SAS/ETS® 13.2 User's Guide
The FORECAST Procedure

This document is an individual chapter from *SAS/ETS® 13.2 User's Guide*.

The correct bibliographic citation for the complete manual is as follows: SAS Institute Inc. 2014. *SAS/ETS® 13.2 User's Guide*. Cary, NC: SAS Institute Inc.

SAS Institute Inc., SAS Campus Drive, Cary, North Carolina 27513.

August 2014

Chapter 16
The FORECAST Procedure

Contents

Overview: FORECAST Procedure

The FORECAST procedure is obsolete and has been superseded by newer SAS/ETS procedures. These newer procedures provide more powerful and flexible versions of the forecasting methods that PROC FORECAST uses, and they also provide additional forecasting methods that are not available in PROC FORECAST.

The FORECAST procedure is still available for use. However, before choosing to use PROC FORECAST, consider the following alternatives:

- For forecasting by using exponential smoothing methods or Winters method, consider using the ESM procedure. The models that can be selected by the PROC FORECAST options METHOD=EXPO, METHOD=WINTERS, and METHOD=ADDWINTERS are provided by PROC ESM, which also provides additional forecasting methods that PROC FORECAST does not support. Unlike PROC FORECAST, the ESM procedure optimizes the smoothing weights for the forecasting model based on the data. Also unlike PROC FORECAST, the ESM procedure can automatically select the form of exponential smoothing model that is most appropriate for your data. For information about forecasting with PROC ESM, see Chapter 14, "The ESM Procedure."

- For forecasting by using time trend models with autoregressive errors, consider using the AUTOREG procedure. The models that can be selected by the PROC FORECAST options METHOD=STEPAR and TREND= can be fit and forecast using PROC AUTOREG, which also allows the inclusion of additional predictor variables in the forecasting model. For information about PROC AUTOREG, see Chapter 8, "The AUTOREG Procedure."

- For forecasting by using more general and sophisticated time series models, consider using the UCM procedure, which fits and forecasts unobserved components models that are not available in PROC FORECAST. Using UCM models, you can fit and forecast much more complex data patterns than you can by using the simple methods that PROC FORECAST provides. Unlike PROC FORECAST, the UCM procedure can also model and forecast the effect of independent predictor variables. For information about PROC UCM, see Chapter 34, "The UCM Procedure."

- For forecasting by using ARIMA models and the Box-Jenkins methodology, consider using the ARIMA procedure. PROC ARIMA identifies, fits, and forecasts general autoregressive integrated moving average models, optionally incorporating transfer function models for the effects of independent predictor variables. (As a special case, you can use seasonal ARMA models for forecasting seasonal series for which the Winters and additive Winters methods might be used.) PROC ARIMA also provides features for automatically identifying the specific ARIMA model that is most appropriate for the data. ARIMA and ARIMAX models are not available in PROC FORECAST. For information about PROC ARIMA, see Chapter 7, "The ARIMA Procedure."

- For forecasting multivariate time series, where two or more related variables need to be forecast jointly, consider using the VARMAX procedure or the SSM procedure. PROC VARMAX fits and forecasts vector autoregressive moving average models, optionally incorporating multivariate transfer function models for the effects of independent predictor variables. For information about PROC VARMAX, see Chapter 35, "The VARMAX Procedure." PROC SSM fits and forecasts general linear state space models. The general state space model encompasses most of the other forecasting models that are mentioned in this section, and it enables generalizations that can model time series data patterns of almost any type and complexity. For information about PROC SSM, see Chapter 27, "The SSM Procedure."

- For forecasting both the future expectation and future volatility or risk, consider using the AUTOREG procedure or the VARMAX procedure. PROC AUTOREG can fit and forecast many types of GARCH models of time-varying volatility, while also fitting and forecasting future expected values of the dependent variable. PROC VARMAX supports multivariate GARCH models. For information about PROC AUTOREG, see Chapter 8, "The AUTOREG Procedure." For information about PROC VARMAX, see Chapter 35, "The VARMAX Procedure."

If you decide to use PROC FORECAST instead of these newer alternatives, this chapter explains the features of the FORECAST procedure.

The FORECAST procedure provides a quick and automatic way to generate forecasts for many time series in one step. The procedure can forecast hundreds of series at a time, with the series organized into separate variables or across BY groups. PROC FORECAST uses extrapolative forecasting methods where the forecasts for a series are functions only of time and past values of the series, not of other variables.

You can use the following forecasting methods. For each of these methods, you can specify linear, quadratic, or no trend.

- The stepwise autoregressive method is used by default. This method combines time trend regression with an autoregressive model and uses a stepwise method to select the lags to use for the autoregressive process.

- The exponential smoothing method produces a time trend forecast. However, in fitting the trend, the parameters are allowed to change gradually over time, and earlier observations are given exponentially declining weights. Single, double, and triple exponential smoothing are supported, depending on whether no trend, linear trend, or quadratic trend, respectively, is specified. Holt two-parameter linear exponential smoothing is supported as a special case of the Holt-Winters method without seasons.

- The Winters method (also called Holt-Winters) combines a time trend with multiplicative seasonal factors to account for regular seasonal fluctuations in a series. Like the exponential smoothing method, the Winters method allows the parameters to change gradually over time, with earlier observations given exponentially declining weights. You can also specify the additive version of the Winters method, which uses additive instead of multiplicative seasonal factors. When seasonal factors are omitted, the Winters method reduces to the Holt two-parameter version of double exponential smoothing.

The FORECAST procedure writes the forecasts and confidence limits to an output data set. It can also write parameter estimates and fit statistics to an output data set. The FORECAST procedure does not produce printed output.

PROC FORECAST is an extrapolation procedure useful for producing practical results efficiently. However, in the interest of speed, PROC FORECAST uses some shortcuts that cause some statistical results (such as confidence limits) to be only approximate. For many time series, the FORECAST procedure, with appropriately chosen methods and weights, can yield satisfactory results. Other SAS/ETS procedures can produce better forecasts.

Getting Started: FORECAST Procedure

To use PROC FORECAST, specify the input and output data sets and the number of periods to forecast in the PROC FORECAST statement, and then list the variables to forecast in a VAR statement.

For example, suppose you have monthly data on the sales of some product in a data set named PAST, as shown in Figure 16.1, and you want to forecast sales for the next 10 months.

Figure 16.1 Example Data Set PAST

Obs	date	sales
1	JUL89	9.5161
2	AUG89	9.6994
3	SEP89	9.2644
4	OCT89	9.6837
5	NOV89	10.0784
6	DEC89	9.9005
7	JAN90	10.2375
8	FEB90	10.6940
9	MAR90	10.6290
10	APR90	11.0332
11	MAY90	11.0270
12	JUN90	11.4165
13	JUL90	11.2918
14	AUG90	11.3475
15	SEP90	11.2913
16	OCT90	11.3771
17	NOV90	11.5457
18	DEC90	11.6433
19	JAN91	11.9293
20	FEB91	11.9752
21	MAR91	11.9283
22	APR91	11.8985
23	MAY91	12.0419
24	JUN91	12.3537
25	JUL91	12.4546

The following statements forecast 10 observations for the variable SALES by using the default STEPAR method and write the results to the output data set PRED:

```
proc forecast data=past lead=10 out=pred;
   var sales;
run;
```

The following statements use the PRINT procedure to print the data set PRED:

```
proc print data=pred;
run;
```

The PROC PRINT listing of the forecast data set PRED is shown in Figure 16.2.

Figure 16.2 Forecast Data Set PRED

Obs	_TYPE_	_LEAD_	sales
1	FORECAST	1	12.6205
2	FORECAST	2	12.7665
3	FORECAST	3	12.9020
4	FORECAST	4	13.0322
5	FORECAST	5	13.1595
6	FORECAST	6	13.2854
7	FORECAST	7	13.4105
8	FORECAST	8	13.5351
9	FORECAST	9	13.6596
10	FORECAST	10	13.7840

Giving Dates to Forecast Values

Normally, your input data set has an ID variable that gives dates to the observations, and you want the forecast observations to have dates also. Usually, the ID variable has SAS date values. (See Chapter 3, "Working with Time Series Data," for information about using SAS date and datetime values.) The ID statement specifies the identifying variable.

If the ID variable contains SAS date or datetime values, the INTERVAL= option should be used on the PROC FORECAST statement to specify the time interval between observations. (See Chapter 4, "Date Intervals, Formats, and Functions," for more information about time intervals.) The FORECAST procedure uses the INTERVAL= option to generate correct dates for forecast observations.

The data set PAST, shown in Figure 16.1, has monthly observations and contains an ID variable DATE with SAS date values identifying each observation. The following statements produce the same forecast as the preceding example and also include the ID variable DATE in the output data set. Monthly SAS date values are extrapolated for the forecast observations.

```
proc forecast data=past interval=month lead=10 out=pred;
   var sales;
   id date;
run;
```

Computing Confidence Limits

Depending on the output options specified, multiple observations are written to the OUT= data set for each time period. The different parts of the results are contained in the VAR statement variables in observations identified by the character variable _TYPE_ and by the ID variable.

For example, the following statements use the OUTLIMIT option to write forecasts and 95% confidence limits for the variable SALES to the output data set PRED. This data set is printed with the PRINT procedure.

```
proc forecast data=past interval=month lead=10
              out=pred outlimit;
   var sales;
   id date;
run;

proc print data=pred;
run;
```

The output data set PRED is shown in Figure 16.3.

Figure 16.3 Output Data Set

Obs	date	_TYPE_	_LEAD_	sales
1	AUG91	FORECAST	1	12.6205
2	AUG91	L95	1	12.1848
3	AUG91	U95	1	13.0562
4	SEP91	FORECAST	2	12.7665
5	SEP91	L95	2	12.2808
6	SEP91	U95	2	13.2522
7	OCT91	FORECAST	3	12.9020
8	OCT91	L95	3	12.4001
9	OCT91	U95	3	13.4039
10	NOV91	FORECAST	4	13.0322
11	NOV91	L95	4	12.5223
12	NOV91	U95	4	13.5421
13	DEC91	FORECAST	5	13.1595
14	DEC91	L95	5	12.6435
15	DEC91	U95	5	13.6755
16	JAN92	FORECAST	6	13.2854
17	JAN92	L95	6	12.7637
18	JAN92	U95	6	13.8070
19	FEB92	FORECAST	7	13.4105
20	FEB92	L95	7	12.8830
21	FEB92	U95	7	13.9379
22	MAR92	FORECAST	8	13.5351
23	MAR92	L95	8	13.0017
24	MAR92	U95	8	14.0686
25	APR92	FORECAST	9	13.6596
26	APR92	L95	9	13.1200
27	APR92	U95	9	14.1993
28	MAY92	FORECAST	10	13.7840
29	MAY92	L95	10	13.2380
30	MAY92	U95	10	14.3301

Form of the OUT= Data Set

The OUT= data set PRED, shown in Figure 16.3, contains three observations for each of the 10 forecast periods. Each of these three observations has the same value of the ID variable DATE, the SAS date value for the month and year of the forecast.

The three observations for each forecast period have different values of the variable _TYPE_. For the _TYPE_=FORECAST observation, the value of the variable SALES is the forecast value for the period indicated by the DATE value. For the _TYPE_=L95 observation, the value of the variable SALES is the lower limit of the 95% confidence interval for the forecast. For the _TYPE_=U95 observation, the value of the variable SALES is the upper limit of the 95% confidence interval.

You can control the types of observations written to the OUT= data set with the PROC FORECAST statement options OUTLIMIT, OUTRESID, OUTACTUAL, OUT1STEP, OUTSTD, OUTFULL, and OUTALL. For example, the OUTFULL option outputs the confidence limit values, the one-step-ahead predictions, and the actual data, in addition to the forecast values. See the sections "Syntax: FORECAST Procedure" on page 913 and "OUTEST= Data Set" on page 931 for more information.

Plotting Forecasts

The forecasts, confidence limits, and actual values can be plotted on the same graph with the SGPLOT procedure. Use the appropriate output control options in the PROC FORECAST statement to include in the OUT= data set the series you want to plot. Use the _TYPE_ variable in the SGPLOT procedure GROUP option to separate the observations for the different plots.

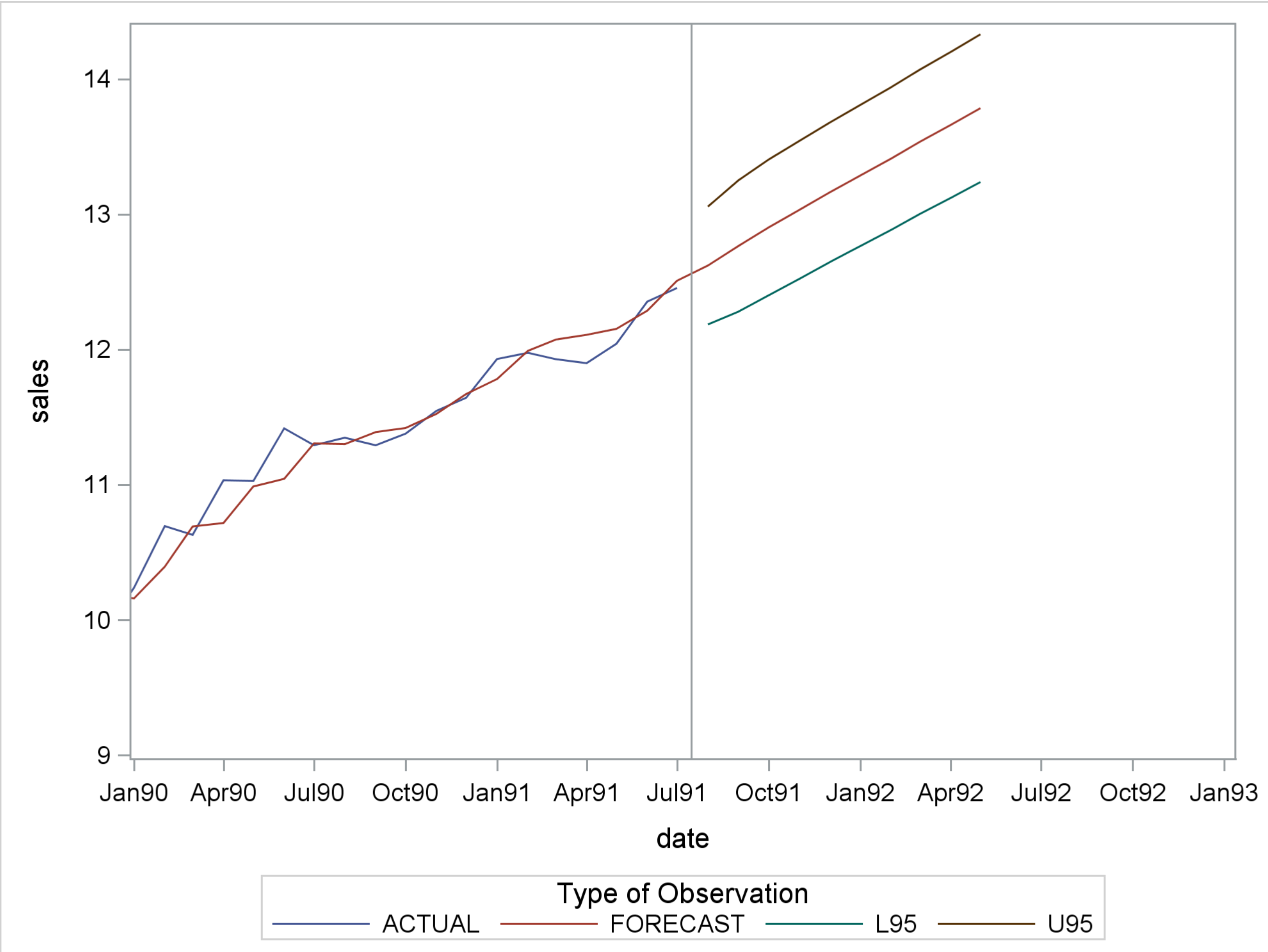
The OUTFULL option is used in the following statements. The resulting output data set contains the actual and predicted values, as well as the upper and lower 95% confidence limits.

```
proc forecast data=past interval=month lead=10
              out=pred outfull;
   id date;
   var sales;
run;

proc sgplot data=pred;
   series x=date y=sales / group=_type_ lineattrs=(pattern=1);
   xaxis values=('1jan90'd to '1jan93'd by qtr);
   refline '15jul91'd / axis=x;
run;
```

The _TYPE_ variable is used in the SGPLOT procedure's PLOT statement to make separate plots over time for each type of value. A reference line marks the start of the forecast period. (See *SAS/GRAPH: Reference* for more information about using PROC SGPLOT.) The WHERE statement restricts the range of the actual data shown in the plot. In this example, the variable SALES has monthly data from July 1989 through July 1991, but only the data for 1990 and 1991 are shown in Figure 16.4.

Figure 16.4 Plot of Forecast with Confidence Limits



Plotting Residuals

You can plot the residuals from the forecasting model by using PROC SGPLOT and a WHERE statement.

1. Use the OUTRESID option or the OUTALL option in the PROC FORECAST statement to include the residuals in the output data set.

2. Use a WHERE statement to specify the observation type of 'RESIDUAL' in the PROC GPLOT code.

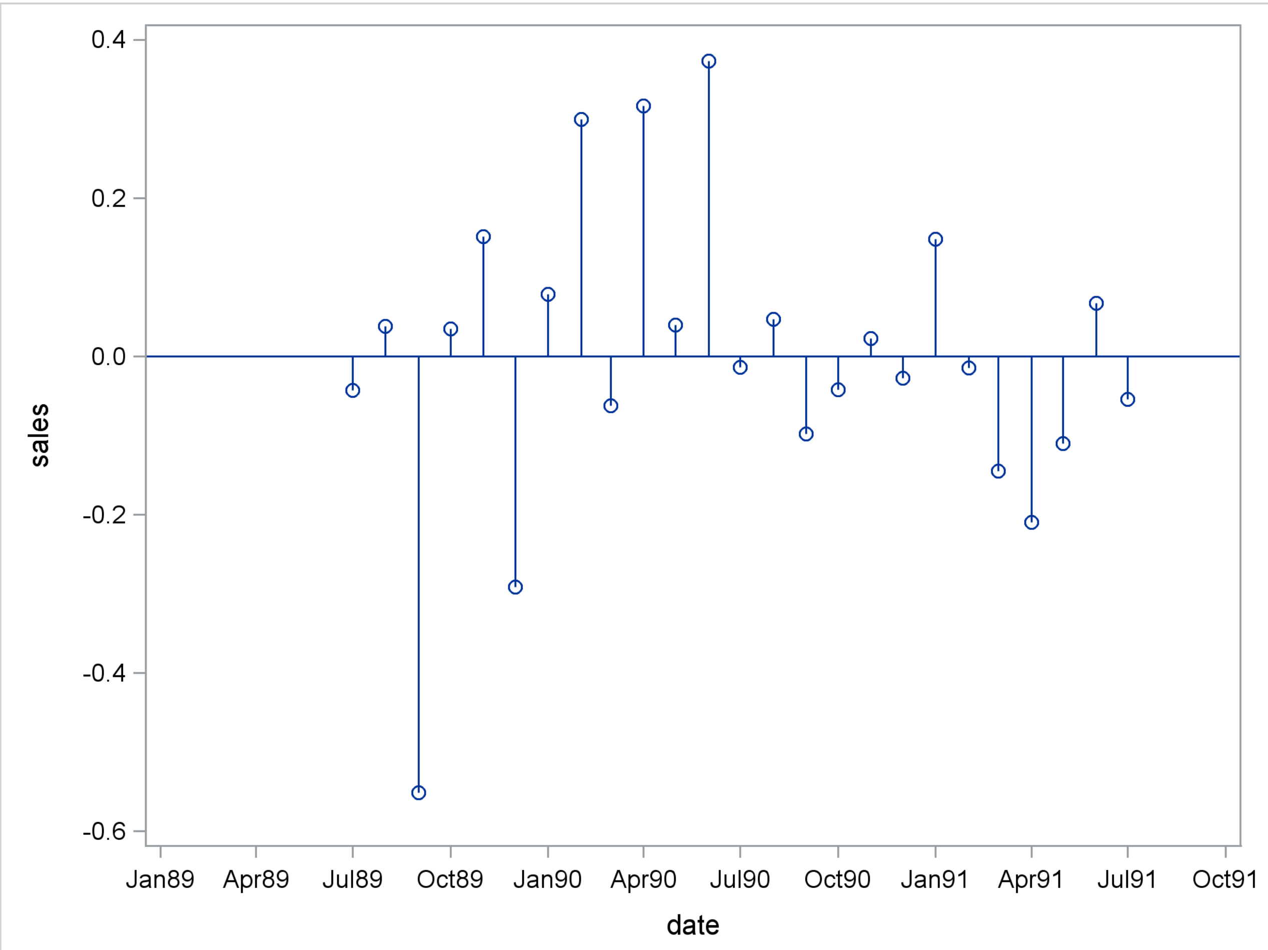
The following statements add the OUTRESID option to the preceding example and plot the residuals:

```
proc forecast data=past interval=month lead=10
              out=pred outfull outresid;
   id date;
   var sales;
run;
```

```
proc sgplot data=pred;
   where _type_='RESIDUAL';
   needle x=date y=sales / markers;
   xaxis values=('1jan89'd to '1oct91'd by qtr);
run;
```

The plot of residuals is shown in Figure 16.5.

Figure 16.5 Plot of Residuals



Model Parameters and Goodness-of-Fit Statistics

You can write the parameters of the forecasting models used, as well as statistics that measure how well the forecasting models fit the data, to an output SAS data set by using the OUTEST= option. The options OUTFITSTATS, OUTESTTHEIL, and OUTESTALL control what goodness-of-fit statistics are added to the OUTEST= data set.

For example, the following statements add the OUTEST= and OUTFITSTATS options to the previous example to create the output statistics data set EST for the results of the default stepwise autoregressive forecasting method:

```
proc forecast data=past interval=month lead=10
              out=pred outfull outresid
              outest=est outfitstats;
   id date;
   var sales;
run;

proc print data=est;
run;
```

The PRINT procedure prints the OUTEST= data set, as shown in Figure 16.6.

Figure 16.6 The OUTEST= Data Set for STEPAR Method

Obs	_TYPE_	date	sales
1	N	JUL91	25
2	NRESID	JUL91	25
3	DF	JUL91	22
4	SIGMA	JUL91	0.2001613
5	CONSTANT	JUL91	9.4348822
6	LINEAR	JUL91	0.1242648
7	AR1	JUL91	0.5206294
8	AR2	JUL91	.
9	AR3	JUL91	.
10	AR4	JUL91	.
11	AR5	JUL91	.
12	AR6	JUL91	.
13	AR7	JUL91	.
14	AR8	JUL91	.
15	SST	JUL91	21.28342
16	SSE	JUL91	0.8793714
17	MSE	JUL91	0.0399714
18	RMSE	JUL91	0.1999286
19	MAPE	JUL91	1.2280089
20	MPE	JUL91	-0.050139
21	MAE	JUL91	0.1312115
22	ME	JUL91	-0.001811
23	MAXE	JUL91	0.3732328
24	MINE	JUL91	-0.551605
25	MAXPE	JUL91	3.2692294
26	MINPE	JUL91	-5.954022
27	RSQUARE	JUL91	0.9586828
28	ADJRSQ	JUL91	0.9549267
29	RW_RSQ	JUL91	0.2657801
30	ARSQ	JUL91	0.9474145
31	APC	JUL91	0.044768
32	AIC	JUL91	-77.68559
33	SBC	JUL91	-74.02897
34	CORR	JUL91	0.9791313

In the OUTEST= data set, the DATE variable contains the ID value of the last observation in the data set used to fit the forecasting model. The variable SALES contains the statistic indicated by the value of the _TYPE_ variable. The _TYPE_=N, NRESID, and DF observations contain, respectively, the number of observations read from the data set, the number of nonmissing residuals used to compute the goodness-of-fit statistics, and the number of nonmissing observations minus the number of parameters used in the forecasting model.

The observation that has _TYPE_=SIGMA contains the estimate of the standard deviation of the one-step prediction error computed from the residuals. The _TYPE_=CONSTANT and _TYPE_=LINEAR observations contain the coefficients of the time trend regression. The _TYPE_=AR1, AR2, ..., AR8 observations contain the estimated autoregressive parameters. A missing autoregressive parameter indicates that the autoregressive term at that lag was not retained in the model by the stepwise model selection method. (See the section "STEPAR Method" on page 921 for more information.)

The other observations in the OUTEST= data set contain various goodness-of-fit statistics that measure how well the forecasting model used fits the given data. See the section "OUTEST= Data Set" on page 931 for details.

Controlling the Forecasting Method

The METHOD= option controls which forecasting method is used. The TREND= option controls the degree of the time trend model used. For example, the following statements produce forecasts of SALES as in the preceding example but use the double exponential smoothing method instead of the default STEPAR method:

```
proc forecast data=past interval=month lead=10
              method=expo trend=2
              out=pred outfull outresid
              outest=est outfitstats;
   var sales;
   id date;
run;

proc print data=est;
run;
```

The PRINT procedure prints the OUTEST= data set for the EXPO method, as shown in Figure 16.7.

Figure 16.7 The OUTEST= Data Set for METHOD=EXPO

Obs	_TYPE_	date	sales
1	N	JUL91	25
2	NRESID	JUL91	25
3	DF	JUL91	23
4	WEIGHT	JUL91	0.1055728
5	S1	JUL91	11.427657
6	S2	JUL91	10.316473
7	SIGMA	JUL91	0.2545069
8	CONSTANT	JUL91	12.538841
9	LINEAR	JUL91	0.1311574
10	SST	JUL91	21.28342
11	SSE	JUL91	1.4897965
12	MSE	JUL91	0.0647738
13	RMSE	JUL91	0.2545069
14	MAPE	JUL91	1.9121204
15	MPE	JUL91	-0.816886
16	MAE	JUL91	0.2101358
17	ME	JUL91	-0.094941
18	MAXE	JUL91	0.3127332
19	MINE	JUL91	-0.460207
20	MAXPE	JUL91	2.9243781
21	MINPE	JUL91	-4.967478
22	RSQUARE	JUL91	0.930002
23	ADJRSQ	JUL91	0.9269586
24	RW_RSQ	JUL91	-0.243886
25	ARSQ	JUL91	0.9178285
26	APC	JUL91	0.0699557
27	AIC	JUL91	-66.50591
28	SBC	JUL91	-64.06816
29	CORR	JUL91	0.9772418

See the section "Syntax: FORECAST Procedure" on page 913 for other options that control the forecasting method. See the section "Introduction to Forecasting Methods" on page 908 and the section "Forecasting Methods" on page 921 for an explanation of the different forecasting methods.

Introduction to Forecasting Methods

This section briefly introduces the forecasting methods used by the FORECAST procedure. See textbooks on forecasting and see the section "Forecasting Methods" on page 921 for more detailed discussions of forecasting methods.

The FORECAST procedure combines three basic models to fit time series:

- time trend models for long-term, deterministic change
- autoregressive models for short-term fluctuations

- seasonal models for regular seasonal fluctuations

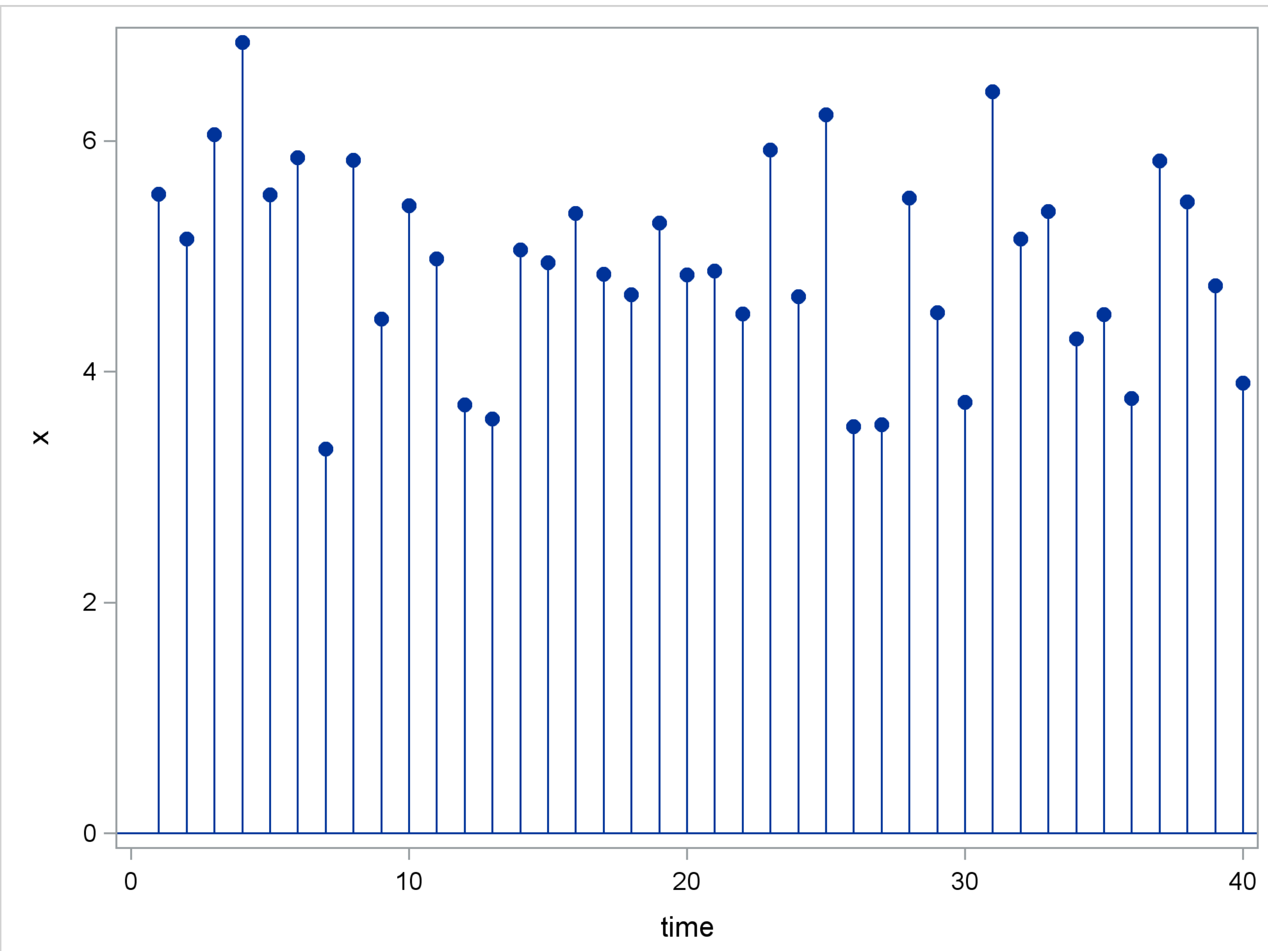
Two approaches to time series modeling and forecasting are *time trend models* and *time series methods*.

Time Trend Models

Time trend models assume that there is some permanent deterministic pattern across time. These models are best suited to data that are not dominated by random fluctuations.

Examining a graphical plot of the time series you want to forecast is often very useful in choosing an appropriate model. The simplest case of a time trend model is one in which you assume the series is a constant plus purely random fluctuations that are independent from one time period to the next. Figure 16.8 shows how such a time series might look.

Figure 16.8 Time Series without Trend



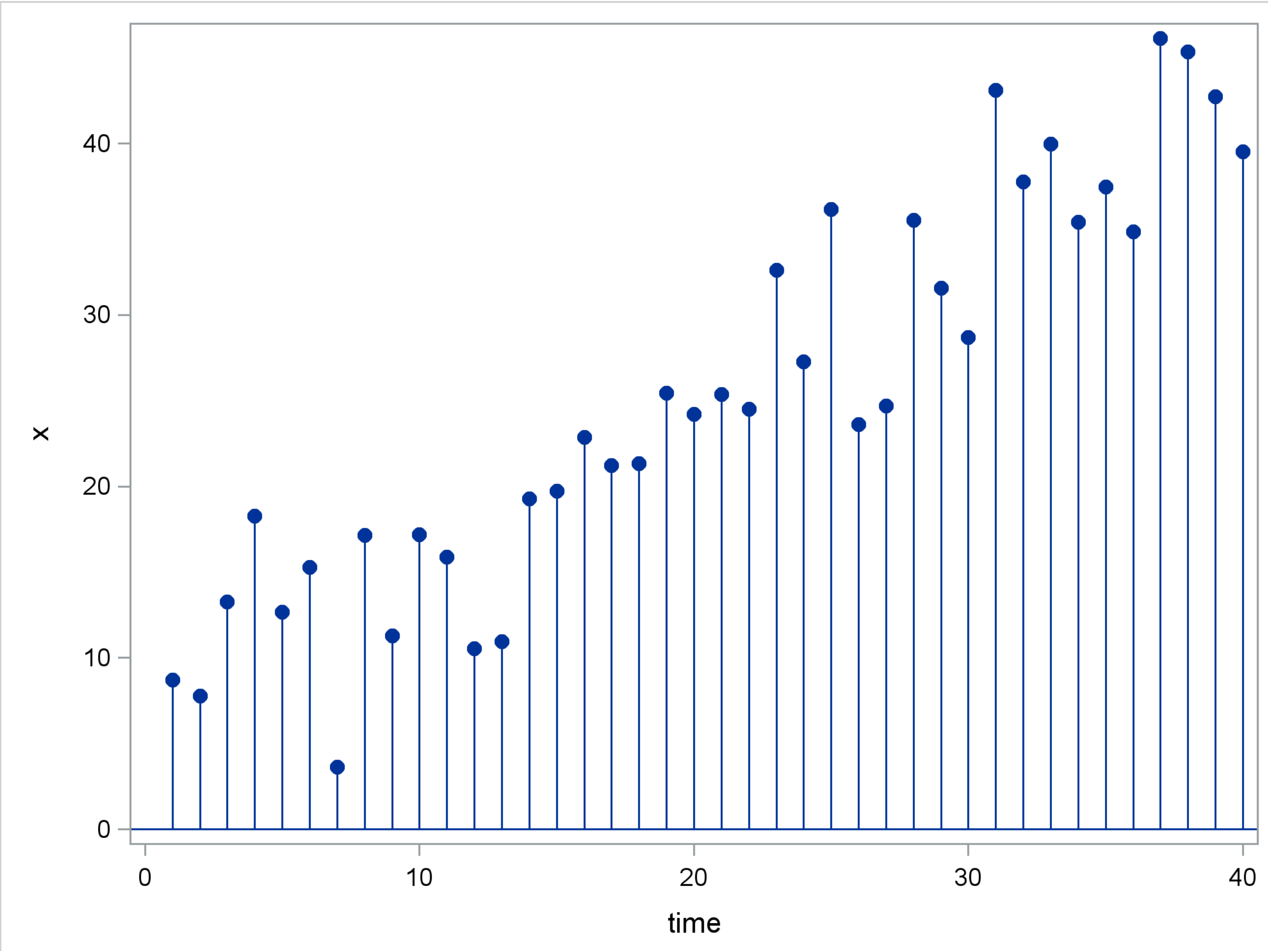
The x_t values are generated according to the equation

$$x_t = b_0 + \epsilon_t$$

where ϵ_t is an independent, zero-mean, random error and b_0 is the true series mean.

Suppose that the series exhibits growth over time, as shown in Figure 16.9.

Figure 16.9 Time Series with Linear Trend



A linear model is appropriate for this data. For the linear model, assume the x_t values are generated according to the equation

$$x_t = b_0 + b_1 t + \epsilon_t$$

The linear model has two parameters. The predicted values for the future are the points on the estimated line. The extension of the polynomial model to three parameters is the quadratic (which forms a parabola). This allows for a constantly changing slope, where the x_t values are generated according to the equation

$$x_t = b_0 + b_1 t + b_2 t^2 + \epsilon_t$$

PROC FORECAST can fit three types of time trend models: constant, linear, and quadratic. For other kinds of trend models, other SAS procedures can be used.

Exponential smoothing fits a time trend model by using a smoothing scheme in which the weights decline geometrically as you go backward in time. The forecasts from exponential smoothing are a time trend, but the trend is based mostly on the recent observations instead of on all the observations equally. How well exponential smoothing works as a forecasting method depends on choosing a good smoothing weight for the series.

To specify the exponential smoothing method, use the METHOD=EXPO option. Single exponential smoothing produces forecasts with a constant trend (that is, no trend). Double exponential smoothing produces forecasts with a linear trend, and triple exponential smoothing produces a quadratic trend. Use the TREND= option with the METHOD=EXPO option to select single, double, or triple exponential smoothing.

The time trend model can be modified to account for regular seasonal fluctuations of the series about the trend. To capture seasonality, the trend model includes a seasonal parameter for each season. Seasonal models can be additive or multiplicative.

$$x_t = b_0 + b_1 t + s(t) + \epsilon_t \qquad \text{(additive)}$$

$$x_t = (b_0 + b_1 t)s(t) + \epsilon_t \qquad \text{(multiplicative)}$$

where *s(t)* is the seasonal parameter for the season that corresponds to time *t*.

The Winters method is similar to exponential smoothing, but it includes seasonal factors. The Winters method can use either additive or multiplicative seasonal factors. Like exponential smoothing, good results with the Winters method depend on choosing good smoothing weights for the series to be forecast.

To specify the multiplicative or additive versions of the Winters method, use the METHOD=WINTERS or METHOD=ADDWINTERS options, respectively. To specify seasonal factors to include in the model, use the SEASONS= option.

Many observed time series do not behave like constant, linear, or quadratic time trends. However, you can partially compensate for the inadequacies of the trend models by fitting time series models to the departures from the time trend, as described in the following sections.

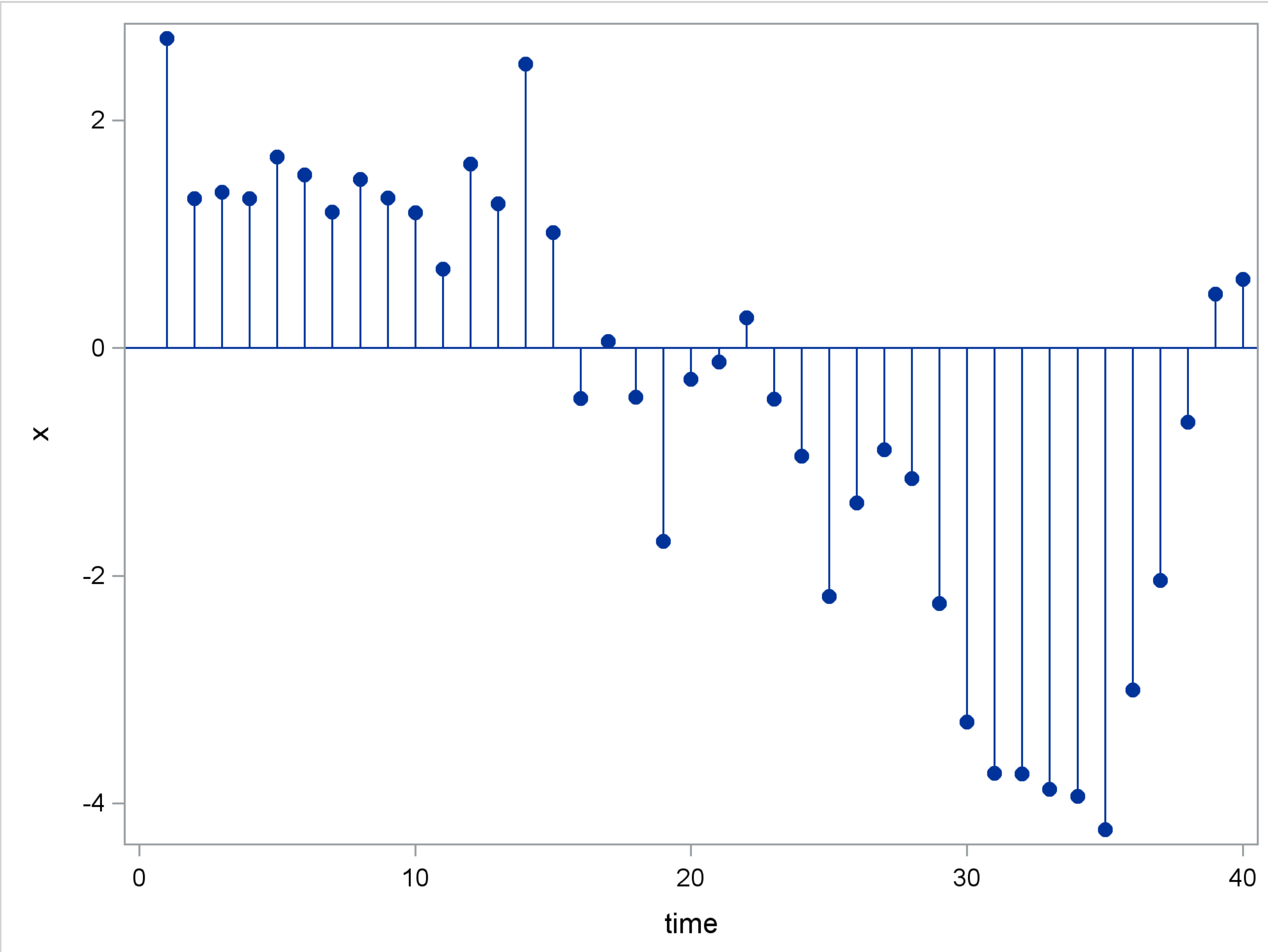
Time Series Methods

Time series models assume the future value of a variable to be a linear function of past values. If the model is a function of past values for a finite number of periods, it is an *autoregressive model* and is written as follows:

$$x_t = a_0 + a_1 x_{t-1} + a_2 x_{t-2} + \ldots + a_p x_{t-p} + \epsilon_t$$

The coefficients a_i are *autoregressive parameters*. One of the simplest cases of this model is the random walk, where the series dances around in purely random jumps. This is illustrated in Figure 16.10.

Figure 16.10 Random Walk Series



The x_t values are generated by the equation

$$x_t = x_{t-1} + \epsilon_t$$

In this type of model, the best forecast of a future value is the present value. However, with other autoregressive models, the best forecast is a weighted sum of recent values. Pure autoregressive forecasts always damp down to a constant (assuming the process is stationary).

Autoregressive time series models can also be used to predict seasonal fluctuations.

Combining Time Trend with Autoregressive Models

Trend models are suitable for capturing long-term behavior, whereas autoregressive models are more appropriate for capturing short-term fluctuations. One approach to forecasting is to combine a deterministic time trend model with an autoregressive model.

The *stepwise autoregressive method* (STEPAR method) combines a time trend regression with an autoregressive model for departures from trend. The combined time trend and autoregressive model is written as follows:

$$x_t = b_0 + b_1 t + b_2 t^2 + u_t$$

$$u_t = a_1 u_{t-1} + a_2 u_{t-2} + \ldots + a_p u_{t-p} + \epsilon_t$$

The autoregressive parameters included in the model for each series are selected by a stepwise regression procedure, so that autoregressive parameters are included only at those lags at which they are statistically significant.

The stepwise autoregressive method is fully automatic. Unlike the exponential smoothing and Winters methods, it does not depend on choosing smoothing weights. However, the STEPAR method assumes that the long-term trend is stable; that is, the time trend regression is fit to the whole series with equal weights for the observations.

The stepwise autoregressive model is used when you specify the METHOD=STEPAR option or do not specify any METHOD= option. To select a constant, linear, or quadratic trend for the time-trend part of the model, use the TREND= option.

Syntax: FORECAST Procedure

The following statements are used with PROC FORECAST:

PROC FORECAST *options* ;
  BY *variables* ;
  ID *variables* ;
  VAR *variables* ;

Functional Summary

Table 16.1 summarizes the statements and options that control the FORECAST procedure.

Table 16.1 FORECAST Functional Summary

Description	Statement	Option
Statements		
specify model and data set options	PROC FORECAST	
specify BY-group processing	BY	
identify observations	ID	
specify the variables to forecast	VAR	
Input Data Set Options		
specify the input SAS data set	PROC FORECAST	DATA=
specify frequency of the input time series	PROC FORECAST	INTERVAL=
specify increment between observations	PROC FORECAST	INTPER=
specify seasonality	PROC FORECAST	SEASONS=
specify number of periods in a season	PROC FORECAST	SINTPER=
treat zeros at beginning of series as missing	PROC FORECAST	ZEROMISS

Description	Statement	Option
Output Data Set Options		
specify the number of periods ahead to forecast	PROC FORECAST	LEAD=
name output data set to contain the forecasts	PROC FORECAST	OUT=
write actual values to the OUT= data set	PROC FORECAST	OUTACTUAL
write confidence limits to the OUT= data set	PROC FORECAST	OUTLIMIT
write residuals to the OUT= data set	PROC FORECAST	OUTRESID
write standard errors of the forecasts to the OUT= data set	PROC FORECAST	OUTSTD
write one-step-ahead predicted values to the OUT= data set	PROC FORECAST	OUT1STEP
write predicted, actual, and confidence limit values to the OUT= data set	PROC FORECAST	OUTFULL
write all available results to the OUT= data set	PROC FORECAST	OUTALL
specify significance level for confidence limits	PROC FORECAST	ALPHA=
control the alignment of SAS date values	PROC FORECAST	ALIGN=
Parameters and Statistics Output Data Set Options		
write parameter estimates and goodness-of-fit statistics to an output data set	PROC FORECAST	OUTEST=
write additional statistics to OUTEST= data set	PROC FORECAST	OUTESTALL
write Theil statistics to OUTEST= data set	PROC FORECAST	OUTESTTHEIL
write forecast accuracy statistics to OUTEST= data set	PROC FORECAST	OUTFITSTATS
Forecasting Method Options		
specify the forecasting method	PROC FORECAST	METHOD=
specify degree of the time trend model	PROC FORECAST	TREND=
specify smoothing weights	PROC FORECAST	WEIGHT=
specify order of the autoregressive model	PROC FORECAST	AR=
specify significance level for adding AR lags	PROC FORECAST	SLENTRY=
specify significance level for keeping AR lags	PROC FORECAST	SLSTAY=
start forecasting before the end of data	PROC FORECAST	START=
specify criterion for judging singularity	PROC FORECAST	SINGULAR=
limit number of error or warning messages	PROC FORECAST	MAXERRORS=

Description	Statement	Option
Initializing Smoothed Values		
specify number of beginning values to use in calculating starting values	PROC FORECAST	NSTART=
specify number of beginning values to use in calculating initial seasonal parameters	PROC FORECAST	NSSTART=
specify starting values for constant term	PROC FORECAST	ASTART=
specify starting values for linear trend	PROC FORECAST	BSTART=
specify starting values for the quadratic trend	PROC FORECAST	CSTART=

PROC FORECAST Statement

PROC FORECAST *options* **;**

The following options can be specified in the PROC FORECAST statement:

ALIGN=*option*

controls the alignment of SAS dates used to identify output observations. The ALIGN= option allows the following values: BEGINNING | BEG | B, MIDDLE | MID | M, and ENDING | END | E. BEGINNING is the default.

ALPHA=*value*

specifies the significance level to use in computing the confidence limits of the forecast. The value of the ALPHA= option must be between 0.01 and 0.99. You should use only two digits for the ALPHA= option because PROC FORECAST rounds the value to the nearest percent (ALPHA=0.101 is the same as ALPHA=0.10). The default is ALPHA=0.05, which produces 95% confidence limits.

AR=*n*

NLAGS=*n*

specifies the maximum order of the autoregressive model. The AR= option is valid only for METHOD=STEPAR. The default value of *n* depends on the INTERVAL= option and on the number of observations in the DATA= data set. See the section "STEPAR Method" on page 921 for details.

ASTART=*value*

ASTART=*(value . . .)*

specifies starting values for the constant term for the exponential smoothing, Winters, and additive Winters methods. This option is ignored if METHOD=STEPAR. The values specified are associated with the variables in the VAR statement in the order in which the variables are listed. See the section "Starting Values for EXPO, WINTERS, and ADDWINTERS Methods" on page 928 for details.

BSTART=*value*

BSTART=*(value . . .)*

specifies starting values for the linear trend for the exponential smoothing, Winters, and additive Winters methods. The values specified are associated with the variables in the VAR statement in the order in which the variables are listed. This option is ignored if METHOD=STEPAR or TREND=1. See the section "Starting Values for EXPO, WINTERS, and ADDWINTERS Methods" on page 928 for details.

CSTART=*value*

CSTART=*(value . . .)*

specifies starting values for the quadratic trend for the exponential smoothing, Winters, and additive Winters methods. The values specified are associated with the variables in the VAR statement in the order in which the variables are listed. This option is ignored if METHOD=STEPAR or TREND=1 or 2. See the section "Starting Values for EXPO, WINTERS, and ADDWINTERS Methods" on page 928 for details.

DATA=*SAS-data-set*

names the SAS data set that contains the input time series for the procedure to forecast. If the DATA= option is not specified, the most recently created SAS data set is used.

INTERVAL=*interval*

specifies the frequency of the input time series. For example, if the input data set consists of quarterly observations, then INTERVAL=QTR should be used. See Chapter 4, "Date Intervals, Formats, and Functions," for more details about the intervals available.

INTPER=*n*

when the INTERVAL= option is not used, specifies an increment (other than 1) to use in generating the values of the ID variable for the forecast observations in the output data set.

LEAD=*n*

specifies the number of periods ahead to forecast. The default is LEAD=12.

The LEAD= value is relative to the last observation in the input data set and not to the end of a particular series. Thus, if a series has missing values at the end, the actual number of forecasts computed for that series will be greater than the LEAD= value.

MAXERRORS=*n*

limits the number of warning and error messages produced during the execution of the procedure to the specified value. The default is MAXERRORS=50.

This option is particularly useful in BY-group processing where it can be used to suppress the recurring messages.

METHOD=*method-name*

specifies the method to use to model the series and generate the forecasts.

METHOD=STEPAR specifies the stepwise autoregressive method.

METHOD=EXPO specifies the exponential smoothing method.

METHOD=WINTERS specifies the Holt-Winters exponentially smoothed trend-seasonal method.

METHOD=ADDWINTERS specifies the additive seasonal factors variant of the Winters method.

For more information, see the section "Forecasting Methods" on page 921. The default is METHOD=STEPAR.

NSTART=*n*

NSTART=MAX

specifies the number of beginning values of the series to use in calculating starting values for the trend parameters in the exponential smoothing, Winters, and additive Winters methods. This option is ignored if METHOD=STEPAR.

For METHOD=EXPO, *n* beginning values of the series are used in forming the exponentially smoothed values S1, S2, and S3, where *n* is the value of the NSTART= option. The parameters are initialized by fitting a time trend regression to the first *n* nonmissing values of the series.

For METHOD=WINTERS or METHOD=ADDWINTERS, *n* beginning complete seasonal cycles are used to compute starting values for the trend parameters. For example, for monthly data the seasonal cycle is one year, and NSTART=2 specifies that the first 24 observations at the beginning of each series are used for the time trend regression used to calculate starting values.

When NSTART=MAX is specified, all the observations are used. The default for METHOD=EXPO is NSTART=8; the default for METHOD=WINTERS or METHOD=ADDWINTERS is NSTART=2. See the section "Starting Values for EXPO, WINTERS, and ADDWINTERS Methods" on page 928 for details.

NSSTART=*n*

NSSTART=MAX

specifies the number of beginning values of the series to use in calculating starting values for seasonal parameters for METHOD=WINTERS or METHOD=ADDWINTERS. The seasonal parameters are initialized by averaging over the first *n* values of the series for each season, where *n* is the value of the NSSTART= option. When NSSTART=MAX is specified, all the observations are used.

If NSTART= is specified, but NSSTART= is not, NSSTART= defaults to the value specified for NSTART=. If neither NSTART= nor NSSTART= is specified, then the default is NSSTART=2. This option is ignored if METHOD=STEPAR or METHOD=EXPO. See the section "Starting Values for EXPO, WINTERS, and ADDWINTERS Methods" on page 928 for details.

OUT=*SAS-data-set*

names the output data set to contain the forecasts. If the OUT= option is not specified, the data set is named by using the DATA*n* convention. See the section "OUTEST= Data Set" on page 931 for details.

OUTACTUAL

writes the actual values to the OUT= data set.

OUTALL

provides all the output control options (OUTLIMIT, OUT1STEP, OUTACTUAL, OUTRESID, and OUTSTD).

OUTEST=*SAS-data-set*

names an output data set to contain the parameter estimates and goodness-of-fit statistics. When the OUTEST= option is not specified, the parameters and goodness-of-fit statistics are not stored. See the section "OUTEST= Data Set" on page 931 for details.

OUTESTALL
writes additional statistics to the OUTEST= data set. This option is the same as specifying both OUTESTTHEIL and OUTFITSTATS.

OUTESTTHEIL
writes Theil forecast accuracy statistics to the OUTEST= data set.

OUTFITSTATS
writes various R-square-type forecast accuracy statistics to the OUTEST= data set.

OUTFULL
provides OUTACTUAL, OUT1STEP, and OUTLIMIT output control options in addition to the forecast values.

OUTLIMIT
writes the forecast confidence limits to the OUT= data set.

OUTRESID
writes the residuals (when available) to the OUT= data set.

OUTSTD
writes the standard errors of the forecasts to the OUT= data set.

OUT1STEP
writes the one-step-ahead predicted values to the OUT= data set.

SEASONS=*interval*

SEASONS= (*interval1 [interval2 [interval3]]* **)**

SEASONS=*n*

SEASONS= (*n1 [n2 [n3]]* **)**
specifies the seasonality for seasonal models. The *interval* can be QTR, MONTH, DAY, or HOUR, or multiples of these (for example, QTR2, MONTH2, MONTH3, MONTH4, MONTH6, HOUR2, HOUR3, HOUR4, HOUR6, HOUR8, and HOUR12).

Alternatively, seasonality can be specified by giving the length of the seasonal cycles. For example, SEASONS=3 means that every group of three observations forms a seasonal cycle. The SEASONS= option is valid only for METHOD=WINTERS or METHOD=ADDWINTERS. See the section "Specifying Seasonality" on page 928 for details.

SINGULAR=*value*
gives the criterion for judging singularity. The default depends on the precision of the computer that you run SAS programs on.

SINTPER=*m*

SINTPER= (*m1 [m2 [m3]]* **)**
specifies the number of periods to combine in forming a season. For example, SEASONS=3 SINTPER=2 specifies that each group of two observations forms a season and that the seasonal cycle repeats every six observations. The SINTPER= option is valid only when the SEASONS= option is used. See the section "Specifying Seasonality" on page 928 for details.

SLENTRY=*value*

controls the significance levels for entry of autoregressive parameters in the STEPAR method. The value of the SLENTRY= option must be between 0 and 1. The default is SLENTRY=0.2. See the section "STEPAR Method" on page 921 for details.

SLSTAY=*value*

controls the significance levels for removal of autoregressive parameters in the STEPAR method. The value of the SLSTAY= option must be between 0 and 1. The default is SLSTAY=0.05. See the section "STEPAR Method" on page 921 for details.

START=*n*

uses the first *n* observations to fit the model and begins forecasting with the *n* +1 observation.

TREND=*n*

specifies the degree of the time trend model. The value of the TREND= option must be 1, 2, or 3. TREND=1 selects the constant trend model; TREND=2 selects the linear trend model; and TREND=3 selects the quadratic trend model. The default is TREND=2, except for METHOD=EXPO, for which the default is TREND=3.

WEIGHT=*w*

WEIGHT= (*w1 [w2 [w3]]* **)**

specifies the smoothing weights for the EXPO, WINTERS, and ADDWINTERS methods. For the EXPO method, only one weight can be specified. For the WINTERS or ADDWINTERS method, *w1* gives the weight for updating the constant component, *w2* gives the weight for updating the linear and quadratic trend components, and *w3* gives the weight for updating the seasonal component. The *w2* and *w3* values are optional. Each value in the WEIGHT= option must be between 0 and 1. For default values, see the section "EXPO Method" on page 922 and the section "WINTERS Method" on page 924.

ZEROMISS

treats zeros at the beginning of a series as missing values. For example, a product can be introduced at a date after the date of the first observation in the data set, and the sales variable for the product can be recorded as zero for the observations prior to the introduction date. The ZEROMISS option says to treat these initial zeros as missing values.

BY Statement

BY *variables* **;**

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

ID Statement

ID *variables* **;**

The first variable listed in the ID statement identifies observations in the input and output data sets. Usually, the first ID variable is a SAS date or datetime variable. Its values are interpreted and extrapolated according to the values of the INTERVAL= option. See the section "Data Periodicity and Time Intervals" on page 920 for details.

If more than one ID variable is specified in the ID statement, only the first is used to identify the observations; the rest are just copied to the OUT= data set and will have missing values for forecast observations.

VAR Statement

VAR *variables* **;**

The VAR statement specifies the variables in the input data set that you want to forecast. If no VAR statement is specified, the procedure forecasts all numeric variables except the ID and BY variables.

Details: FORECAST Procedure

Missing Values

The treatment of missing values varies by method. For METHOD=STEPAR, missing values are tolerated in the series; the autocorrelations are estimated from the available data and tapered, if necessary. For the EXPO, WINTERS, and ADDWINTERS methods, missing values after the start of the series are replaced with one-step-ahead predicted values, and the predicted values are applied to the smoothing equations. For the WINTERS method, negative or zero values are treated as missing.

Data Periodicity and Time Intervals

The INTERVAL= option is used to establish the frequency of the time series. For example, INTERVAL=MONTH specifies that each observation in the input data set represents one month. If INTERVAL=MONTH2, each observation represents two months. Thus, there is a two-month time interval between each pair of successive observations, and the data frequency is bimonthly.

See Chapter 4, "Date Intervals, Formats, and Functions," for details about the interval values supported.

The INTERVAL= option is used together with the ID statement to fully describe the observations that make up the time series. The first variable specified in the ID statement is used to identify the observations. Usually, SAS date or datetime values are used for this variable. PROC FORECAST uses the ID variable in the following ways:

- to validate the data periodicity. When the INTERVAL= option is specified, the ID variable is used to check the data and verify that successive observations have valid ID values that correspond to successive time intervals. When the INTERVAL= option is not used, PROC FORECAST verifies that the ID values are nonmissing and in ascending order. A warning message is printed when an invalid ID value is found in the input data set.

- to check for gaps in the input observations. For example, if INTERVAL=MONTH and an input observation for January 1970 is followed by an observation for April 1970, there is a gap in the input data, with two observations omitted. When a gap in the input data is found, a warning message is printed, and PROC FORECAST processes missing values for each omitted input observation.
- to label the forecast observations in the output data set. The values of the ID variable for the forecast observations after the end of the input data set are extrapolated according to the frequency specifications of the INTERVAL= option. If the INTERVAL= option is not specified, the ID variable is extrapolated by incrementing the ID variable value for the last observation in the input data set by the INTPER= value, if specified, or by one.

The ALIGN= option controls the alignment of SAS dates. See Chapter 4, “Date Intervals, Formats, and Functions,” for more information.

Forecasting Methods

This section explains the forecasting methods used by PROC FORECAST.

STEPAR Method

In the STEPAR method, PROC FORECAST first fits a time trend model to the series and takes the difference between each value and the estimated trend. (This process is called *detrending*.) Then, the remaining variation is fit by using an autoregressive model.

The STEPAR method fits the autoregressive process to the residuals of the trend model by using a backwards-stepping method to select parameters. Because the trend and autoregressive parameters are fit in sequence rather than simultaneously, the parameter estimates are not optimal in a statistical sense. However, the estimates are usually close to optimal, and the method is computationally inexpensive.

The STEPAR Algorithm

The STEPAR method consists of the following computational steps:

1. Fit the trend model as specified by the TREND= option by using ordinary least-squares regression. This step detrends the data. The default trend model for the STEPAR method is TREND=2, a linear trend model.
2. Take the residuals from step 1 and compute the autocovariances to the number of lags specified by the NLAGS= option.
3. Regress the current values against the lags, using the autocovariances from step 2 in a Yule-Walker framework. Do not bring in any autoregressive parameter that is not significant at the level specified by the SLENTRY= option. (The default is SLENTRY=0.20.) Do not bring in any autoregressive parameter that results in a nonpositive-definite Toeplitz matrix.
4. Find the autoregressive parameter that is least significant. If the significance level is greater than the SLSTAY= value, remove the parameter from the model. (The default is SLSTAY=0.05.) Continue this process until only significant autoregressive parameters remain. If the OUTEST= option is specified, write the estimates to the OUTEST= data set.

5. Generate the forecasts by using the estimated model and output to the OUT= data set. Form the confidence limits by combining the trend variances with the autoregressive variances.

Missing values are tolerated in the series; the autocorrelations are estimated from the available data and tapered if necessary.

This method requires at least three passes through the data: two passes to fit the model and a third pass to initialize the autoregressive process and write to the output data set.

Default Value of the NLAGS= Option

If the NLAGS= option is not specified, the default value of the NLAGS= option is chosen based on the data frequency specified by the INTERVAL= option and on the number of observations in the input data set, if this can be determined in advance. (PROC FORECAST cannot determine the number of input observations before reading the data when a BY statement or a WHERE statement is used or if the data are from a tape format SAS data set or external database. The NLAGS= value must be fixed before the data are processed.)

If the INTERVAL= option is specified, the default NLAGS= value includes lags for up to three years plus one, subject to the maximum of 13 lags or one-third of the number of observations in your data set, whichever is less. If the number of observations in the input data set cannot be determined, the maximum NLAGS= default value is 13. If the INTERVAL= option is not specified, the default is NLAGS=13 or one-third the number of input observations, whichever is less.

If the Toeplitz matrix formed by the autocovariance matrix at a given step is not positive definite, the maximal number of autoregressive lags is reduced.

For example, for INTERVAL=QTR, the default is NLAGS=13 (that is, $4 \times 3 + 1$) provided that there are at least 39 observations. The NLAGS= option default is always at least 3.

EXPO Method

Exponential smoothing is used when the METHOD=EXPO option is specified. The term *exponential smoothing* is derived from the computational scheme developed by Brown and others (Brown and Meyer 1961; Brown 1962). Estimates are computed with updating formulas that are developed across time series in a manner similar to smoothing.

The EXPO method fits a trend model such that the most recent data are weighted more heavily than data in the early part of the series. The weight of an observation is a geometric (exponential) function of the number of periods that the observation extends into the past relative to the current period. The weight function is

$$w_\tau = \omega(1-\omega)^{t-\tau}$$

where τ is the observation number of the past observation, t is the current observation number, and ω is the weighting constant specified with the WEIGHT= option.

You specify the model with the TREND= option as follows:

- TREND=1 specifies single exponential smoothing (a constant model)
- TREND=2 specifies double exponential smoothing (a linear trend model)
- TREND=3 specifies triple exponential smoothing (a quadratic trend model)

Updating Equations

The single exponential smoothing operation is expressed by the formula

$$S_t = \omega x_t + (1-\omega)S_{t-1}$$

where S_t is the smoothed value at the current period, t is the time index of the current period, and x_t is the current actual value of the series. The smoothed value S_t is the forecast of x_{t+1} and is calculated as the smoothing constant ω times the value of the series, x_t, in the current period plus $(1-\omega)$ times the previous smoothed value S_{t-1}, which is the forecast of x_t computed at time $t-1$.

Double and triple exponential smoothing are derived by applying exponential smoothing to the smoothed series, obtaining smoothed values as follows:

$$S_t^{[2]} = \omega S_t + (1-\omega)S_{t-1}^{[2]}$$

$$S_t^{[3]} = \omega S_t^{[2]} + (1-\omega)S_{t-1}^{[3]}$$

Missing values after the start of the series are replaced with one-step-ahead predicted values, and the predicted value is then applied to the smoothing equations.

The polynomial time trend parameters CONSTANT, LINEAR, and QUAD in the OUTEST= data set are computed from S_T, $S_T^{[2]}$, and $S_T^{[3]}$, the final smoothed values at observation *T*, the last observation used to fit the model. In the OUTEST= data set, the values of S_T, $S_T^{[2]}$, and $S_T^{[3]}$ are identified by _TYPE_=S1, _TYPE_=S2, and _TYPE_=S3, respectively.

Smoothing Weights

Exponential smoothing forecasts are forecasts for an integrated moving-average process; however, the weighting parameter is specified by the user rather than estimated from the data. Experience has shown that good values for the WEIGHT= option are between 0.05 and 0.3. As a general rule, smaller smoothing weights are appropriate for series with a slowly changing trend, while larger weights are appropriate for volatile series with a rapidly changing trend. If unspecified, the weight defaults to $(1 - 0.8^{1/trend})$, where *trend* is the value of the TREND= option. This produces defaults of WEIGHT=0.2 for TREND=1, WEIGHT=0.10557 for TREND=2, and WEIGHT=0.07168 for TREND=3.

The ESM procedure can be used to forecast time series by using exponential smoothing with smoothing weights that are optimized automatically. See Chapter 14, "The ESM Procedure."

The Time Series Forecasting System provides for exponential smoothing models and enables you to either specify or optimize the smoothing weights. See Chapter 46, "Getting Started with Time Series Forecasting," for details.

Confidence Limits

The confidence limits for exponential smoothing forecasts are calculated as they would be for an exponentially weighted time trend regression, using the simplifying assumption of an infinite number of observations. The variance estimate is computed by using the mean square of the unweighted one-step-ahead forecast residuals.

More detailed descriptions of the forecast computations can be found in Montgomery and Johnson (1976); Brown (1962).

WINTERS Method

The WINTERS method uses updating equations similar to exponential smoothing to fit parameters for the model

$$x_t = (a + bt)s(t) + \epsilon_t$$

where a and b are the trend parameters and the function $s(t)$ selects the seasonal parameter for the season that corresponds to time t.

The WINTERS method assumes that the series values are positive. If negative or zero values are found in the series, a warning is printed and the values are treated as missing.

The preceding standard WINTERS model uses a linear trend. However, PROC FORECAST can also fit a version of the WINTERS method that uses a quadratic trend. When TREND=3 is specified for METHOD=WINTERS, PROC FORECAST fits the following model:

$$x_t = (a + bt + ct^2)s(t) + \epsilon_t$$

The quadratic trend version of the Winters method is often unstable, and its use is not recommended.

When TREND=1 is specified, the following constant trend version is fit:

$$x_t = as(t) + \epsilon_t$$

The default for the WINTERS method is TREND=2, which produces the standard linear trend model.

Seasonal Factors

The notation $s(t)$ represents the selection of the seasonal factor used for different time periods. For example, if INTERVAL=DAY and SEASONS=MONTH, there are 12 seasonal factors, one for each month in the year, and the time index t is measured in days. For any observation, t is determined by the ID variable and $s(t)$ selects the seasonal factor for the month that t falls in. For example, if t is 9 February 1993 then $s(t)$ is the seasonal parameter for February.

When there are multiple seasons specified, $s(t)$ is the product of the parameters for the seasons. For example, if SEASONS=(MONTH DAY), then $s(t)$ is the product of the seasonal parameter for the month that corresponds to the period t and the seasonal parameter for the day of the week that corresponds to period t. When the SEASONS= option is not specified, the seasonal factors $s(t)$ are not included in the model. See the section "Specifying Seasonality" on page 928 for more information about specifying multiple seasonal factors.

Updating Equations

This section shows the updating equations for the Winters method. In the following formula, x_t is the actual value of the series at time t; a_t is the smoothed value of the series at time t; b_t is the smoothed trend at time t; c_t is the smoothed quadratic trend at time t; $s_{t-1}(t)$ selects the old value of the seasonal factor that corresponds to time t before the seasonal factors are updated.

The estimates of the constant, linear, and quadratic trend parameters are updated by using the following equations:

For TREND=3,

$$a_t = \omega_1 \frac{x_t}{s_{t-1}(t)} + (1 - \omega_1)(a_{t-1} + b_{t-1} + c_{t-1})$$

$$b_t = \omega_2(a_t - a_{t-1} + c_{t-1}) + (1 - \omega_2)(b_{t-1} + 2c_{t-1})$$

$$c_t = \omega_2 \frac{1}{2}(b_t - b_{t-1}) + (1 - \omega_2)c_{t-1}$$

For TREND=2,

$$a_t = \omega_1 \frac{x_t}{s_{t-1}(t)} + (1 - \omega_1)(a_{t-1} + b_{t-1})$$

$$b_t = \omega_2(a_t - a_{t-1}) + (1 - \omega_2)b_{t-1}$$

For TREND=1,

$$a_t = \omega_1 \frac{x_t}{s_{t-1}(t)} + (1 - \omega_1)a_{t-1}$$

In this updating system, the trend polynomial is always centered at the current period so that the intercept parameter of the trend polynomial for predicted values at times after t is always the updated intercept parameter a_t. The predicted value for τ periods ahead is

$$x_{t+\tau} = (a_t + b_t\tau)s_t(t + \tau)$$

The seasonal parameters are updated when the season changes in the data, using the mean of the ratios of the actual to the predicted values for the season. For example, if SEASONS=MONTH and INTERVAL=DAY, then when the observation for the first of February is encountered, the seasonal parameter for January is updated by using the formula

$$s_t(t-1) = \omega_3 \frac{1}{31} \sum_{i=t-31}^{t-1} \frac{x_i}{a_i} + (1 - \omega_3)s_{t-1}(t-1)$$

where t is February 1 of the current year, $s_t(t-1)$ is the seasonal parameter for January updated with the data available at time t, and $s_{t-1}(t-1)$ is the seasonal parameter for January of the previous year.

When multiple seasons are used, $s_t(t)$ is a product of seasonal factors. For example, if SEASONS=(MONTH DAY) then $s_t(t)$ is the product of the seasonal factors for the month and for the day of the week: $s_t(t) = s_t^m(t)s_t^d(t)$.

The factor $s_t^m(t)$ is updated at the start of each month by using a modification of the preceding formula that adjusts for the presence of the other seasonal by dividing the summands $\frac{x_i}{a_i}$ by the that corresponds to day of the week effect $s_i^d(i)$.

Similarly, the factor $s_t^d(t)$ is updated by using the following formula:

$$s_t^d(t) = \omega_3 \frac{x_t}{a_t s_t^m(t)} + (1 - \omega_3)s_{t-1}^d(t)$$

where $s_{t-1}^d(t)$ is the seasonal factor for the same day of the previous week.

Missing values after the start of the series are replaced with one-step-ahead predicted values, and the predicted value is substituted for x_i and applied to the updating equations.

Normalization

The parameters are normalized so that the seasonal factors for each cycle have a mean of 1.0. This normalization is performed after each complete cycle and at the end of the data. Thus, if INTERVAL=MONTH and SEASONS=MONTH are specified and a series begins with a July value, then the seasonal factors for the series are normalized at each observation for July and at the last observation in the data set. The normalization is performed by dividing each of the seasonal parameters, and multiplying each of the trend parameters, by the mean of the unnormalized seasonal parameters.

Smoothing Weights

The weight for updating the seasonal factors, ω_3, is given by the third value specified in the WEIGHT= option. If the WEIGHT= option is not used, then ω_3 defaults to 0.25; if the WEIGHT= option is used but does not specify a third value, then ω_3 defaults to ω_2. The weight for updating the linear and quadratic trend parameters, ω_2, is given by the second value specified in the WEIGHT= option; if the WEIGHT= option does not specify a second value, then ω_2 defaults to ω_1. The updating weight for the constant parameter, ω_1, is given by the first value specified in the WEIGHT= option. As a general rule, smaller smoothing weights are appropriate for series with a slowly changing trend, while larger weights are appropriate for volatile series with a rapidly changing trend.

If the WEIGHT= option is not used, then ω_1 defaults to $(1 - 0.8^{1/trend})$, where *trend* is the value of the TREND= option. This produces defaults of WEIGHT=0.2 for TREND=1, WEIGHT=0.10557 for TREND=2, and WEIGHT=0.07168 for TREND=3.

The ESM procedure and the Time Series Forecasting System provide for generating forecast models that use Winters Method and enable you to specify or optimize the weights. (See Chapter 14, "The ESM Procedure," and Chapter 46, "Getting Started with Time Series Forecasting," for details.)

Confidence Limits

A method for calculating exact forecast confidence limits for the WINTERS method is not available. Therefore, the approach taken in PROC FORECAST is to assume that the true seasonal factors have small variability about a set of fixed seasonal factors and that the remaining variation of the series is small relative to the mean level of the series. The equations are written

$$s_t(t) = \mathrm{I}(t)(1 + \delta_t)$$

$$x_t = \mu\mathrm{I}(t)(1 + \gamma_t)$$

$$a_t = \xi(1 + \alpha_t)$$

where μ is the mean level and $\mathrm{I}(t)$ are the fixed seasonal factors. Assuming that α_t and δ_t are small, the forecast equations can be linearized and only first-order terms in δ_t and α_t kept. In terms of forecasts for γ_t, this linearized system is equivalent to a seasonal ARIMA model. Confidence limits for γ_t are based on this ARIMA model and converted into confidence limits for x_t using $s_t(t)$ as estimates of $\mathrm{I}(t)$.

The exponential smoothing confidence limits are based on an approximation to a weighted regression model, whereas the preceding Winters confidence limits are based on an approximation to an ARIMA model. You can use METHOD=WINTERS without the SEASONS= option to do exponential smoothing and get confidence limits for the EXPO forecasts based on the ARIMA model approximation. These are generally more pessimistic than the weighted regression confidence limits produced by METHOD=EXPO.

ADDWINTERS Method

The ADDWINTERS method is like the WINTERS method except that the seasonal parameters are added to the trend instead of multiplied with the trend. The default TREND=2 model is as follows:

$$x_t = a + bt + s(t) + \epsilon_t$$

The WINTERS method for updating equation and confidence limits calculations described in the preceding section are modified accordingly for the additive version.

Holt Two-Parameter Exponential Smoothing

If the seasonal factors are omitted (that is, if the SEASONS= option is not specified), the WINTERS (and ADDWINTERS) method reduces to the Holt two-parameter version of exponential smoothing. Thus, the WINTERS method is often referred to as the Holt-Winters method.

Double exponential smoothing is a special case of the Holt two-parameter smoother. The double exponential smoothing results can be duplicated with METHOD=WINTERS by omitting the SEASONS= option and appropriately setting the WEIGHT= option. Letting $\alpha = \omega(2 - \omega)$ and $\beta = \omega/(2 - \omega)$, the following statements produce the same forecasts:

```
proc forecast method=expo trend=2 weight=ω ...;

proc forecast method=winters trend=2 weight=(α,β) ...;
```

Although the forecasts are the same, the confidence limits are computed differently.

Choice of Weights for EXPO, WINTERS, and ADDWINTERS Methods

For the EXPO, WINTERS, and ADDWINTERS methods, properly chosen smoothing weights are of critical importance in generating reasonable results. There are several factors to consider in choosing the weights.

The noisier the data, the lower should be the weight given to the most recent observation. Another factor to consider is how quickly the mean of the time series is changing. If the mean of the series is changing rapidly, relatively more weight should be given to the most recent observation. The more stable the series over time, the lower should be the weight given to the most recent observation.

Note that the smoothing weights should be set separately for each series; weights that produce good results for one series might be poor for another series. Since PROC FORECAST does not have a feature to use different weights for different series, when forecasting multiple series with the EXPO, WINTERS, or ADDWINTERS method it might be desirable to use different PROC FORECAST steps with different WEIGHT= options.

For the Winters method, many combinations of weight values might produce unstable *noninvertible* models, even though all three weights are between 0 and 1. When the model is noninvertible, the forecasts depend strongly on values in the distant past, and predictions are determined largely by the starting values. Unstable models usually produce poor forecasts. The Winters model can be unstable even if the weights are optimally chosen to minimize the in-sample MSE. See Archibald (1990) for a detailed discussion of the unstable region of the parameter space of the Winters model.

Optimal weights and forecasts for exponential smoothing models can be computed by using the ESM and ARIMA procedures and by the Time Series Forecasting System.

Starting Values for EXPO, WINTERS, and ADDWINTERS Methods

The exponential smoothing method requires starting values for the smoothed values S_0, $S_0^{[2]}$, and $S_0^{[3]}$. The Winters and additive Winters methods require starting values for the trend coefficients and seasonal factors.

By default, starting values for the trend parameters are computed by a time trend regression over the first few observations for the series. Alternatively, you can specify the starting value for the trend parameters with the ASTART=, BSTART=, and CSTART= options.

The number of observations used in the time trend regression for starting values depends on the NSTART= option. For METHOD=EXPO, NSTART= beginning values of the series are used, and the coefficients of the time trend regression are then used to form the initial smoothed values S_0, $S_0^{[2]}$, and $S_0^{[3]}$.

For METHOD=WINTERS or METHOD=ADDWINTERS, n complete seasonal cycles are used to compute starting values for the trend parameter, where n is the value of the NSTART= option. For example, for monthly data the seasonal cycle is one year, so NSTART=2 specifies that the first 24 observations at the beginning of each series are used for the time trend regression used to calculate starting values.

The starting values for the seasonal factors for the WINTERS and ADDWINTERS methods are computed from seasonal averages over the first few complete seasonal cycles at the beginning of the series. The number of seasonal cycles averaged to compute starting seasonal factors is controlled by the NSSTART= option. For example, for monthly data with SEASONS=12 or SEASONS=MONTH, the first n January values are averaged to get the starting value for the January seasonal parameter, where n is the value of the NSSTART= option.

The $s_0(i)$ seasonal parameters are set to the ratio (for WINTERS) or difference (for ADDWINTERS) of the mean for the season to the overall mean for the observations used to compute seasonal starting values.

For example, if METHOD=WINTERS, INTERVAL=DAY, SEASON=(MONTH DAY), and NSTART=2 (the default), the initial seasonal parameter for January is the ratio of the mean value over days in the first two Januaries after the start of the series (that is, after the first nonmissing value) to the mean value for all days read for initialization of the seasonal factors. Likewise, the initial factor for Sundays is the ratio of the mean value for Sundays to the mean of all days read.

For the ASTART=, BSTART=, and CSTART= options, the values specified are associated with the variables in the VAR statement in the order in which the variables are listed (the first value with the first variable, the second value with the second variable, and so on). If there are fewer values than variables, default starting values are used for the later variables. If there are more values than variables, the extra values are ignored.

Specifying Seasonality

Seasonality of a time series is a regular fluctuation about a trend. This is called seasonality because the time of year is the most common source of periodic variation. For example, sales of home heating oil are regularly greater in winter than during other times of the year.

Seasonality can be caused by many things other than weather. In the United States, sales of nondurable goods are greater in December than in other months because of the Christmas shopping season. The term seasonality is also used for cyclical fluctuation at periods other than a year. Often, certain days of the week cause regular fluctuation in daily time series, such as increased spending on leisure activities during weekends.

Three kinds of seasonality are supported in PROC FORECAST: time-of-year, day-of-week, and time-of-day. The seasonal part of the model is specified by using the SEASONS= option. The values for the SEASONS= option are listed in Table 16.2.

Table 16.2 The SEASONS= Option

SEASONS= Value	Cycle Length	Type of Seasonality
QTR	yearly	time of year
MONTH	yearly	time of year
DAY	weekly	day of week
HOUR	daily	time of day

The three kinds of seasonality can be combined. For example, SEASONS=(MONTH DAY HOUR) specifies that 24 hour-of-day seasons are nested within 7 day-of-week seasons, which in turn are nested within 12 month-of-year seasons. The different kinds of intervals can be listed in the SEASONS= option in any order. Thus, SEASONS=(HOUR DAY MONTH) is the same as SEASONS=(MONTH DAY HOUR). Note that the Winters method smoothing equations might be less stable when multiple seasonal factors are used.

Multiple period seasons can also be used. For example, SEASONS=QTR2 specifies two semiannual time-of-year seasons. The grouping of observations into multiple period seasons starts with the first interval in the seasonal cycle. Thus, MONTH2 seasons are January–February, March–April, and so on. (There is no provision for shifting seasonal intervals; thus, there is no way to specify seasons December–January, February–March, April–May, and so on.)

For multiple period seasons, the number of intervals combined to form the seasons must evenly divide and be less than the basic cycle length. For example, with SEASONS=MONTH*n*, the basic cycle length is 12, so MONTH2, MONTH3, MONTH4, and MONTH6 are valid SEASONS= values (because 2, 3, 4, and 6 evenly divide 12 and are less than 12), but MONTH5 and MONTH12 are not valid SEASONS= values.

The frequency of the seasons must not be greater than the frequency of the input data. For example, you cannot specify SEASONS=MONTH if INTERVAL=QTR or SEASONS=MONTH if INTERVAL=MONTH2. You also cannot specify two seasons of the same basic cycle. For example, SEASONS=(MONTH QTR) or SEASONS=(MONTH2 MONTH4) is not allowed.

Alternatively, the seasonality can be specified by giving the number of seasons in the SEASONS= option. SEASONS=*n* specifies that there are *n* seasons, with observations 1, $n+1$, $2n+1$, and so on in the first season, observations 2, $n+2$, $2n+2$, and so on in the second season, and so forth.

The options SEASONS=*n* and SINTPER=*m* cause PROC FORECAST to group the input observations into *n* seasons, with *m* observations to a season, which repeat every *nm* observations. The options SEASONS=(n_1 n_2) and SINTPER=(m_1 m_2) produce n_1 seasons with m_1 observations to a season nested within n_2 seasons with $n_1 m_1 m_2$ observations to a season.

If the SINTPER=*m* option is used with the SEASONS= option, the SEASONS= interval is multiplied by the SINTPER= value. For example, specifying both SEASONS=(QTR HOUR) and SINTPER=(2 3) is the same as specifying SEASONS=(QTR2 HOUR3) and also the same as specifying SEASONS=(HOUR3 QTR2).

Data Requirements

You should have ample data for the series that you forecast by using PROC FORECAST. However, the results might be poor unless you have a good deal more than the minimum amount of data the procedure allows. The minimum number of observations required for the different methods is as follows:

- If METHOD=STEPAR is used, the minimum number of nonmissing observations required for each series forecast is the TREND= option value plus the value of the NLAGS= option. For example, using NLAGS=13 and TREND=2, at least 15 nonmissing observations are needed.
- If METHOD=EXPO is used, the minimum is the TREND= option value.
- If METHOD=WINTERS or ADDWINTERS is used, the minimum number of observations is either the number of observations in a complete seasonal cycle or the TREND= option value, whichever is greater. (However, there should be data for several complete seasonal cycles, or the seasonal factor estimates might be poor.) For example, for the seasonal specifications SEASONS=MONTH, SEASONS=(QTR DAY), or SEASONS=(MONTH DAY HOUR), the longest cycle length is one year, so at least one year of data is required. At least two years of data is recommended.

OUT= Data Set

The FORECAST procedure writes the forecast to the output data set named by the OUT= option. The OUT= data set contains the following variables:

- the BY variables
- _TYPE_, a character variable that identifies the type of observation
- _LEAD_, a numeric variable that indicates the number of steps ahead in the forecast. The value of _LEAD_ is 0 for the one-step-ahead forecasts before the start of the forecast period.
- the ID statement variables
- the VAR statement variables, which contain the result values as indicated by the _TYPE_ variable value for the observation

The FORECAST procedure processes each of the input variables listed in the VAR statement and writes several observations for each forecast period to the OUT= data set. The observations are identified by the value of the _TYPE_ variable. The options OUTACTUAL, OUTALL, OUTLIMIT, OUTRESID, OUT1STEP, OUTFULL, and OUTSTD control which types of observations are included in the OUT= data set.

The values of the variable _TYPE_ are as follows:

ACTUAL — The VAR statement variables contain actual values from the input data set. The OUTACTUAL option writes the actual values. By default, only the observations for the forecast period are output.

FORECAST — The VAR statement variables contain forecast values. The OUT1STEP option writes the one-step-ahead predicted values for the observations used to fit the model.

RESIDUAL — The VAR statement variables contain residuals. The residuals are computed by subtracting the forecast value from the actual value ($residual = actual - forecast$). The OUTRESID option writes observations for the residuals.

L*nn* — The VAR statement variables contain lower *nn* % confidence limits for the forecast values for the future observations specified by the LEAD= option. The value of *nn* depends on the ALPHA= option; with the default ALPHA=0.05, the _TYPE_ value is L95 for the lower confidence limit observations. The OUTLIMIT option writes observations for the upper and lower confidence limits.

U*nn* — The VAR statement variables contain upper *nn* % confidence limits for the forecast values for the future observations specified by the LEAD= option. The value of *nn* depends on the ALPHA= option; with the default ALPHA=0.05, the _TYPE_ value is U95 for the upper confidence limit observations. The OUTLIMIT option writes observations for the upper and lower confidence limits.

STD — The VAR statement variables contain standard errors of the forecast values. The OUTSTD option writes observations for the standard errors of the forecast.

If no output control options are specified, PROC FORECAST outputs only the forecast values for the forecast periods.

The _TYPE_ variable can be used to subset the OUT= data set. For example, the following data step splits the OUT= data set into two data sets, one that contains the forecast series and the other that contains the residual series. For example

```
proc forecast out=out outresid ...;
   ...
run;

data fore resid;
   set out;
   if _TYPE_='FORECAST' then output fore;
   if _TYPE_='RESIDUAL' then output resid;
run;
```

See Chapter 3, “Working with Time Series Data,” for more information about processing time series data sets in this format.

OUTEST= Data Set

The FORECAST procedure writes the parameter estimates and goodness-of-fit statistics to an output data set when the OUTEST= option is specified. The OUTEST= data set contains the following variables:

- the BY variables
- the first ID variable, which contains the value of the ID variable for the last observation in the input data set used to fit the model

- _TYPE_, a character variable that identifies the type of each observation
- the VAR statement variables, which contain statistics and parameter estimates for the input series. The values contained in the VAR statement variables depend on the _TYPE_ variable value for the observation.

The observations contained in the OUTEST= data set are identified by the _TYPE_ variable. The OUTEST= data set might contain observations with the following _TYPE_ values:

AR1–AR*n* The observation contains estimates of the autoregressive parameters for the series. Two-digit lag numbers are used if the value of the NLAGS= option is 10 or more; in that case these _TYPE_ values are AR01–AR*n*. These observations are output for the STEPAR method only.

CONSTANT The observation contains the estimate of the constant or intercept parameter for the time trend model for the series. For the exponential smoothing and the Winters' methods, the trend model is centered (that is, t =0) at the last observation used for the fit.

LINEAR The observation contains the estimate of the linear or slope parameter for the time trend model for the series. This observation is output only if you specify TREND=2 or TREND=3.

N The observation contains the number of nonmissing observations used to fit the model for the series.

QUAD The observation contains the estimate of the quadratic parameter for the time trend model for the series. This observation is output only if you specify TREND=3.

SIGMA The observation contains the estimate of the standard deviation of the error term for the series.

S1–S3 The observations contain exponentially smoothed values at the last observation. _TYPE_=S1 is the final smoothed value of the single exponential smooth. _TYPE_=S2 is the final smoothed value of the double exponential smooth. _TYPE_=S3 is the final smoothed value of the triple exponential smooth. These observations are output for METHOD=EXPO only.

S_*name* The observation contains estimates of the seasonal parameters. For example, if SEASONS=MONTH, the OUTEST= data set contains observations with _TYPE_=S_JAN, _TYPE_=S_FEB, _TYPE_=S_MAR, and so forth.

For multiple-period seasons, the names of the first and last interval of the season are concatenated to form the season name. Thus, for SEASONS=MONTH4, the OUTEST= data set contains observations with _TYPE_=S_JANAPR, _TYPE_=S_MAYAUG, and _TYPE_=S_SEPDEC.

When the SEASONS= option specifies numbers, the seasonal factors are labeled _TYPE_=S_*i_j*. For example, SEASONS=(2 3) produces observations with _TYPE_ values of S_1_1, S_1_2, S_2_1, S_2_2, and S_2_3. The observation with _TYPE_=S_*i_j* contains the seasonal parameters for the *j*th season of the *i*th seasonal cycle.

These observations are output only for METHOD=WINTERS or METHOD=ADDWINTERS.

WEIGHT The observation contains the smoothing weight used for exponential smoothing. This is the value of the WEIGHT= option. This observation is output for METHOD=EXPO only.

WEIGHT1 | WEIGHT2 | WEIGHT3 The observations contain the weights used for smoothing the WINTERS or ADDWINTERS method parameters (specified by the WEIGHT= option). _TYPE_=WEIGHT1 is the weight used to smooth the CONSTANT parameter. _TYPE_=WEIGHT2 is the weight used to smooth the LINEAR and QUAD parameters. _TYPE_=WEIGHT3 is the weight used to smooth the seasonal parameters. These observations are output only for the WINTERS and ADDWINTERS methods.

NRESID
: The observation contains the number of nonmissing residuals, n, used to compute the goodness-of-fit statistics. The residuals are obtained by subtracting the one-step-ahead predicted values from the observed values.

SST
: The observation contains the total sum of squares for the series, corrected for the mean. $SST = \sum_{t=1}^{n} (y_t - \overline{y})^2$, where $\overline{y}$ is the series mean.

SSE
: The observation contains the sum of the squared residuals, uncorrected for the mean. $SSE = \sum_{t=1}^{n} (y_t - \hat{y}_t)^2$, where $\hat{y}$ is the one-step predicted value for the series.

MSE
: The observation contains the mean squared error, calculated from one-step-ahead forecasts. $MSE = \frac{1}{n-k} SSE$, where k is the number of parameters in the model.

RMSE
: The observation contains the root mean squared error. $RMSE = \sqrt{MSE}$.

MAPE
: The observation contains the mean absolute percent error.

  $MAPE = \frac{100}{n} \sum_{t=1}^{n} |(y_t - \hat{y}_t)/y_t|$.

MPE
: The observation contains the mean percent error.

  $MPE = \frac{100}{n} \sum_{t=1}^{n} (y_t - \hat{y}_t)/y_t$.

MAE
: The observation contains the mean absolute error. $MAE = \frac{1}{n} \sum_{t=1}^{n} |y_t - \hat{y}_t|$.

ME
: The observation contains the mean error. $MAE = \frac{1}{n} \sum_{t=1}^{n} (y_t - \hat{y}_t)$.

MAXE
: The observation contains the maximum error (the largest residual).

MINE
: The observation contains the minimum error (the smallest residual).

MAXPE
: The observation contains the maximum percent error.

MINPE
: The observation contains the minimum percent error.

RSQUARE
: The observation contains the R square statistic, $R^2 = 1 - SSE/SST$. If the model fits the series badly, the model error sum of squares *SSE* might be larger than *SST* and the R square statistic will be negative.

ADJRSQ
: The observation contains the adjusted R square statistic.

  $ADJRSQ = 1 - (\frac{n-1}{n-k})(1 - R^2)$.

ARSQ
: The observation contains Amemiya's adjusted R square statistic.

  $ARSQ = 1 - (\frac{n+k}{n-k})(1 - R^2)$.

RW_RSQ
: The observation contains the random walk R square statistic (Harvey's R_{D}^2 statistic that uses the random walk model for comparison).

  $RW\_RSQ = 1 - (\frac{n-1}{n}) SSE/RWSSE$,

  where $RWSSE = \sum_{t=2}^{n} (y_t - y_{t-1} - \mu)^2$ and $\mu = \frac{1}{n-1} \sum_{t=2}^{n} (y_t - y_{t-1})$.

AIC
: The observation contains Akaike's information criterion.

  $AIC = n \ln(SSE/n) + 2k$.

SBC — The observation contains Schwarz's Bayesian criterion.
$SBC = n\ln(SSE/n) + k\ln(n)$.

APC — The observation contains Amemiya's prediction criterion.
$APC = \frac{1}{n}SST(\frac{n+k}{n-k})(1 - R^2) = (\frac{n+k}{n-k})\frac{1}{n}SSE$.

CORR — The observation contains the correlation coefficient between the actual values and the one-step-ahead predicted values.

THEILU — The observation contains Theil's U statistic that uses original units. See Maddala (1977, pp. 344–345), and Pindyck and Rubinfeld (1981, pp. 364–365) for more information about Theil statistics.

RTHEILU — The observation contains Theil's U statistic calculated using relative changes.

THEILUM — The observation contains the bias proportion of Theil's U statistic.

THEILUS — The observation contains the variance proportion of Theil's U statistic.

THEILUC — The observation contains the covariance proportion of Theil's U statistic.

THEILUR — The observation contains the regression proportion of Theil's U statistic.

THEILUD — The observation contains the disturbance proportion of Theil's U statistic.

RTHEILUM — The observation contains the bias proportion of Theil's U statistic, calculated by using relative changes.

RTHEILUS — The observation contains the variance proportion of Theil's U statistic, calculated by using relative changes.

RTHEILUC — The observation contains the covariance proportion of Theil's U statistic, calculated by using relative changes.

RTHEILUR — The observation contains the regression proportion of Theil's U statistic, calculated by using relative changes.

RTHEILUD — The observation contains the disturbance proportion of Theil's U statistic, calculated by using relative changes.

Examples: FORECAST Procedure

Example 16.1: Forecasting Auto Sales

This example uses the Winters method to forecast the monthly U. S. sales of passenger cars series (VEHICLES) from the data set SASHELP.USECON. These data are taken from *Business Statistics*, published by the U. S. Bureau of Economic Analysis.

The following statements plot the series. The plot is shown in Output 16.1.1.

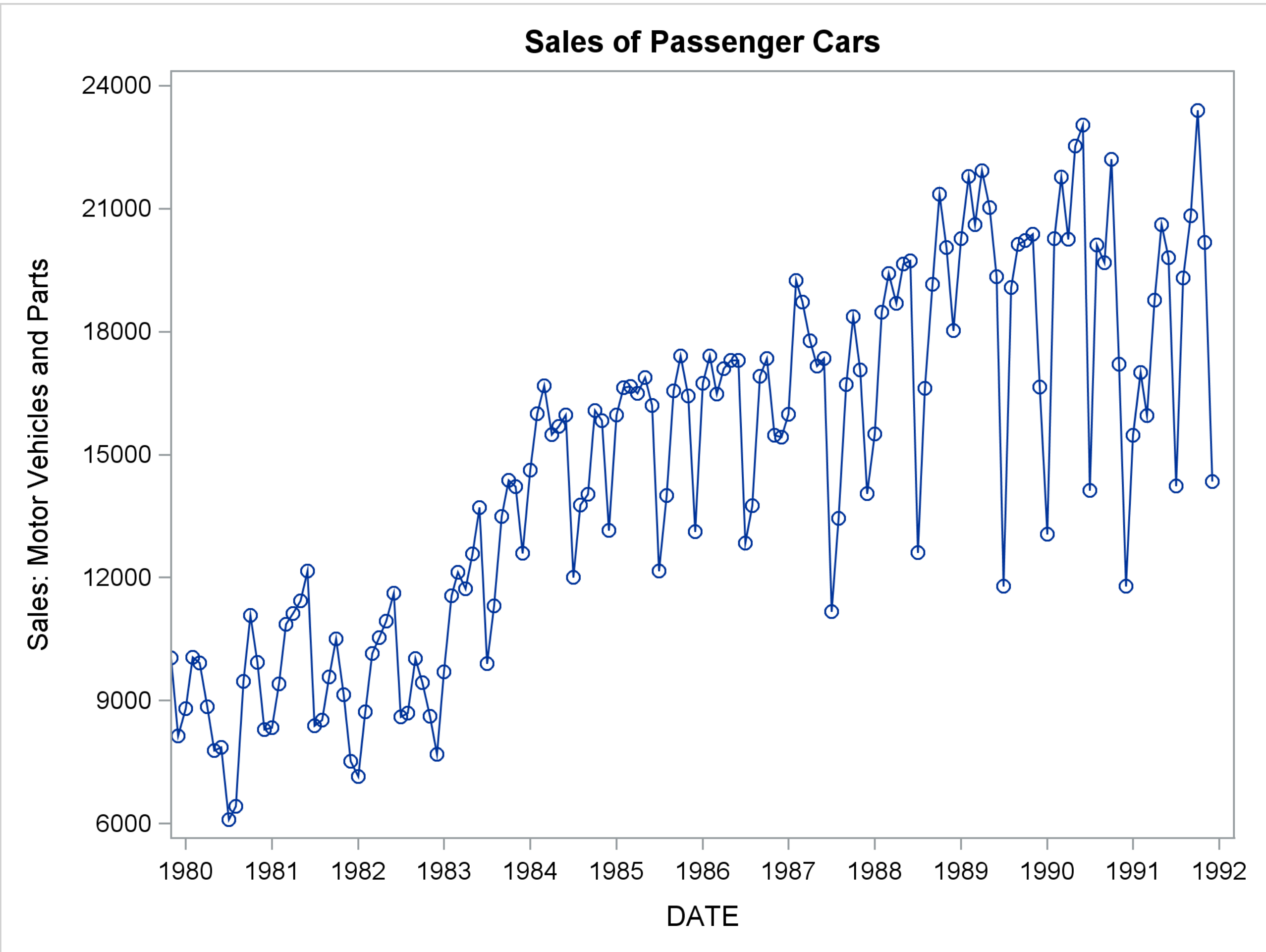
```
title1 "Sales of Passenger Cars";

symbol1 i=spline v=dot;
```

```
axis2 label=(a=-90 r=90 "Vehicles and Parts" )
      order=(6000 to 24000 by 3000);

title1 "Sales of Passenger Cars";
proc sgplot data=sashelp.usecon;
   series x=date y=vehicles / markers;
   xaxis values=('1jan80'd to '1jan92'd by year);
   yaxis values=(6000 to 24000 by 3000);
   format date year4.;
run;
```

Output 16.1.1 Monthly Passenger Car Sales



The following statements produce the forecast:

```
proc forecast data=sashelp.usecon interval=month
              method=winters seasons=month lead=12
              out=out outfull outresid outest=est;
   id date;
   var vehicles;
   where date >= '1jan80'd;
run;
```

The INTERVAL=MONTH option indicates that the data are monthly, and the ID DATE statement gives the dating variable. The METHOD=WINTERS specifies the Winters smoothing method. The LEAD=12 option forecasts 12 months ahead. The OUT=OUT option specifies the output data set, while the OUTFULL and OUTRESID options include in the OUT= data set the predicted and residual values for the historical period and the confidence limits for the forecast period. The OUTEST= option stores various statistics in an output data set. The WHERE statement is used to include only data from 1980 on.

The following statements print the OUT= data set (first 20 observations):

```
title2 'The OUT= Data Set';
proc print data=out (obs=20) noobs;
run;
```

The listing of the output data set produced by PROC PRINT is shown in part in Output 16.1.2.

Output 16.1.2 The OUT= Data Set Produced by PROC FORECAST (First 20 Observations)

Sales of Passenger Cars
The OUT= Data Set

DATE	_TYPE_	_LEAD_	VEHICLES
JAN80	ACTUAL	0	8808.00
JAN80	FORECAST	0	8046.52
JAN80	RESIDUAL	0	761.48
FEB80	ACTUAL	0	10054.00
FEB80	FORECAST	0	9284.31
FEB80	RESIDUAL	0	769.69
MAR80	ACTUAL	0	9921.00
MAR80	FORECAST	0	10077.33
MAR80	RESIDUAL	0	-156.33
APR80	ACTUAL	0	8850.00
APR80	FORECAST	0	9737.21
APR80	RESIDUAL	0	-887.21
MAY80	ACTUAL	0	7780.00
MAY80	FORECAST	0	9335.24
MAY80	RESIDUAL	0	-1555.24
JUN80	ACTUAL	0	7856.00
JUN80	FORECAST	0	9597.50
JUN80	RESIDUAL	0	-1741.50
JUL80	ACTUAL	0	6102.00
JUL80	FORECAST	0	6833.16

The following statements print the OUTEST= data set:

```
title2 'The OUTEST= Data Set: WINTERS Method';
proc print data=est;
run;
```

The PROC PRINT listing of the OUTEST= data set is shown in Output 16.1.3.

Output 16.1.3 The OUTEST= Data Set Produced by PROC FORECAST

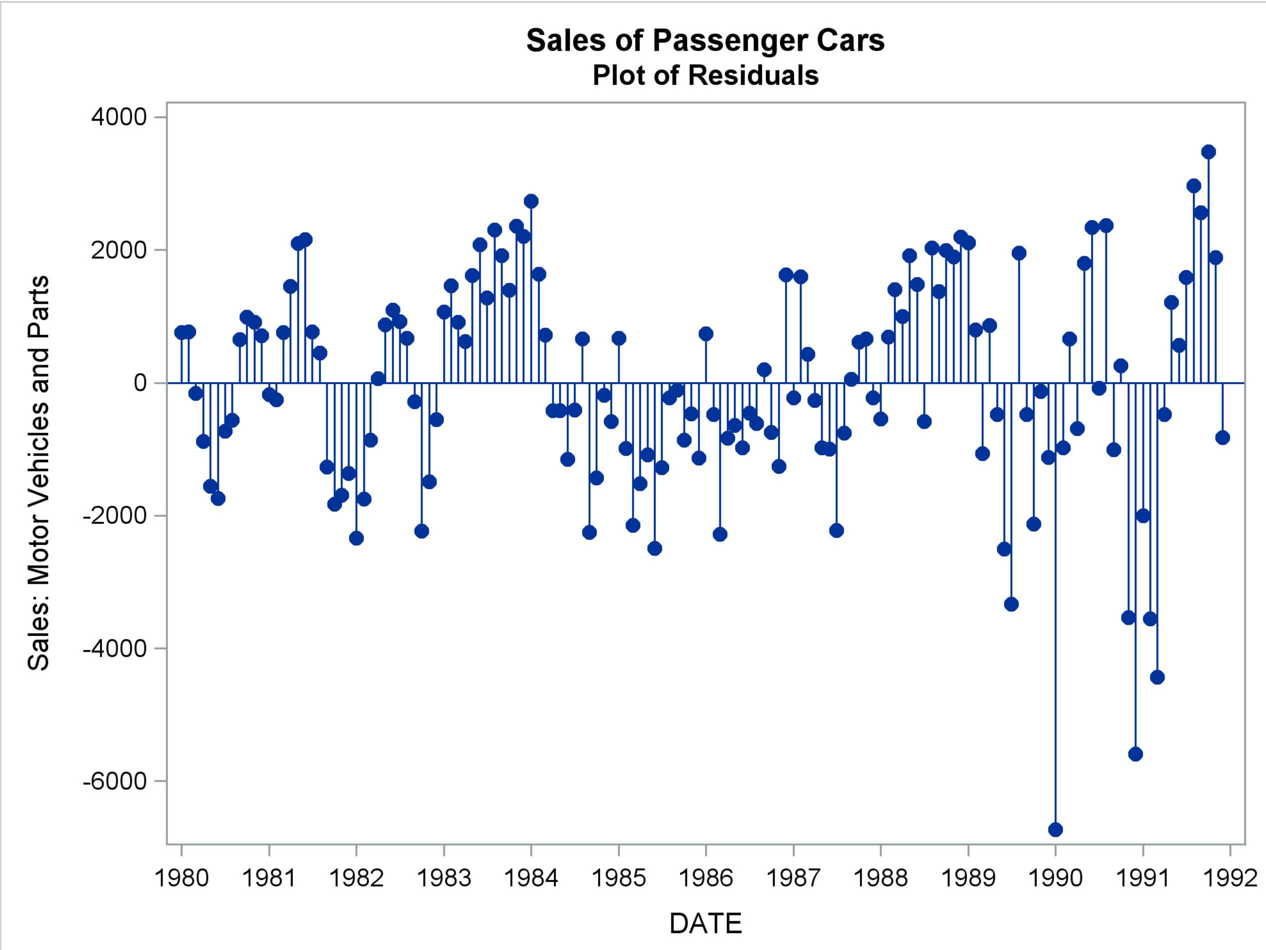
Sales of Passenger Cars
The OUTEST= Data Set: WINTERS Method

Obs	_TYPE_	DATE	VEHICLES
1	N	DEC91	144
2	NRESID	DEC91	144
3	DF	DEC91	130
4	WEIGHT1	DEC91	0.1055728
5	WEIGHT2	DEC91	0.1055728
6	WEIGHT3	DEC91	0.25
7	SIGMA	DEC91	1741.481
8	CONSTANT	DEC91	18577.368
9	LINEAR	DEC91	4.804732
10	S_JAN	DEC91	0.8909173
11	S_FEB	DEC91	1.0500278
12	S_MAR	DEC91	1.0546539
13	S_APR	DEC91	1.074955
14	S_MAY	DEC91	1.1166121
15	S_JUN	DEC91	1.1012972
16	S_JUL	DEC91	0.7418297
17	S_AUG	DEC91	0.9633888
18	S_SEP	DEC91	1.051159
19	S_OCT	DEC91	1.1399126
20	S_NOV	DEC91	1.0132126
21	S_DEC	DEC91	0.802034
22	SST	DEC91	2.63312E9
23	SSE	DEC91	394258270
24	MSE	DEC91	3032755.9
25	RMSE	DEC91	1741.481
26	MAPE	DEC91	9.4800217
27	MPE	DEC91	-1.049956
28	MAE	DEC91	1306.8534
29	ME	DEC91	-42.95376
30	RSQUARE	DEC91	0.8502696

The following statements plot the residuals. The plot is shown in Output 16.1.4.

```
title1 "Sales of Passenger Cars";
title2 'Plot of Residuals';
proc sgplot data=out;
   where _type_ = 'RESIDUAL';
   needle x=date y=vehicles / markers markerattrs=(symbol=circlefilled);
   xaxis values=('1jan80'd to '1jan92'd by year);
   format date year4.;
run;
```

Output 16.1.4 Residuals from Winters Method

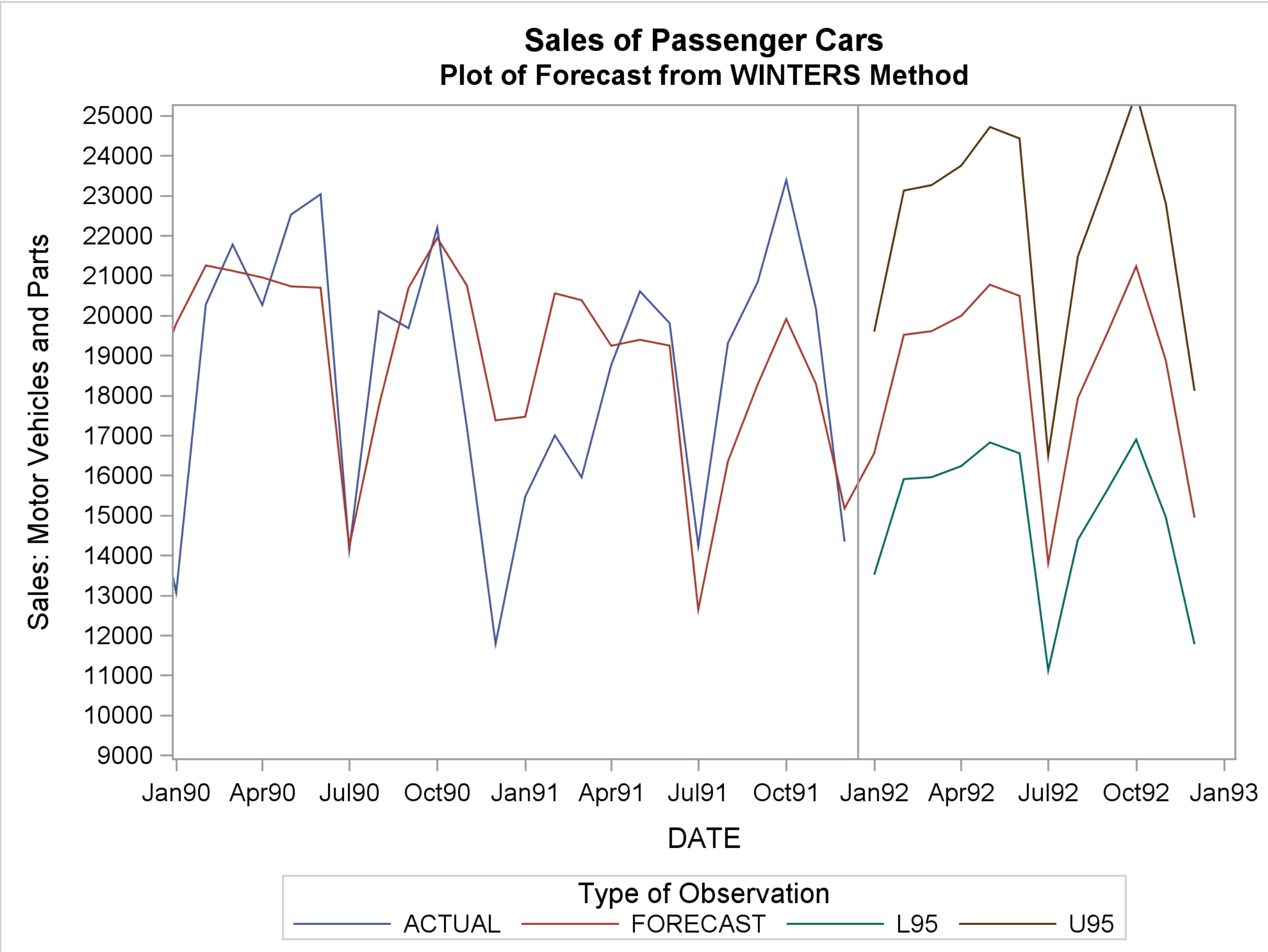


The following statements plot the forecast and confidence limits. The last two years of historical data are included in the plot to provide context for the forecast plot. A reference line is drawn at the start of the forecast period.

```
title1 "Sales of Passenger Cars";
title2 'Plot of Forecast from WINTERS Method';
proc sgplot data=out;
   series x=date y=vehicles / group=_type_ lineattrs=(pattern=1);
   where _type_ ^= 'RESIDUAL';
   refline '15dec91'd / axis=x;
   yaxis values=(9000 to 25000 by 1000);
   xaxis values=('1jan90'd to '1jan93'd by qtr);
run;
```

The plot is shown in Output 16.1.5.

Output 16.1.5 Forecast of Passenger Car Sales



Example 16.2: Forecasting Retail Sales

This example uses the stepwise autoregressive method to forecast the monthly U. S. sales of durable goods (DURABLES) and nondurable goods (NONDUR) from the SASHELP.USECON data set. The data are from *Business Statistics*, published by the U.S. Bureau of Economic Analysis. The following statements plot the series:

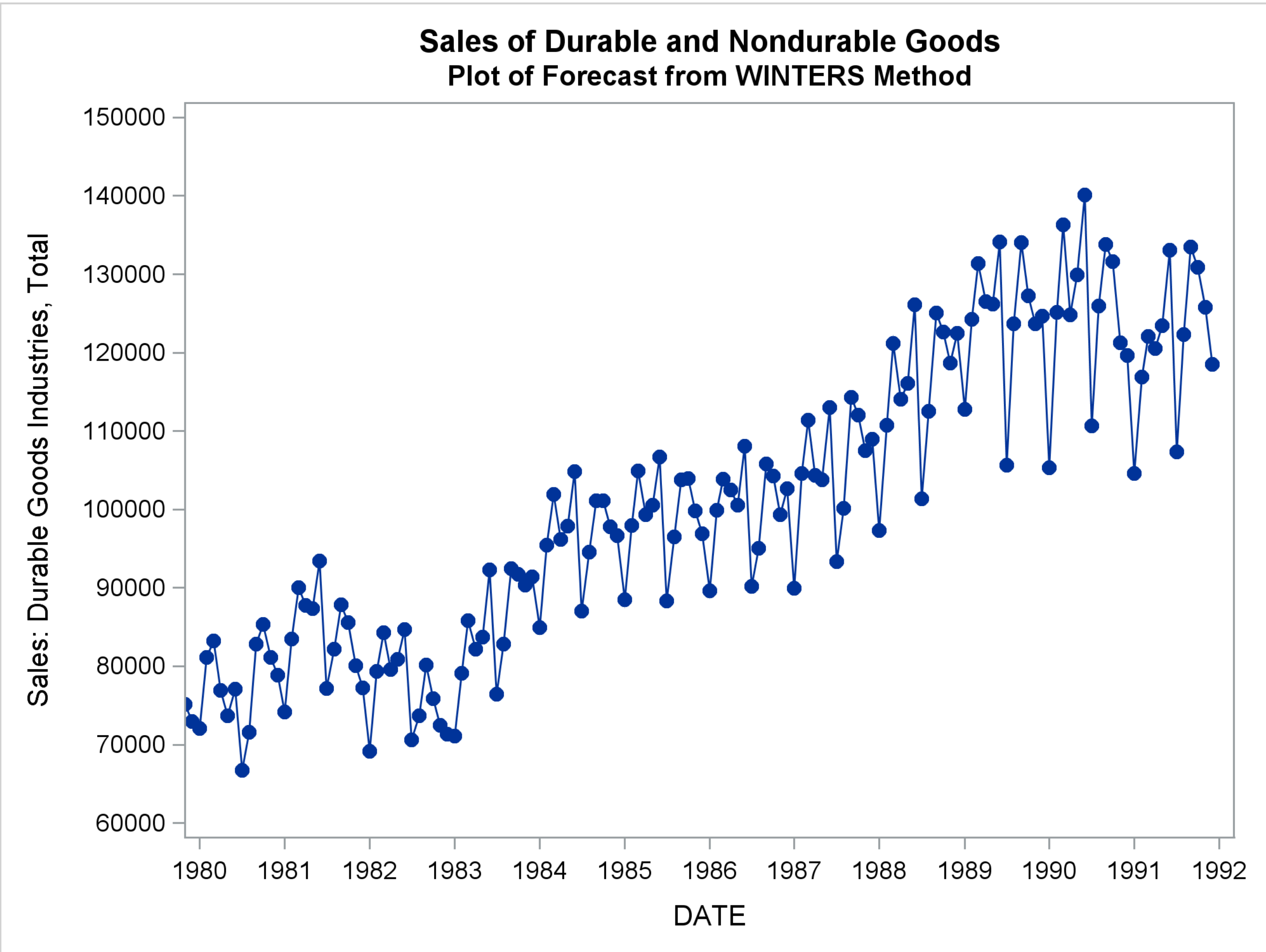
```
title1 'Sales of Durable and Nondurable Goods';
title2 'Plot of Forecast from WINTERS Method';
proc sgplot data=sashelp.usecon;
   series x=date y=durables / markers markerattrs=(symbol=circlefilled);
   xaxis values=('1jan80'd to '1jan92'd by year);
   yaxis values=(60000 to 150000 by 10000);
   format date year4.;
run;

title1 'Sales of Durable and Nondurable Goods';
title2 'Plot of Forecast from WINTERS Method';
proc sgplot data=sashelp.usecon;
```

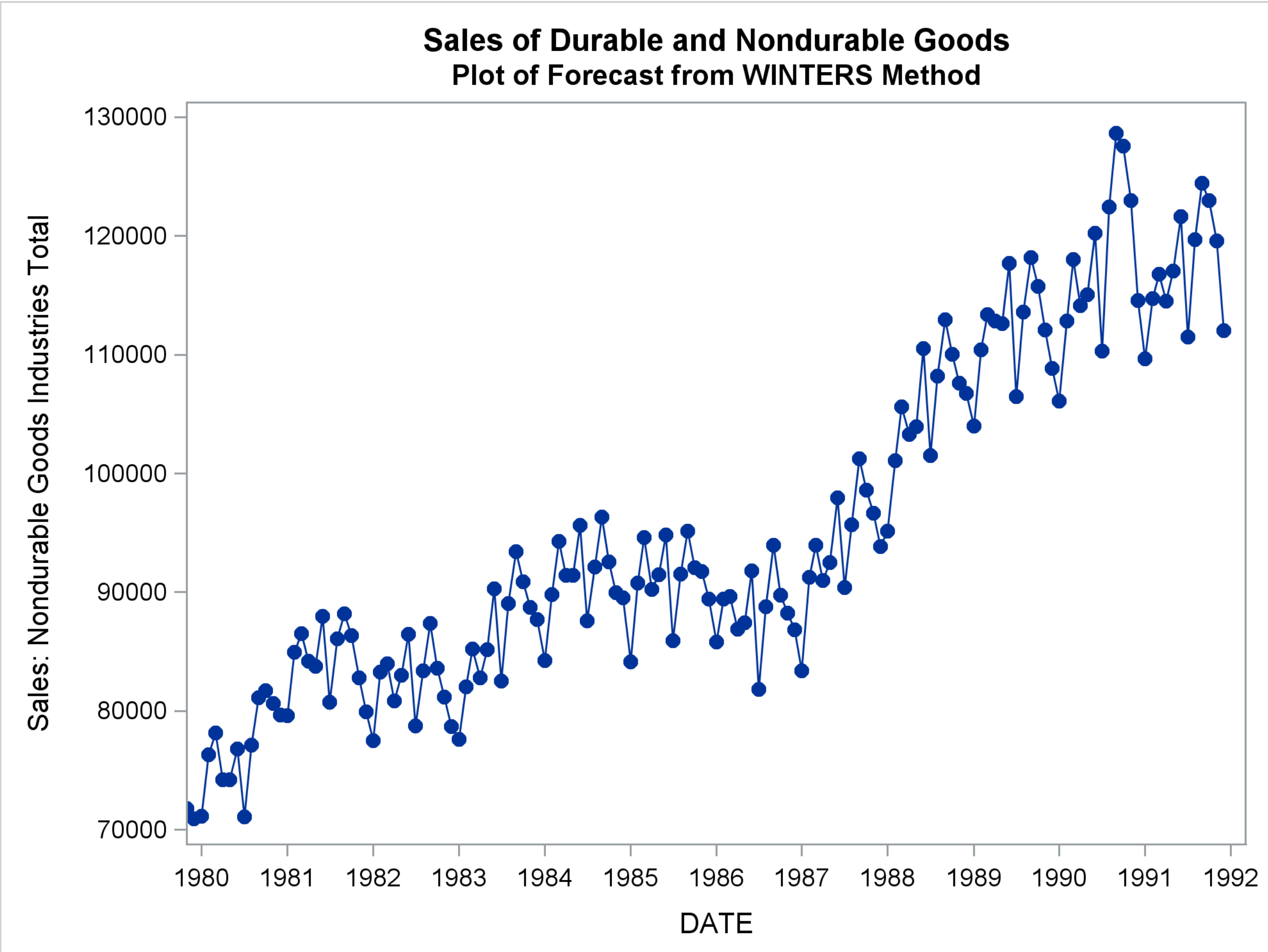
```
      series x=date y=nondur / markers markerattrs=(symbol=circlefilled);
      xaxis values=('1jan80'd to '1jan92'd by year);
      yaxis values=(70000 to 130000 by 10000);
      format date year4.;
   run;
```

The plots are shown in Output 16.2.1 and Output 16.2.2.

Output 16.2.1 Durable Goods Sales



Output 16.2.2 Nondurable Goods Sales



The following statements produce the forecast:

```
title1 "Forecasting Sales of Durable and Nondurable Goods";

proc forecast data=sashelp.usecon interval=month
              method=stepar trend=2 lead=12
              out=out outfull outest=est;
   id date;
   var durables nondur;
   where date >= '1jan80'd;
run;
```

The following statements print the OUTEST= data set.

```
title2 'OUTEST= Data Set: STEPAR Method';
proc print data=est;
run;
```

The PROC PRINT listing of the OUTEST= data set is shown in Output 16.2.3.

Output 16.2.3 The OUTEST= Data Set Produced by PROC FORECAST

Forecasting Sales of Durable and Nondurable Goods
OUTEST= Data Set: STEPAR Method

Obs	_TYPE_	DATE	DURABLES	NONDUR
1	N	DEC91	144	144
2	NRESID	DEC91	144	144
3	DF	DEC91	137	139
4	SIGMA	DEC91	4519.451	2452.2642
5	CONSTANT	DEC91	71884.597	73190.812
6	LINEAR	DEC91	400.90106	308.5115
7	AR01	DEC91	0.5844515	0.8243265
8	AR02	DEC91	.	.
9	AR03	DEC91	.	.
10	AR04	DEC91	.	.
11	AR05	DEC91	.	.
12	AR06	DEC91	0.2097977	.
13	AR07	DEC91	.	.
14	AR08	DEC91	.	.
15	AR09	DEC91	.	.
16	AR10	DEC91	-0.119425	.
17	AR11	DEC91	.	.
18	AR12	DEC91	0.6138699	0.8050854
19	AR13	DEC91	-0.556707	-0.741854
20	SST	DEC91	4.923E10	2.8331E10
21	SSE	DEC91	1.88157E9	544657337
22	MSE	DEC91	13734093	3918398.1
23	RMSE	DEC91	3705.9538	1979.4944
24	MAPE	DEC91	2.9252601	1.6555935
25	MPE	DEC91	-0.253607	-0.085357
26	MAE	DEC91	2866.675	1532.8453
27	ME	DEC91	-67.87407	-29.63026
28	RSQUARE	DEC91	0.9617803	0.9807752

The following statements plot the forecasts and confidence limits. The last two years of historical data are included in the plots to provide context for the forecast. A reference line is drawn at the start of the forecast period.

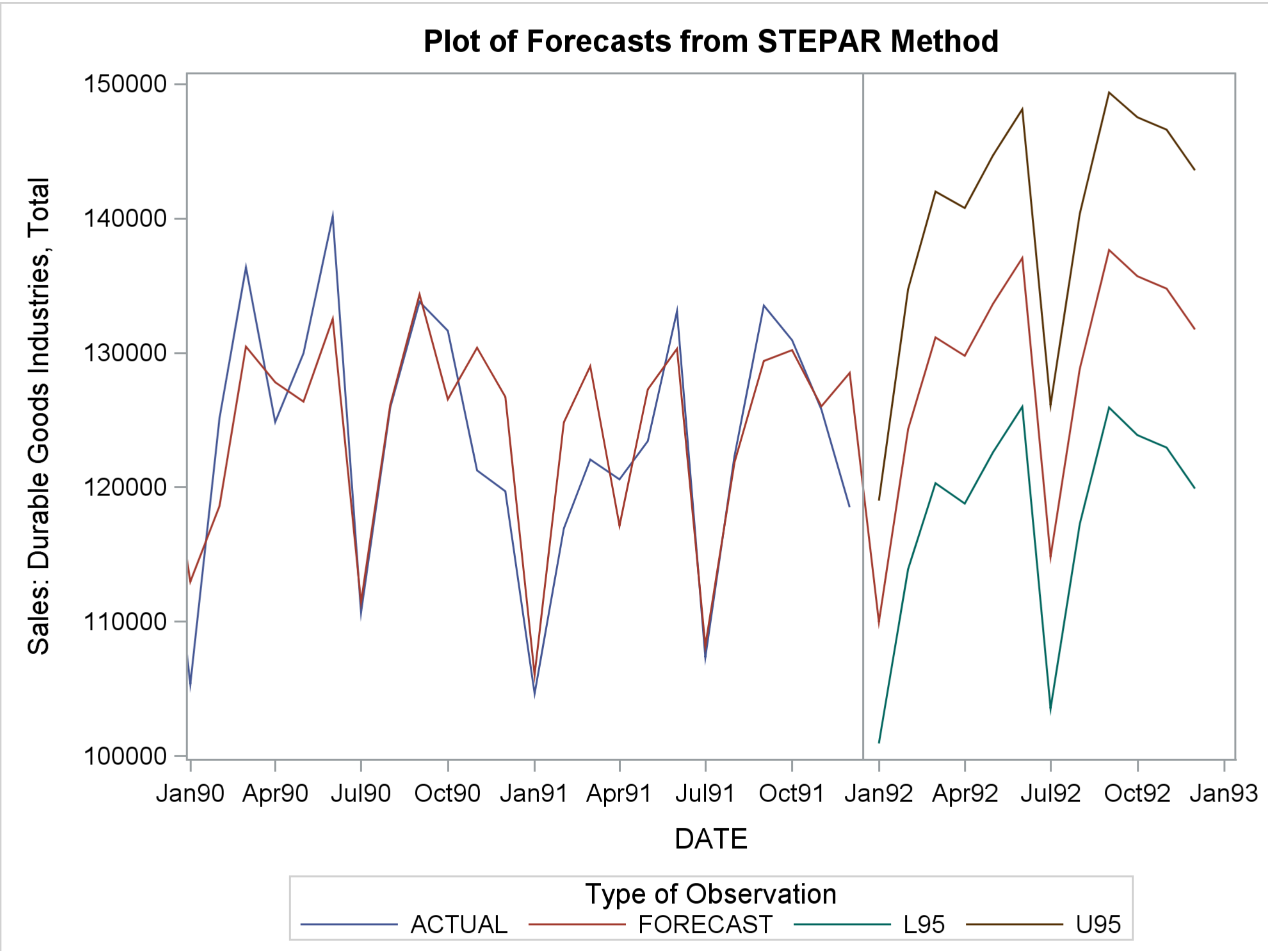
```
title1 'Plot of Forecasts from STEPAR Method';
proc sgplot data=out;
   series x=date y=durables / group=_type_;
   xaxis values=('1jan90'd to '1jan93'd by qtr);
   yaxis values=(100000 to 150000 by 10000);
   refline '15dec91'd / axis=x;
run;

proc sgplot data=out;
   series x=date y=nondur / group=_type_;
   xaxis values=('1jan90'd to '1jan93'd by qtr);
   yaxis values=(100000 to 140000 by 10000);
```

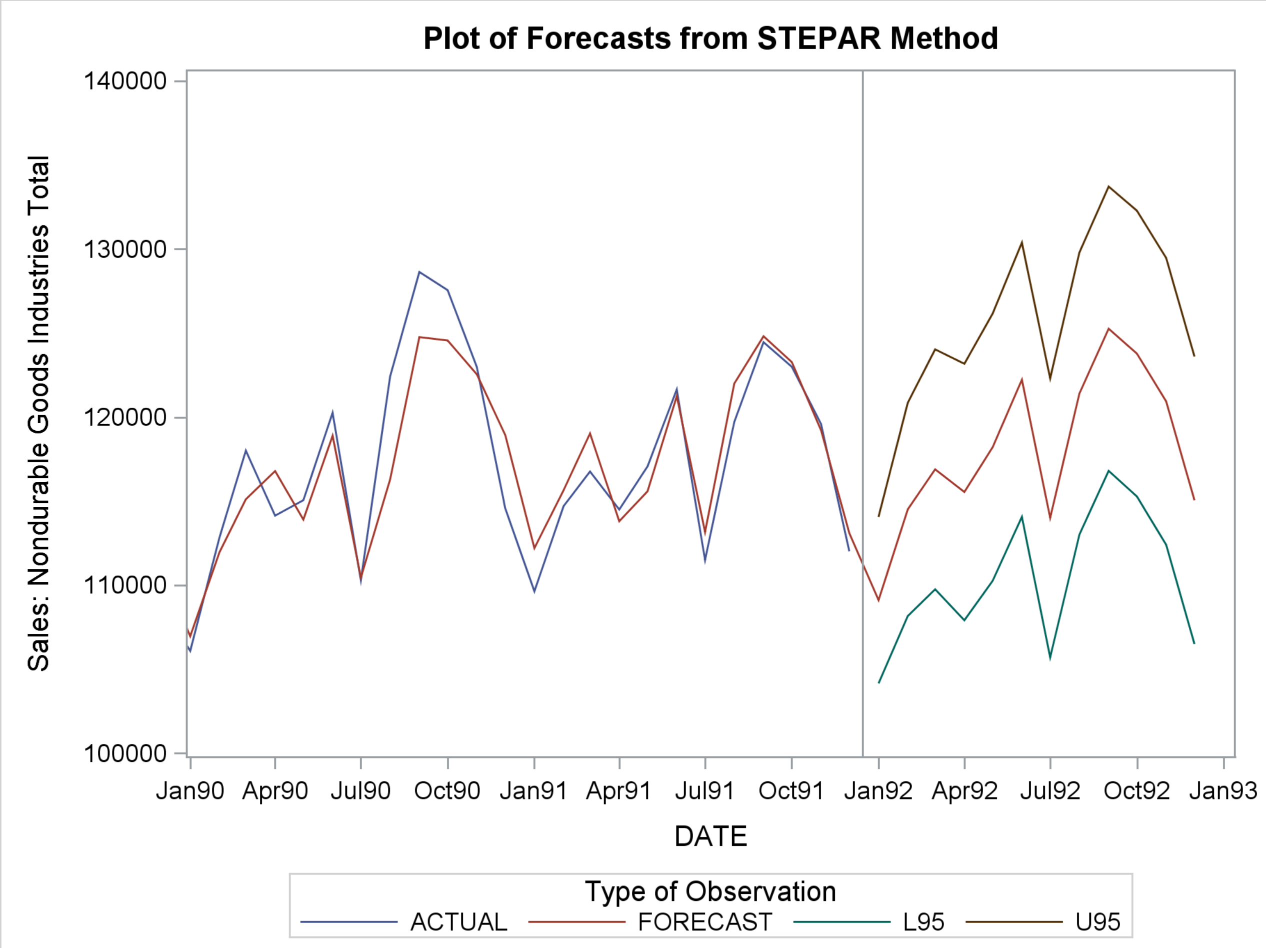
```
    refline '15dec91'd / axis=x;
run;
```

The plots are shown in Output 16.2.4 and Output 16.2.5.

Output 16.2.4 Forecast of Durable Goods Sales



Output 16.2.5 Forecast of Nondurable Goods Sales



Example 16.3: Forecasting Petroleum Sales

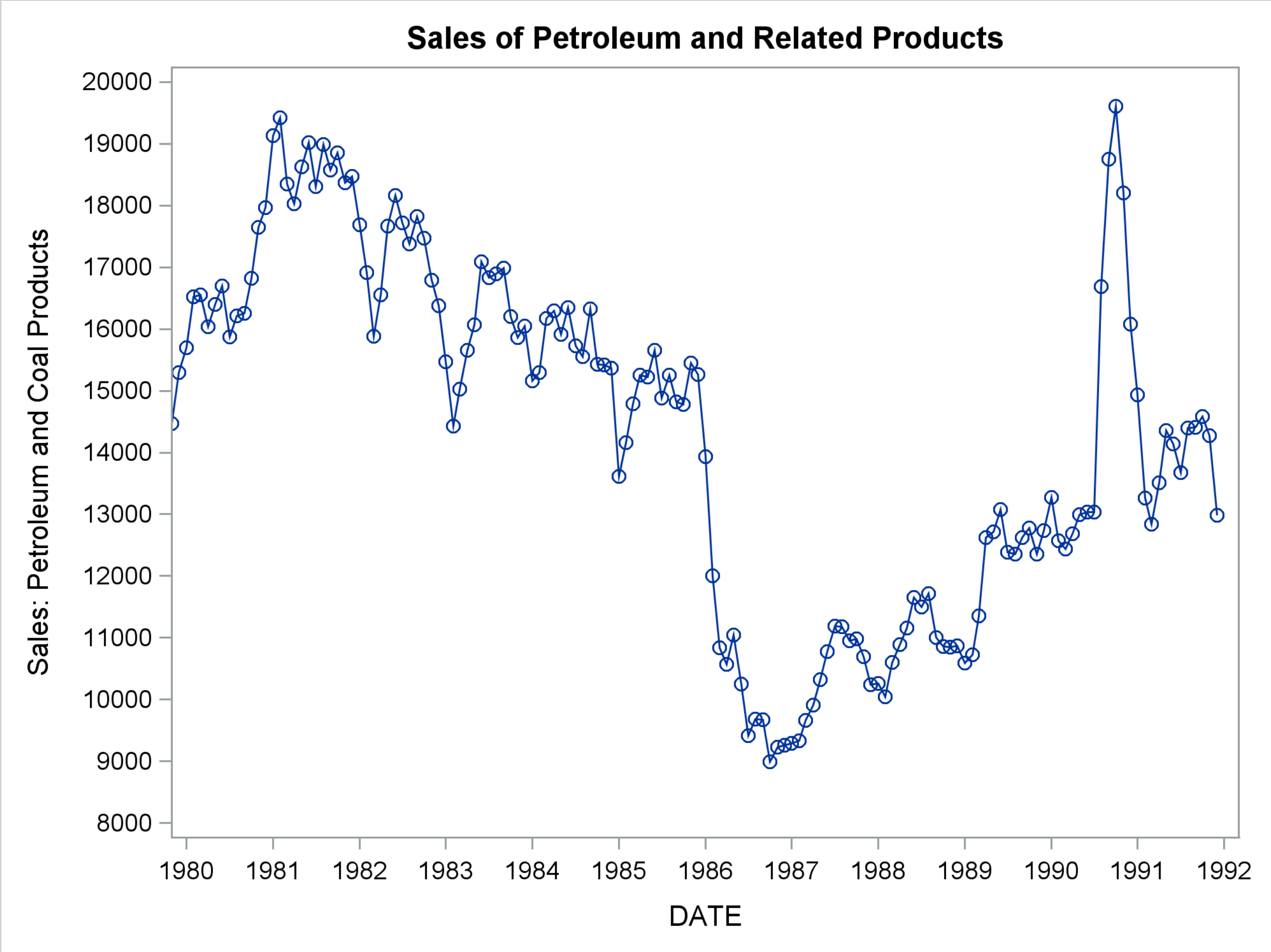
This example uses the double exponential smoothing method to forecast the monthly U. S. sales of petroleum and related products series (PETROL) from the data set SASHELP.USECON. These data are taken from *Business Statistics*, published by the U.S. Bureau of Economic Analysis.

The following statements plot the PETROL series:

```
title1 "Sales of Petroleum and Related Products";
proc sgplot data=sashelp.usecon;
   series x=date y=petrol / markers;
   xaxis values=('1jan80'd to '1jan92'd by year);
   yaxis values=(8000 to 20000 by 1000);
   format date year4.;
run;
```

The plot is shown in Output 16.3.1.

Output 16.3.1 Sales of Petroleum and Related Products



The following statements produce the forecast:

```
proc forecast data=sashelp.usecon interval=month
              method=expo trend=2 lead=12
              out=out outfull outest=est;
   id date;
   var petrol;
   where date >= '1jan80'd;
run;
```

The following statements print the OUTEST= data set:

```
title2 'OUTEST= Data Set: EXPO Method';
proc print data=est;
run;
```

The PROC PRINT listing of the output data set is shown in Output 16.3.2.

Output 16.3.2 The OUTEST= Data Set Produced by PROC FORECAST

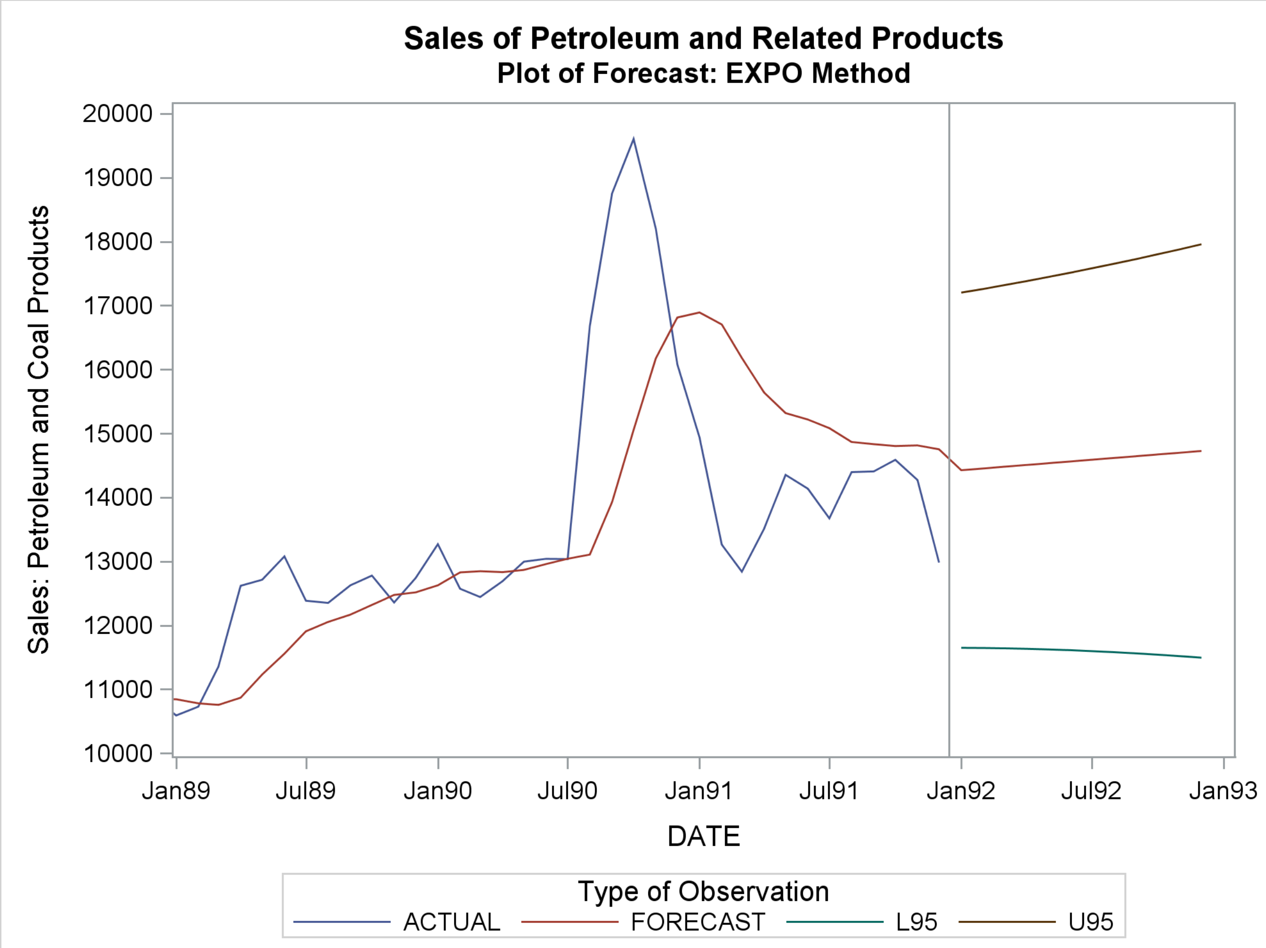
Sales of Petroleum and Related Products
OUTEST= Data Set: EXPO Method

Obs	_TYPE_	DATE	PETROL
1	N	DEC91	144
2	NRESID	DEC91	144
3	DF	DEC91	142
4	WEIGHT	DEC91	0.1055728
5	S1	DEC91	14165.259
6	S2	DEC91	13933.435
7	SIGMA	DEC91	1281.0945
8	CONSTANT	DEC91	14397.084
9	LINEAR	DEC91	27.363164
10	SST	DEC91	1.17001E9
11	SSE	DEC91	233050838
12	MSE	DEC91	1641203.1
13	RMSE	DEC91	1281.0945
14	MAPE	DEC91	6.5514467
15	MPE	DEC91	-0.147168
16	MAE	DEC91	891.04243
17	ME	DEC91	8.2148584
18	RSQUARE	DEC91	0.8008122

The plot of the forecast is shown in Output 16.3.3.

```
title1 "Sales of Petroleum and Related Products";
title2 'Plot of Forecast: EXPO Method';
proc sgplot data=out;
   series x=date y=petrol / group=_type_;
   xaxis values=('1jan89'd to '1jan93'd by qtr);
   yaxis values=(10000 to 20000 by 1000);
   refline '15dec91'd / axis=x;
run;
```

Output 16.3.3 Forecast of Petroleum and Related Products



References

Ahlburg, D. A. (1984), "Forecast Evaluation and Improvement Using Theil's Decomposition," *Journal of Forecasting*, 3, 345–351.

Aldrin, M. and Damsleth, E. (1989), "Forecasting Non-seasonal Time Series with Missing Observations," *Journal of Forecasting*, 8, 97–116.

Archibald, B. C. (1990), "Parameter Space of the Holt-Winters' Model," *International Journal of Forecasting*, 6, 199–209.

Bails, D. G. and Peppers, L. C. (1982), *Business Fluctuations: Forecasting Techniques and Applications*, New Jersey: Prentice-Hall.

Bartolomei, S. M. and Sweet, A. L. (1989), "A Note on the Comparison of Exponential Smoothing Methods for Forecasting Seasonal Series," *International Journal of Forecasting*, 5, 111–116.

Bliemel, F. (1973), “Theil’s Forecast Accuracy Coefficient: A Clarification,” *Journal of Marketing Research*, 10, 444–446.

Bowerman, B. L. and O’Connell, R. T. (1979), *Time Series and Forecasting: An Applied Approach*, North Scituate, MA: Duxbury Press.

Box, G. E. P. and Jenkins, G. M. (1976), *Time Series Analysis: Forecasting and Control*, Rev. Edition, San Francisco: Holden-Day.

Bretschneider, S. I., Carbone, R., and Longini, R. L. (1979), “An Adaptive Approach to Time Series Forecasting,” *Decision Sciences*, 10, 232–244.

Brown, R. G. (1962), *Smoothing, Forecasting, and Prediction of Discrete Time Series*, New York: Prentice-Hall.

Brown, R. G. and Meyer, R. F. (1961), “The Fundamental Theorem of Exponential Smoothing,” *Operations Research*, 9, 673–685.

Bureau of Economic Analysis (1992), *Business Statistics*, 27th Edition, Washington, DC: U.S. Department of Commerce.

Chatfield, C. (1978), “The Holt-Winters Forecasting Procedure,” *Applied Statistics*, 27, 264–279.

Chatfield, C. and Prothero, D. L. (1973), “Box-Jenkins Seasonal Forecasting: Problems in a Case Study,” *Journal of the Royal Statistical Society, Series A*, 136, 295–315.

Chow, W. M. (1965), “Adaptive Control of the Exponential Smoothing Constant,” *Journal of Industrial Engineering*, 16, 314–317.

Cogger, K. O. (1974), “The Optimality of General-Order Exponential Smoothing,” *Operations Research*, 22, 858–867.

Cox, D. R. (1961), “Prediction by Exponentially Weighted Moving Averages and Related Methods,” *Journal of the Royal Statistical Society, Series B*, 23, 414–422.

Fair, R. C. (1986), “Evaluating the Predictive Accuracy of Models,” in Z. Griliches and M. D. Intriligator, eds., *Handbook of Econometrics*, volume 3, New York: North-Holland.

Fildes, R. (1979), “Quantitative Forecasting—the State of the Art: Extrapolative Models,” *Journal of Operational Research Society*, 30, 691–710.

Gardner, E. S., Jr. (1984), “The Strange Case of the Lagging Forecasts,” *Interfaces*, 14, 47–50.

Gardner, E. S., Jr. (1985), “Exponential Smoothing: The State of the Art,” *Journal of Forecasting*, 4, 1–38.

Granger, C. W. J. and Newbold, P. (1977), *Forecasting Economic Time Series*, New York: Academic Press.

Harvey, A. C. (1984), “A Unified View of Statistical Forecasting Procedures,” *Journal of Forecasting*, 3, 245–275.

Ledolter, J. and Abraham, B. (1984), “Some Comments on the Initialization of Exponential Smoothing,” *Journal of Forecasting*, 3, 79–84.

Maddala, G. S. (1977), *Econometrics*, New York: McGraw-Hill.

Makridakis, S. G., Wheelwright, S. C., and McGee, V. E. (1983), *Forecasting: Methods and Applications*, 2nd Edition, New York: John Wiley & Sons.

McKenzie, E. (1984), "General Exponential Smoothing and the Equivalent ARMA Process," *Journal of Forecasting*, 3, 333–344.

Montgomery, D. C. and Johnson, L. A. (1976), *Forecasting and Time Series Analysis*, New York: McGraw-Hill.

Muth, J. F. (1960), "Optimal Properties of Exponentially Weighted Forecasts," *Journal of the American Statistical Association*, 55, 299–306.

Pierce, D. A. (1979), "R^2 Measures for Time Series," *Journal of the American Statistical Association*, 74, 901–910.

Pindyck, R. S. and Rubinfeld, D. L. (1981), *Econometric Models and Econometric Forecasts*, 2nd Edition, New York: McGraw-Hill.

Raine, J. E. (1971), "Self-Adaptive Forecasting Reconsidered," *Decision Sciences*, 2, 181–191.

Roberts, S. A. (1982), "A General Class of Holt-Winters Type Forecasting Models," *Management Science*, 28, 808–820.

Theil, H. (1966), *Applied Economic Forecasting*, Amsterdam: North-Holland.

Trigg, D. W. and Leach, A. G. (1967), "Exponential Smoothing with an Adaptive Response Rate," *Operational Research Quarterly*, 18, 53–59.

Winters, P. R. (1960), "Forecasting Sales by Exponentially Weighted Moving Averages," *Management Science*, 6, 324–342.

Subject Index

Syntax Index