SAS/STAT®
User's Guide
Introduction

2020.1.5*

Chapter 2
Introduction

Contents

Overview of SAS/STAT Software

SAS/STAT software provides comprehensive statistical tools for a wide range of statistical analyses, including analysis of variance, categorical data analysis, cluster analysis, multiple imputation, multivariate analysis, nonparametric analysis, power and sample size computations, psychometric analysis, regression, survey data analysis, and survival analysis. A few examples include nonlinear mixed models, generalized linear models, correspondence analysis, and robust regression. The software is constantly being updated to reflect new methodology. SAS/STAT software provides more than 90 procedures for statistical analysis.

SAS/STAT software also provides high-performance predictive modeling tools that have been specially developed to take advantage of parallel processing in multithreaded single-machine mode. Predictive modeling methods include regression, logistic regression, generalized linear models, generalized additive models, linear mixed models, nonlinear models, and decision trees. The procedures provide model selection, dimension reduction, and identification of important variables whenever this is appropriate for the analysis.

In addition to the high-performance statistical procedures described in this book, SAS/STAT includes high-performance utility procedures, which are described in *Base SAS Procedures Guide: High-Performance Procedures*.

Experimental Software

Experimental software is sometimes included as part of a production-release product. It is provided to (sometimes targeted) customers in order to obtain feedback. All experimental uses are marked Experimental in this document. Whenever an experimental procedure, statement, or option is used, a message is printed to the SAS log to indicate that it is experimental.

The design and syntax of experimental software might change before any production release. Experimental software has been tested prior to release, but it has not necessarily been tested to production-quality standards, and so should be used with care.

About This Book

Since SAS/STAT software is a part of the SAS System, this book assumes that you are familiar with Base SAS software and with the books *SAS Programmers Guide: Essentials* and the *Base SAS Procedures Guide*. It also assumes that you are familiar with basic SAS System concepts such as creating SAS data sets with the DATA step and manipulating SAS data sets with the procedures in Base SAS software (for example, the PRINT and SORT procedures).

Chapter Organization

This book is organized as follows.

Chapter 2, this chapter, provides an overview of SAS/STAT software and summarizes related information, products, and services. The remaining introductory chapters (Chapter 3, "Introduction to Statistical Modeling with SAS/STAT Software," through Chapter 18, "Introduction to Structural Equation Modeling with Latent Variables") provide an introduction to the broad areas covered by SAS/STAT software.

Chapter 19, "Introduction to Power and Sample Size Analysis," provides documentation for the Power and Sample Size Application (PSS).

Chapter 20, "Shared Concepts and Topics," provides information about topics that are common to multiple procedures. Topics include parameterization of model effects, the EFFECT statement, and the NLOPTIONS statement. This chapter also documents the following statements that are used for postfitting analysis and are common across many modeling procedures: EFFECTPLOT, ESTIMATE, LSMEANS, LSMESTIMATE, SLICE, STORE, and TEST.

Chapter 21, "Shared Concepts in High-Performance Computing," describes common concepts that are used by SAS/STAT high-performance procedures.

Chapter 22, "Shared Concepts and Topics in High-Performance Statistical Procedures," describes common syntax elements that are supported by SAS/STAT high-performance procedures.

Chapter 23, "Using the Output Delivery System," explains the fundamentals of using the Output Delivery System (ODS) to manage your SAS output.

Chapter 24, "Statistical Graphics Using ODS," Chapter 25, "ODS Graphics Template Modification," and Chapter 26, "Customizing the Kaplan-Meier Survival Plot," describe the extension to ODS that enables many statistical procedures to create statistical graphics as easily as they create tables.

Subsequent chapters describe the SAS procedures that make up SAS/STAT software. These chapters appear in alphabetical order by procedure name and are organized as follows:

- The "Overview" section provides a brief description of the analysis provided by the procedure.

- The "Getting Started" section provides a quick introduction to the procedure through a simple example.

- The "Syntax" section describes the SAS statements and options that control the procedure.

- The "Details" section discusses methodology and miscellaneous details, such as ODS tables and ODS graphics.

- The "Examples" section contains examples that use the procedure.

- The "References" section contains references for the methodology and for examples of the procedure.

Following the chapters on the SAS/STAT procedures, Appendix A, "Special SAS Data Sets," documents the special SAS data sets that are associated with SAS/STAT procedures and Appendix B, "Sashelp Data Sets," documents Sashelp data sets that are used in this book.

Syntax Conventions

Each procedure's "Syntax" section follows the conventions that are described in this section. Consider the following MODEL and CONTRAST statements from the GLIMMIX procedure:

> **MODEL** *response* < **(** *response-options* **)** > **=** < *fixed-effects* > < */ model-options* > **;**

> **CONTRAST** '*label*' *contrast-specification* < **,** ... > **=** < *fixed-effects* > < */ options* > **;**

These statements demonstrate the syntax conventions that are described in the following list:

UPPERCASE BOLD	is used for keywords in lists of SAS statements and options in "Syntax" sections. When you type a keyword in SAS code, you type it as shown (although any mix of uppercase and lowercase is valid). In the two preceding examples, the statement names (MODEL and CONTRAST) are keywords. Note that keywords are displayed only in uppercase (not bold) when they are used in text.
oblique	is used in syntax definitions and in text to represent arguments for which you supply a value. The preceding MODEL statement indicates that *response*, *response-options*, *fixed-effects*, and *model-options* are arguments for which you can supply values. The values that you can supply are defined later in the description of the MODEL statement.
< >	(angle brackets) identify optional arguments. Arguments that are not enclosed in angle brackets are required. In the preceding MODEL statement, you must supply a value for *response* because it is not enclosed in angle brackets. However, supplying values for *response-options*, *fixed-effects*, and *model-options* is optional.
...	(ellipsis dots) indicate that the preceding argument can be repeated. Sometimes the argument is shown again for clarity. The "..." in the preceding CONTRAST statement indicate that you can supply additional *contrast-specifications*. The comma before the ellipsis dots indicates that each additional *contrast-specification* must be preceded by a comma.
'*value*'	(straight quotes around a *value*) indicate that the *value* must be enclosed in quotation marks (which can be single or double quotes). In the preceding CONTRAST statement, straight quotes around *label* indicate that you must use quotation marks when you specify a label.
( )	(parentheses) indicate arguments that must be grouped together. In the preceding MODEL statement, you must type parentheses around the *response-options*. Some statements do not require parentheses when you specify only one value

for the argument that is shown in parentheses; these cases are indicated in the statement description.

| (vertical bar) indicates that you can choose one value from a group of values. Values that are separated by a vertical bar are mutually exclusive. A vertical bar indicates mutually exclusive values for an option or indicates aliases for an option name.

; (semicolon) indicates the end of a statement.

Other special characters—such as an equal sign (=), tilde ($\sim$), colon (:), and slash (/)—indicate where in the syntax you must type those characters.

Typographical Conventions

This book uses several type styles for presenting information. The following list explains the meaning of the typographical conventions used in this book:

UPPERCASE ROMAN   is used for SAS statements, options, and other SAS language elements when they appear in the text. However, you can enter these elements in your own SAS programs in lowercase, uppercase, or a mixture of the two.

VariableName   is used for the names of variables and data sets when they appear in the text.

bold   is used to refer to matrices and vectors.

italic   is used for terms that are defined in the text, for emphasis, and for references to publications.

`monospace`   is used for example code. In most cases, this book uses lowercase type for SAS code.

Options Used in Examples

The HTMLBLUE style is used to create the graphs and the HTML tables that appear in the online documentation. The PEARLJ style is used to create the PDF tables that appear in the documentation. A style template controls stylistic elements such as colors, fonts, and presentation attributes. You can specify a style template in an ODS destination statement as follows:

```
ods html style=HTMLBlue;
. . .
ods html close;

ods pdf style=PearlJ;
. . .
ods pdf close;
```

For more information about styles, see Chapter 24, "Statistical Graphics Using ODS."

Most of the PDF tables are produced by using the following SAS System option:

```
options papersize=(6.5in 9in);
```

If you run the examples, you might get slightly different output. This is a function of the SAS System options that are used and the precision that your computer uses for floating-point calculations.

Where to Turn for More Information

This section describes other sources of information about SAS/STAT software.

Accessing the SAS/STAT Sample Library

The SAS/STAT sample library includes many examples that illustrate the use of SAS/STAT software. These examples include the examples in this documentation; at the beginning of each example in this documentation, you can find a link to the sample code that is used for it.

You can also access the SAS/STAT sample library at http://support.sas.com/documentation/onlinedoc/stat/ex_code/index.html.

Sashelp Data Sets

SAS provides over 200 data sets in the Sashelp library. These data sets are available for you to use for examples and for testing code. For example, the following step uses the Sashelp.Class data set:

```
proc reg data=sashelp.class;
   model weight = height;
quit;
```

You do not need to provide a DATA step to use Sashelp data sets.

The following steps list all of the data sets that are available in Sashelp:

```
ods select none;
proc contents data=sashelp._all_;
   ods output members=m;
run;
ods select all;

proc print;
   where memtype = 'DATA';
run;
```

The results of these steps (over 200 data set names) are not displayed.

The following steps provide detailed information about the Sashelp data sets:

```
proc contents data=sashelp._all_;
run;
```

The results of this step (hundreds of pages of PROC CONTENTS information) are not displayed. See Chapter B, "Sashelp Data Sets," for more information about Sashelp data sets.

SAS Technical Support Services

As with all SAS products, the SAS Technical Support staff is available to respond to problems and answer technical questions regarding the use of SAS/STAT software. Go to `http://support.sas.com/techsup` for more information.

Related SAS Software

Many features not found in SAS/STAT software are available in other parts of the SAS System. If you do not find something you need in SAS/STAT software, try looking for the feature in the following SAS software products.

SAS/IML Software

SAS/IML software gives you access to a powerful and flexible programming language (Interactive Matrix Language) in a dynamic, interactive environment. The fundamental object of the language is a data matrix. You can use SAS/IML software interactively (at the statement level) to see results immediately, or you can store statements in a module and execute them later. The programming is dynamic because necessary activities such as memory allocation and dimensioning of matrices are done automatically. SAS/IML software is of interest to users of SAS/STAT software because it enables you to program your methods in the SAS System.

Base SAS Software

The features provided by SAS/STAT software are in addition to the features provided by Base SAS software. Many data management and reporting capabilities you will need are part of Base SAS software. See *SAS Programmers Guide: Essentials* and *Base SAS Procedures Guide* for documentation of Base SAS software.

Base SAS software includes many useful SAS procedures. Base SAS procedures are documented in the *Base SAS Procedures Guide*. The following is a list of Base SAS procedures you might find useful:

CORR	computes correlations.
RANK	computes rankings or order statistics.
STANDARD	standardizes variables to a fixed mean and variance.
MEANS	computes descriptive statistics and summarizes or collapses data over cross sections.

| TABULATE | prints descriptive statistics in tabular format. |
| UNIVARIATE | computes descriptive statistics. |

ODS Graphics

Base SAS software provides the following:

- The SG family of procedures provides a simple syntax for creating stand-alone statistical graphics. These procedures include SGPLOT, SGSCATTER, and SGPANEL, which provide a simple and convenient syntax for producing many types of displays. They are particularly convenient for exploring and presenting data. See the *SAS ODS Graphics: Procedures Guide* for more information.

- The GTL (Graph Template Language) and the SGRENDER procedure provide a powerful syntax for creating customized graphs. See the *SAS Graph Template Language: User's Guide* and the *SAS Graph Template Language: Reference* for more information. You can also use the GTL to modify the SAS templates that are provided for use with SAS/STAT procedures. See Chapter 25, "ODS Graphics Template Modification," for more information about template modification.

See Chapter 24, "Statistical Graphics Using ODS," for more information about ODS Graphics.

SAS DATA Step

The DATA step is your primary tool for reading and processing data in the SAS System. The DATA step provides a powerful general purpose programming language that enables you to perform all kinds of data processing tasks. The DATA step is documented in *SAS Programmers Guide: Essentials*.

Base SAS Procedures

Base SAS software includes many useful SAS procedures. Base SAS procedures are documented in the *Base SAS Procedures Guide*. The following is a list of Base SAS procedures you might find useful:

CORR	computes correlations.
RANK	computes rankings or order statistics.
STANDARD	standardizes variables to a fixed mean and variance.
MEANS	computes descriptive statistics and summarizes or collapses data over cross sections.
TABULATE	prints descriptive statistics in tabular format.
UNIVARIATE	computes descriptive statistics.

SAS/ETS Software

SAS/ETS software provides SAS procedures for econometrics and time series analysis. It includes capabilities for forecasting, systems modeling and simulation, seasonal adjustment, and financial analysis and reporting. In addition, SAS/ETS software includes an interactive time series forecasting system.

SAS/GRAPH Software

SAS/GRAPH software includes procedures that create two- and three-dimensional plots and charts.

SAS/OR Software

SAS/OR software provides SAS procedures for operations research and project planning and includes a point-and-click interface to project management. Its capabilities include the following:

- solving transportation problems
- linear, integer, and mixed-integer programming
- nonlinear programming
- scheduling projects
- plotting Gantt charts
- drawing network diagrams
- solving optimal assignment problems
- network flow programming

SAS/OR software might be of interest to users of SAS/STAT software for its mathematical programming features. In particular, the NLP procedure in SAS/OR software solves nonlinear programming problems, and it can be used for constrained and unconstrained maximization of user-defined likelihood functions.

SAS/QC Software

SAS/QC software provides a variety of procedures for statistical quality control and quality improvement. SAS/QC software includes procedures for the following:

- Shewhart control charts

- cumulative sum control charts

- moving average control charts

- process capability analysis

- Pareto charts

- experimental design

SAS/IML Studio

Many users of SAS/STAT software will be interested in SAS/IML Studio, which is new in SAS 9.2 software. Formerly known as SAS Stat Studio, SAS/IML Studio is a tool for data exploration and analysis; it provides a highly flexible programming environment in which you can run SAS/STAT or SAS/IML analyses and display the results with dynamically linked graphics and data tables. You can also move seamlessly between interactive analysis and programatically driven analysis. SAS/IML Studio is intended for data analysts who write SAS programs to solve statistical problems but need more versatility for data exploration and model building.

The programming language in SAS/IML Studio, which is called *IMLPlus*, is an enhanced version of the SAS/IML programming language. IMLPlus extends SAS/IML by providing features such as the ability to create and manipulate dynamically linked graphs and the ability to call SAS procedures.

SAS/IML Studio also includes an interface to the R language. The IMLPlus language provides functions that transfer data between SAS data sets and R data frames, and between SAS/IML matrices and R matrices.

SAS/IML Studio runs on a PC in the Microsoft Windows operating environment. For more information about SAS/IML Studio, see the *SAS/IML Studio: User's Guide* and *SAS/IML Studio for SAS/STAT Users*.

Index