,

\[ \text{MAIC}(p) = \log(\hat{\sigma}_p^2) + \frac{2 \left( T(\hat{\tau}_p(p) + p) \right)}{T - p_{\text{max}}} \]

where \( \hat{\tau}_p(p) = (\hat{\sigma}_p^2)^{-1} \psi_0^2 \sum_{t=p+1}^{T} (y_{t-1}^{(d)})^2 \) and \( \hat{\sigma}_p^2 = (T - p_{\text{max}})^{-1} \sum_{t=p_{\text{max}}+1}^{T} \hat{\psi}_p^2 \). For fixed lag length \( p \), an estimate of \( \sigma_p^2 \) is given by

\[ \hat{\sigma}_p^2 = \frac{(T - p)^{-1} \sum_{t=p+1}^{T} \hat{\psi}_p^2}{1 - \sum_{t=p+1}^{T} \hat{\psi}_p^2} \]

DF-GLS is indeed a superior unit root test, according to Stock (1994), Schwert (1989), and Elliott, Rothenberg, and Stock (1996). In terms of the size of the test, DF-GLS is almost as good as the ADF \( t \) test DF, and better than the PP \( \tilde{Z}_\alpha \) and \( \tilde{Z}_x \) test. In addition, the power of the DF-GLS is larger than the ADF \( t \) test and \( \rho \) test.

Ng and Perron (2001) also apply GLS detrending to obtain the following M-tests:

\[ M_{Z_{\alpha}} = (T^{-1}(y_T^{(d)} - \hat{\lambda})^2) \left( 2T^{-2} \sum_{t=1}^{T} (y_{t-1}^{(d)})^2 \right)^{-1} \]

\[ M_{SB} = \left( \sum_{t=1}^{T} (y_{t-1}^{(d)})^2 / T \sigma_p^2 \right)^{1/2} \]

\[ M_{Z_{\rho}} = M_{Z_{\alpha}} \times M_{SB} \]

The first one is a modified version of Phillips-Perron \( Z_{\rho} \) test

\[ M_{Z_{\rho}} = Z_{\rho} + \frac{T}{2}(\alpha - 1)^2 \]

where the detrended data \( \{y_t^{(d)}\} \) is used. The second is a modified Bhargava (1986) \( R_1 \) test statistic. The third can be perceived as a modified
Phillips-Perron $Z_t$ statistic because of the relationship $Z_t = MSB \times Z_p$

The modified point optimal tests using the GLS detrended data are

$$M_{GLS}^T = \frac{-T^{-2} \sum_{t=1}^T (\hat{\phi}_t) \sum_{t=1}^T \hat{\phi}_t}{\sum_{t=1}^T \hat{\phi}_t^2}$$  for $\hat{\phi}_t = 1$

$$M_{GLS}^T = \frac{-T^{-2} \sum_{t=1}^T (\hat{\phi}_t) \sum_{t=1}^T (1-\phi_t)^{-1} \hat{\phi}_t^2}{\sum_{t=1}^T \hat{\phi}_t^2}$$  for $\hat{\phi}_t = (1, t)$

The DF-GLS test and the $MZ$ test have the same limiting distribution

$$DF-GLS \approx MZ \Rightarrow 0.5 \frac{(L_t(1)^{-1} \sum_{t=1}^T L_t(1)^2)}{\sum_{t=1}^T L_t(1)^2}$$  for $\hat{\phi}_t = 1$

$$DF-GLS \approx MZ \Rightarrow 0.5 \frac{(L_t(1)^{-1} \sum_{t=1}^T V_t(1)^2)}{\sum_{t=1}^T V_t(1)^2}$$  for $\hat{\phi}_t = (1, t)$

The point optimal test and the modified point optimal test have the same limiting distribution

$$P_{GLS}^T \approx M_{GLS}^T \Rightarrow c^2 \sum_{t=1}^T J_t(1)^2 \sum_{t=1}^T J_t(1)^2$$  for $\hat{\phi}_t = 1$

$$P_{GLS}^T \approx M_{GLS}^T \Rightarrow c^2 \sum_{t=1}^T V_t(1)^2 \sum_{t=1}^T V_t(1)^2$$  for $\hat{\phi}_t = (1, t)$

where $W_t$ is a standard Brownian motion and $J_t$ is an Ornstein-Uhlenbeck process defined by $dJ_t = cJ_t dr + dW_t$ with $J_t(0) = 0$.

Overall, the M-tests has the smallest size distortion, with the ADF t test having the next smallest. The ADF $t$-test, $t$-test, and $t$-test have the worst size distortion. In addition, the power of the DF-GLS and M-tests are larger than that of the ADF $t$ test and $t$ test. The ADF $t$ test has more severe size distortion than the ADF $t$ test, but larger power for a fixed lag length.

**Kwiatkowski, Phillips, Schmidt, and Shin (KPSS) Unit Root Test**

There are less existent tests for the null hypothesis of trend stationary I(0). The main reason is the difficulty in the theoretical development. The KPSS test was introduced in Kwiatkowski et al. (1992) to test the null hypothesis that an observable series is stationary around a deterministic trend. Please note, that for consistency reasons, the notation used here is different from the notation used in the original paper. The setup of the problem is as follows: it is assumed that the series is expressed as the sum of the deterministic trend, random walk $r_t$, and stationary error $u_t$; that is,

$$y_t = \mu + \delta t + r_t + u_t$$

$$r_t = r_{t-1} + \epsilon_t$$

where $\epsilon_t \sim iid (0, \sigma^2_\epsilon)$ and an intercept $\mu$ (in the original paper, the authors use $\mu_0$ instead of $\mu$; here we assume $\mu_0 = 0$). The null hypothesis of trend stationary is specified by $H_0 : \sigma^2_\epsilon = 0$, while the null of level stationary is the same as above with the model restriction $\delta = 0$. Under the alternative that $\sigma^2_\epsilon \neq 0$, there is a random walk component in the observed series $y_t$.

Under stronger assumptions of normality and iid of $u_t$ and $\epsilon_t$, a one-sided LM test of the null that there is no random walk ($\epsilon_t = 0, \forall t$) can be constructed as follows:

$$\hat{L}_M = \frac{1}{T^2} \sum_{t=1}^T \frac{S_t^2}{S_t^2}$$

$$\hat{L}_M (I) = \frac{1}{T^2} \sum_{t=1}^T \hat{a}_t^2 + \frac{1}{T} \sum_{t=1}^T w(s, l) \sum_{t=s+1}^T \hat{a}_t \hat{a}_{t-s}$$

$$S_t = \sum_{t=1}^T \hat{a}_t$$

Notice that under the null hypothesis, $\hat{a}_t$ can be estimated by ordinary least squares regression of $y_t$ on an intercept and the time trend. Following the original work of Kwiatkowski, Phillips, Schmidt, and Shin, under the null ($\sigma^2_\epsilon = 0$), $L_M$ statistic converges asymptotically to three different distributions depending on whether the model is trend-stationary, level-stationary ($\delta = 0$), or zero-mean stationary ($\delta = 0$, $\mu = 0$). The trend-stationary model is denoted by subscript $\tau$ and the level-stationary model is denoted by subscript $\mu$. The case when there is no trend and zero intercept is denoted as $0$. The last case, although rarely used in practice, is considered in Hobijn, Franses, and Ooms (2004).
\[ y_t = \mu + u_t : \lim_{n \to \infty} \frac{1}{n} \sum_{i=1}^{n} V^2(r)dr \]
\[ y_t = \mu + \delta t + u_t : \lim_{n \to \infty} \frac{1}{n} \sum_{i=1}^{n} V^2_2(r)dr \]

with
\[ V(r) = B(r) - rB(1) \]
\[ V_2(r) = B(r) + (2r - 3r^2)B(1) + \left[-6r + 6r^2 \right] \int_0^1 B(s)ds \]

where \( B(t) \) is a Brownian motion (Wiener process), and \( \lim_{n \to \infty} \frac{1}{n} \sum_{i=1}^{n} V^2_2(r)dr \) is convergence in distribution. Note that \( V(r) \) is a standard Brownian bridge, \( V_2(r) \) is a Brownian bridge of a second-level.

Using the notation of Kwiatkowski et al. (1992) the \( \hat{M} \) statistic is named as \( \tilde{Z} \). This test depends on the computational method used to compute the long-run variance \( s(I) \) — that is, the window width \( I \) and the kernel type \( w(\cdot, \cdot) \). You can specify the kernel used in the test, using the KERNEL option:

- Newey-West/Bartlett (KERNEL=NW BART), default
- Quadratic spectral (KERNEL=QS)

You can specify the number of lags, \( l \), in three different ways:

- Schwert (SCHW=c) (default for NW, c=4)
- Manual (LAG =)

The last option (AUTO) needs more explanation, summarized in the following table. For each of the kernel function, a formula for optimal window width \( I \) is provided.

<table>
<thead>
<tr>
<th>NW Kernel</th>
<th>QS Kernel</th>
</tr>
</thead>
<tbody>
<tr>
<td>( l = \min(T, \text{floor}(T^{1/3})) )</td>
<td>( l = \min(T, \text{floor}(T^{1/3})) )</td>
</tr>
<tr>
<td>where ( T ) is the number of observations,</td>
<td>where ( T ) is the number of observations,</td>
</tr>
<tr>
<td>( \delta = 1.1447 \left( \frac{T}{100} \right)^{1/3} )</td>
<td>( \delta = 1.321 \left( \frac{T}{100} \right)^{1/3} )</td>
</tr>
<tr>
<td>( \gamma = 1 - \frac{25}{17\pi^2} )</td>
<td>( \gamma = 1 - \frac{6\pi^2}{17\pi^2} )</td>
</tr>
<tr>
<td>( n = \text{floor} \left( T^{2/9} \right) )</td>
<td>( n = \text{floor} \left( T^{2/9} \right) )</td>
</tr>
</tbody>
</table>

where \( \delta_{n,j} = 1 \) if \( j = 0 \) and 0, otherwise; \( \gamma_{n,j} = 1 + \frac{4}{T} \sum_{i=1}^{T} u_{i+j} \).

Simulation evidence shows that the KPSS has size distortion in finite samples. For example, see Caner and Kilian (2001). The power is reduced when the sample size is large, which can be derived theoretically (see Breitung (1995)). Another problem of the KPSS test is that the power depends on the choice of the truncation lag used in the Newey-West estimator of the long run variance \( s(I) \).

**Testing for Statistical Independence**

Independence tests are widely used in model selection, residual analysis, and model diagnostics because models are usually based on the assumption of independently distributed errors. If a given time series (for example, a series of residuals) is independent, then no deterministic model is necessary for this completely random process; otherwise, there must exist some relationship in the series to be addressed. In the following section, four independence tests are introduced: the BDS test, the runs test, the turning point test, and the rank version of von Neumann ratio test.
BDS Test

Broock, Dechert, and Scheinkman (1987) propose a test (BDS test) of independence based on the correlation dimension. Broock et al. (1996) show that the first-order asymptotic distribution of the test statistic is independent of the estimation error provided that the parameters of the model under test can be estimated consistently. Hence, the BDS test can be used as a model selection tool and as a specification test.

Given the sample size \( T \), the embedding dimension \( m \), and the value of the radius \( r \), the BDS statistic is

\[
S_{\text{BDS}}(T, m, r) = \sqrt{T - m + 1} \frac{c_n(n, \delta, r) - c_m(n, T, r)}{\sigma_m(T, r)}
\]

where

\[
c_n(n, \delta, r) = \frac{2}{(N-n+1)(N-n)} \sum_{k=0}^{N-1} \prod_{j=0}^{n-1} I_k(z_k - z_{k+1})
\]

\[
I_k(z, z') = \begin{cases} 
1 & \text{if } |z - z'| < r \\
0 & \text{otherwise}
\end{cases}
\]

\[
\sigma_m(T, r) = 4 \left( k^2 + 2 \sum_{i=1}^{m-1} k^{n-i} e^{2i} + (m-1)^2 e^{2m} - m^2 e^{2m-2} \right)
\]

\[
c = c_{1,1,T}(r)
\]

\[
k = k_T(r) = \frac{6}{T(T-1)(T-2)} \sum_{i=1}^{T} \sum_{j=1}^{i-1} \sum_{x=1}^{T} h_x(z_i, z_j, z)
\]

\[
h_x(z_i, z_j, z) = \frac{1}{3} \left( I_x(z_i, z_j) I_x(z_i, z) + I_x(z_j, z_i) I_x(z_j, z) + I_x(z_i, z_j) I_x(z_j, z) \right)
\]

The statistic has a standard normal distribution if the sample size is large enough. For small sample size, the distribution can be approximately obtained through simulation. Kanzler (1999) has a comprehensive discussion on the implementation and empirical performance of BDS test.

Runs Test and Turning Point Test

The runs test and turning point test are two widely used tests for independence (Cromwell, Labys, and Terraza; 1994).

The runs test needs several steps. First, convert the original time series into the sequence of signs, \( \{+ - + - \ldots + - - \} \), that is, map \( \{z_t\} \) into \( \{\text{sign}(z_t) - 2M\} \) where \( z_M \) is the sample mean of \( z_t \) and \( \text{sign}(x) \) is “+” if \( x \) is nonnegative and “-” if \( x \) is negative. Second, count the number of runs, \( R \), in the sequence. A run of a sequence is a maximal non-empty segment of the sequence that consists of adjacent equal elements. For example, the following sequence contains \( R = 8 \) runs:

```
++-++-+-+-+-++-+--
```

Third, count the number of pluses and minuses in the sequence and denote them as \( N_+ \) and \( N_- \), respectively. In the preceding example sequence, \( N_+ = 11 \) and \( N_- = 8 \). Note that the sample size \( T = N_+ + N_- \). Finally, compute the statistic of runs test,

\[
S_{\text{runs}} = \frac{R - \mu}{\sigma}
\]

where

\[
\mu = \frac{2N_+N_-}{T} + 1
\]

\[
\sigma^2 = \frac{(\mu - 1)(\mu - 2)}{T - 1}
\]

The statistic of the turning point test is defined as follows:

\[
S_{\text{TP}} = \frac{\sum_{i=2}^{T-1} TP_i - 2(T-2)/3}{\sqrt{(10T - 29)/90}}
\]

where the indicator function of the turning point \( TP_i \) is 1 if \( z_i > z_{i+1} \) or \( z_i < z_{i-1} \) (that is, both the previous and next values are greater or less
than the current value); otherwise, 0.

The statistics of both the runs test and the turning point test have the standard normal distribution under the null hypothesis of independence.

**Rank Version of von Neumann Ratio Test**

Since the runs test completely ignores the magnitudes of the observations, Bartels (1982) proposes a rank version of the von Neumann Ratio test for independence:

\[
S_{RVN} = \frac{\sqrt{T}}{2} \left( \frac{\sum_{i=1}^{T-1} (R_{i+1} - R_i)^2}{(T^2 + 1)/12} - 2 \right)
\]

where \( R_i \) is the rank of \( i \)th observation in the sequence of \( T \) observations. For large sample, the statistic follows the standard normal distribution under the null hypothesis of independence. For small samples of size between 11 and 100, the critical values through simulation would be more precise; for samples of size no more than 10, the exact CDF is applied.

**Testing for Normality**

Based on skewness and kurtosis, Jarque and Bera (1980) calculated the test statistic

\[
T_N = \left[ \frac{N}{6} b_1^2 + \frac{N}{24} (b_2 - 3)^2 \right]
\]

where

\[
b_1 = \frac{\sqrt{\sum_{i=1}^{N} \hat{e}_i^4}}{(\sum_{i=1}^{N} \hat{e}_i^2)^{1/2}}
\]

\[
b_2 = \frac{\sum_{i=1}^{N} \hat{e}_i^4}{(\sum_{i=1}^{N} \hat{e}_i^2)^2}
\]

The \( \chi^2(2) \) distribution gives an approximation to the normality test \( T_N \).

When the GARCH model is estimated, the normality test is obtained using the standardized residuals \( \hat{e}_t = \hat{\varepsilon}_t / \sqrt{R} \). The normality test can be used to detect misspecification of the family of ARCH models.

**Testing for Linear Dependence**

**Generalized Durbin-Watson Tests**

Consider the following linear regression model:

\[
Y = X\beta + \nu
\]

where \( X \) is an \( N \times k \) data matrix, \( \beta \) is a \( k \times 1 \) coefficient vector, and \( \nu \) is a \( N \times 1 \) disturbance vector. The error term \( \nu \) is assumed to be generated by the \( j \)th-order autoregressive process \( \tilde{\varepsilon}_t = \varepsilon_t - \phi_1 \varepsilon_{t-1} \) where \( |\phi_1| < 1 \). For a sequence of independent normal error terms with mean 0 and variance \( \sigma^2 \) Usually, the Durbin-Watson statistic is used to test the null hypothesis \( H_0 : \phi_1 = 0 \) against \( H_1 : -\phi_1 > 0 \). Vinod (1973) generalized the Durbin-Watson statistic:

\[
d_j = \frac{\sum_{i=j+1}^{N} (\tilde{\varepsilon}_i - \tilde{\varepsilon}_{i-j})^2}{\sum_{i=1}^{N} \tilde{\varepsilon}_i^2}
\]

where \( \tilde{\varepsilon} \) are OLS residuals. Using the matrix notation,

\[
d_j = \frac{\text{MYMA}_j \text{MY}}{\text{YM}}
\]

where \( M = I_N - X(X'X)^{-1}X' \) and \( A_j \) is a \((N - j) \times N\) matrix:
\[
A_j = \begin{bmatrix}
-1 & 0 & \cdots & 0 & 1 & 0 & \cdots & 0 \\
0 & -1 & 0 & \cdots & 0 & 1 & \cdots & 0 \\
\vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots \\
0 & \cdots & 0 & -1 & 0 & \cdots & 0 & 1
\end{bmatrix}
\]

and there are \( j - 1 \) zeros between -1 and 1 in each row of matrix \( A_j \).

The QR factorization of the design matrix \( X \) yields an \( N \times N \) orthogonal matrix \( Q \):

\[
X = QR
\]

where \( R \) is an \( N \times k \) upper triangular matrix. There exists an \( N \times (N - k) \) submatrix of \( Q \) such that \( Q_iQ_i' = M \) and \( Q_i'Q_1 = I_{N-k} \). Consequently, the generalized Durbin-Watson statistic is stated as a ratio of two quadratic forms:

\[
d_j = \frac{\sum_{i=1}^{n} \lambda_i \xi_i^2}{\sum_{i=1}^{n} \xi_i^2}
\]

where \( \lambda_1, \ldots, \lambda_n \) are upper \( n \) eigenvalues of \( MA_j'A_jM \) and \( \xi_i \) is a standard normal variate, and \( n = \min(N-k, N-j) \). These eigenvalues are obtained by a singular value decomposition of \( Q_i'A_j' \) (Golub and Van Loan; 1989; Savin and White, 1978).

The marginal probability (or \( p \)-value) for \( d_j \) given \( c_0 \) is

\[
\text{Prob} \left( \frac{\sum_{i=1}^{n} \lambda_i \xi_i^2}{\sum_{i=1}^{n} \xi_i^2} < c_0 \right) = \text{Prob}(q_j < 0)
\]

where

\[
q_j = \sum_{i=1}^{n} (\lambda_i - c_0) \xi_i^2
\]

When the null hypothesis \( H_0 : \phi_j = 0 \) holds, the quadratic form \( q_j \) has the characteristic function

\[
\phi_j(t) = \prod_{i=1}^{n} \left( 1 - 2(\lambda_i - c_0)it \right)^{-1/2}
\]

The distribution function is uniquely determined by this characteristic function:

\[
F(x) = \frac{1}{2} + \frac{1}{2\pi} \int_0^\infty \frac{e^{itx} \phi_j(-t) - e^{-itx} \phi_j(t)}{it} dt
\]

For example, to test \( H_0 : \phi_4 = 0 \) given \( \phi_1 = \phi_2 = \phi_3 = 0 \) against \( H_1 : -\phi_4 > 0 \), the marginal probability (\( p \)-value) can be used:

\[
F(0) = \frac{1}{2} + \frac{1}{2\pi} \int_0^\infty \frac{\phi_4(-t) - \phi_4(t)}{it} dt
\]

where

\[
\phi_4(t) = \prod_{i=1}^{n} (1 - 2(\lambda_i - d_4)it)^{-1/2}
\]

and \( d_4 \) is the calculated value of the fourth-order Durbin-Watson statistic.

In the Durbin-Watson test, the marginal probability indicates positive autocorrelation (\( -\phi_j > 0 \)) if it is less than the level of significance (\( \alpha \)), while you can conclude that a negative autocorrelation (\( -\phi_j < 0 \)) exists if the marginal probability based on the computed Durbin-Watson statistic is greater than \( 1 - \alpha \). Wallis (1972) presented tables for bounds tests of fourth-order autocorrelation, and Vinod (1973) has given tables for a 5% significance level for orders two to four. Using the AUTOREG procedure, you can calculate the exact \( p \)-values for the general order of Durbin-Watson test statistics. Tests for the absence of autocorrelation of order \( p \) can be performed sequentially; at the \( j \)th step, test \( H_0 : \phi_j = 0 \) given \( \phi_1 = \ldots = \phi_{j-1} = 0 \) against \( \phi_j \neq 0 \). However, the size of the sequential test is not known.

The Durbin-Watson statistic is computed from the OLS residuals, while that of the autoregressive error model uses residuals that are the
difference between the predicted values and the actual values. When you use the Durbin-Watson test from the residuals of the autoregressive error model, you must be aware that this test is only an approximation. See Autoregressive Error Model earlier in this chapter. If there are missing values, the Durbin-Watson statistic is computed using all the nonmissing values and ignoring the gaps caused by missing residuals. This does not affect the significance level of the resulting test, although the power of the test against certain alternatives may be adversely affected. Savin and White (1978) have examined the use of the Durbin-Watson statistic with missing values.

The Durbin-Watson probability calculations have been enhanced to compute the p-value of the generalized Durbin-Watson statistic for large sample sizes. Previously, the Durbin-Watson probabilities were only calculated for small sample sizes.

Consider the following linear regression model:

\[ Y = X\beta + u \]

\[ u_t + \phi_t u_{t-j} = \xi_t, \quad t = 1, \ldots, N \]

where \( X \) is an \( N \times k \) data matrix, \( \beta \) is a \( k \times 1 \) coefficient vector, \( u \) is a \( N \times 1 \) disturbance vector, and \( \xi_t \) is a sequence of independent normal error terms with mean 0 and variance \( \sigma^2 \).

The generalized Durbin-Watson statistic is written as

\[ DW_j = \frac{\hat{u}' A_j' A_j \hat{u}}{\hat{u}' \hat{u}} \]

where \( \hat{u} \) is a vector of OLS residuals and \( A_j \) is a \( (T-j) \times T \) matrix. The generalized Durbin-Watson statistic \( DW_j \) can be rewritten as

\[ DW_j = \frac{Y' M A_j' A_j M Y}{Y' M Y} = \frac{\eta'(Q_j' A_j' A_j Q_j)\eta}{\eta' \eta} \]

where \( Q_j' Q_j = I_{T-j} \), \( Q_j' X = 0 \), and \( \eta = Q_j' u \)

The marginal probability for the Durbin-Watson statistic is

\[ \Pr(DW_j < c) = \Pr(h < 0) \]

where \( h = \eta'(Q_j' A_j' A_j Q_j - c I)\eta \).

The p-value or the marginal probability for the generalized Durbin-Watson statistic is computed by numerical inversion of the characteristic function \( \phi(u) \) of the quadratic form \( h = \eta'(Q_j' A_j' A_j Q_j - c I)\eta \). The trapezoidal rule approximation to the marginal probability \( \Pr(h < 0) \) is

\[ \Pr(h < 0) = \frac{1}{2} \sum_{k=0}^{K} \text{Im} \left[ \frac{\phi \left( \frac{k+1}{2} \Delta \right)}{\pi \left( k + \frac{1}{2} \right)} \right] + E_t(\Delta) + E_t(K) \]

where \( \text{Im}[\phi(\cdot)] \) is the imaginary part of the characteristic function, \( E_t(\Delta) \) and \( E_t(K) \) are integration and truncation errors, respectively. Refer to Davies (1973) for numerical inversion of the characteristic function.

Ansley, Kohn, and Shively (1992) proposed a numerically efficient algorithm that requires \( O(N) \) operations for evaluation of the characteristic function \( \phi(u) \). The characteristic function is denoted as

\[ \phi(u) = \left| 1 - 2iu Q_j' A_j' A_j Q_j - c I_{N-j} \right|^{-1/2} \]

\[ = \left| V \right|^{1/2} \left| X' V^{-1} X \right|^{-1/2} \left| X' X \right|^{1/2} \]

where \( V = (1 + 2iu) I_j - 2iu A_j' A_j \) and \( i = \sqrt{-1} \). By applying the Cholesky decomposition to the complex matrix \( V \), you can obtain the lower triangular matrix \( G \) that satisfies \( V = GG' \). Therefore, the characteristic function can be evaluated in \( O(N) \) operations by using the following formula:

\[ \phi(u) = \left| G^{-1} \right| \left| X' X \right|^{-1/2} \left| X' X \right|^{1/2} \]

where \( X' = G^{-1} X \) Refer to Ansley, Kohn, and Shively (1992) for more information on evaluation of the characteristic function.

Tests for Serial Correlation with Lagged Dependent Variables
When regressors contain lagged dependent variables, the Durbin-Watson statistic ($d_1$) for the first-order autocorrelation is biased toward 2 and has reduced power. Wallis (1972) shows that the bias in the Durbin-Watson statistic ($d_1$) for the fourth-order autocorrelation is smaller than the bias in $d_1$ in the presence of a first-order lagged dependent variable. Durbin (1970) proposes two alternative statistics ($\bar{d}$ and $t$) that are asymptotically equivalent. The $h$ statistic is written as

$$h = \beta \sqrt{N / (1 - N^r)}$$

where $\beta = \sum_{t=2}^{N} \hat{V}_t \hat{u}_{t-1} / \sum_{t=1}^{N} \hat{V}_t^2$ and $\hat{V}$ is the least squares variance estimate for the coefficient of the lagged dependent variable. Durbin’s $t$ test consists of regressing the OLS residuals $\hat{e}_t$ on explanatory variables and $\hat{V}_{t-1}$ and testing the significance of the estimate for coefficient of $\hat{V}_{t-1}$.

Inder (1984) shows that the Durbin-Watson test for the absence of first-order autocorrelation is generally more powerful than the $h$ test in finite samples. Refer to Inder (1986) and King and Wu (1991) for the Durbin-Watson test in the presence of lagged dependent variables.

Godfrey LM test

The GODFREY= option in the MODEL statement produces the Godfrey Lagrange multiplier test for serially correlated residuals for each equation (Godfrey 1978a and 1978b). $r$ is the maximum autoregressive order, and specifies that Godfrey’s tests be computed for lags 1 through $r$. The default number of lags is four.

Testing for Nonlinear Dependence: Ramsey’s Reset Test

Ramsey’s reset test is a misspecification test associated with the functional form of models to check whether power transforms need to be added to a model. The original linear model, henceforth called the restricted model, is

$$y_t = x_0 \beta + u_t$$

To test for misspecification in the functional form, the unrestricted model is

$$y_t = x_0 \beta + \sum_{j=2}^{p} \phi_j \hat{y}_t^j + u_t$$

where $\hat{y}_t$ is the predicted value from the linear model and $p$ is the power of $\hat{y}_t$ in the unrestricted model equation starting from 2. The number of higher-ordered terms to be chosen depends on the discretion of the analyst. The RESET option produces test results for $p = 2, 3,$ and 4.

The reset test is an $F$ statistic for testing $H_0: \phi_j = 0$, for all $j = 2, \ldots, p$ against $H_1: \phi_j \neq 0$ for at least one $j = 2, \ldots, p$ in the unrestricted model and is computed as follows:

$$F_{(p-1, n-p+1)} = \frac{(SSE_R - SSE_U) / (p - 1)}{SSE_U / (n - F - p + 1)}$$

where $SSE_R$ is the sum of squared errors due to the restricted model, $SSE_U$ is the sum of squared errors due to the unrestricted model, $n$ is the total number of observations, and $F$ is the number of parameters in the original linear model.

Ramsey’s test can be viewed as a linearity test that checks whether any nonlinear transformation of the specified independent variables has been omitted, but it need not help in identifying a new relevant variable other than those already specified in the current model.

Testing for Nonlinear Dependence: Heteroscedasticity Tests

Portmanteau Q Test

For nonlinear time series models, the portmanteau test statistic based on squared residuals is used to test for independence of the series (McLeod and Li; 1983):

$$Q(q) = N(N + 2) \sum_{i=1}^{N-q} \frac{r(i; \hat{V}_i^2)}{(N - i)}$$

where

$$r(i; \hat{V}_i^2) = \frac{\sum_{t=1}^{N-i} (\hat{V}_t^2 - \hat{\sigma}^2)(\hat{V}_{t+i}^2 - \hat{\sigma}^2)}{\sum_{t=1}^{N} (\hat{V}_t^2 - \hat{\sigma}^2)^2}$$

$$\hat{\sigma}^2 = \frac{1}{N} \sum_{t=1}^{N} \hat{V}_t^2$$
This Q statistic is used to test the nonlinear effects (for example, GARCH effects) present in the residuals. The GARCH\(_{(p,q)}\) process can be considered as an ARMA\(\max(p,q)\) process. See the section Predicting the Conditional Variance later in this chapter. Therefore, the Q statistic calculated from the squared residuals can be used to identify the order of the GARCH process.

**Engle’s Lagrange Multiplier Test for ARCH Disturbances**

Engle (1982) proposed a Lagrange multiplier test for ARCH disturbances. The test statistic is asymptotically equivalent to the test used by Breusch and Pagan (1979). Engle's Lagrange multiplier test for the \(q\)th order ARCH process is written

\[
LM(q) = \frac{NW'Z(Z'Z)^{-1}Z'W}{W'W}
\]

where

\[
W = \left( \frac{\hat{\sigma}_1^2}{\hat{\sigma}_2^2 - 1}, \ldots, \frac{\hat{\sigma}_q^2}{\hat{\sigma}_2^2 - 1} \right),
\]

and

\[
Z = \begin{bmatrix}
1 & \hat{\sigma}_0^2 & \cdots & \hat{\sigma}_q^2 - 1 \\
\vdots & \vdots & \ddots & \vdots \\
\vdots & \vdots & \ddots & \vdots \\
1 & \hat{\sigma}_{q-1}^2 - 1 & \cdots & \hat{\sigma}_q^2 - 1
\end{bmatrix}
\]

The presample values \((\hat{\sigma}_0^2, \hat{\sigma}_1^2, \ldots, \hat{\sigma}_q^2)\) have been set to 0. Note that the LM\((q)\) tests might have different finite-sample properties depending on the presample values, though they are asymptotically equivalent regardless of the presample values.

**Lee and King’s Test for ARCH Disturbances**

Engle's Lagrange multiplier test for ARCH disturbances is a two-sided test; that is, it ignores the inequality constraints for the coefficients in ARCH models. Lee and King (1993) propose a one-sided test and prove that the test is locally most mean powerful. Let \(e_t, t = 1, \ldots, T\), denote the residuals to be tested. Lee and King’s test checks

\[
H_0 : \alpha_t = 0, i = 1, \ldots, q \\
H_1 : \alpha_t > 0, i = 1, \ldots, q
\]

where \(\alpha_t, i = 1, \ldots, q\) are in the following ARCH\((q)\) model:

\[
e_t = \sqrt{h_t} \epsilon_t, i.d(0, 1) \\
h_t = \alpha_0 + \sum_{i=1}^{q} \alpha_i \epsilon_{t-1}^2
\]

The statistic is written as

\[
S = \left[ \frac{\sum_{t=q+1}^{T}(\hat{e}_t^2 - 1) \sum_{i=1}^{q} \hat{e}_{t-i}^2}{2 \sum_{t=q+1}^{T}(\sum_{i=1}^{q} \hat{e}_{t-i}^2)^2 - \frac{1}{q} \left( \sum_{i=1}^{q} \hat{e}_{t-i}^2 \right)^2} \right]^{1/2}
\]

**Wong and Li’s Test for ARCH Disturbances**

Wong and Li (1995) propose a rank portmanteau statistic to minimize the effect of the existence of outliers in the test for ARCH disturbances. They first rank the squared residuals; that is, \(R_t = \text{rank}(\hat{e}_t^2)\). Then they calculate the rank portmanteau statistic

\[
Q_R = \sum_{i=1}^{q} \frac{(r_i - \mu_i)^2}{\sigma_i^2}
\]

where \(r_i, \mu_i, \) and \(\sigma_i^2\) are defined as follows:
The Q, Engle’s LM, Lee and King’s, and Wong and Li’s statistics are computed from the OLS residuals, or residuals if the NLAG= option is specified, assuming that disturbances are white noise. The Q, Engle’s LM, and Wong and Li’s statistics have an approximate $\chi^2(\nu)$ distribution under the white-noise null hypothesis, while the Lee and King’s statistic has a standard normal distribution under the white-noise null hypothesis.

**Testing for Structural Change: Chow Test**

Consider the linear regression model

$$y = X\beta + u$$

where the parameter vector $\beta$ contains $k$ elements.

Split the observations for this model into two subsets at the break point specified by the CHOW= option, so that

$$y = (y_1', y_2')'$$

$$X = (X_1', X_2')'$$

$$u = (u_1', u_2')'$$

Now consider the two linear regressions for the two subsets of the data modeled separately,

$$y_1 = X_1\beta_1 + u_1$$

$$y_2 = X_2\beta_2 + u_2$$

where the number of observations from the first set is $n_1$ and the number of observations from the second set is $n_2$.

The Chow test statistic is used to test the null hypothesis $H_0 : \beta_1 = \beta_2$ conditional on the same error variance $V(u_1) = V(u_2)$. The Chow test is computed using three sums of square errors:

$$F_{chow} = \frac{(\hat{u}'\hat{u} - \hat{u}_1'\hat{u}_1 - \hat{u}_2'\hat{u}_2)/k}{(\hat{u}_1'\hat{u}_1 + \hat{u}_2'\hat{u}_2)/(n_1 + n_2 - 2k)}$$

where $\hat{u}$ is the regression residual vector from the full set model, $\hat{u}_1$ is the regression residual vector from the first set model, and $\hat{u}_2$ is the regression residual vector from the second set model. Under the null hypothesis, the Chow test statistic has an $F$ distribution with $k$ and $(n_1 + n_2 - 2k)$ degrees of freedom, where $k$ is the number of elements in $\beta$.

Chow (1960) suggested another test statistic that tests the hypothesis that the mean of prediction errors is 0. The predictive Chow test can also be used when $n_2 < k$.

The PCHOW= option computes the predictive Chow test statistic

$$F_{pre-chow} = \frac{(\hat{u}'\hat{u} - \hat{u}_1'\hat{u}_1)/n_2}{\hat{u}_1'\hat{u}_1/(n_1 - k)}$$

The predictive Chow test has an $F$ distribution with $n_2$ and $(n_1 - k)$ degrees of freedom.
The AUTOREG Procedure

Predicted Values

The AUTOREG procedure can produce two kinds of predicted values for the response series and corresponding residuals and confidence limits. The residuals in both cases are computed as the actual value minus the predicted value. In addition, when GARCH models are estimated, the AUTOREG procedure can output predictions of the conditional error variance.

Predicting the Unconditional Mean

The first type of predicted value is obtained from only the structural part of the model, \( \hat{\mathbf{y}}_t = \mathbf{x}_t^\prime \mathbf{b} \). These are useful in predicting values of new response time series, which are assumed to be described by the same model as the current response time series. The predicted values, residuals, and upper and lower confidence limits for the structural predictions are requested by specifying the PREDICTEDM=, RESIDUALM=, UCLM=, or LCLM= option in the OUTPUT statement. The ALPHACLM= option controls the confidence level for UCLM= and LCLM=.

These confidence limits are for estimation of the mean of the dependent variable, \( \hat{\mathbf{y}}_t \), where \( \mathbf{x}_t \) is the column vector of independent variables at observation \( t \).

The predicted values are computed as

\[
\hat{y}_t = \mathbf{x}_t^\prime \mathbf{b}
\]

and the upper and lower confidence limits as

\[
\hat{y}_t \pm t_{\alpha/2} \hat{v} = \hat{y}_t \pm t_{\alpha/2} \sqrt{\sum_{i=1}^{p} \hat{\phi}_i t_{i/2} \hat{v}}
\]

where \( \hat{v}^2 \) is an estimate of the variance of \( \hat{y}_t \) and \( t_{\alpha/2} \) is the upper \( \alpha/2 \) percentage point of the \( t \) distribution.

\[
\text{Prob}(T > t_{\alpha/2}) = \alpha/2
\]

where \( T \) is an observation from a \( t \) distribution with \( q \) degrees of freedom. The value of \( \alpha \) can be set with the ALPHACLM= option. The degrees of freedom parameter, \( q \), is taken to be the number of observations minus the number of free parameters in the regression and autoregression parts of the model. For the YW estimation method, the value of \( \hat{v} \) is calculated as

\[
\hat{v} = \sqrt{\hat{X}_t X_t^{-1} X_t}
\]

where \( \hat{X}_t \) is the error sum of squares divided by \( q \). For the ULS and ML methods, it is calculated as

\[
\hat{v} = \sqrt{\hat{X}_t W \hat{X}_t}
\]

where \( W \) is the \( k \times k \) submatrix of \( (\mathbf{J}^\prime \mathbf{J})^{-1} \) that corresponds to the regression parameters. For details, see the section Computational Methods earlier in this chapter.

Predicting Future Series Realizations

The other predicted values use both the structural part of the model and the predicted values of the error process. These conditional mean values are useful in predicting future values of the current response time series. The predicted values, residuals, and upper and lower confidence limits for future observations conditional on past values are requested by the PREDICTED=, RESIDUAL=, UCL=, or LCL= option in the OUTPUT statement. The ALPHACL= option controls the confidence level for UCL= and LCL=.

These confidence limits are for the predicted value,

\[
\hat{y}_t = \mathbf{x}_t^\prime \mathbf{b} + V_{y_{t-1}}
\]

where \( \mathbf{x}_t \) is the vector of independent variables if all independent variables at time \( t \) are nonmissing, and \( V_{y_{t-1}} \) is the minimum variance linear predictor of the error term, which is defined in the following recursive way given the autoregressive model, AR(\( m \)) model, for \( V_{y_t} \):

\[
V_{y_t} = \begin{cases} 
    0 
    & s > t \text{ or observation s is missing} \\
    \sum_{i=s}^{t} \phi_i V_{y_{t-i}} 
    & 0 < s \leq t \text{ and observation s is nonmissing} \\
    \mathbf{x}_s - \mathbf{x}_t^\prime \mathbf{b} 
    & s \leq 0
\end{cases}
\]

where \( \phi_i, i = 1, \ldots, m \) are the estimated AR parameters. Observation \( s \) is considered to be missing if the dependent variable or at least one
independent variable is missing. If some of the independent variables at time \( t \) are missing, the predicted \( \hat{y}_t \) is also missing. With the same definition of \( v_{sp}^2 \), the prediction method can be easily extended to the multistep forecast of \( \hat{y}_{t+d} | d > 0 \):

\[
\hat{y}_{t+d} = x_{t+d}^T b + v_{t+d|t-1}
\]

The prediction method is implemented through the Kalman filter.

If \( \hat{y}_t \) is not missing, the upper and lower confidence limits are computed as

\[
\hat{u}_t = \hat{y}_t + t_{(\alpha/2)} v \\
\hat{l}_t = \hat{y}_t - t_{(\alpha/2)} v
\]

where \( v \), in this case, is computed as

\[
v = \sqrt{\hat{y}^T \hat{V}_\beta \hat{y} + \hat{s}^2 r}
\]

where \( \hat{V}_\beta \) is the variance-covariance matrix of the estimation of regression parameter \( \beta \). \( \hat{z}_t \) is defined as

\[
\hat{z}_t = x_t + \sum_{s=1}^{m} \hat{\phi}_s x_{t-s|t-1}
\]

and \( \hat{x}_{s|t} \) is defined in a similar way as \( v_{sp}^2 \):

\[
\hat{x}_{s|t} = \begin{cases} 
0 & s > t \text{ or observation } s \text{ is missing} \\
- \sum_{j=1}^{n} \hat{\phi}_j x_{t-s-j|t} & 0 < s \leq t \text{ and observation } s \text{ is nonmissing} \\
- \sum_{j=1}^m \hat{\phi}_j x_{t-s-j|t} & s \leq 0
\end{cases}
\]

The value \( \hat{s}^2 r \) is the estimate of the conditional prediction error variance. At the start of the series, and after missing values, \( r \) is generally greater than 1. See the section Predicting the Conditional Variance for the computational details of \( r \). The plot of residuals and confidence limits in Example 8.4 illustrates this behavior.

Except to adjust the degrees of freedom for the error sum of squares, the preceding formulas do not account for the fact that the autoregressive parameters are estimated. In particular, the confidence limits are likely to be somewhat too narrow. In large samples, this is probably not an important effect, but it might be appreciable in small samples. Refer to Harvey (1981) for some discussion of this problem for AR(1) models.

At the beginning of the series (the first \( m \) observations, where \( m \) is the value of the NLAG= option) and after missing values, these residuals do not match the residuals obtained by using OLS on the transformed variables. This is because, in these cases, the predicted noise values must be based on less than a complete set of past noise values and, thus, have larger variance. The GLS transformation for these observations includes a scale factor in addition to a linear combination of past values. Put another way, the \( L^{-1} \) matrix defined in the section Computational Methods has the value 1 along the diagonal, except for the first \( m \) observations and after missing values.

Predicting the Conditional Variance

The GARCH process can be written

\[
epsilon_t^2 = \omega + \sum_{i=1}^{n} (\alpha_i + \eta_i) \epsilon_{t-i}^2 - \sum_{j=1}^{p} \gamma_j |\eta_{t-j}| + \epsilon_t
\]

where \( \eta_t = \epsilon_t^2 - h_t \) and \( n = \max(p, q) \). This representation shows that the squared residual \( \epsilon_t^2 \) follows an ARMA\( (n, p) \) process. Then for any \( d > 0 \), the conditional expectations are as follows:

\[
E(\epsilon_{t+d}^2 | W_t) = \omega + \sum_{i=1}^{n} (\alpha_i + \eta_i) E(\epsilon_{t-i+d}^2 | W_t) - \sum_{j=1}^{p} \gamma_j E(\eta_{t+d-j} | W_t)
\]

The \( d \)-step-ahead prediction error, \( \tilde{\epsilon}_{t+d} = y_{t+d} - y_{t+d|t} \), has the conditional variance

\[
V(\tilde{\epsilon}_{t+d} | W_t) = \sum_{j=0}^{d-1} \sigma_j^2 \alpha_{d-j}^2
\]
where
\[
\sigma^2_{t+d-j|t} = E(\varepsilon_{t+d-j}^2|\Psi_t)
\]

Coefficients in the conditional \(d\)-step prediction error variance are calculated recursively using the formula
\[
g_j = -\phi_1 g_{j-1} - \cdots - \phi_m g_{j-m}
\]

where \(g_0 = 1\) and \(g_j = 0\) if \(j < 0\); \(\phi_1, \ldots, \phi_m\) are autoregressive parameters. Since the parameters are not known, the conditional variance is computed using the estimated autoregressive parameters. The \(d\)-step-ahead prediction error variance is simplified when there are no autoregressive terms:
\[
V(\varepsilon_{t+d}|\Psi_t) = \sigma^2_{t+d|t}
\]

Therefore, the one-step-ahead prediction error variance is equivalent to the conditional error variance defined in the GARCH process:
\[
h_t = E(\varepsilon_t^2|\Psi_{t-1}) = \sigma^2_{t|t-1}
\]

The multistep forecast of conditional error variance of the EGARCH, QGARCH, TGARCH, PGARCH, and GARCH-M models cannot be calculated using the preceding formula for the GARCH model. The following formulas are recursively implemented to obtain the multistep forecast of conditional error variance of these models:

\(\varepsilon\) for the EGARCH\((p, q)\) model:
\[
\ln(\sigma^2_{t+d|t}) = \omega + \sum_{i=1}^{d-1} g_i \ln(\sigma^2_{t+d-i}) + \sum_{i=1}^{d-1} \gamma_i \ln(h^*_{t+d-i}) + \sum_{i=1}^{d-1} \gamma_i \ln(h^*_{t+d-i})
\]

where
\[
g_i = \beta_i \varepsilon_i + |\varepsilon_i| - E|\varepsilon_i|
\]
\[
\varepsilon_i = \varepsilon_i / \sqrt{h_i}
\]

\(\varepsilon\) for the QGARCH\((p, q)\) model:
\[
\sigma^2_{t+d|t} = \omega + \sum_{i=1}^{d-1} g_i (\sigma^2_{t+d-i} + \psi^2) + \sum_{i=1}^{d-1} g_i (\sigma^2_{t+d-i} - \psi^2)
\]
\[
+ \sum_{i=1}^{d-1} g_i \sigma^2_{t+d-i} + \sum_{i=1}^{d-1} \gamma_i h^*_{t+d-i}
\]

\(\varepsilon\) for the TGARCH\((p, q)\) model:
\[
\sigma^2_{t+d|t} = \omega + \sum_{i=1}^{d-1} (\alpha_i + \psi_i/2) \sigma^2_{t+d-i} + \sum_{i=1}^{d-1} (\alpha_i + 1/\omega_i) \sigma^2_{t+d-i} - \sum_{i=1}^{d-1} \gamma_i \sigma^2_{t+d-i} + \sum_{i=1}^{d-1} \gamma_i h^*_{t+d-i}
\]

\(\varepsilon\) for the PGARCH\((p, q)\) model:
\[
(\sigma^2_{t+d|t})^{2\lambda} = \omega + \sum_{i=1}^{d-1} g_i (1 + \psi_i)^{2\lambda} + (1 - \psi_i)^{2\lambda} (\sigma^2_{t+d-i})^{2\lambda} / 2 + \sum_{i=1}^{d-1} \alpha_i (1 + \psi_i)^{2\lambda} + (1 - \psi_i)^{2\lambda} (\sigma^2_{t+d-i})^{2\lambda} / 2
\]
\[
+ \sum_{i=1}^{d-1} g_i (1 + \psi_i)^{2\lambda} + (1 - \psi_i)^{2\lambda} (\sigma^2_{t+d-i})^{2\lambda} / 2 + \sum_{i=1}^{d-1} \gamma_i (\sigma^2_{t+d-i})^{2\lambda} + \sum_{i=1}^{d-1} \gamma_i h^2_{t+d-i}
\]

\(\varepsilon\) for the GARCH-M model: ignoring the mean effect and directly using the formula of the corresponding GARCH model.

If the conditional error variance is homoscedastic, the conditional prediction error variance is identical to the unconditional prediction error variance
\[ \mathbf{V}^e(\mathbf{\hat{y}}_t|\Psi_t) - \mathbf{V}^e(\mathbf{\hat{y}}_t) - \sigma^2 \sum_{j=0}^{d-1} \hat{\sigma}_j^2 \]

since \( \sigma_{t-d-j}^2 \hat{\sigma}_j^2 = \sigma^2 \). You can compute \( \sigma^2 \) (which is the second term of the variance for the predicted value \( \mathbf{\hat{y}}_t \) explained in the section Predicting Future Series Realizations) by using the formula \( \sigma^2 \sum_{j=0}^{d-1} \hat{\sigma}_j^2 \) and \( r \) is estimated from \( \sum_{j=0}^{d-1} \hat{\sigma}_j^2 \) by using the estimated autoregressive parameters.

Consider the following conditional prediction error variance:

\[ \mathbf{V}^e(\mathbf{\hat{y}}_t|\Psi_t) = \sigma^2 \sum_{j=0}^{d-1} \hat{\sigma}_j^2 + \sum_{j=0}^{d-1} \hat{\sigma}_j^2 (\sigma_{t-d-j}^2 \hat{\sigma}_j^2 - \sigma^2) \]

The second term in the preceding equation can be interpreted as the noise from using the homoscedastic conditional variance when the errors follow the GARCH process. However, it is expected that if the GARCH process is covariance stationary, the difference between the conditional prediction error variance and the unconditional prediction error variance disappears as the forecast horizon \( d \) increases.
the estimate of the \( \psi \) parameter in the TGARCH model, if a TGARCH model is specified. There are \( q \) such variables \_AHT_ 1 through \_AHT_ \( q \), where \( q \) is the value of the Q= option.

\_DELTA_

the estimated mean parameter for the GARCH-M model if a GARCH-in-mean model is specified

\_DEPVAR_

the name of the dependent variable

\_GH_ \( i \)

the \( i \)th order GARCH parameter estimate, if the GARCH= option is specified. There are \( p \) such variables \_GH_ 1 through \_GH_ \( p \), where \( p \) is the value of the P= option.

\_HET_ \( i \)

the \( i \)th heteroscedasticity model parameter specified by the HETERO statement

INTERCEPT

the intercept estimate. INTERCEPT contains a missing value for models for which the NOINT option is specified.

\_METHOD_

the estimation method that is specified in the METHOD= option

\_MODEL_

the label of the MODEL statement if one is given, or blank otherwise

\_MSE_

the value of the mean square error for the model

\_NAME_

the name of the row of covariance matrix for the parameter estimate, if the COVOUT option is specified

\_LAMBDA_

the estimate of the power parameter \( \lambda \) in the PGARCH model, if a PGARCH model is specified.

\_LIKLHD_

the log-likelihood value of the GARCH model

\_SSE_

the value of the error sum of squares

\_START_

the estimated start-up value for the conditional variance when GARCH= (STARTUP=ESTIMATE) option is specified

\_STATUS_

This variable indicates the optimization status. \_STATUS_ = 0 indicates that there were no errors during the optimization and the algorithm converged. \_STATUS_ = 1 indicates that the optimization could not improve the function value and means that the results should be interpreted with caution. \_STATUS_ = 2 indicates that the optimization failed due to the number of iterations exceeding either the maximum default or the specified number of iterations or the number of function calls allowed. \_STATUS_ = 3 indicates that an error occurred during the optimization process. For example, this error message is obtained when a function or its derivatives cannot be calculated at the initial values or during the iteration process, when an optimization step is outside of the feasible region or when active constraints are linearly dependent.

\_STDERR_
standard error of the parameter estimate, if the COVOUT option is specified.

_THDFI_

the estimate of the inverted degrees of freedom for Student’s $t$ distribution, if DIST=T is specified.

_THETA_

the estimate of the $\theta$ parameter in the EGARCH model, if an EGARCH model is specified.

_TYPE_

OLS for observations containing parameter estimates, or COV for observations containing covariance matrix elements.

The OUTEST= data set contains one observation for each MODEL statement giving the parameter estimates for that model. If the COVOUT option is specified, the OUTEST= data set includes additional observations for each MODEL statement giving the rows of the covariance of parameter estimates matrix. For covariance observations, the value of the _TYPE_ variable is COV, and the _NAME_ variable identifies the parameter associated with that row of the covariance matrix.

The AUTOREG procedure prints the following items:

1. the name of the dependent variable
2. the ordinary least squares estimates
3. Estimates of autocorrelations, which include the estimates of the autocovariances, the autocorrelations, and (if there is sufficient space) a graph of the autocorrelation at each LAG
4. if the PARTIAL option is specified, the partial autocorrelations
5. the preliminary MSE, which results from solving the Yule-Walker equations. This is an estimate of the final MSE.
6. the estimates of the autoregressive parameters (Coefficient), their standard errors (Standard Error), and the ratio of estimate to standard error (t Value)
7. the statistics of fit for the final model. These include the error sum of squares (SSE), the degrees of freedom for error (DFE), the mean square error (MSE), the mean absolute error (MAE), the mean absolute percentage error (MAPE), the root mean square error (Root MSE), the Schwarz information criterion (SBC), the Hannan-Quinn information criterion (HQC), the Akaike information criterion (AIC), the corrected Akaike information criterion (AICC), the Durbin-Watson statistic (Durbin-Watson), the regression $R^2$ (Regress R-square), and the total $R^2$ (Total R-square). For GARCH models, the following additional items are printed:
   - the value of the log-likelihood function (Log Likelihood)
   - the number of observations that are used in estimation (Observations)
   - the unconditional variance (Uncond Var)
   - the normality test statistic and its $p$-value (Normality Test and Pr > ChiSq)
8. the parameter estimates for the structural model (Estimate), a standard error estimate (Standard Error), the ratio of estimate to standard error (t Value), and an approximation to the significance probability for the parameter being 0 (Approx Pr > |t|)
9. If the NLAG= option is specified with METHOD=ULS or METHOD=ML, the regression parameter estimates are printed again, assuming that the autoregressive parameter estimates are known. In this case, the Standard Error and related statistics for the regression estimates will, in general, be different from the case when they are estimated. Note that from a standpoint of estimation, Yule-Walker and iterated Yule-Walker methods (NLAG= with METHOD=YW, ITYW) generate only one table, assuming AR parameters are given.
10. If you specify the NORMAL option, the Bera-Jarque normality test statistics are printed. If you specify the LAGDEP option, Durbin’s $h$
or Durbin’s $f$ is printed.

## The AUTOREG Procedure

### ODS Table Names

PROC AUTOREG assigns a name to each table it creates. You can use these names to reference the table when using the Output Delivery System (ODS) to select tables and create output data sets. These names are listed in the Table 8.2.

### Table 8.2 ODS Tables Produced in PROC AUTOREG

<table>
<thead>
<tr>
<th>ODS Table Name</th>
<th>Description</th>
<th>Option</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>ODS Tables Created by the MODEL Statement</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ClassLevels</td>
<td>Class Levels</td>
<td>default</td>
</tr>
<tr>
<td>FitSummary</td>
<td>Summary of regression</td>
<td>default</td>
</tr>
<tr>
<td>SummaryDepVarCen</td>
<td>Summary of regression (centered dependent var)</td>
<td>CENTER</td>
</tr>
<tr>
<td>SummaryNoIntercept</td>
<td>Summary of regression (no intercept)</td>
<td>NOINT</td>
</tr>
<tr>
<td>YWIterSSE</td>
<td>Yule-Walker iteration sum of squared error</td>
<td>METHOD=ITYW</td>
</tr>
<tr>
<td>PreMSE</td>
<td>Preliminary MSE</td>
<td>NLAG=</td>
</tr>
<tr>
<td>Dependent</td>
<td>Dependent variable</td>
<td>default</td>
</tr>
<tr>
<td>DependenceEquations</td>
<td>Linear dependence equation</td>
<td></td>
</tr>
<tr>
<td>ARCHTest</td>
<td>Tests for ARCH disturbances based on OLS residuals</td>
<td>ARCHTEST=</td>
</tr>
<tr>
<td>ARCHTestAR</td>
<td>Tests for ARCH disturbances based on residuals</td>
<td>ARCHTEST= (with NLAG=)</td>
</tr>
<tr>
<td>BDSTest</td>
<td>BDS test for independence</td>
<td>BDS&lt;=()</td>
</tr>
<tr>
<td>RunsTest</td>
<td>Runs test for independence</td>
<td>RUNS&lt;=()</td>
</tr>
<tr>
<td>TurningPointTest</td>
<td>Turning Point test for independence</td>
<td>TP&lt;=()</td>
</tr>
<tr>
<td>VNRRankTest</td>
<td>Rank version of von Neumann ratio test for independence</td>
<td>VNRRANK&lt;=()</td>
</tr>
<tr>
<td>ChowTest</td>
<td>Chow test and predictive Chow test</td>
<td>CHOW= PCHOW=</td>
</tr>
<tr>
<td>Godfrey</td>
<td>Godfrey’s serial correlation test</td>
<td>GODFREY&lt;=()</td>
</tr>
<tr>
<td>PhilPerron</td>
<td>Phillips-Perron unit root test</td>
<td>STATIONARITY= (PHILIPS&lt;=() (no regressor)</td>
</tr>
<tr>
<td>PhilOul</td>
<td>Phillips-Ouliaris cointegration test</td>
<td>STATIONARITY= (PHILIPS&lt;=() (has regressor)</td>
</tr>
<tr>
<td>ADF</td>
<td>Augmented Dickey-Fuller unit root test</td>
<td>STATIONARITY= (ADF&lt;=() (no regressor)</td>
</tr>
<tr>
<td>EngGran</td>
<td>Engle-Granger cointegration test</td>
<td>STATIONARITY= (ADF&lt;=() (has regressor)</td>
</tr>
<tr>
<td>ERS</td>
<td>ERS unit root test</td>
<td>STATIONARITY= (ERS&lt;=()</td>
</tr>
<tr>
<td>NgPerron</td>
<td>Ng-Perron Unit root tests</td>
<td>STATIONARITY= (NP&lt;=()</td>
</tr>
<tr>
<td>KPSS</td>
<td>Kwiatkowski, Phillips, Schmidt, and Shin test</td>
<td>STATIONARITY= (KPSS&lt;=()</td>
</tr>
<tr>
<td>ResetTest</td>
<td>Ramsey’s RESET test</td>
<td>RESET</td>
</tr>
<tr>
<td>ARParameterEstimates</td>
<td>Estimates of autoregressive parameters</td>
<td>NLAG=</td>
</tr>
<tr>
<td>CorrGraph</td>
<td>estimates of autocorrelations</td>
<td>NLAG=</td>
</tr>
<tr>
<td>BackStep</td>
<td>Backward elimination of autoregressive terms</td>
<td>BACKSTEP</td>
</tr>
<tr>
<td>ExpAutocorr</td>
<td>Expected autocorrelations</td>
<td>NLAG=</td>
</tr>
<tr>
<td>IterHistory</td>
<td>Iteration history</td>
<td>ITPRINT</td>
</tr>
<tr>
<td>ParameterEstimates</td>
<td>Parameter estimates</td>
<td>default</td>
</tr>
<tr>
<td>ParameterEstimatesGivenAR</td>
<td>Parameter estimates assuming AR parameters are given</td>
<td>NLAG=, METHOD= ULS</td>
</tr>
<tr>
<td>PartialAutoCov</td>
<td>Partial autocorrelation</td>
<td>PARTIAL</td>
</tr>
<tr>
<td>CovB</td>
<td>Covariance of parameter estimates</td>
<td>COVB</td>
</tr>
<tr>
<td>CorrB</td>
<td>Correlation of parameter estimates</td>
<td>CORRB</td>
</tr>
<tr>
<td>CholeskyFactor</td>
<td>Cholesky root of gamma</td>
<td>ALL</td>
</tr>
<tr>
<td>Coefficients</td>
<td>Coefficients for first NLAG observations</td>
<td>COEF</td>
</tr>
<tr>
<td>GammaInverse</td>
<td>Gamma inverse</td>
<td>GINV</td>
</tr>
<tr>
<td>ConvergenceStatus</td>
<td>Convergence status table</td>
<td>default</td>
</tr>
<tr>
<td>MiscStat</td>
<td>MiscStat Durbin $f$ or Durbin $h$, Bera-Jarque normality test</td>
<td>LAGDEP&lt;=() NORMAL</td>
</tr>
<tr>
<td>DWTest</td>
<td>Durbin-Watson statistics</td>
<td>DW=</td>
</tr>
<tr>
<td><strong>ODS Tables Created by the RESTRICT Statement</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Restrict</td>
<td>Restriction table</td>
<td>default</td>
</tr>
<tr>
<td><strong>ODS Tables Created by the TEST Statement</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>FTest</td>
<td>$F$ test</td>
<td>default, TYPE=ALL</td>
</tr>
<tr>
<td>WaldTest</td>
<td>Wald test</td>
<td>TYPE=WALD/ALL</td>
</tr>
</tbody>
</table>
This section describes the use of ODS for creating graphics with the AUTOREG procedure. To request these graphs, you must specify the ODS GRAPHICS statement. By default, only the residual, predicted versus actual, and autocorrelation of residuals plots are produced. If, in addition to the ODS GRAPHICS statement, you also specify the ALL option in either the PROC AUTOREG statement or MODEL statement, all plots are created. For HETERO, GARCH, and AR models, studentized residuals are replaced by standardized residuals. For the autoregressive models, the conditional variance of the residuals is computed as described in the section Predicting Future Series Realizations. For the GA RCH and HETERO models, residuals are assumed to have conditional variance invoked by the HT= option of the OUTPUT statement. For all these cases, the Cook’s D plot is not produced.

**ODS Graph Names**

PROC AUTOREG assigns a name to each graph it creates using ODS. You can use these names to reference the graphs when using ODS. The names are listed in Table 8.3.

<table>
<thead>
<tr>
<th>ODS Graph Name</th>
<th>Plot Description</th>
<th>Option</th>
</tr>
</thead>
<tbody>
<tr>
<td>ACFPlot</td>
<td>Autocorrelation of residuals</td>
<td>ACF</td>
</tr>
<tr>
<td>FitPlot</td>
<td>Predicted versus actual plot</td>
<td>Default</td>
</tr>
<tr>
<td>CooksD</td>
<td>Cook’s D plot</td>
<td>ALL (no NLAG=)</td>
</tr>
<tr>
<td>IACFPlot</td>
<td>Inverse autocorrelation of residuals</td>
<td>ALL</td>
</tr>
<tr>
<td>QQPlot</td>
<td>Q-Q plot of residuals</td>
<td>ALL</td>
</tr>
<tr>
<td>PACFPlot</td>
<td>Partial autocorrelation of residuals</td>
<td>ALL</td>
</tr>
<tr>
<td>ResidualHistogram</td>
<td>Histogram of the residuals</td>
<td>ALL</td>
</tr>
<tr>
<td>ResidualPlot</td>
<td>Residual plot</td>
<td>Default</td>
</tr>
<tr>
<td>StudentResidualPlot</td>
<td>Studentized residual plot</td>
<td>ALL (no NLAG=/HETERO=/GARCH=)</td>
</tr>
<tr>
<td>StandardResidualPlot</td>
<td>Standardized residual plot</td>
<td>ALL</td>
</tr>
<tr>
<td>WhiteNoiseLogProbPlot</td>
<td>Tests for white noise residuals</td>
<td>ALL</td>
</tr>
</tbody>
</table>

**Examples: AUTOREG Procedure**

- Analysis of Real Output Series
- Comparing Estimates and Models
- Lack-of-Fit Study
- Missing Values
- Money Demand Model
- Estimation of ARCH(2) Process
- Estimation of GARCH-Type Models
- Illustration of ODS Graphics
Example 8.1 Analysis of Real Output Series

In this example, the annual real output series is analyzed over the period 1901 to 1983 (Balke and Gordon; 1986, pp. 581–583). With the following DATA step, the original data are transformed using the natural logarithm, and the differenced series DY is created for further analysis. The log of real output is plotted in Output 8.1.1.

```plaintext
title 'Analysis of Real GNP';
data gnp;
  date = intnx( 'year', '01jan1901'd, _n_-1 );
  format date year4.;
  input x @@;
  y  = log(x);
  dy = dif(y);
  t  = _n_;  
  label y  = 'Real GNP'
             dy = 'First Difference of Y'
             t  = 'Time Trend';
datalines;

proc sgplot data=gnp noautolegend;
  scatter x=date y=y;
  xaxis grid values=('01jan1901'd '01jan1911'd '01jan1921'd '01jan1931'd '01jan1941'd '01jan1951'd '01jan1961'd '01jan1971'd '01jan1981'd '01jan1991'd)
run;
```

**Output 8.1.1 Real Output Series: 1901 – 1983**

The (linear) trend-stationary process is estimated using the following form:

\[ y_t = \beta_0 + \beta_1 t + \nu_t \]

where

\[ \nu_t = \phi_1 \nu_{t-1} - \phi_2 \nu_{t-2} \]
\[ \varepsilon_t \sim \text{IN}(0, \sigma^2_e) \]

The preceding trend-stationary model assumes that uncertainty over future horizons is bounded since the error term, \( \varepsilon_t \), has a finite variance.

The maximum likelihood AR estimates from the statements that follow are shown in Output 8.1.2:

```plaintext
proc autoreg data=gnp;
   model y = t / nlag=2 method=ml;
run;
```

**Output 8.1.2 Estimating the Linear Trend Model**

Analysis of Real GNP

| Parameter Estimates | Variable DF | Estimate | Standard Error | t Value | Approx Pr > |t| | Variable Label |
|---------------------|-------------|----------|----------------|---------|-------------|---------------|----------------|
| Intercept           | 1           | 4.8206   | 0.0661         | 72.88   | <.0001      |               | Time Trend     |
| t                   | 1           | 0.0302   | 0.001346       | 22.45   | <.0001      |               | Time Trend     |
| AR1                 | 1           | -1.2041  | 0.1040         | -11.58  | <.0001      |               | Time Trend     |
| AR2                 | 1           | 0.3748   | 0.1039         | 3.61    | 0.0005      |               | Time Trend     |

Nelson and Plosser (1982) failed to reject the hypothesis that macroeconomic time series are nonstationary and have no tendency to return to a trend line. In this context, the simple random walk process can be used as an alternative process:

\[ y_t = \beta_0 + y_{t-1} + \varepsilon_t \]

where \( \varepsilon_t = \xi_t \) and \( y_0 = 0 \). In general, the difference-stationary process is written as

\[ \phi(L)(1-L)y_t = \beta_0 + \theta(L)\varepsilon_t \]

where \( L \) is the lag operator. You can observe that the class of a difference-stationary process should have at least one unit root in the AR polynomial \( \phi(L)(1-L) \).

The Dickey-Fuller procedure is used to test the null hypothesis that the series has a unit root in the AR polynomial. Consider the following equation for the augmented Dickey-Fuller test:

\[ \Delta y_t = \beta_0 + \delta_t + \beta_1 y_{t-1} + \sum_{i=1}^{m} \gamma_i \Delta y_{t-i} + \xi_t \]

where \( \Delta = 1 - L \). The test statistic \( \tau_t \) is the usual \( t \) ratio for the parameter estimate \( \hat{\beta}_0 \), but the \( \tau_t \) does not follow a \( t \) distribution.

The following code performs the augmented Dickey-Fuller test with \( m = 3 \) and we are interested in the test results in the linear time trend case.
since the previous plot reveals there is a linear trend.

```
proc autoreg data = gnp;
  model y = / stationarity =(adf =3);
run;
```

The augmented Dickey-Fuller test indicates that the output series may have a difference-stationary process. The statistic Tau with linear time trend has a value of -2.6190 and its p-value is 0.2732. The statistic Rho has a p-value of 0.0817 which also indicates the null of unit root is accepted at the 5% level. (See Output 8.1.3.)

**Output 8.1.3 Augmented Dickey-Fuller Test Results**

Analysis of Real GNP

```
The AUTOREG Procedure

Augmented Dickey-Fuller Unit Root Tests

<table>
<thead>
<tr>
<th>Type</th>
<th>Lags</th>
<th>Rho</th>
<th>Pr &lt; Rho</th>
<th>Tau</th>
<th>Pr &lt; Tau</th>
<th>F</th>
<th>Pr &gt; F</th>
</tr>
</thead>
<tbody>
<tr>
<td>Zero Mean</td>
<td>3</td>
<td>0.3827</td>
<td>0.7732</td>
<td>3.3342</td>
<td>0.9997</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Single Mean</td>
<td>3</td>
<td>-0.1674</td>
<td>0.9465</td>
<td>-0.2046</td>
<td>0.9326</td>
<td>5.7521</td>
<td>0.0211</td>
</tr>
<tr>
<td>Trend</td>
<td>3</td>
<td>-18.0246</td>
<td>0.0817</td>
<td>-2.6190</td>
<td>0.2732</td>
<td>3.4472</td>
<td>0.4957</td>
</tr>
</tbody>
</table>
```

The AR(1) model for the differenced series DY is estimated using the maximum likelihood method for the period 1902 to 1983. The difference-stationary process is written

\[
\Delta y_t = \beta_0 + \nu_t \\
\nu_t = \phi_1 \nu_{t-1}
\]

The estimated value of \(\phi_1\) is -0.297 and that of \(\beta_0\) is 0.0293. All estimated values are statistically significant. The PROC step follows:

```
proc autoreg data=gnp;
  model dy = / nlag=1 method=ml;
run;
```

The printed output produced by the PROC step is shown in **Output 8.1.4**.

**Output 8.1.4 Estimating the Differenced Series with AR(1) Error**

Analysis of Real GNP

```
The AUTOREG Procedure

Maximum Likelihood Estimates

<table>
<thead>
<tr>
<th></th>
<th>SSE</th>
<th>DFE</th>
<th>DFE</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>0.27107673</td>
<td>80</td>
<td></td>
</tr>
<tr>
<td>MSE</td>
<td>0.00339</td>
<td></td>
<td></td>
</tr>
<tr>
<td>SBC</td>
<td>-226.77848</td>
<td></td>
<td></td>
</tr>
<tr>
<td>MAE</td>
<td>0.04333026</td>
<td></td>
<td></td>
</tr>
<tr>
<td>MAPE</td>
<td>153.637587</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Durbin-Watson</td>
<td>1.9268</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Regress R-Square</td>
<td>0.0000</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Total R-Square</td>
<td>0.0900</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
```

```
Parameter Estimates

| Variable | DF | Estimate | Standard Error | t Value | Approx Pr > |t| |
|----------|----|----------|----------------|---------|-------------|---|
| Intercept| 1  | 0.0293   | 0.009093       | 3.22    | 0.0018      |
| AR1      | 1  | -0.2967  | 0.1067         | -2.78   | 0.0067      |
```

```
Autoregressive parameters assumed given

| Variable | DF | Estimate | Standard Error | t Value | Approx Pr > |t| |
|----------|----|----------|----------------|---------|-------------|---|
| Intercept| 1  | 0.0293   | 0.009093       | 3.22    | 0.0018      |
```

file:///C:/Users/CSchulman/AppData/Local/Temp/~hhE81E.htm 9/23/2013
Example 8.2 Comparing Estimates and Models

In this example, the Grunfeld series are estimated using different estimation methods. Refer to Maddala (1977) for details of the Grunfeld investment data set. For comparison, the Yule-Walker method, ULS method, and maximum likelihood method estimates are shown. With the DWPROB option, the p-value of the Durbin-Watson statistic is printed. The Durbin-Watson test indicates the positive autocorrelation of the regression residuals. The DATA and PROC steps follow:

title 'Grunfeld''s Investment Models Fit with Autoregressive Errors';
data grunfeld;
  input year gei gef gec;
  label gei = 'Gross investment GE'
    gef = 'Lagged Value of GE shares';
datalines;
... more lines ...

proc autoreg data=grunfeld;
  model gei = gef gec / nlag=1 dwprob;
  model gei = gef gec / nlag=1 method=uls;
  model gei = gef gec / nlag=1 method=ml;
run;

The printed output produced by each of the MODEL statements is shown in Output 8.2.1 through Output 8.2.4.

Output 8.2.1 OLS Analysis of Residuals
Grunfeld's Investment Models Fit with Autoregressive Errors

<table>
<thead>
<tr>
<th>Dependent Variable</th>
<th>gei</th>
<th>Gross investment GE</th>
</tr>
</thead>
<tbody>
<tr>
<td>Ordinary Least Squares Estimates</td>
<td></td>
<td></td>
</tr>
<tr>
<td>SSE</td>
<td>13216.5878</td>
<td></td>
</tr>
<tr>
<td>MSE</td>
<td>777.44634</td>
<td></td>
</tr>
<tr>
<td>SBC</td>
<td>195.614652</td>
<td></td>
</tr>
<tr>
<td>MAE</td>
<td>19.9433255</td>
<td></td>
</tr>
<tr>
<td>MAPE</td>
<td>23.2049793</td>
<td></td>
</tr>
<tr>
<td>Durbin-Watson</td>
<td>1.0721</td>
<td></td>
</tr>
<tr>
<td>Regress R-Square</td>
<td>0.7053</td>
<td></td>
</tr>
<tr>
<td>Total R-Square</td>
<td>0.7053</td>
<td></td>
</tr>
</tbody>
</table>

Parameter Estimates

<table>
<thead>
<tr>
<th>Variable</th>
<th>DF</th>
<th>Estimate</th>
<th>Standard Error</th>
<th>t Value</th>
<th>Approx Pr &gt;</th>
<th>Variable Label</th>
</tr>
</thead>
<tbody>
<tr>
<td>Intercept</td>
<td>1</td>
<td>-9.9563</td>
<td>31.3742</td>
<td>-0.32</td>
<td>0.7548</td>
<td></td>
</tr>
<tr>
<td>gef</td>
<td>1</td>
<td>0.0266</td>
<td>0.0156</td>
<td>1.71</td>
<td>0.1063</td>
<td>Lagged Value of GE shares</td>
</tr>
<tr>
<td>gec</td>
<td>1</td>
<td>0.1517</td>
<td>0.0257</td>
<td>5.90</td>
<td>&lt;.0001</td>
<td>Lagged Capital Stock GE</td>
</tr>
</tbody>
</table>

Estimates of Autocorrelations

<table>
<thead>
<tr>
<th>Lag</th>
<th>Covariance</th>
<th>Correlation</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>660.8</td>
<td>1.000000</td>
</tr>
<tr>
<td>1</td>
<td>304.6</td>
<td>0.460867</td>
</tr>
</tbody>
</table>
Output 8.2.2 Regression Results Using Default Yule-Walker Method

<table>
<thead>
<tr>
<th>Lag</th>
<th>Coefficient</th>
<th>Standard Error</th>
<th>t Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>-0.460867</td>
<td>0.221867</td>
<td>-2.08</td>
</tr>
</tbody>
</table>

Yule-Walker Estimates

<table>
<thead>
<tr>
<th></th>
<th>SSE</th>
<th>DFE</th>
<th>16</th>
</tr>
</thead>
<tbody>
<tr>
<td>MSE</td>
<td>639.89344</td>
<td>Root MSE</td>
<td>25.29612</td>
</tr>
<tr>
<td>SBC</td>
<td>193.742396</td>
<td>AIC</td>
<td>189.759467</td>
</tr>
<tr>
<td>MAE</td>
<td>18.0715195</td>
<td>AICC</td>
<td>192.426133</td>
</tr>
<tr>
<td>MAPE</td>
<td>21.0772644</td>
<td>HQC</td>
<td>190.536976</td>
</tr>
<tr>
<td>Durbin-Watson</td>
<td>1.3321</td>
<td>Regress R-Square</td>
<td>0.5717</td>
</tr>
<tr>
<td></td>
<td>Total R-Square</td>
<td></td>
<td>0.7717</td>
</tr>
</tbody>
</table>

Algorithm converged.

Output 8.2.3 Regression Results Using Unconditional Least Squares Method

<table>
<thead>
<tr>
<th>Lag</th>
<th>Coefficient</th>
<th>Standard Error</th>
<th>t Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>-0.460867</td>
<td>0.221867</td>
<td>-2.08</td>
</tr>
</tbody>
</table>

Unconditional Least Squares Estimates

<table>
<thead>
<tr>
<th></th>
<th>SSE</th>
<th>DFE</th>
<th>16</th>
</tr>
</thead>
<tbody>
<tr>
<td>MSE</td>
<td>10220.8455</td>
<td>Root MSE</td>
<td>25.27455</td>
</tr>
<tr>
<td>SBC</td>
<td>638.80284</td>
<td>AIC</td>
<td>189.773763</td>
</tr>
<tr>
<td>MAE</td>
<td>193.756692</td>
<td>AICC</td>
<td>192.44043</td>
</tr>
<tr>
<td>MAPE</td>
<td>21.149176</td>
<td>HQC</td>
<td>190.551273</td>
</tr>
<tr>
<td>Durbin-Watson</td>
<td>1.3523</td>
<td>Regress R-Square</td>
<td>0.5511</td>
</tr>
<tr>
<td></td>
<td>Total R-Square</td>
<td></td>
<td>0.7721</td>
</tr>
</tbody>
</table>

Parameter Estimates

<table>
<thead>
<tr>
<th>Variable Label</th>
<th>DF</th>
<th>Estimate</th>
<th>Standard Error</th>
<th>t Value</th>
<th>Approx Pr &gt;</th>
<th>Variable Label</th>
</tr>
</thead>
<tbody>
<tr>
<td>Intercepts</td>
<td>1</td>
<td>-18.6582</td>
<td>33.7567</td>
<td>-0.55</td>
<td>0.5881</td>
<td>Lagged Value of GE shares</td>
</tr>
<tr>
<td>gef</td>
<td>1</td>
<td>0.0339</td>
<td>0.0179</td>
<td>1.89</td>
<td>0.0769</td>
<td>Lagged Value of GE shares</td>
</tr>
<tr>
<td>gec</td>
<td>1</td>
<td>0.1369</td>
<td>0.0449</td>
<td>3.05</td>
<td>0.0076</td>
<td>Lagged Capital Stock GE</td>
</tr>
<tr>
<td>AR1</td>
<td>1</td>
<td>-0.4996</td>
<td>0.2592</td>
<td>-1.93</td>
<td>0.0718</td>
<td>Lagged Capital Stock GE</td>
</tr>
</tbody>
</table>

Autoregressive parameters assumed given

<table>
<thead>
<tr>
<th>Variable Label</th>
<th>DF</th>
<th>Estimate</th>
<th>Standard Error</th>
<th>t Value</th>
<th>Approx Pr &gt;</th>
<th>Variable Label</th>
</tr>
</thead>
<tbody>
<tr>
<td>Intercept</td>
<td>1</td>
<td>-18.6582</td>
<td>33.7567</td>
<td>-0.55</td>
<td>0.5881</td>
<td>Lagged Value of GE shares</td>
</tr>
</tbody>
</table>

Preliminary MSE

520.5
Many time series exhibit high positive autocorrelation, having the smooth appearance of a random walk. This behavior can be explained by the partial adjustment and adaptive expectation hypotheses.

Short-term forecasting applications often use autoregressive models because these models absorb the behavior of this kind of data. In the case of a first-order AR process where the autoregressive parameter is exactly 1 (a random walk), the best prediction of the future is the immediate past.

PROC AUTOREG can often greatly improve the fit of models, not only by adding additional parameters but also by capturing the random walk tendencies. Thus, PROC AUTOREG can be expected to provide good short-term forecast predictions.

However, good forecasts do not necessarily mean that your structural model contributes anything worthwhile to the fit. In the following example, random noise is fit to part of a sine wave. Notice that the structural model does not fit at all, but the autoregressive process does quite well and is very nearly a first difference (AR(1) = 0.970). The DATA step, PROC AUTOREG step, and PROC SGPLOT step follow:

### Example 8.3 Lack-of-Fit Study

| Variable | DF | Estimate | Standard Error | t Value | Approx Pr > |t| | Variable Label |
|----------|----|----------|----------------|---------|-------------|----------------|----------------|
| Intercept | 1 | -18.3751 | 34.5941 | -0.53 | 0.6026 | | Lagged Value of GE shares |
| gef | 1 | 0.0334 | 0.0179 | 1.87 | 0.0799 | Lagged Value of GE shares |
| gec | 1 | 0.1385 | 0.0428 | 3.23 | 0.0052 | Lagged Capital Stock GE |
| AR1 | 1 | -0.4728 | 0.2582 | -1.83 | 0.0858 | |

| Variable | DF | Estimate | Standard Error | t Value | Approx Pr > |t| | Variable Label |
|----------|----|----------|----------------|---------|-------------|----------------|----------------|
| Intercept | 1 | -18.3751 | 33.3931 | -0.55 | 0.5897 | | Lagged Value of GE shares |
| gef | 1 | 0.0334 | 0.0158 | 2.11 | 0.0512 | Lagged Value of GE shares |
| gec | 1 | 0.1385 | 0.0389 | 3.56 | 0.0026 | Lagged Capital Stock GE |

### Output 8.2.4 Regression Results Using Maximum Likelihood Method

#### Estimates of Autoregressive Parameters

<table>
<thead>
<tr>
<th>Lag</th>
<th>Coefficient</th>
<th>Standard Error</th>
<th>t Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>-0.460867</td>
<td>0.221867</td>
<td>-2.08</td>
</tr>
</tbody>
</table>

Algorithm converged.

### Maximum Likelihood Estimates

<table>
<thead>
<tr>
<th>SSE</th>
<th>10229.2303</th>
<th>DFE</th>
<th>16</th>
</tr>
</thead>
<tbody>
<tr>
<td>MSE</td>
<td>639.32689</td>
<td>DFE</td>
<td>16</td>
</tr>
<tr>
<td>SBC</td>
<td>193.738877</td>
<td>AIC</td>
<td>189.755947</td>
</tr>
<tr>
<td>MAE</td>
<td>18.0892426</td>
<td>AICC</td>
<td>192.422614</td>
</tr>
<tr>
<td>MAFE</td>
<td>21.0978407</td>
<td>HQC</td>
<td>190.533457</td>
</tr>
<tr>
<td>Durbin-Watson</td>
<td>1.3385</td>
<td>Regress R-Square</td>
<td>0.5656</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Total R-Square</td>
<td>0.7719</td>
</tr>
</tbody>
</table>

### Parameter Estimates

| Variable | DF | Estimate | Standard Error | t Value | Approx Pr > |t| |
|----------|----|----------|----------------|---------|-------------|----------------|
| Intercept | 1 | -18.3751 | 34.5941 | -0.53 | 0.6026 |
| gef | 1 | 0.0334 | 0.0179 | 1.87 | 0.0799 |
| gec | 1 | 0.1385 | 0.0428 | 3.23 | 0.0052 |
| AR1 | 1 | -0.4728 | 0.2582 | -1.83 | 0.0858 |

### Autoregressive parameters assumed given

| Variable | DF | Estimate | Standard Error | t Value | Approx Pr > |t| |
|----------|----|----------|----------------|---------|-------------|----------------|
| Intercept | 1 | -18.3751 | 33.3931 | -0.55 | 0.5897 |
| gef | 1 | 0.0334 | 0.0158 | 2.11 | 0.0512 |
| gec | 1 | 0.1385 | 0.0389 | 3.56 | 0.0026 |
title1 'Lack of Fit Study';
title2 'Fitting White Noise Plus Autoregressive Errors to a Sine Wave';

data a;
pi=3.14159;
do time = 1 to 75;
  if time > 75 then y = .;
  else y = sin( pi * ( time / 50 ) );
x = ranuni( 1234567 );
output;
end;
run;

proc autoreg data=a plots;
  model y = x / nlag=1;
  output out=b p=pred pm=xbeta;
run;

proc sgplot data=b;
  scatter y=y x=time / markerattrs=(color=black);
  series y=pred x=time / lineattrs=(color=blue);
  series y=xbeta x=time / lineattrs=(color=red);
run;

The printed output produced by PROC AUTOREG is shown in Output 8.3.1 and Output 8.3.2. Plots of observed and predicted values are shown in Output 8.3.3 and Output 8.3.4. Note: the plot Output 8.3.3 can be viewed in the Autoreg.Model.FitDiagnosticPlots category by selecting View Results.

Output 8.3.1 Results of OLS Analysis: No Autoregressive Model Fit
Lack of Fit Study
Fitting White Noise Plus Autoregressive Errors to a Sine Wave

The AUTOREG Procedure

<table>
<thead>
<tr>
<th>Ordinary Least Squares Estimates</th>
</tr>
</thead>
<tbody>
<tr>
<td>SSE</td>
</tr>
<tr>
<td>MSE</td>
</tr>
<tr>
<td>Root MSE</td>
</tr>
<tr>
<td>SBC</td>
</tr>
<tr>
<td>AIC</td>
</tr>
<tr>
<td>MAE</td>
</tr>
<tr>
<td>AICC</td>
</tr>
<tr>
<td>MAPE</td>
</tr>
<tr>
<td>HQC</td>
</tr>
<tr>
<td>Durbin-Watson</td>
</tr>
<tr>
<td>Regress R-Square</td>
</tr>
<tr>
<td>Total R-Square</td>
</tr>
</tbody>
</table>

Parameter Estimates

| Variable | DF | Estimate | Standard Error | t Value | Approx Pr > |t| |
|----------|----|----------|----------------|---------|-------------|---|
| Intercept| 1  | 0.2383   | 0.1584         | 1.50    | 0.1367      |
| x        | 1  | -0.0665  | 0.2771         | -0.24   | 0.8109      |

Estimates of Autocorrelations

<table>
<thead>
<tr>
<th>Lag</th>
<th>Covariance</th>
<th>Correlation</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0.4641</td>
<td>1.000000</td>
</tr>
<tr>
<td>1</td>
<td>0.4531</td>
<td>0.976386</td>
</tr>
</tbody>
</table>

Preliminary MSE 0.0217

Output 8.3.2 Regression Results with AR(1) Error Correction

<table>
<thead>
<tr>
<th>Estimates of Autoregressive Parameters</th>
</tr>
</thead>
<tbody>
<tr>
<td>Lag</td>
</tr>
<tr>
<td>-----</td>
</tr>
<tr>
<td>1</td>
</tr>
</tbody>
</table>
Yule-Walker Estimates

<p>| | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>SSE</td>
<td>0.18304264</td>
<td>DFE</td>
<td>72</td>
</tr>
<tr>
<td>MSE</td>
<td>0.00254</td>
<td>Root MSE</td>
<td>0.05042</td>
</tr>
<tr>
<td>SBC</td>
<td>-222.30643</td>
<td>AIC</td>
<td>-229.2589</td>
</tr>
<tr>
<td>MAE</td>
<td>0.04551667</td>
<td>AICC</td>
<td>-228.92087</td>
</tr>
<tr>
<td>MAPE</td>
<td>29145.3526</td>
<td>HQC</td>
<td>-226.48285</td>
</tr>
<tr>
<td>Durbin-Watson</td>
<td>0.0942</td>
<td>Regress R-Square</td>
<td>0.0001</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Total R-Square</td>
<td>0.9947</td>
</tr>
</tbody>
</table>

Parameter Estimates

| Variable | DF | Estimate | Standard Error | t Value | Approx Pr > |t|   |
|----------|----|----------|----------------|---------|-------------|-----|
| Intercept| 1  | -0.1473  | 0.1702         | -0.87   | 0.3898      |     |
| x        | 1  | -0.001219| 0.0141         | -0.09   | 0.9315      |     |

Output 8.3.3 Diagnostics Plots

Output 8.3.4 Plot of Autoregressive Prediction
Example 8.4 Missing Values

In this example, a pure autoregressive error model with no regressors is used to generate 50 values of a time series. Approximately 15% of the values are randomly chosen and set to missing. The following statements generate the data:

```sas
title 'Simulated Time Series with Roots:';
title2 ' (X-1.25)(X**4-1.25)';
title3 'With 15% Missing Values';
data ar;
do i=1 to 550;
e = rannor(12345);
n = sum( e, .8*n1, .8*n4, -.64*n5 );  /* ar process */
y = n;
if ranuni(12345) > .85 then y = .;    /* 15% missing */
n5=n4; n4=n3; n3=n2; n2=n1; n1=n;     /* set lags    */
if i>500 then output;
end;
run;
```

The model is estimated using maximum likelihood, and the residuals are plotted with 99% confidence limits. The PARTIAL option prints the partial autocorrelations. The following statements fit the model:

```sas
proc autoreg data=ar partial;
   model y = / nlag=(1 4 5) method=ml;
   output out=a predicted=p residual=r ucl=u lcl=l alphacli=.01;
run;
```

The printed output produced by the AUTOREG procedure is shown in Output 8.4.1 and Output 8.4.2. Note: the plot Output 8.4.2 can be viewed in the Autoreg.Model.FitDiagnosticPlots category by selecting View Results.

**Output 8.4.1 Autocorrelation-Corrected Regression Results**

Simulated Time Series with Roots:

(X-1.25)(X**4-1.25)
With 15% Missing Values

The AUTOREG Procedure

<table>
<thead>
<tr>
<th>Dependent Variable</th>
<th>y</th>
</tr>
</thead>
</table>

Ordinary Least Squares Estimates

<p>| | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>SSE</td>
<td>182.972379</td>
</tr>
<tr>
<td>MSE</td>
<td>4.57431</td>
</tr>
<tr>
<td>SBC</td>
<td>181.39282</td>
</tr>
<tr>
<td>MAE</td>
<td>1.80469152</td>
</tr>
<tr>
<td>MAPE</td>
<td>270.104379</td>
</tr>
<tr>
<td>Durbin-Watson</td>
<td>1.3962</td>
</tr>
</tbody>
</table>

<p>| | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>DFE</td>
<td>40</td>
</tr>
<tr>
<td>Root MSE</td>
<td>2.13876</td>
</tr>
<tr>
<td>AIC</td>
<td>179.679248</td>
</tr>
<tr>
<td>AICC</td>
<td>179.781813</td>
</tr>
<tr>
<td>HQC</td>
<td>180.303237</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Parameter Estimates</th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Variable</td>
<td>DF</td>
<td>Estimate</td>
<td>Standard Error</td>
</tr>
<tr>
<td>Intercept</td>
<td>1</td>
<td>-2.2387</td>
<td>0.3340</td>
</tr>
</tbody>
</table>

Estimates of Autocorrelations

<table>
<thead>
<tr>
<th>Lag</th>
<th>Covariance</th>
<th>Correlation</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>4.4627</td>
<td>1.000000</td>
</tr>
<tr>
<td>1</td>
<td>1.4241</td>
<td>0.319109</td>
</tr>
<tr>
<td>2</td>
<td>1.6505</td>
<td>0.369829</td>
</tr>
<tr>
<td>3</td>
<td>0.6808</td>
<td>0.152551</td>
</tr>
<tr>
<td>4</td>
<td>2.9167</td>
<td>0.653556</td>
</tr>
<tr>
<td>5</td>
<td>-0.3816</td>
<td>-0.085519</td>
</tr>
</tbody>
</table>

Partial Autocorrelations

<table>
<thead>
<tr>
<th>Lag</th>
<th>Autocorr</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0.319109</td>
</tr>
<tr>
<td>4</td>
<td>0.619288</td>
</tr>
<tr>
<td>5</td>
<td>-0.821179</td>
</tr>
</tbody>
</table>

Preliminary MSE 0.7609

Estimates of Autoregressive Parameters

<table>
<thead>
<tr>
<th>Lag</th>
<th>Coefficient</th>
<th>Standard Error</th>
<th>t Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>-0.733182</td>
<td>0.089966</td>
<td>-8.15</td>
</tr>
<tr>
<td>4</td>
<td>-0.803754</td>
<td>0.071849</td>
<td>-11.19</td>
</tr>
<tr>
<td>5</td>
<td>0.821179</td>
<td>0.093818</td>
<td>8.75</td>
</tr>
</tbody>
</table>

Expected Autocorrelations

<table>
<thead>
<tr>
<th>Lag</th>
<th>Autocorr</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>1.0000</td>
</tr>
<tr>
<td>1</td>
<td>0.4204</td>
</tr>
<tr>
<td>2</td>
<td>0.2480</td>
</tr>
<tr>
<td>3</td>
<td>0.3160</td>
</tr>
<tr>
<td>4</td>
<td>0.6903</td>
</tr>
<tr>
<td>5</td>
<td>0.0228</td>
</tr>
</tbody>
</table>
Algorithm converged.

### Maximum Likelihood Estimates

<p>| | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>SSE</td>
<td>48.4396756</td>
<td>DFE</td>
</tr>
<tr>
<td>MSE</td>
<td>1.30918</td>
<td>Root MSE</td>
</tr>
<tr>
<td>SBC</td>
<td>146.879013</td>
<td>AIC</td>
</tr>
<tr>
<td>MAE</td>
<td>0.88786192</td>
<td>AICC</td>
</tr>
<tr>
<td>MAPE</td>
<td>141.377721</td>
<td>HQC</td>
</tr>
<tr>
<td>Durbin-Watson</td>
<td>2.9457</td>
<td>Regress R-Square</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Total R-Square</td>
</tr>
</tbody>
</table>

### Parameter Estimates

| Variable  | DF | Estimate | Standard Error | t Value | Approx Pr > |t| |
|-----------|----|----------|----------------|---------|-------------|---|
| Intercept | 1  | -2.2370  | 0.5239         | -4.27   | 0.0001      |
| AR1       | 1  | -0.6201  | 0.1129         | -5.49   | <.0001      |
| AR4       | 1  | -0.7237  | 0.0914         | -7.92   | <.0001      |
| AR5       | 1  | 0.6550   | 0.1202         | 5.45    | <.0001      |

### Expected Autocorrelations

<table>
<thead>
<tr>
<th>Lag</th>
<th>Autocorr</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>1.0000</td>
</tr>
<tr>
<td>1</td>
<td>0.4204</td>
</tr>
<tr>
<td>2</td>
<td>0.2423</td>
</tr>
<tr>
<td>3</td>
<td>0.2958</td>
</tr>
<tr>
<td>4</td>
<td>0.6318</td>
</tr>
<tr>
<td>5</td>
<td>0.0411</td>
</tr>
</tbody>
</table>

### Autoregressive parameters assumed given

| Variable  | DF | Estimate | Standard Error | t Value | Approx Pr > |t| |
|-----------|----|----------|----------------|---------|-------------|---|
| Intercept | 1  | -2.2370  | 0.5225         | -4.28   | 0.0001      |

Output 8.4.2 Diagnostic Plots
The following statements plot the residuals and confidence limits:

```r
data reshapel;
  set a;
  miss = .;
  if r= . then do;
    miss = p;
    p = .;
  end;
run;
```

title 'Predicted Values and Confidence Limits';

proc sgplot data=reshape1 NOAUTOLEGEND;
  band x=i upper=u lower=l;
  scatter y=miss x=i/ MARKERATTRS =(symbol=x color=red);
  series y=p x=i/markers MARKERATTRS =(color=blue) lineattrs=(color=blue);
run;

The plot of the predicted values and the upper and lower confidence limits is shown in Output 8.4.3. Note that the confidence interval is wider at the beginning of the series (when there are no past noise values to use in the forecast equation) and after missing values where, again, there is an incomplete set of past residuals.

Output 8.4.3 Plot of Predicted Values and Confidence Interval
Example 8.5 Money Demand Model

This example estimates the log-log money demand equation by using the maximum likelihood method. The money demand model contains four explanatory variables. The lagged nominal money stock $M_1$ is divided by the current price level $GDF$ to calculate a new variable $M1CP$ since the money stock is assumed to follow the partial adjustment process. The variable $M1CP$ is then used to estimate the coefficient of adjustment. All variables are transformed using the natural logarithm with a DATA step. Refer to Balke and Gordon (1986) for a data description.

The first eight observations are printed using the PRINT procedure and are shown in Output 8.5.1. Note that the first observation of the variables $M1CP$ and $INFR$ are missing. Therefore, the money demand equation is estimated for the period 1968:2 to 1983:4 since PROC AUTOREG ignores the first missing observation. The DATA step that follows generates the transformed variables.

data money;
  date = intnx( 'qtr', '01jan1968'd, _n_-1 );
  format date yyqc6.;
  input m1 gnp gdf ycb @@;
  m = log( 100 * m1 / gdf );
  m1cp = log( 100 * lag(m1) / gdf );
  y = log( gnp );
  intr = log( ycb );
  infr = 100 * log( gdf / lag(gdf) );
  label m    = 'Real Money Stock (M1)'
                m1cp = 'Lagged M1/Current GDF'
                y    = 'Real GNP'
                intr = 'Yield on Corporate Bonds'
                infr = 'Rate of Prices Changes';
datalines;
... more lines ...

Output 8.5.1 Money Demand Data Series – First 8 Observations
Predicted Values and Confidence Limits
The money demand equation is first estimated using OLS. The DW=4 option produces generalized Durbin-Watson statistics up to the fourth order. Their exact marginal probabilities (p-values) are also calculated with the DWPROB option. The Durbin-Watson test indicates positive first-order autocorrelation at, say, the 10% confidence level. You can use the Durbin-Watson table, which is available only for 1% and 5% significance points. The relevant upper (U) and lower (L) bounds are 1.731 and 1.471, respectively, at 5% significance level. However, the bounds test is inconvenient, since sometimes you may get the statistic in the inconclusive region while the interval between the upper and lower bounds becomes smaller with the increasing sample size. The PROC step follows:

```plaintext
title 'Partial Adjustment Money Demand Equation';
title2 'Quarterly Data - 1968:2 to 1983:4';
proc autoreg data=money outest=est covout;
model m = m1cp y intr infr / dw=4 dwprob;
run;
```

Output 8.5.2 OLS Estimation of the Partial Adjustment Money Demand Equation

Partial Adjustment Money Demand Equation
Quarterly Data - 1968:2 to 1983:4

The AUTOREG Procedure

<table>
<thead>
<tr>
<th>Obs</th>
<th>date</th>
<th>m1</th>
<th>gnp</th>
<th>gdf</th>
<th>ycb</th>
<th>m</th>
<th>m1cp</th>
<th>y</th>
<th>intr</th>
<th>infr</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1968:1</td>
<td>187.15</td>
<td>1036.22</td>
<td>81.18</td>
<td>6.84</td>
<td>5.44041</td>
<td>6.94333</td>
<td>1.92279</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>1968:2</td>
<td>190.63</td>
<td>1056.02</td>
<td>82.12</td>
<td>6.97</td>
<td>5.44732</td>
<td>5.42890</td>
<td>6.96226</td>
<td>1.94162</td>
<td>1.15127</td>
</tr>
<tr>
<td>3</td>
<td>1968:3</td>
<td>194.30</td>
<td>1068.72</td>
<td>82.80</td>
<td>6.98</td>
<td>5.45815</td>
<td>5.43908</td>
<td>6.97422</td>
<td>1.94305</td>
<td>0.82465</td>
</tr>
<tr>
<td>4</td>
<td>1968:4</td>
<td>198.55</td>
<td>1071.28</td>
<td>84.04</td>
<td>6.84</td>
<td>5.46492</td>
<td>5.44328</td>
<td>6.97661</td>
<td>1.92279</td>
<td>1.48648</td>
</tr>
<tr>
<td>5</td>
<td>1969:1</td>
<td>201.73</td>
<td>1084.15</td>
<td>84.97</td>
<td>7.32</td>
<td>5.45815</td>
<td>5.43908</td>
<td>6.97422</td>
<td>1.99061</td>
<td>1.10054</td>
</tr>
<tr>
<td>6</td>
<td>1969:2</td>
<td>203.18</td>
<td>1088.73</td>
<td>86.10</td>
<td>7.54</td>
<td>5.46375</td>
<td>5.45659</td>
<td>6.99277</td>
<td>2.02022</td>
<td>1.32112</td>
</tr>
<tr>
<td>7</td>
<td>1969:3</td>
<td>204.18</td>
<td>1091.90</td>
<td>87.49</td>
<td>7.70</td>
<td>5.45265</td>
<td>5.44774</td>
<td>6.99567</td>
<td>2.04122</td>
<td>1.60151</td>
</tr>
<tr>
<td>8</td>
<td>1969:4</td>
<td>206.10</td>
<td>1085.53</td>
<td>88.62</td>
<td>8.22</td>
<td>5.44917</td>
<td>5.43981</td>
<td>6.98982</td>
<td>2.10657</td>
<td>1.28331</td>
</tr>
</tbody>
</table>

The money demand equation is first estimated using OLS. The DW=4 option produces generalized Durbin-Watson statistics up to the fourth order. Their exact marginal probabilities (p-values) are also calculated with the DWPROB option. The Durbin-Watson test indicates positive first-order autocorrelation at, say, the 10% confidence level. You can use the Durbin-Watson table, which is available only for 1% and 5% significance points. The relevant upper (U) and lower (L) bounds are 1.731 and 1.471, respectively, at 5% significance level. However, the bounds test is inconvenient, since sometimes you may get the statistic in the inconclusive region while the interval between the upper and lower bounds becomes smaller with the increasing sample size. The PROC step follows:

```
title 'Partial Adjustment Money Demand Equation';
title2 'Quarterly Data - 1968:2 to 1983:4';
proc autoreg data=money outest=est covout;
model m = m1cp y intr infr / dw=4 dwprob;
run;
```

Output 8.5.2 OLS Estimation of the Partial Adjustment Money Demand Equation

Partial Adjustment Money Demand Equation
Quarterly Data - 1968:2 to 1983:4

The AUTOREG Procedure

<table>
<thead>
<tr>
<th>Dependent Variable</th>
<th>m</th>
</tr>
</thead>
</table>

**Ordinary Least Squares Estimates**

<table>
<thead>
<tr>
<th>SSE</th>
<th>0.00271902</th>
</tr>
</thead>
<tbody>
<tr>
<td>DFE</td>
<td>58</td>
</tr>
<tr>
<td>MSE</td>
<td>0.0000469</td>
</tr>
<tr>
<td>Root MSE</td>
<td>0.00685</td>
</tr>
<tr>
<td>SBC</td>
<td>-433.68709</td>
</tr>
<tr>
<td>AIC</td>
<td>-444.40276</td>
</tr>
<tr>
<td>MAE</td>
<td>0.00483389</td>
</tr>
<tr>
<td>AICC</td>
<td>-443.35013</td>
</tr>
<tr>
<td>MAPE</td>
<td>0.08888324</td>
</tr>
<tr>
<td>HQC</td>
<td>-440.18824</td>
</tr>
<tr>
<td>Regress R-Square</td>
<td>0.9546</td>
</tr>
<tr>
<td>Total R-Square</td>
<td>0.9546</td>
</tr>
</tbody>
</table>

**Durbin-Watson Statistics**

<table>
<thead>
<tr>
<th>Order</th>
<th>DW</th>
<th>Pr &lt; DW</th>
<th>Pr &gt; DW</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1.7355</td>
<td>0.0607</td>
<td>0.9393</td>
</tr>
<tr>
<td>2</td>
<td>2.1058</td>
<td>0.5519</td>
<td>0.4481</td>
</tr>
<tr>
<td>3</td>
<td>2.0286</td>
<td>0.5002</td>
<td>0.4998</td>
</tr>
<tr>
<td>4</td>
<td>2.2835</td>
<td>0.8880</td>
<td>0.1120</td>
</tr>
</tbody>
</table>

Note: Pr<DW is the p-value for testing positive autocorrelation, and Pr>DW is the p-value for testing negative autocorrelation.

**Parameter Estimates**

<table>
<thead>
<tr>
<th>Variable</th>
<th>DF</th>
<th>Estimate</th>
<th>Standard Error</th>
<th>t Value</th>
<th>Approx Pr &gt;</th>
<th>Variable Label</th>
</tr>
</thead>
<tbody>
<tr>
<td>Intercept</td>
<td>1</td>
<td>0.3084</td>
<td>0.2359</td>
<td>1.31</td>
<td>0.1963</td>
<td></td>
</tr>
</tbody>
</table>
The autoregressive model is estimated using the maximum likelihood method. Though the Durbin-Watson test statistic is calculated after correcting the autocorrelation, it should be used with care since the test based on this statistic is not justified theoretically. The PROC step follows:

```sas
proc autoreg data=money;
  model m = m1cp y intr infr / nlag=1 method=ml maxit=50;
  output out=a p=p pm=pm r=r rm=rm ucl=ucl lcl=lcl;
run;

proc print data=a(obs=8);
  var p pm r rm ucl lcl uclm lclm;
run;
```

A difference is shown between the OLS estimates in Output 8.5.2 and the AR(1)-ML estimates in Output 8.5.3. The estimated autocorrelation coefficient is significantly negative (−0.88345). Note that the negative coefficient of AR(1) should be interpreted as a positive autocorrelation.

Two predicted values are produced: predicted values computed for the structural model and predicted values computed for the full model. The full model includes both the structural and error-process parts. The predicted values and residuals are stored in the output data set A, as are the upper and lower 95% confidence limits for the predicted values. Part of the data set A is shown in Output 8.5.4. The first observation is missing since the explanatory variables, M1CP and INFR, are missing for the corresponding observation.

### Output 8.5.3 Estimated Partial Adjustment Money Demand Equation
Partial Adjustment Money Demand Equation
Quarterly Data - 1968:2 to 1983:4

#### The AUTOREG Procedure

<table>
<thead>
<tr>
<th>Lag</th>
<th>Coefficient</th>
<th>Standard Error</th>
<th>t Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>-0.126273</td>
<td>0.131393</td>
<td>-0.96</td>
</tr>
</tbody>
</table>

Algorithm converged.

#### Maximum Likelihood Estimates

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Estimate</th>
<th>Standard Error</th>
<th>t Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>SSE</td>
<td>0.00226719</td>
<td></td>
<td></td>
</tr>
<tr>
<td>MSE</td>
<td>0.0000398</td>
<td></td>
<td></td>
</tr>
<tr>
<td>SBC</td>
<td>-439.47665</td>
<td></td>
<td></td>
</tr>
<tr>
<td>MAE</td>
<td>0.00506044</td>
<td></td>
<td></td>
</tr>
<tr>
<td>MAPE</td>
<td>0.09302277</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Durbin-Watson</td>
<td>2.1778</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

#### Parameter Estimates

<table>
<thead>
<tr>
<th>Variable</th>
<th>DF</th>
<th>Estimate</th>
<th>Standard Error</th>
<th>t Value</th>
<th>Approx Pr &gt;</th>
<th>Variable Label</th>
</tr>
</thead>
<tbody>
<tr>
<td>Intercept</td>
<td>1</td>
<td>2.4121</td>
<td>0.4880</td>
<td>4.94</td>
<td>&lt;.0001</td>
<td></td>
</tr>
<tr>
<td>m1cp</td>
<td>1</td>
<td>0.4086</td>
<td>0.0908</td>
<td>4.50</td>
<td>&lt;.0001</td>
<td>Lagged M1/Current GDF</td>
</tr>
<tr>
<td>y</td>
<td>1</td>
<td>0.1509</td>
<td>0.0411</td>
<td>3.67</td>
<td>0.0005</td>
<td>Real GNP</td>
</tr>
<tr>
<td>intr</td>
<td>1</td>
<td>-0.1101</td>
<td>0.0159</td>
<td>-6.92</td>
<td>&lt;.0001</td>
<td>Yield on Corporate Bonds</td>
</tr>
<tr>
<td>infr</td>
<td>1</td>
<td>-0.006348</td>
<td>0.001834</td>
<td>-3.46</td>
<td>0.0010</td>
<td>Rate of Prices Changes</td>
</tr>
</tbody>
</table>

### Output 8.5.2

#### Output 8.5.2

The AUTOREG Procedure: The AUTOREG Procedure

<table>
<thead>
<tr>
<th>Lag Coefficient</th>
<th>Standard Error</th>
<th>t Value</th>
<th>Approx Pr &gt; t</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.8952</td>
<td>0.0439</td>
<td>20.38</td>
<td>&lt;.0001</td>
</tr>
<tr>
<td>0.0476</td>
<td>0.0122</td>
<td>3.89</td>
<td>0.0003</td>
</tr>
<tr>
<td>-0.0238</td>
<td>0.007933</td>
<td>-3.00</td>
<td>0.0040</td>
</tr>
<tr>
<td>-0.005646</td>
<td>0.001584</td>
<td>-3.56</td>
<td>0.0070</td>
</tr>
</tbody>
</table>

The estimated autocorrelation coefficient is significantly negative (−0.88345). Note that the negative coefficient of AR(1) should be interpreted as a positive autocorrelation.

Two predicted values are produced: predicted values computed for the structural model and predicted values computed for the full model. The full model includes both the structural and error-process parts. The predicted values and residuals are stored in the output data set A, as are the upper and lower 95% confidence limits for the predicted values. Part of the data set A is shown in Output 8.5.4. The first observation is missing since the explanatory variables, M1CP and INFR, are missing for the corresponding observation.
Example 8.6 Estimation of ARCH(2) Process

Stock returns show a tendency for small changes to be followed by small changes while large changes are followed by large changes. The plot of daily price changes of IBM common stock (Box and Jenkins; 1976, p. 527) is shown in Output 8.6.1. The time series look serially uncorrelated, but the plot makes us skeptical of their independence.

With the following DATA step, the stock (capital) returns are computed from the closing prices. To forecast the conditional variance, an additional 46 observations with missing values are generated.

title 'IBM Stock Returns (daily)';
title2 '29jun1959 - 30jun1960';

data ibm;
  infile datalines eof=last;
  r = dif( log( x ) );
  time = _n_-1;
  output;
return;
last:
  do i = 1 to 46;
    r = .;
    time + 1;
    output;
  end;
return;
datalines;
... more lines ...

proc sgplot data=ibm;

Output 8.5.4 Partial List of the Predicted Values

Partial Adjustment Money Demand Equation
Quarterly Data - 1968:2 to 1983:4
The simple ARCH(2) model is estimated using the AUTOREG procedure. The MODEL statement option GARCH=(Q=2) specifies the ARCH (2) model. The OUTPUT statement with the CEV= option produces the conditional variances $V$. The conditional variance and its forecast are calculated using parameter estimates:

$$ h_t = \theta + \alpha_1 \epsilon_{t-1}^2 + \alpha_2 \epsilon_{t-2}^2 $$

$$ \mathbb{E}(\epsilon_{t+1}^2 | \Psi_t) = \theta + \sum_{j=1}^{2} \alpha_j \mathbb{E}(\epsilon_{t+j}^2 | \Psi_t) $$

where $d > 1$. This model can be estimated as follows:

```plaintext
proc autoreg data=ibm maxit=50;
  model r = / noint garch=(q=2);
  output out=a cev=v;
run;
```

The parameter estimates for $\theta$, $\alpha_1$, and $\alpha_2$ are 0.00011, 0.04136, and 0.06976, respectively. The normality test indicates that the conditional normal distribution may not fully explain the leptokurtosis in the stock returns (Bollerslev; 1987).

The ARCH model estimates are shown in Output 8.6.2, and conditional variances are also shown in Output 8.6.3. The code that generates Output 8.6.3 is shown below.

```plaintext
data b; set a;
  length type $ 8.;
  if r ^= . then do;
    type = 'ESTIMATE'; output; end;
  else do;
    type = 'FORECAST'; output; end;
run;
proc sgplot data=b;
  series x=time y=v/group=type;
  reffline 254/ axis = x LINEATTRS = (pattern=ShortDash);
run;
```

Output 8.6.2 ARCH(2) Estimation Results
IBM Stock Returns (daily)
29jun1959 - 30jun1960
The AUTOREG Procedure

**Dependent Variable:** \( r \)

<table>
<thead>
<tr>
<th>Ordinary Least Squares Estimates</th>
</tr>
</thead>
<tbody>
<tr>
<td>SSE</td>
</tr>
<tr>
<td>MSE</td>
</tr>
<tr>
<td>SBC</td>
</tr>
<tr>
<td>MAE</td>
</tr>
<tr>
<td>MAPE</td>
</tr>
<tr>
<td>Durbin-Watson</td>
</tr>
</tbody>
</table>

**Root MSE:** 0.01125

**GARCH Estimates**

| SSE                        | 0.03214307 |
| MSE                        | 0.0001265  |
| Log Likelihood             | 781.017441 |
| SBC                        | -1545.4229 |
| MAE                        | 0.00805675 |
| MAPE                       | 100        |

**Uncond Var:** 0.0012632

**Total R-Square:** 0.0000

**NOTE:** No intercept term is used. R-squares are redefined.

Algorithm converged.

<table>
<thead>
<tr>
<th>GARCH Estimates</th>
</tr>
</thead>
<tbody>
<tr>
<td>MSE</td>
</tr>
<tr>
<td>Log Likelihood</td>
</tr>
<tr>
<td>SBC</td>
</tr>
<tr>
<td>MAE</td>
</tr>
<tr>
<td>MAPE</td>
</tr>
</tbody>
</table>

**Uncond Var:** 0.0012632

**Total R-Square:** 0.0000

**NOTE:** No intercept term is used. R-squares are redefined.

**Parameter Estimates**

| Variable | DF Estimate | Standard Error | t Value | Approx Pr > |t| |
|----------|-------------|----------------|---------|-------------|---|
| ARCH0    | 1           | 0.000112       | 7.6059E-6 | 14.76       | <.0001 |
| ARCH1    | 1           | 0.0414         | 0.0514  | 0.81        | 0.4208 |
| ARCH2    | 1           | 0.0698         | 0.0434  | 1.61        | 0.1082 |

Output 8.6.3 Conditional Variance for IBM Stock Prices
This example extends Example 8.6 to include more volatility models and to perform model selection and diagnostics.

Following is the data of daily IBM stock prices for the long period from 1962 to 2009.

```sas
data ibm_long;
  infile datalines;
  format date MMDDYY10.;
  input date:MMDDYY10. price_ibm;
  r = 100*diff( log( price_ibm ) );
datalines;
01/02/1962 2.68
01/03/1962 2.7
01/04/1962 2.67
01/05/1962 2.62
... more lines ...
08/12/2009 119.29
;
```

The time series of IBM returns is depicted graphically in Output 8.7.1.

**Output 8.7.1 IBM Stock Returns: Daily**
The following statements perform estimation of different kinds of GARCH-type models. First, ODS listing output that contains fit summary tables for each single model is captured by using an ODS OUTPUT statement with the appropriate ODS table name assigned to a new SAS data set. Along with these new data sets, another one that contains parameter estimates is created by using the OUTEST= option in AUTOREG statement.

/* Capturing ODS tables into SAS data sets */
ods output Autoreg.ar_1.FinalModel.FitSummary
   =fitsum_ar_1;
ods output Autoreg.arch_2.FinalModel.Results.FitSummary
   =fitsum_arch_2;
ods output Autoreg.garch_1_1.FinalModel.Results.FitSummary
   =fitsum_garch_1_1;
ods output Autoreg.st_garch_1_1.FinalModel.Results.FitSummary
   =fitsum_st_garch_1_1;
ods output Autoreg.ar_1_garch_1_1.FinalModel.Results.FitSummary
   =fitsum_ar_1_garch_1_1;
ods output Autoreg.igarch_1_1.FinalModel.Results.FitSummary
   =fitsum_igarch_1_1;
ods output Autoreg.garchm_1_1.FinalModel.Results.FitSummary
   =fitsum_garchm_1_1;
ods output Autoreg.egarch_1_1.FinalModel.Results.FitSummary
   =fitsum_egarch_1_1;
ods output Autoreg.qgarch_1_1.FinalModel.Results.FitSummary
   =fitsum_qgarch_1_1;
ods output Autoreg.tgarch_1_1.FinalModel.Results.FitSummary
   =fitsum_tgarch_1_1;
ods output Autoreg.pgarch_1_1.FinalModel.Results.FitSummary
   =fitsum_pgarch_1_1;
/* Estimating multiple GARCH-type models */
title "GARCH family";
proc autoreg data=ibm_long outest=garch_family;
   ar_1 : model r = / noint nlag=1 method=ml;
   arch_2 : model r = / noint garch=(q=2);
   st_garch_1_1 : model r = / noint garch=(p=1,q=1,type=stationary);
   ar_1_garch_1_1 : model r = / noint nlag=1 garch=(p=1,q=1);
   igarch_1_1 : model r = / noint garch=(p=1,q=1,type=integ,noint);
   egarch_1_1 : model r = / noint garch=(p=1,q=1,type=egarch);
   garchm_1_1 : model r = / noint garch=(p=1,q=1,mean=log);
   qgarch_1_1 : model r = / noint garch=(p=1,q=1,type=qgarch);
   tgarch_1_1 : model r = / noint garch=(p=1,q=1,type=tgarch);
   pgarch_1_1 : model r = / noint garch=(p=1,q=1,type=pgarch);
run;

The following statements print partial contents of the data set GARCH_FAMILY. The columns of interest are explicitly specified in the VAR statement.
/* Printing summary table of parameter estimates */
title "Parameter Estimates for Different Models";
proc print data=garch_family;
  var _MODEL_ _A_1 _AH_0 _AH_1 _AH_2 _GH_1 _AHQ_1 _AHT_1 _AHP_1 _THETA_ _LAMBDA_ _DELTA_
run;

These statements produce the results shown in **Output 8.7.2**.

**Output 8.7.2 GARCH-Family Estimation Results**

Parameter Estimates for Different Models

<table>
<thead>
<tr>
<th>Obs</th>
<th><em>MODEL</em></th>
<th>_A_1</th>
<th>_AH_0</th>
<th>_AH_1</th>
<th>_AH_2</th>
<th>_GH_1</th>
<th>_AHQ_1</th>
<th>_AHT_1</th>
<th>_AHP_1</th>
<th><em>THETA</em></th>
<th><em>LAMBDA</em></th>
<th><em>DELTA</em></th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>ar_1</td>
<td>0.017112</td>
<td></td>
<td>.</td>
<td>.</td>
<td>.</td>
<td>.</td>
<td>.</td>
<td>.</td>
<td>.</td>
<td>.</td>
<td>.</td>
</tr>
<tr>
<td>2</td>
<td>arch_2</td>
<td></td>
<td>1.60288</td>
<td>0.23235</td>
<td>0.21407</td>
<td>.</td>
<td>.</td>
<td>.</td>
<td>.</td>
<td>.</td>
<td>.</td>
<td>.</td>
</tr>
<tr>
<td>3</td>
<td>garch_1_1</td>
<td></td>
<td>0.02730</td>
<td>0.06984</td>
<td>.</td>
<td>0.92294</td>
<td>.</td>
<td>.</td>
<td>.</td>
<td>.</td>
<td>.</td>
<td>.</td>
</tr>
<tr>
<td>4</td>
<td>st_garch_1_1</td>
<td></td>
<td>0.02831</td>
<td>0.06913</td>
<td>.</td>
<td>0.92260</td>
<td>.</td>
<td>.</td>
<td>.</td>
<td>.</td>
<td>.</td>
<td>.</td>
</tr>
<tr>
<td>5</td>
<td>ar_1_garch_1_1</td>
<td>-0.005995</td>
<td>0.02734</td>
<td>0.06994</td>
<td>.</td>
<td>0.92282</td>
<td>.</td>
<td>.</td>
<td>.</td>
<td>.</td>
<td>.</td>
<td>.</td>
</tr>
<tr>
<td>6</td>
<td>igarch_1_1</td>
<td></td>
<td>.</td>
<td>0.00000</td>
<td>.</td>
<td>1.00000</td>
<td>.</td>
<td>.</td>
<td>.</td>
<td>.</td>
<td>.</td>
<td>.</td>
</tr>
<tr>
<td>7</td>
<td>egarch_1_1</td>
<td></td>
<td>0.01541</td>
<td>0.12882</td>
<td>.</td>
<td>0.98914</td>
<td>.</td>
<td>.</td>
<td>-0.41706</td>
<td>.</td>
<td>.</td>
<td>.</td>
</tr>
<tr>
<td>8</td>
<td>garchm_1_1</td>
<td></td>
<td>0.02897</td>
<td>0.07139</td>
<td>.</td>
<td>0.92079</td>
<td>.</td>
<td>.</td>
<td>.</td>
<td>.</td>
<td>0.094773</td>
<td>.</td>
</tr>
<tr>
<td>9</td>
<td>qgarch_1_1</td>
<td></td>
<td>0.00120</td>
<td>0.05792</td>
<td>.</td>
<td>0.93458</td>
<td>0.66461</td>
<td>.</td>
<td>.</td>
<td>.</td>
<td>.</td>
<td>.</td>
</tr>
<tr>
<td>10</td>
<td>tgarch_1_1</td>
<td></td>
<td>0.02706</td>
<td>0.02966</td>
<td>.</td>
<td>0.92765</td>
<td>0.074815</td>
<td>.</td>
<td>.</td>
<td>.</td>
<td>.</td>
<td>.</td>
</tr>
<tr>
<td>11</td>
<td>pgarch_1_1</td>
<td></td>
<td>0.01623</td>
<td>0.06724</td>
<td>.</td>
<td>0.93952</td>
<td>.</td>
<td>.</td>
<td>0.43445</td>
<td>0.53625</td>
<td>.</td>
<td>.</td>
</tr>
</tbody>
</table>

The table shown in **Output 8.7.2** is convenient for reporting the estimation result of multiple models and their comparison.

The following statements merge multiple tables that contain fit statistics for each estimated model, leaving only columns of interest, and rename them.

/* Merging ODS output tables and extracting AIC and SBC measures */
data sbc_aic;
  set fitsum_arch_2 fitsum_garch_1_1 fitsum_st_garch_1_1
  fitsum_ar_1 fitsum_ar_1_garch_1_1 fitsum_igarch_1_1
  fitsum_egarch_1_1 fitsum_garchm_1_1 fitsum_qgarch_1_1;
  keep Model SBC AIC;
  if Label1="SBC" then do; SBC=input(cValue1,BEST12.4); end;
  if Label2="SBC" then do; SBC=input(cValue2,BEST12.4); end;
  if Label1="AIC" then do; AIC=input(cValue1,BEST12.4); end;
  if Label2="AIC" then do; AIC=input(cValue2,BEST12.4); end;
  if not (SBC=.) then output;
runc;

Next, sort the models by one of the criteria, for example, by AIC:

/* Sorting data by AIC criterion */
proc sort data=sbc_aic;
  by AIC;
runc;

Finally, print the sorted data set:

title "Selection Criteria for Different Models";
proc print data=sbc_aic;
  format _NUMERIC_ BEST12.4;
runc;

The result is given in **Output 8.7.3**.

**Output 8.7.3 GARCH-Family Model Selection on the Basis of AIC and SBC**

Selection Criteria for Different Models

<table>
<thead>
<tr>
<th>Obs</th>
<th>Model</th>
<th>SBC</th>
<th>AIC</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>pgarch_1_1</td>
<td>42907.7292</td>
<td>42870.7722</td>
</tr>
</tbody>
</table>
According to the smaller-is-better rule for the information criteria, the PGARCH(1,1) model is the leader by AIC while the EGARCH(1,1) is the model of choice according to SBC.

Next, check whether the power GARCH model is misspecified, especially, if dependence exists in the standardized residuals that correspond to the assumed independently and identically distributed (iid) disturbance. The following statements reestimate the power GARCH model and use the BDS test to check the independence of the standardized residuals.

```plaintext
proc autoreg data=ibm_long;
   model r = / noint garch=(p=1,q=1,type=pgarch) BDS=(Z=SR,D=2.0);
run;
```

The partial results listing of the preceding statements is given in Output 8.7.4.

**Output 8.7.4 Diagnostic Checking of the PGARCH(1,1) Model**

<table>
<thead>
<tr>
<th>Selection Criteria for Different Models</th>
</tr>
</thead>
<tbody>
<tr>
<td>BDS Test for Independence</td>
</tr>
<tr>
<td>Distance</td>
</tr>
<tr>
<td>----------</td>
</tr>
<tr>
<td>2.0000</td>
</tr>
<tr>
<td>3</td>
</tr>
<tr>
<td>4</td>
</tr>
<tr>
<td>5</td>
</tr>
<tr>
<td>6</td>
</tr>
<tr>
<td>7</td>
</tr>
<tr>
<td>8</td>
</tr>
<tr>
<td>9</td>
</tr>
<tr>
<td>10</td>
</tr>
<tr>
<td>11</td>
</tr>
<tr>
<td>12</td>
</tr>
<tr>
<td>13</td>
</tr>
<tr>
<td>14</td>
</tr>
<tr>
<td>15</td>
</tr>
<tr>
<td>16</td>
</tr>
<tr>
<td>17</td>
</tr>
<tr>
<td>18</td>
</tr>
<tr>
<td>19</td>
</tr>
<tr>
<td>20</td>
</tr>
</tbody>
</table>

The results in **Output 8.7.4** indicate that when embedded size is greater than 9, you fail to reject the null hypothesis of independence at 1% significance level, which is a good indicator that the PGARCH model is not misspecified.
The AUTOREG Procedure: The AUTOREG Procedure

Example 8.8 Illustration of ODS Graphics

This example illustrates the use of ODS GRAPHICS. This is a continuation of the section Forecasting Autoregressive Error Models.

These graphical displays are requested by specifying the ODS GRAPHICS statement. For information about the graphs available in the AUTOREG procedure, see the section ODS Graphics.

The following statements show how to generate ODS GRAPHICS plots with the AUTOREG procedure. In this case, all plots are requested using the ALL option in the PROC AUTOREG statement, in addition to the ODS GRAPHICS statement. The plots are displayed in Output 8.8.1 through Output 8.8.8. Note: these plots can be viewed in the Autoreg.Model.FitDiagnosticPlots category by selecting View Results.

data a;
  ul = 0; ull = 0;
  do time = -10 to 36;
    u = 1.3 * ul - .5 * ull + 2*rannor(12346);
    y = 10 + .5 * time + u;
    if time > 0 then output;
    ull = ul; ul = u;
  end;
run;

data b;
  y = .;
  do time = 37 to 46; output; end;
run;

data b;
  merge a b;
  by time;
run;

proc autoreg data=b all plots(unpack);
  model y = time / nlag=2 method=ml;
  output out=p p=yhat pm=ytrend
    lcl=lcl ucl=ucl;
run;

Output 8.8.1 Residuals Plot
Output 8.8.2 Predicted versus Actual Plot

Output 8.8.3 Autocorrelation of Residuals Plot
Output 8.8.4 Partial Autocorrelation of Residuals Plot

Output 8.8.5 Inverse Autocorrelation of Residuals Plot
Output 8.8.6 Tests for White Noise Residuals Plot

Output 8.8.7 Q-Q Plot of Residuals
Output 8.8.8 Histogram of Residuals


Hildreth, C. and Lu, J. Y. (1960), *Demand Relations with Autocorrelated Disturbances*, Technical Report 276, Michigan State University Agricultural Experiment Station, East Lansing, MI.


