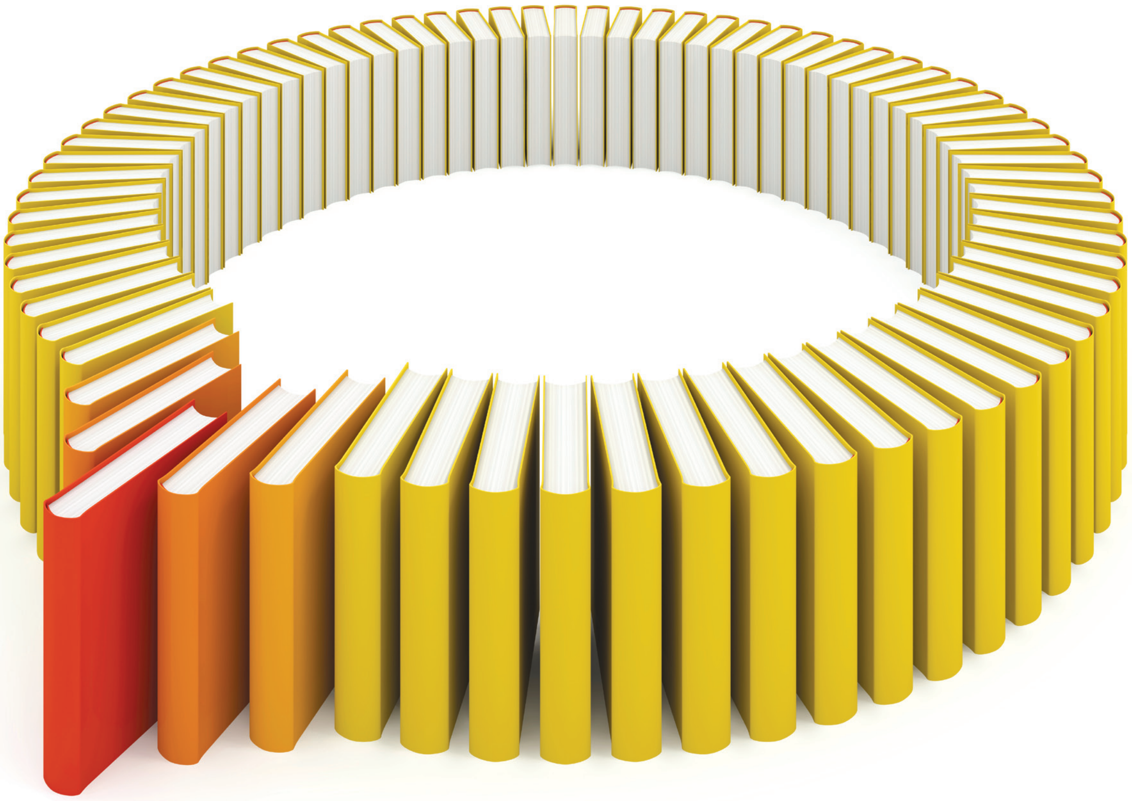
SAS/Genetics™ 13.1
User's Guide

The correct bibliographic citation for this manual is as follows: SAS Institute Inc. 2013. *SAS/Genetics™ 13.1 User's Guide*. Cary, NC: SAS Institute Inc.

SAS/Genetics™ 13.1 User's Guide

SAS Institute Inc., SAS Campus Drive, Cary, North Carolina 27513-2414.

December 2013



Contents

Acknowledgments

Credits

Documentation

Writing	Wendy Czika, Christine Duarte, Padraic Neville, and Xiang Yu
Editing	Ed Huddleston
Technical Review	Yujun Shao and Russell D. Wolfinger
Documentation Production	Tim Arnold

Software

ALLELE	Wendy Czika
BTL	Wendy Czika, Christine Duarte
CASECONTROL	Wendy Czika
FAMILY	Wendy Czika
GENESELECT	Padraic Neville
HAPLOTYPE	Wendy Czika, Xiang Yu
HTSNP	Wendy Czika, Xiang Yu
INBREED	Wendy Czika, Anthony Baiching An, Meltem Narter
PSMOOTH	Wendy Czika
TPLOT Macro	Wendy Czika
TPLOT Results (Frame)	Art Barnes
TPLOT Results (SCL)	Susan E. Haller

Support Groups

Quality Assurance	Jack J. Berry and Yujun Shao
Technical Support	Rob Agnelli and Kathleen Kiernan

Acknowledgments

Many people make significant and continuing contributions to the development of SAS software products. The following are some of those who have contributed significant amounts of their time to help us make improvements to SAS/Genetics software. This includes research and consulting, testing, or reviewing documentation. We are grateful for the involvement of these members of the statistical community and the many others who are not mentioned here for their feedback, suggestions, and consulting.

Peter Boyd	GlaxoSmithKline
Cynthia Coffman	Duke University School of Medicine
Margaret G. Ehm	GlaxoSmithKline
Greg Gibson	North Carolina State University
Shizue Izumi	Radiation Effects Research Foundation
Bret Musser	Merck
Dahlia Nielsen	North Carolina State University
Bruce S. Weir	University of Washington
Dmitri Zaykin	NIEHS

The simulated GAW12 data used in this book are supported by NIGMS grant GM31575.

The final responsibility for the SAS System lies with SAS alone. We hope that you will always let us know your opinions about the SAS System and its documentation. It is through your participation that SAS software is continuously improved.

Please enter your comments at Feedback at http://www.sas.com/statistics/.

Chapter 1
Introduction

Contents

Overview of SAS/Genetics Software

Statistical analyses of genetic data are now central to medicine, agriculture, evolutionary biology, and forensic science. The inherent variation in genetic data, together with the substantial increase in the scale of genetic data following the human genome project, has created a need for reliable computer software to perform these analyses. The procedures offered by SAS/Genetics and described here represent an initial response of SAS Institute to this need.

Although many of the statistical techniques used in the new procedures are standard, others have had to be developed to reflect the genetic nature of the data. All the procedures are designed to operate on data sets that have a familiar structure to geneticists, and that mirror those used in existing software. The syntax for these genetic analyses follows that familiar to SAS users, and the output can be tabular or graphical. The objective of the procedures is to bring the full power of SAS analyses to bear on the characterization of fundamental genetic parameters, and most importantly on the detection of associations between genetic markers and disease status.

Most of the analyses in SAS/Genetics are concerned with detecting patterns of covariation in genetic marker data. These data generally consist of pairs of discrete categories; this pairing derives from the underlying biology, namely the fact that complex organisms have pairs of chromosomes. Each marker refers to the genetic status of a *locus,* each marker type is called an *allele,* and each pair of alleles in an individual is called a *genotype.* A set of alleles present on a single chromosome is called a *haplotype.* Genetic markers can be single nucleotide polymorphisms (SNPs), which are sites in the DNA where the nucleotide varies among individuals, usually with only two alleles possible; microsatellites, which are simple sequence repeats that generate usually between 2 and 20 categories; and other classes of DNA variation.

Two of the procedures in SAS/Genetics are concerned solely with the analysis of genetic marker data. The ALLELE procedure calculates descriptive statistics such as the frequency and variance of alleles and genotypes, as well as estimating measures of marker informativeness, and testing whether genotype frequencies are consistent with Hardy-Weinberg equilibrium (HWE). This procedure also supports four methods for calculation of the degree and significance of *linkage disequilibrium* (LD) among markers at pairs of loci, where LD refers to the propensity of alleles to co-segregate. The HAPLOTYPE procedure is used to infer the most likely multilocus haplotype frequencies in a set of genotypes. Since genetic markers are usually measured independently of one another, there is no direct way to determine which two alleles were on the same chromosome. The algorithm implemented in this procedure converges on the haplotype frequencies that have the highest probability of generating the observed genotypes. These estimated haplotype frequencies can be used as inputs to the HTSNP procedure where *haplotype-tagging SNPs* (htSNPs) that explain much of the haplotype diversity in a block or region can be identified.

Many genetic data sets are now used to study the relationship between genetic markers and complex phenotypes, particularly disease susceptibility. In general terms, traits can be measured as continuous variables (for example, weight or serum glucose concentration), as discrete numerical categories (for example, meristic measures or psychological class), or as affected/unaffected indicator variables. The two procedures CASECONTROL and FAMILY both take simple dichotomous indicators of disease status and use standard algorithms to compute statistics of association between these indicators and the genetic markers. The CASECONTROL procedure is designed to contrast allele and genotype frequencies between affected and unaffected populations, using three types of chi-square tests and options for controlling correlation of allele frequencies among members of the same subpopulation. Significant associations can indicate that the marker is linked to a locus that contributes to disease susceptibility, though population structure in conjunction with environmental or cultural variables can also lead to associations, and the statistical results must be interpreted with caution. The FAMILY procedure employs several transmission/disequilibrium tests of nonrandom association between disease status and linkage to markers transmitted from heterozygous parents to affected offspring (TDT) or pairs of affected and unaffected siblings (S-TDT and SDT). A joint analysis known as the reconstruction-combined TDT (RC-TDT) can also accommodate missing parental genotypes and families lacking unaffected children under some circumstances.

The output of these procedures can be further explored by using the PSMOOTH procedure to adjust p-values from association tests performed on large numbers of markers obtained in a genome scan, or by creating a graphical representation of the procedures' output, namely p-values from tests for LD, HWE, and marker-disease associations, using the %TPLOT macro.

In addition to testing for associations between traits and single markers, testing for effects of multiple markers or epistatic effects between two or more markers might be of interest. PROC HAPLOTYPE can perform testing between a binary trait and several markers at the haplotype level, either looking across all haplotypes at a set of markers or testing each possible haplotype separately. The GENESELECT procedure can determine the best subset of markers, phenotypic variables, and their interactions to include in a model for predicting a trait.

About This Book

Since SAS/Genetics software is a part of the SAS System, this book assumes that you are familiar with Base SAS software and with the books *SAS Language Reference: Dictionary*, *SAS Language Reference: Concepts,* 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 1, this chapter, provides an overview of SAS/Genetics software and summarizes related information, products, and services. The next ten chapters describe the SAS procedures and macro that make up SAS/Genetics software. These chapters appear in alphabetical order by procedure name. They are followed by a chapter documenting a SAS macro provided with SAS/Genetics software.

The chapters documenting the SAS/Genetics procedures 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.
- The *Examples* section contains examples that use the procedure.
- The *References* section contains references for the methodology and examples of the procedure.

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:

roman — is the standard type style used for most text.

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.

UPPERCASE BOLD — is used in the "Syntax" sections' initial lists of SAS statements and options.

oblique — is used for user-supplied values for options in the syntax definitions. In the text, these values are written in *italic*.

helvetica — 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

Output of Examples

For each example, the procedure output is numbered consecutively starting with 1, and each output is given a title. Each page of output produced by a procedure is enclosed in a box. Most of the output shown in this book is produced with the following SAS System options:

```
options linesize=80 pagesize=200 nonumber nodate;
```

In some cases, if you run the examples, you get slightly different output depending on the SAS system options you use and the precision used for floating-point calculations by your computer. This does not indicate a problem with the software. In all situations, any differences should be minor.

Graphics Options

The examples that contain graphical output are created with a specific set of options and symbol statements. The code you see in the examples creates the color graphics that appear in the online (CD) version of this book. A slightly different set of options and statements is used to create the black-and-white graphics that appear in the printed version of the book.

If you run the examples, you might get slightly different results. This can occur because not all graphic options for color devices translate directly to black-and-white output formats. For complete information about SAS/GRAPH software and graphics options, see *SAS/GRAPH Software: Reference*.

The following GOPTIONS statement is used to create the online (color) version of the graphic output.

```
filename GSASFILE '<file-specification>';

goptions gsfname=GSASFILE   gsfmode =replace
         fileonly
         transparency       dev     = gif
         ftext    = swiss   lfactor = 1
         htext    = 4.0pct  htitle  = 4.5pct
         hsize    = 5.625in vsize   = 3.5in
         noborder           cback   = white
         horigin = 0in      vorigin = 0in ;
```

The following GOPTIONS statement is used to create the black-and-white version of the graphic output, which appears in the printed version of this manual.

```
filename GSASFILE '<file-specification>';

goptions gsfname=GSASFILE   gsfmode =replace
         gaccess = sasgaedt fileonly
         dev     = pslepsf
         ftext   = swiss    lfactor = 1
         htext   = 3.0pct   htitle  = 3.5pct
         hsize   = 5.625in  vsize   = 3.5in
         border             cback   = white
         horigin = 0in      vorigin = 0in ;
```

In most of the online examples, the plot symbols are specified as follows:

```
symbol1 value=dot color=white height=3.5pct;
```

The SYMBOL*n* statements used in online examples order the symbol colors as follows: white, yellow, cyan, green, orange, blue, and black.

In the examples appearing in the printed manual, symbol statements specify COLOR=BLACK and order the plot symbols as follows: dot, square, triangle, circle, plus, x, diamond, and star.

Where to Turn for More Information

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

Online Help System

You can access online Help for SAS/Genetics software in two ways. You can select **SAS System Help** from the **Help** menu and then select **SAS/Genetics Software** from the list of available topics. Or you can bring up a command line and issue the command **help Genetics** to access an index to the statistical procedures, or issue the command **help ALLELE** (or another procedure name) to access the Help for that particular procedure. Note that the online Help includes syntax and some essential overview and detail material.

SAS Technical Support Services

As with all SAS Institute products, the SAS Technical Support staff is available to respond to problems and answer technical questions regarding the use of SAS/Genetics software.

Related SAS Software

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

Base SAS Software

The features provided by SAS/Genetics 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. Refer to *SAS Language Reference: Concepts*, *SAS Language Reference: Dictionary*, and the *Base SAS Procedures Guide* for documentation of Base SAS software.

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 Language Reference: Concepts*.

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:

CHART	for printing charts and histograms
CONTENTS	for displaying the contents of SAS data sets
CORR	for computing correlations
FREQ	for computing frequency crosstabulations
MEANS	for computing descriptive statistics and summarizing or collapsing data over cross sections
PRINT	for printing SAS data sets
SORT	for sorting SAS data sets
TABULATE	for printing descriptive statistics in tabular format
TRANSPOSE	for transposing SAS data sets
UNIVARIATE	for computing descriptive statistics

SAS/GRAPH Software

SAS/GRAPH software includes procedures that create two- and three-dimensional high-resolution color graphics plots and charts. You can generate output that graphs the relationship of data values to one another, enhance existing graphs, or simply create graphics output that is not tied to data.

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/Genetics software because it enables you to program your own methods in the SAS System.

SAS/INSIGHT Software

SAS/INSIGHT software is a highly interactive tool for data analysis. You can explore data through a variety of interactive graphs including bar charts, scatter plots, box plots, and three-dimensional rotating plots. You can examine distributions and perform parametric and nonparametric regression, analyze general linear models and generalized linear models, examine correlation matrices, and perform principal component analyses. Any changes you make to your data show immediately in all graphs and analyses. You can also configure SAS/INSIGHT software to produce graphs and analyses tailored to the way you work.

SAS/INSIGHT software might be of interest to users of SAS/Genetics software for interactive graphical viewing of data, editing data, exploratory data analysis, and checking distributional assumptions.

SAS/STAT Software

SAS/STAT software includes procedures for a wide range of statistical methodologies including the following:

- logistic and linear regression
- censored regression
- principal component analysis
- variance component analysis
- cluster analysis
- contingency table analysis
- categorical data analysis: log-linear and conditional logistic models
- general linear models
- linear and nonlinear mixed models
- generalized linear models
- multiple hypothesis testing

SAS/STAT software is of interest to users of SAS/Genetics software because many statistical methods for analyzing genetics data not included in SAS/Genetics software are provided in SAS/STAT software.

Chapter 2
The ALLELE Procedure

Contents

Overview: ALLELE Procedure

The ALLELE procedure performs preliminary analyses on genetic marker data. These analyses serve to characterize the markers themselves or the population from which they were sampled, and can also serve as the basis for joint analyses on markers and traits. A *genetic marker* is any heritable unit that obeys the laws of transmission genetics, and the analyses presented here assume the marker genotypes are determined without error. With an underlying assumption of random sampling, the analyses rest on the multinomial distribution of marker alleles, and many standard statistical techniques can be invoked with little modification.

The ALLELE procedure uses the notation and concepts described by Weir (1996); this is the reference for all equations and methods not otherwise cited.

Data are usually collected at the genotypic level, but interest is likely to be centered on the constituent alleles, so the first step is to construct tables of allele and genotype frequencies. When alleles are independent within individuals—that is, when there is Hardy-Weinberg equilibrium (HWE)—analyses can be conducted at the allelic level. For this reason the ALLELE procedure allows for Hardy-Weinberg testing, although testing is also recommended as a means for detecting possible errors in data.

PROC ALLELE calculates the PIC, heterozygosity, and allelic diversity measures that serve to give an indication of marker informativeness. Such measures can be useful in determining which markers to use for further linkage or association testing with a trait. High values of these measures are a sign of marker informativeness, which is a desirable property in linkage and association tests.

Associations between markers might also be of interest. PROC ALLELE provides tests and various statistics for the association, also called the linkage disequilibrium, between each pair of markers. These statistics can be formed either by using haplotypes that are given in the data, by estimating the haplotype frequencies, or by using only genotypic information.

Population structure can also be analyzed using the F statistics that are reported by PROC ALLELE to describe functions of the various covariance components related to the population group effects.

Getting Started: ALLELE Procedure

Example

Suppose you have genotyped 25 individuals at five markers. You want to examine some basic properties of these markers, such as whether they are in HWE, how many alleles each has, what genotypes appear in the data, and whether there is linkage disequilibrium between any pairs of markers. You have ten columns of data, with the first two columns containing the set of alleles at the first marker, the next two columns containing the set of alleles for the second marker, and so on. There is one row per each individual. You input your data as follows:

```
data markers;
   input (a1-a10) ($);
   datalines;
B  B  A  B  B  B  A  A  B  B
A  A  B  B  A  B  A  B  C  C
B  B  A  A  B  B  B  B  A  C
A  B  A  B  A  B  A  B  A  B
A  A  A  B  A  B  B  B  C  C
B  B  A  A  A  B  A  B  C  C
A  B  B  B  A  B  A  A  A  B
A  B  A  A  A  A  A  A  A  A
B  B  A  A  A  A  A  B  B  B
A  B  A  B  A  B  B  B  A  C
A  A  A  B  A  A  A  B  B  C
B  B  A  B  A  B  A  B  A  C
A  B  B  B  A  A  A  B  A  C
B  B  B  B  A  A  A  A  A  B
A  B  A  A  A  B  A  A  C  C
A  B  A  A  A  B  A  B  C  C
B  B  A  A  A  A  A  B  A  A
A  A  A  B  A  A  A  B  A  B
A  B  A  A  A  A  B  B  C  C
A  A  A  A  A  A  A  A  B  B
A  B  B  B  A  A  A  A  C  C
A  B  A  B  A  B  A  A  B  B
B  B  A  B  A  B  A  A  A  C
A  B  A  A  A  B  A  B  A  C
A  B  B  B  B  B  A  B  B  B
;
```

You can now use PROC ALLELE to examine the frequencies of alleles and genotypes in your data, and see if these frequencies are occurring in proportions you would expect. The following statements perform the analysis you want:

```
proc allele data=markers outstat=ld prefix=Marker
            perms=10000 boot=1000 seed=123;
   var a1-a10;
run;

proc print data=ld;
run;
```

This analysis is using 10,000 permutations to approximate an exact p-value for the HWE test, as well as 1,000 bootstrap samples to obtain the confidence interval for the allele frequencies and one-locus Hardy-Weinberg disequilibrium (HWD) coefficients. The starting seed for the random number generator is 123. The PREFIX= option requests that the five markers be named Marker1–Marker5. Since the BOOTSTRAP= option is specified but the ALPHA= option is omitted, a 95% confidence interval is calculated by default.

All five markers are included in the analysis since the ten variables containing the alleles for those five markers were specified in the VAR statement.

The marker data can alternatively be read in as columns of genotypes instead of columns of alleles by using the GENOCOL and DELIMITER= options in the PROC ALLELE statement, with just one column per each marker. The following DATA step and SAS code could be used to produce the same output by using data in this alternative format:

```
data markers;
   input (g1-g5) ($);
   datalines;
B/B  A/B  B/B  A/A  B/B
A/A  B/B  A/B  A/B  C/C
B/B  A/A  B/B  B/B  A/C

   ... more lines ...

B/B  A/B  A/B  A/A  A/C
A/B  A/A  A/B  A/B  A/C
A/B  B/B  B/B  A/B  B/B
;

proc allele data=markers outstat=ld prefix=Marker
            perms=10000 boot=1000 seed=123 genocol delimiter='/';
   var g1-g5;
run;

proc print data=ld;
run;
```

Note that the DELIMITER= option, which indicates the character or string that separates the alleles that compose a genotype, could have been omitted in this example since '/' is the default.

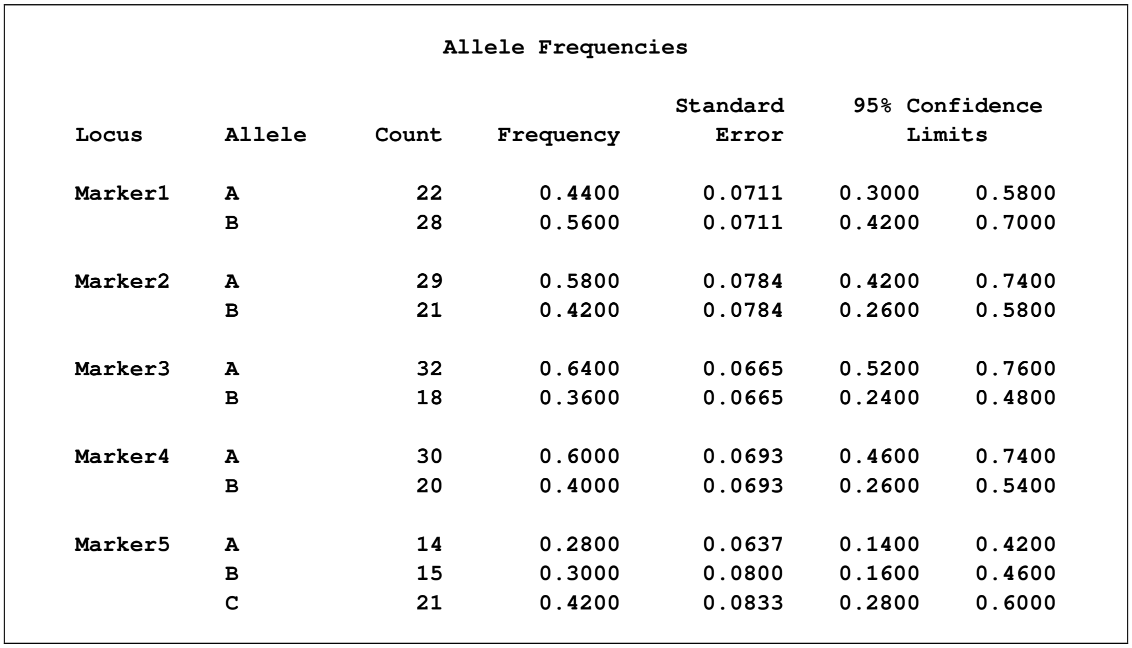
The results from the analysis are shown in Figure 2.1 through Figure 2.4.

Figure 2.1 Marker Summary for the ALLELE Procedure

```
                               The ALLELE Procedure

                                  Marker Summary

                                                     ----------Test for HWE---------
         Number  Number
             of      of          Hetero-   Allelic      Chi-            Pr >    Prob
Locus     Indiv Alleles      PIC zygosity Diversity     Square      DF  ChiSq  Exact

Marker1      25       2 0.3714    0.4800    0.4928     0.0169       1 0.8967 1.0000
Marker2      25       2 0.3685    0.3600    0.4872     1.7041       1 0.1918 0.2262
Marker3      25       2 0.3546    0.4800    0.4608     0.0434       1 0.8350 1.0000
Marker4      25       2 0.3648    0.4800    0.4800     0.0000       1 1.0000 1.0000
Marker5      25       3 0.5817    0.4400    0.6552     9.3537       3 0.0249 0.0106
```

Figure 2.1 displays information about the five markers. From this output, you can conclude that Marker5 is the only one showing significant departure from HWE.

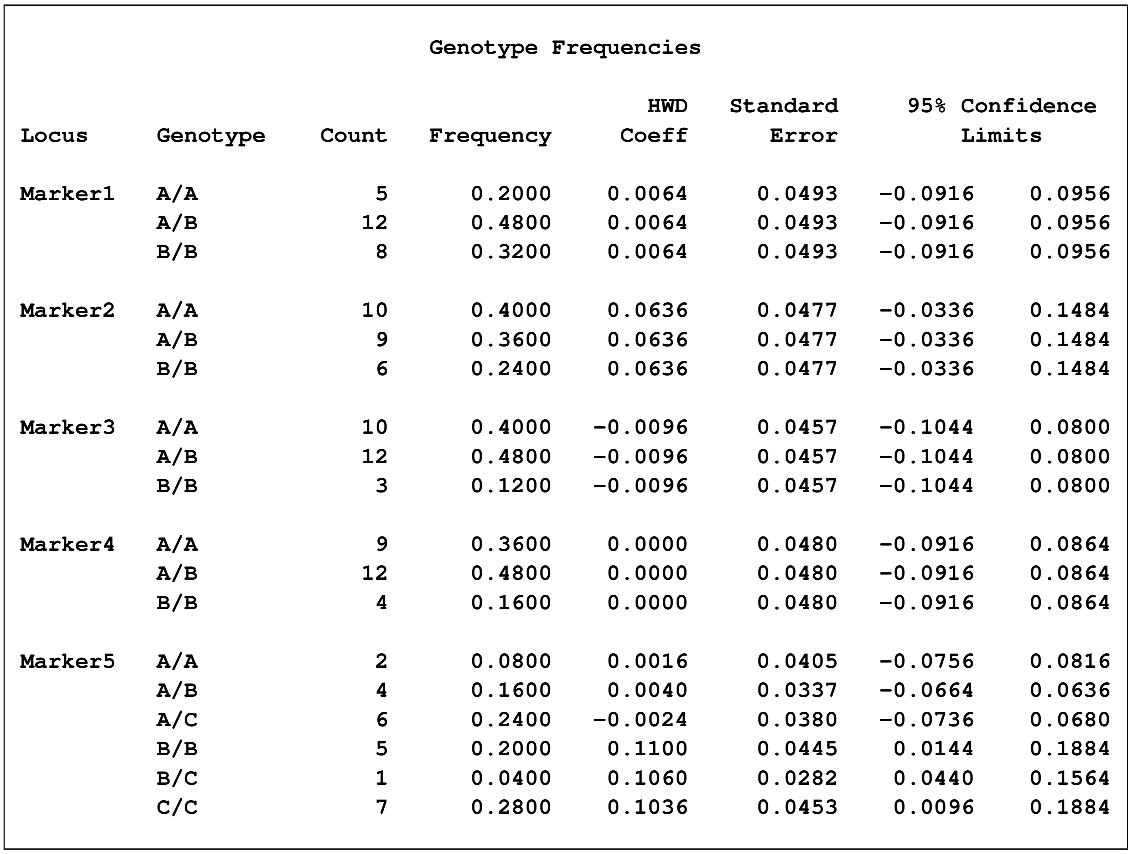
Figure 2.2 Allele Frequencies for the ALLELE Procedure



```
                              Allele Frequencies

                                                    Standard      95% Confidence
Locus         Allele       Count      Frequency        Error          Limits

Marker1       A               22         0.4400       0.0711     0.3000     0.5800
              B               28         0.5600       0.0711     0.4200     0.7000

Marker2       A               29         0.5800       0.0784     0.4200     0.7400
              B               21         0.4200       0.0784     0.2600     0.5800

Marker3       A               32         0.6400       0.0665     0.5200     0.7600
              B               18         0.3600       0.0665     0.2400     0.4800

Marker4       A               30         0.6000       0.0693     0.4600     0.7400
              B               20         0.4000       0.0693     0.2600     0.5400

Marker5       A               14         0.2800       0.0637     0.1400     0.4200
              B               15         0.3000       0.0800     0.1600     0.4600
              C               21         0.4200       0.0833     0.2800     0.6000
```

Figure 2.2 displays the allele frequencies for each marker with their standard errors and the lower and upper limits of the 95% confidence interval.

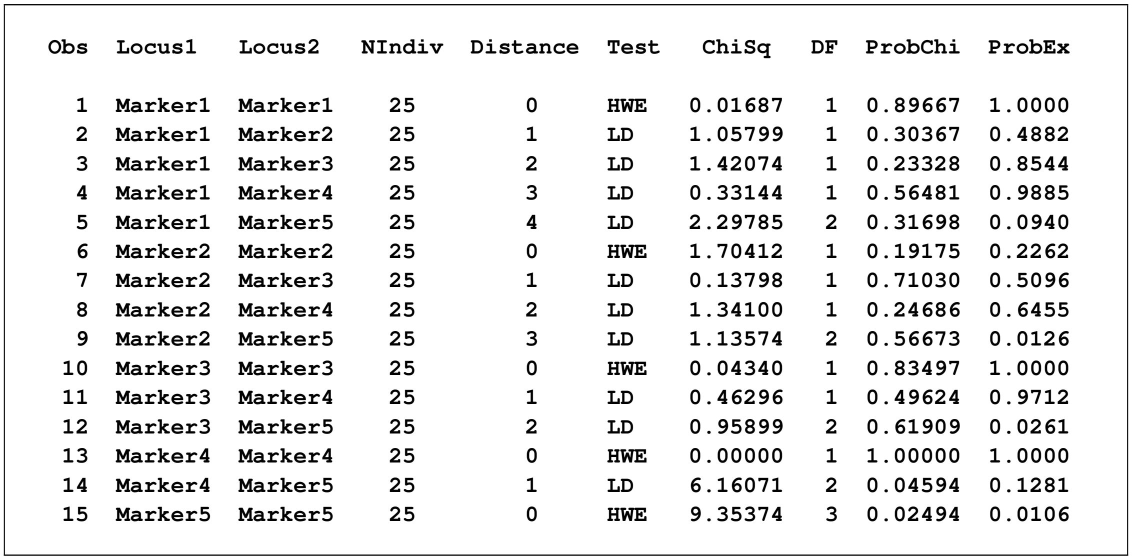
Figure 2.3 Genotype Frequencies for the ALLELE Procedure



```
                             Genotype Frequencies

                                              HWD    Standard      95% Confidence
Locus      Genotype     Count    Frequency      Coeff       Error          Limits

Marker1    A/A              5       0.2000     0.0064      0.0493    -0.0916     0.0956
           A/B             12       0.4800     0.0064      0.0493    -0.0916     0.0956
           B/B              8       0.3200     0.0064      0.0493    -0.0916     0.0956

Marker2    A/A             10       0.4000     0.0636      0.0477    -0.0336     0.1484
           A/B              9       0.3600     0.0636      0.0477    -0.0336     0.1484
           B/B              6       0.2400     0.0636      0.0477    -0.0336     0.1484

Marker3    A/A             10       0.4000    -0.0096      0.0457    -0.1044     0.0800
           A/B             12       0.4800    -0.0096      0.0457    -0.1044     0.0800
           B/B              3       0.1200    -0.0096      0.0457    -0.1044     0.0800

Marker4    A/A              9       0.3600     0.0000      0.0480    -0.0916     0.0864
           A/B             12       0.4800     0.0000      0.0480    -0.0916     0.0864
           B/B              4       0.1600     0.0000      0.0480    -0.0916     0.0864

Marker5    A/A              2       0.0800     0.0016      0.0405    -0.0756     0.0816
           A/B              4       0.1600     0.0040      0.0337    -0.0664     0.0636
           A/C              6       0.2400    -0.0024      0.0380    -0.0736     0.0680
           B/B              5       0.2000     0.1100      0.0445     0.0144     0.1884
           B/C              1       0.0400     0.1060      0.0282     0.0440     0.1564
           C/C              7       0.2800     0.1036      0.0453     0.0096     0.1884
```

Figure 2.3 displays the genotype frequencies for each marker with the associated disequilibrium coefficient, its standard error, and the 95% confidence limits.

Figure 2.4 Testing for Disequilibrium Using the ALLELE Procedure



```
Obs  Locus1   Locus2   NIndiv  Distance  Test   ChiSq   DF  ProbChi  ProbEx

  1  Marker1  Marker1    25        0     HWE   0.01687   1  0.89667  1.0000
  2  Marker1  Marker2    25        1     LD    1.05799   1  0.30367  0.4882
  3  Marker1  Marker3    25        2     LD    1.42074   1  0.23328  0.8544
  4  Marker1  Marker4    25        3     LD    0.33144   1  0.56481  0.9885
  5  Marker1  Marker5    25        4     LD    2.29785   2  0.31698  0.0940
  6  Marker2  Marker2    25        0     HWE   1.70412   1  0.19175  0.2262
  7  Marker2  Marker3    25        1     LD    0.13798   1  0.71030  0.5096
  8  Marker2  Marker4    25        2     LD    1.34100   1  0.24686  0.6455
  9  Marker2  Marker5    25        3     LD    1.13574   2  0.56673  0.0126
 10  Marker3  Marker3    25        0     HWE   0.04340   1  0.83497  1.0000
 11  Marker3  Marker4    25        1     LD    0.46296   1  0.49624  0.9712
 12  Marker3  Marker5    25        2     LD    0.95899   2  0.61909  0.0261
 13  Marker4  Marker4    25        0     HWE   0.00000   1  1.00000  1.0000
 14  Marker4  Marker5    25        1     LD    6.16071   2  0.04594  0.1281
 15  Marker5  Marker5    25        0     HWE   9.35374   3  0.02494  0.0106
```

Figure 2.4 displays the output data set created using the OUTSTAT= option of the PROC ALLELE statement. This data set contains the statistics for testing individual markers for HWE and marker pairs for linkage disequilibrium.

Syntax: ALLELE Procedure

The following statements are available in PROC ALLELE.

PROC ALLELE < *options* > ;
  BY *variables* ;
  POP *variable* < / *options* > ;
  VAR *variables* ;
  WITH *variables* ;

Items within angle brackets (< >) are optional, and statements following the PROC ALLELE statement can appear in any order. The VAR statement is required.

PROC ALLELE Statement

PROC ALLELE < *options* > ;

You can specify the following options in the PROC ALLELE statement.

ALLELEMIN=*number*

AMIN=*number*

indicates that only alleles with a frequency greater than or equal to *number* should be included in the "Allele Frequencies" table. By default, any allele that appears in a nonmissing genotype in the sample is included in the table. The value of *number* must be between 0 and 1.

ALPHA=*number*

specifies that a confidence level of $100(1-\textit{number}\,)\%$ is to be used in forming bootstrap confidence intervals for estimates of allele frequencies and disequilibrium coefficients. The value of *number* must be between 0 and 1, and is set to 0.05 by default.

BOOTSTRAP=*number*

BOOT=*number*

indicates that bootstrap confidence intervals should be formed for the estimates of allele frequencies and one-locus disequilibrium coefficients by using *number* random samples. One thousand samples are usually recommended to form confidence intervals. If this statement is omitted, no confidence limits are reported.

CORRCOEFF

requests that the "Linkage Disequilibrium Measures" table be displayed and contain the correlation coefficient r, a linkage disequilibrium measure.

DATA=*SAS-data-set*

names the input SAS data set to be used by PROC ALLELE. The default is to use the most recently created data set.

DELIMITER=*'string'*

indicates the string that is used to separate the two alleles that compose the genotypes contained in the variables specified in the VAR statement. This option is ignored if GENOCOL is not specified.

DELTA

requests that the "Linkage Disequilibrium Measures" table be displayed and contain the population attributable risk δ, a linkage disequilibrium measure. This option is ignored if HAPLO=NONE or NONEHWD.

DPRIME

requests that the "Linkage Disequilibrium Measures" table be displayed and contain Lewontin's D', a linkage disequilibrium measure.

GENOCOL

indicates that columns specified in the VAR statement contain genotypes instead of alleles. When this option is specified, there is one column per marker. The genotypes must consist of the two alleles separated by a delimiter. For a genotype with one missing allele, use a blank space to indicate a missing value; if both alleles are missing, either use a single missing value for the entire genotype or use the delimiter alone.

GENOMIN=*number*

GMIN=*number*

indicates that only genotypes with a frequency greater than or equal to *number* should be included in the "Genotype Frequencies" table. By default, any genotype that appears at least once in the sample is included in the table. The value of *number* must be between 0 and 1.

HAPLO=NONE

HAPLO=EST

HAPLO=GIVEN

HAPLO=NONEHWD

indicates whether haplotype frequencies should not be used, haplotype frequencies should be estimated, or observed haplotype frequencies in the data should be used. This option affects all linkage disequilibrium tests and measures. By default or when HAPLO=NONE or NONEHWD is specified, the composite linkage disequilibrium (CLD) coefficient is used in place of the usual linkage disequilibrium (LD) coefficient. In addition, the composite haplotype frequencies are used to form the linkage disequilibrium measures indicated by the options CORRCOEFF and DPRIME. When HAPLO=EST, the maximum likelihood estimates of the haplotype frequencies are used to calculate the LD test statistic as well as the LD measures. The HAPLO=GIVEN option indicates that the haplotypes have been observed, and thus the observed haplotype frequencies are used in the LD test statistic and measures.

When HAPLO=GIVEN, haplotypes are denoted in the data in the following manner according to the type of input data used:

- If you omit the TALL option in the PROC ALLELE statement, then all alleles contained in one of an individual's two haplotypes must be in the first of the two variables listed for each marker, and alleles of the other haplotype must be in the second of the two variables listed for each marker. Similarly, if the GENOCOL option is used, the alleles composing one haplotype should all be the first allele listed in each genotype, and alleles of the other haplotype should be listed second.
- If you specify the TALL option, then the alleles that compose one haplotype for an individual must all be in the first variable in the VAR statement, and all the alleles in the other haplotype must be in the second variable in the VAR statement. When the GENOCOL option is also specified, the alleles of one haplotype should all be the first allele listed in the genotype, and alleles of the other haplotype should be listed second.

HAPLOMIN=*number*

HMIN=*number*

indicates that only haplotypes with a frequency greater than or equal to *number* should be included in the "Linkage Disequilibrium Measures" table. By default, any haplotype that appears in the sample (or is estimated to appear at least once) is included in the table. The value of *number* must be between 0 and 1.

INDIVIDUAL=*variable*

INDIV=*variable*

specifies the individual ID variable when using the TALL option. This variable can be character or numeric.

LOGNOTE

requests that notes be written to the log indicating the status of the LD calculations.

MARKER=*variable*

specifies the marker ID variable when using the TALL option. This variable contains the names of the markers that are used in all output and can be character or numeric.

MAXDIST=*number*

specifies the maximum distance possible between a pair of markers in order for linkage disequilibrium calculations to be performed on that pair. If the NDATA= option is not specified or the NDATA= data set does not contain a LOCATION variable, then the distance between a pair of markers is the number of markers apart that they are, assuming that markers are specified in the VAR statement in the physical order in which they appear on a chromosome. For example, if MAXDIST=1 is specified, linkage disequilibrium measures and statistics are calculated only for pairs of markers that are one marker apart, such as M1 and M2, M2 and M3, and so on. If there is an NDATA= data set specified that contains a LOCATION variable, then distances between markers are calculated as differences between values of this variable. Note that markers with missing values for the LOCATION variable are paired with all other markers for LD calculations. The default value is MAXDIST=50.

NDATA=*SAS-data-set*

names the input SAS data set containing names, or identifiers, for the markers used in the output. There must be a NAME variable in this data set, which should contain the same number of rows as there are markers in the input data set specified in the DATA= option. When there are fewer rows than there are markers, markers without a name are named using the PREFIX= option. Likewise, if there is no NDATA= data set specified, the PREFIX= option is used. Note that this data set is ignored if the TALL option is specified in the PROC ALLELE statement. In that case, the marker variable names are taken from the marker ID variable specified in the MARKER= option.

If there is a LOCATION variable in the NDATA= data set, the value given for the MAXDIST= option for LD measures and testing is applied to distances between markers based on values of this variable.

NOFREQ

suppresses the display of the "Allele Frequencies" and the "Genotype Frequencies" tables.

See the section "Displayed Output" on page 28 for a detailed description of these tables.

NOPRINT

suppresses the normal display of results. Note that this option temporarily disables the Output Delivery System (ODS).

OUTSTAT=*SAS-data-set*

names the output SAS data set containing the disequilibrium statistics, for both within-marker and between-marker disequilibria.

PERMS=*number*

EXACT=*number*

indicates that Monte Carlo estimates of the exact p-values for the disequilibrium tests should be calculated using *number* permutations. Large values of *number* (10,000 or more) are usually recommended for accuracy, but long execution times can result, particularly with large data sets. When this option is omitted, no permutations are performed and asymptotic p-values are reported. If HAPLO=EST, then only the exact tests for Hardy-Weinberg equilibrium are performed; the exact tests for linkage disequilibrium cannot be performed since haplotypes are unknown.

PREFIX=*prefix*

specifies a prefix to use in constructing names for marker variables in all output. For example, if PREFIX=VAR, the names of the variables are VAR1, VAR2, . . . , VAR*n*. Note that this option is ignored when the NDATA= option is specified, unless there are fewer names in the NDATA data set than there are markers; it is also ignored if the TALL option is specified, in which case the marker variable names are taken from the marker ID variable specified in the MARKER= option. Otherwise, if this option is omitted, PREFIX=M is the default when variables contain alleles; if GENOCOL is specified, then the names of the variables specified in the VAR statement are used as the marker names.

PROPDIFF

requests that the "Linkage Disequilibrium Measures" table be displayed and contain the proportional difference d, a linkage disequilibrium measure. This option is ignored if HAPLO=NONE or NONEHWD.

RHO

requests that the "Linkage Disequilibrium Measures" table be displayed and contain ρ, a linkage disequilibrium measure, and its information K_ρ. This option is ignored if HAPLO=NONE or NONEHWD.

SEED=*number*

specifies the initial seed for the random number generator used for permuting the data in the exact tests and for the bootstrap samples. The value for *number* must be an integer; the computer clock time is used if the option is omitted or the integer specified is less than or equal to 0. For more details about seed values, see *SAS Language Reference: Concepts*.

TALL

indicates that the input data set is of an alternative format. This format contains the following columns: two containing marker alleles (or one containing marker genotypes if GENOCOL is specified), one for the marker identifier, and one for the individual identifier. The MARKER= and INDIV= options must also be specified for this option to be in effect. Note that when this option is used, the DATA= data set must first be sorted by any BY variables, then sorted by the marker ID variable, and then sorted by the individual ID variable.

YULESQ

requests that the "Linkage Disequilibrium Measures" table be displayed and contain Yule's Q, a linkage disequilibrium measure. This option is ignored if HAPLO=NONE or NONEHWD.

BY Statement

BY *variables* **;**

You can specify a BY statement with PROC ALLELE to obtain separate analyses on observations in groups defined by the BY variables. When a BY statement appears, the procedure expects the input data set to be sorted in the order of the BY variables. The *variables* are one or more variables in the input data set.

If your input data set is not sorted in ascending order, use one of the following alternatives:

- Sort the data by using the SORT procedure with a similar BY statement.
- Specify the BY statement option NOTSORTED or DESCENDING in the BY statement for the ALLELE procedure. The NOTSORTED option does not mean that the data are unsorted but rather that the data are arranged in groups (according to values of the BY variables) and that these groups are not necessarily in alphabetical or increasing numeric order.
- Create an index on the BY variables by using the DATASETS procedure (in Base SAS software).

For more information about the BY statement, see *SAS Language Reference: Concepts*. For more information about the DATASETS procedure, see the *Base SAS Procedures Guide*.

POP Statement

POP *variable < / options >* **;**

One variable can be specified in the POP statement to designate population groups of individuals. By default, the appropriate F statistics, combined across all alleles and loci, are reported in the "Combined F Statistics" table. This statement is ignored if the TALL option is used.

The following options can be specified after a slash (/):

FPERMS=*number*

indicates that *number* permutations of the data should be used to calculate p-values for testing that the F statistics are significantly greater than 0. Large values of *number* (10,000 or more) are usually recommended for accuracy, but long execution times can result, particularly with large data sets. When FPERMS=0 (the default) or the option is omitted, no p-values are reported.

INDIVLOCI

requests that the F statistics be reported for each individual marker locus that is represented in the VAR statement. When this option is specified, the "Marker F Statistics" table is created.

ZEROF

indicates that HWE should be assumed at all loci for the calculation of the F statistics, thus reducing the number of parameters to be estimated.

VAR Statement

VAR *variables* **;**

The VAR statement identifies the variables containing either the marker alleles or, if GENOCOL is specified, the marker genotypes. The following number of variables should be specified in this statement for a data set containing m markers according to whether the options GENOCOL and TALL are used:

- When both GENOCOL and TALL are specified, there should be one variable named containing marker genotypes.
- When only TALL is specified, there should be two variables named containing marker alleles.
- When only GENOCOL is specified, there should be m variables named, one for each marker containing marker genotypes.
- When neither option is specified, there should be $2m$ variables named, two for each marker containing marker alleles.

All variables specified must be of the same type, either character or numeric.

WITH Statement

WITH *variables* **;**

The WITH statement has the same syntax as the VAR statement. It contains variables of alleles, or genotypes if GENOCOL is specified, from markers that you want to pair with those specified in the VAR statement for linkage disequilibrium calculations. Each marker from the VAR statement is paired with each marker from the WITH statement for these two-marker statistics when the WITH statement is specified. The markers represented by variables in this statement are not included in any of the single-marker calculations (marker summary statistics, allele frequencies, or genotype frequencies) and the associated ODS tables. This statement facilitates the parallelization of the LD calculations.

When the WITH statement is used, the MAXDIST= option is ignored. It cannot be used for data in the tall format.

Details: ALLELE Procedure

Statistical Computations

Frequency Estimates

A marker locus **M** can have a series of alleles $M_u, u = 1, \ldots, k$. A sample of n individuals can therefore have several different genotypes at the locus, with n_{uv} copies of type M_u/M_v. The number n_u of copies of allele M_u can be found directly by summation: $n_u = 2n_{uu} + \sum_{v \neq u} n_{uv}$. The sample frequencies are written as $\tilde{p}_u = n_u/(2n)$ and $\tilde{P}_{uv} = n_{uv}/n$. The $\tilde{P}_{uv}$'s are unbiased maximum likelihood estimates (MLEs) of the population proportions P_{uv}.

The variance of the sample allele frequency $\tilde{p}_u$ is calculated as

$$\mathrm{Var}(\tilde{p}_u) = \frac{1}{2n}(p_u + P_{uu} - 2p_u^2)$$

and can be estimated by replacing p_u and P_{uu} with their sample values $\tilde{p}_u$ and $\tilde{P}_{uu}$. The variance of the sample genotype frequency $\tilde{P}_{uv}$ is not generally calculated; instead, an MLE of the HWD coefficient d_{uv} for alleles M_u and M_v is calculated as

$$\hat{d}_{uv} = \begin{cases} \tilde{P}_{uv} - \tilde{p}_u\tilde{p}_v, & u = v \\ \tilde{p}_u\tilde{p}_v - \frac{1}{2}\tilde{P}_{uv}, & u \neq v \end{cases}$$

and the MLE's variance is estimated using one of the following formulas, depending on whether the two alleles are the same or different:

$$\begin{aligned}
\mathrm{Var}(\hat{d}_{uu}) &= \frac{1}{n}\Big[\tilde{p}_u^2(1-\tilde{p}_u)^2 + (1-2\tilde{p}_u)^2\hat{d}_{uu} - \hat{d}_{uu}^2\Big] \\
\mathrm{Var}(\hat{d}_{uv}) &= \frac{1}{2n}\Big\{\tilde{p}_u\tilde{p}_v(1-\tilde{p}_u)(1-\tilde{p}_v) + \sum_{w\neq u,v}(\tilde{p}_u^2\hat{d}_{vw} + \tilde{p}_v^2\hat{d}_{uw}) \\
&\quad - \big[(1-\tilde{p}_u-\tilde{p}_v)^2 - 2(\tilde{p}_u-\tilde{p}_v)^2\big]\hat{d}_{uv} + \tilde{p}_u^2\tilde{p}_v^2 - 2\hat{d}_{uv}^2\Big\}
\end{aligned}$$

The standard error, the square root of the variance, is reported for the sample allele frequencies and the disequilibrium coefficient estimates. When the BOOTSTRAP= option of the PROC ALLELE statement is specified, bootstrap confidence intervals are formed by resampling individuals from the data set and are reported for these estimates, with the $100(1-\alpha)\%$ confidence level given by the ALPHA=α option (or $\alpha = 0.05$ by default).

Measures of Marker Informativeness

Polymorphism Information Content

The polymorphism information content (PIC) measures the probability of differentiating the allele transmitted by a given parent to its child given the marker genotype of father, mother, and child (Botstein et al. 1980). It is computed as

$$\mathrm{PIC} = 1 - \sum_{u=1}^{k}\tilde{p}_u^2 - \sum_{u=1}^{k-1}\sum_{v=u+1}^{k} 2\tilde{p}_u^2\tilde{p}_v^2$$

Heterozygosity

The heterozygosity, sometimes called the observed heterozygosity, is simply the proportion of heterozygous individuals in the data set and is calculated as

$$\mathrm{Het} = 1 - \sum_{u=1}^{k}\tilde{P}_{uu}$$

Allelic Diversity

The allelic diversity, sometimes called the expected heterozygosity, is the expected proportion of heterozygous individuals in the data set when HWE holds. It is calculated as

$$\mathrm{Div} = 1 - \sum_{u=1}^{k}\tilde{p}_u^2$$

Testing for Hardy-Weinberg Equilibrium

Under ideal population conditions, the two alleles an individual receives, one from each parent, are independent so that $P_{uu} = p_u^2$ and $P_{uv} = 2p_u p_v, u \neq v$. The factor of 2 for heterozygotes recognizes the fact that M_u/M_v and M_v/M_u genotypes are generally indistinguishable. This statement about allelic independence within loci is called Hardy-Weinberg equilibrium (HWE). Forces such as selection, mutation, and migration in a population or nonrandom mating can cause departures from HWE. Two methods are used here for testing a marker for HWE, both of which can accommodate any number of alleles. Both methods are testing the hypothesis that $P_{uu} = p_u^2$ and $P_{uv} = 2p_u p_v, u \neq v$ for all $u, v = 1, \ldots, k$.

Chi-Square Goodness-of-Fit Test

The chi-square goodness-of-fit test can be used to test markers for HWE. The chi-square statistic

$$X_T^2 = \sum_u \frac{(n_{uu} - n\tilde{p}_u^2)^2}{n\tilde{p}_u^2} + \sum_u \sum_{v>u} \frac{(n_{uv} - 2n\tilde{p}_u \tilde{p}_v)^2}{2n\tilde{p}_u \tilde{p}_v}$$

has $k(k-1)/2$ degrees of freedom (df), where k is the number of alleles at the marker locus.

Permutation Version of Exact Test

The permutation version of the exact test given by Guo and Thompson (1992) is based on the conditional probability of genotype counts given allelic counts and the hypothesis of allelic independence. The probability of the observed genotype counts under this hypothesis is

$$T = \frac{n!}{(2n)!} \frac{2^h \prod_u n_u!}{\prod_{u,v} n_{uv}!}$$

where $h = \sum_u \sum_{v \neq u} n_{uv}$ is the number of heterozygous individuals. Significance levels are calculated by the Monte Carlo permutation procedure. The $2n$ alleles are randomly permuted the number of times indicated in the PERMS= option to form new sets of n genotypes. The significance level is then calculated as the proportion of times the value of T for each set of permuted data does not exceed the value of T for the actual data. You can indicate the random seed used to randomly permute the data in the SEED= option of the PROC ALLELE statement.

Linkage Disequilibrium (LD)

The set of genetic material an individual receives from each parent contains an allele at every locus, and statements can be made about these allelic combinations, or haplotypes. The probability p_{uv} (called the gametic or haplotype frequency) that an individual receives the haplotype $M_u N_v$ for marker loci **M** and **N** can be compared to the product of the probabilities that each allele is received. The difference is the linkage, or gametic, disequilibrium (LD) coefficient D_{uv} for those two alleles: $D_{uv} = p_{uv} - p_u p_v$. There is a general expectation that the amount of linkage disequilibrium is inversely related to the distance between the two loci, but there are many other factors that can affect disequilibrium. There can even be disequilibrium between alleles at loci that are located on different chromosomes. Note that these tests and measures are calculated only for pairs of markers at most d markers (or the unit used in the LOCATION variable of the NDATA= data set) apart, where d is the value specified in the MAXDIST= option of the PROC ALLELE statement (or 50 by default) when the WITH statement is omitted; otherwise, all pairs of markers containing one marker from the VAR statement and one from the WITH statement are examined.

Table 2.1 displays how the HAPLO= option of the PROC ALLELE statement interacts with the linkage disequilibrium calculations. These calculations are discussed in more detail in the following two sections.

Table 2.1 Interaction of HAPLO= Option with LD Calculations

HAPLO= Option	LD Test Statistic	LD Exact Test	Estimate of Haplotype Freq
GIVEN	$\tilde{D}_{uv}$	Permutes alleles to form new 2-locus haplotypes	Observed freq, $\tilde{p}_{uv}$
EST	$\hat{D}_{uv}$	Not performed	Estimated freq, $\hat{p}_{uv}$
NONE	$\tilde{\Delta}_{uv}$	Permutes alleles to form new 2-locus genotypes	Composite freq, $\tilde{p}^*_{uv}$
NONEHWD	$\tilde{\Delta}_{uv}$	Permutes genotypes to form new 2-locus genotypes	Composite freq, $\tilde{p}^*_{uv}$

Tests

When haplotypes are known, the HAPLO=GIVEN option should be included in the PROC ALLELE statement so that the linkage disequilibrium can be computed directly by substituting the observed frequencies $\tilde{p}_{uv}$, $\tilde{p}_u$, and $\tilde{p}_v$ into the equation in the preceding section for D_{uv}. This creates the MLE, $\tilde{D}_{uv}$, of the LD coefficient between a pair of alleles at different markers. PROC ALLELE calculates an overall chi-square statistic to test that all of the D_{uv}'s between two markers are zero as follows:

$$X_T^2 = \sum_{u=1}^{k} \sum_{v=1}^{l} \frac{(2n)\tilde{D}_{uv}^2}{\tilde{p}_u \tilde{p}_v}$$

which has $(k-1)(l-1)$ degrees of freedom for markers with k and l alleles, respectively.

There is also a Monte Carlo estimate of the exact test available when haplotypes are known. An estimate of the exact p-value for testing the hypothesis in the preceding paragraph can be calculated by conditioning on the allele counts as with the permutation version of the exact test for HWE. The conditional probability of the haplotype counts is then

$$T = \frac{\prod_u n_u! \prod_v n_v!}{(2n)! \prod_{u,v} n_{uv}!}$$

and the significance level is obtained again by permuting the alleles at one locus to form $2n$ new two-locus haplotypes. You can indicate the number of permutations that are used in the PERMS= option of the PROC ALLELE statement and the random seed used to randomly permute the data in the SEED= option of the PROC ALLELE statement.

When it is requested that haplotype frequencies be estimated with the HAPLO=EST option, D_{uv} is estimated using $\hat{D}_{uv} = \hat{p}_{uv} - \tilde{p}_u \tilde{p}_v$, where $\hat{p}_{uv}$ is the MLE of p_{uv} assuming HWE. The estimate $\hat{p}_{uv}$ is calculated according to the method described by Weir and Cockerham (1979). Again, a chi-square test statistic can be calculated to test that all of the D_{uv}'s between a pair of markers are zero as

$$X_T^2 = \sum_{u=1}^{k} \sum_{v=1}^{l} \frac{n\hat{D}_{uv}^2}{\tilde{p}_u \tilde{p}_v}$$

which has $(k-1)(l-1)$ degrees of freedom for markers with k and l alleles, respectively. No exact test is available when haplotype frequencies are estimated.

The HAPLO=NONE and HAPLO=NONEHWD options indicate that haplotypes are unknown and $\hat{D}_{uv}$ should not be used in the tests for LD between pairs of markers. Instead of using the estimated haplotype frequencies which assumes HWE, a test can be formed using the composite linkage disequilibrium (CLD) coefficient Δ_{uv} that does not require this assumption and uses only allele and two-locus genotype frequencies. The MLE $\tilde{\Delta}_{uv}$ of Δ_{uv} can be calculated as described by Weir (1979), and a chi-square statistic that tests all Δ_{uv}'s between a pair of markers are zero can be formed as follows:

$$X_T^2 = \sum_{u=1}^{k} \sum_{v=1}^{l} \frac{n\tilde{\Delta}_{uv}^2}{\tilde{p}_u \tilde{p}_v}$$

which has $(k-1)(l-1)$ degrees of freedom for markers with k and l alleles, respectively. This statistic is used when HAPLO=NONE is specified. When each marker in the pair being analyzed is biallelic, a correction in this test statistic for departures from HWE can be requested with the HAPLO=NONEHWD option. The 1 df chi-square statistic is then represented as

$$X_T^2 = \frac{n\tilde{\Delta}_{uv}^2}{[\tilde{p}_u(1-\tilde{p}_u) + \hat{d}_{uu}][\tilde{p}_v(1-\tilde{p}_v) + \hat{d}_{vv}]}$$

with $u = v = 1$.

Permutation versions of exact tests for CLD are given by Zaykin, Zhivotovsky, and Weir (1995), either assuming HWE or accounting for departures from HWE. The conditional probability of the two-locus genotypes given the one-locus alleles assuming HWE is

$$T = \frac{n! \prod_r n_r! \prod_u n_u! \prod_{r,s,u,v} 2^{n_{rsuv} H_{rsuv}}}{(2n!)^2 \prod_{r,s,u,v} n_{rsuv}!}$$

where n_{rsuv} is the count of $M_r M_s N_u N_v$ genotypes, n_r and n_u are the counts of M_r and N_u alleles, respectively, and H_{rsuv} represents the number of loci that are heterozygous for genotype $M_r M_s N_u N_v$ (0, 1, or 2). An estimate of the exact significance level is obtained by permuting the alleles at both of the loci and counting a permuted sample toward the p-value when its probability T is not larger than for the observed sample.

When departures from HWE are accounted for, the conditional probability of the two-locus genotypes given the one-locus genotypes is

$$T_{HWD} = \frac{\prod_{r,s} n_{rs}! \prod_{u,v} n_{uv}!}{n! \prod_{r,s,u,v} n_{rsuv}!}$$

with n_{rs} and n_{uv} as the counts of M_r/M_s and N_u/N_v genotypes, respectively. An estimate of the exact significance level is obtained by permuting the genotypes at one of the loci and calculating the probability T_{HWD} for each permuted sample. When HAPLO=NONEHWD is specified, the p-value is reported as the proportion of samples that have a T_{HWD} less than or equal to the one from the original sample. **NOTE:** T_{HWD} can be used for multiallelic markers, while the formula for the chi-square statistic cannot. When HAPLO=NONEHWD, the chi-square statistic and asymptotic p-value that are reported for a marker with more than two alleles do not account for departures from HWE; however, the estimate of the exact p-value does make this adjustment as expected.

Measures

PROC ALLELE offers several linkage disequilibrium measures to be calculated for each pair of alleles M_u and N_v located at loci **M** and **N**, respectively. Devlin and Risch (1995) discuss the correlation coefficient r, the population attributable risk δ, Lewontin's D', the proportional difference d, and Yule's Q; Morton et al. (2001) define ρ and its information K_ρ, which is calculated under the null hypothesis that $D = 0$ and also included in the "Linkage Disequilibrium Measures" table when the RHO option is specified. Since these measures are designed for biallelic markers, the measures are calculated for each allele at locus **M** with each allele at locus **N**, where all other alleles at each loci are combined to represent one allele. Thus for each allele M_u in turn, $\tilde{p}_1$ is used as the frequency of allele M_u, and $\tilde{p}_2$ represents the frequency of "not M_u"; similarly for each N_v in turn, $\tilde{q}_1$ represents the frequency of allele N_v, and $\tilde{q}_2$ represents the frequency of "not N_v." All measures have the same numerator, $D = p_{11}p_{22} - p_{12}p_{21}$, the LD coefficient, which can be directly estimated using the observed haplotype frequencies $\tilde{p}_{uv}$ when HAPLO=GIVEN, or estimated using the MLEs of the haplotype frequencies $\hat{p}_{uv}$ assuming HWE when HAPLO=EST. The computations for the measures are as follows:

$$\begin{aligned}
r &= \frac{D}{(p_1 p_2 q_1 q_2)^{1/2}} \\
\delta &= \frac{D}{q_1 p_{22}} \\
D' &= \frac{D}{D_{\max}}, \; D_{\max} = \begin{cases} \min(p_1 q_2, q_1 p_2), & D > 0 \\ \min(p_1 q_1, q_2 p_2), & D < 0 \end{cases} \\
d &= \frac{D}{q_1 q_2} \\
\rho &= \frac{D}{\text{denom}}, \; \text{denom} = \begin{cases} \min(p_1, p_2) \times \max(q_1, q_2), & \min(p_1, p_2) \le \min(q_1, q_2) \\ \min(q_1, q_2) \times \max(p_1, p_2), & \min(p_1, p_2) > \min(q_1, q_2) \end{cases} \\
Q &= \frac{D}{p_{11}p_{22} + p_{12}p_{21}}
\end{aligned}$$

with estimates of measures calculated by replacing parameters with their appropriate estimates. Under the option HAPLO=NONE (the default) or HAPLO=NONEHWD, the numerator D can be replaced by the CLD coefficient Δ, described in the preceding section, for measures r and D'. In place of the preceding formula for the denominator of D', the bounds used for Δ ($\Delta_{\max}$) are given by: Hamilton and Cole (2004); Zaykin (2004). The denominator of the correlation coefficient r is adjusted for departures from HWE when HAPLO=NONEHWD in the same manner as the corresponding chi-square statistic, so that $r = \Delta_{uv}/\{[p_u(1-p_u) + d_{uu}][q_v(1-q_v) + d_{vv}]\}^{1/2}$. The measures δ, d, ρ, and Q cannot be calculated for either of these two options. The information K_ρ is estimated by $nQ(1-R)/(R(1-Q))$, where $Q = \min(p_1, p_2, q_1, q_2)$ and R is the smaller allele frequency ($\min(p_1, p_2)$ or $\min(q_1, q_2)$) at the locus not used for Q.

Population Structure

The genetic structure of populations can be characterized by Wright's F statistics (1951) measuring the degree of relatedness between different types of allele pairs. Cockerham (1969, 1973) defines these same quantities in an analysis-of-variance (ANOVA) framework. For a population hierarchy defined by the variable in the POP statement, these measures include θ_P and, when HWE is not assumed, F and f, corresponding to Wright's F_{ST}, F_{IT}, and F_{IS}, respectively. A weighted average of these measures over loci can be reported as an overall measure, and measures for individual loci can be requested as well. The estimates of these parameters are calculated using an ANOVA structure along with a method-of-moments approach.

For genotypic data with unknown phase from r populations, variation can be partitioned into three sources: between populations, between individuals within populations, and within individuals, with respective observed mean squares MSP, MSI, and MSG. Using the method of moments to equate estimates of the variance components with functions of the observed mean squares, the coancestry coefficients can be estimated as follows:

$$\begin{aligned}
\hat{F} &= 1 - \frac{2n_c MSG}{MSP + (n_c - 1)MSI + n_c MSG} \\
\hat{\theta}_P &= \frac{MSP - MSI}{MSP + (n_c - 1)MSI + n_c MSG} \\
\hat{f} &= \frac{\hat{F} - \hat{\theta}_p}{1 - \hat{F}}
\end{aligned}$$

where $n_c = \frac{1}{r-1}(\sum_{i=1}^{r} n_i - \frac{\sum_i n_i^2}{\sum_i n_i})$ for r populations.

If HWE is assumed in a two-level population hierarchy, the data can be treated as haploid data where allele, not genotype, frequencies are used in the calculations. Also, in this scenario, θ_P and F are equal and $f = 0$. Thus, there is only one parameter to estimate, θ_P, which represents the covariance of alleles from the same population relative to the covariance between alleles from different populations, estimated as follows:

$$\hat{\theta}_P = \frac{MSP - MSG}{MSP + (n_c - 1)MSG}$$

where the counts used in n_c are now in terms of alleles instead of individuals.

Tests of hypotheses that these parameters are 0 can be executed via permutation tests. A different permutation scheme is used for each parameter under each population structure scenario. The schemes displayed in Table 2.2, "Permutation Schemes for Population Structure Parameters," are derived from Excoffier and Lischer (2011).

Table 2.2 Permutation Schemes for Population Structure Parameters

Parameter	f = 0 ?	Permutation Scheme
θ_P	Yes	Individuals among populations
θ_P	No	Individuals among populations
F	No	Alleles among populations
f	No	Alleles within populations

Missing Values

An individual's genotype for a marker is considered missing if at least one of the alleles at the marker is missing. Any missing genotypes are excluded from all calculations, including the linkage disequilibrium statistics for all pairs that include the marker. However, the individual's nonmissing genotypes at other markers can be used as part of the calculations.

If the BOOTSTRAP= option is specified, any individuals with missing genotypes for all markers are excluded from resampling. All other individuals are included, which could result in different numbers of individuals with nonmissing genotypes for the same marker across different samples.

When the POP statement is used, individuals with a missing value for the variable specified are excluded from the calculation of the population F statistics, but they are included in all other analyses.

OUTSTAT= Data Set

The OUTSTAT= data set contains the following variables:

- the BY variables, if any
- Locus1 and Locus2, which contain the pair of markers for which the disequilibrium statistics are calculated
- NIndiv, which contains the number of individuals that have been genotyped at both the markers listed in Locus1 and Locus2 (that is, the number of individuals that have no missing alleles for the two loci)
- Test, which indicates which disequilibrium test is performed, HWE for individual markers (when Locus1 and Locus2 contain the same value) or LD for marker pairs
- ChiSq, which contains the chi-square statistic for testing for disequilibrium. If Locus1 and Locus2 contain the same marker, the test is for HWE within that locus. Otherwise, the test is for linkage disequilibrium between the two loci.
- DF, which contains the degrees of freedom for the chi-square test
- ProbChi, which contains the p-value for the chi-square test
- ProbEx, which contains an estimate of the exact p-value for testing the pair of markers in Locus1 and Locus2 for disequilibrium. This variable is included in the OUTSTAT= data set only when the PERMS= parameter in the PROC ALLELE statement is a positive integer and HAPLO=EST is not specified.

Displayed Output

This section describes the displayed output from PROC ALLELE. See the section "ODS Table Names" on page 29 for details about how this output interfaces with the Output Delivery System.

Marker Summary

The "Marker Summary" table lists information about each of the markers, including the following:

- NIndiv, the number of individuals genotyped at the marker
- NAllele, the number of alleles at the marker
- PIC, the polymorphism information content (PIC) measure
- Het, the heterozygosity measure
- Div, the allelic diversity measure

The table also contains the following columns for the test for HWE:

- ChiSq, the chi-square statistic
- DF, the degrees of freedom for the chi-square test
- ProbChiSq, the p-value for the chi-square test
- ProbExact, an estimate of the exact p-value for the HWE test (only if the PERMS= option is specified in the PROC ALLELE statement)

Allele Frequencies

The "Allele Frequencies" table lists all the observed alleles for each marker, with the observed allele count and frequency, the standard error of the frequency, and when the BOOTSTRAP= option is specified, the bootstrap lower and upper limits of the confidence interval for the frequency based on the confidence level determined by the ALPHA= option of the PROC ALLELE statement (0.95 by default).

Genotype Frequencies

The "Genotype Frequencies" table lists all the observed genotypes (denoted by the two alleles separated by a "/") for each marker, with the observed genotype count and frequency, an estimate of the disequilibrium coefficient d, the standard error of the estimate, and when the BOOTSTRAP= option is specified, the lower and upper limits of the bootstrap confidence interval for d based on the confidence level determined by the ALPHA= option of the PROC ALLELE statement (0.95 by default).

Linkage Disequilibrium Measures

The "Linkage Disequilibrium Measures" table lists for each marker pair the number of individuals with nonmissing genotypes, frequency of each haplotype (observed frequency when HAPLO=GIVEN and estimated frequency otherwise), an estimate of the LD coefficient D_{uv}, and the linkage disequilibrium measures corresponding to the options included in the PROC ALLELE statement (CORRCOEFF, DELTA, DPRIME, PROPDIFF, RHO, and YULESQ). Haplotypes are represented by the allele at the marker locus listed in Locus1 and the allele at the marker locus listed in Locus2, separated by a "-." Note that this table can be quite large when there are many markers or markers with many alleles. For a data set with m markers, each having k_i alleles, $i = 1, \ldots, m$, the number of rows in the table is $\sum_{i=1}^{m-1} \sum_{j=i+1}^{m} k_i k_j$. The MAXDIST= option of the PROC ALLELE statement or the WITH statement can be used to keep this table to a manageable size.

Population Summary

When the POP statement is specified, the "Population Summary" table displays the number and names of populations defined by the population variable.

Combined *F* Statistics

When the POP statement is specified, the "Combined F Statistics" table displays the estimated values of the following statistics combined over all marker loci:

- WithinPopf and OverallF, the inbreeding coefficients within populations and over all populations, respectively, when the ZEROF option is omitted
- PopTheta, the degree of relatedness between individuals within populations

When the value specified for FPERMS is greater than 0, corresponding p-values for each of these estimates are also reported.

Marker *F* Statistics

When the POP statement is specified with the INDIVLOCI option, the "Marker F Statistics" table displays the same columns as those in the "Combined F Statistics" table for each of the marker variables represented in the VAR statement.

ODS Table Names

PROC ALLELE assigns a name to each table it creates, and you must use this name to reference the table when using the Output Delivery System (ODS). These names are listed in Table 2.3.

Table 2.3 ODS Tables Created by the ALLELE Procedure

ODS Table Name	Description	Statement	Option
MarkerSumm	Marker summary		
AlleleFreq	Allele frequencies		
GenotypeFreq	Genotype frequencies		
LDMeasures	Linkage disequilibrium measures	PROC	CORRCOEFF, DELTA, DPRIME, PROPDIFF, RHO, or YULESQ
PopulationSummary	Population summary	POP	
CombinedFStats	Combined *F* statistics	POP	
MarkerFStats	Marker *F* statistics	POP	INDIVLOCI

Examples: ALLELE Procedure

Example 2.1: Using the NDATA= Option with Microsatellites

The following is a subset of data from GAW12 (Wijsman et al. 2001) and contains 17 individuals' genotypes at 14 microsatellite markers.

```
data gaw;
   input id m1-m14 / m15-m28;
   datalines;
 1 11 14  6  8  2  5  9  4  6  1  9  9  9  7
    3  5 10  1  4  6  5  9  1  1  3  5  6  2
 2  2 12  1  4  6  6  3  3  2  1 11 11  4 11
    2  2 13 11  2  1  9  9  1  5  6  1  2  5
 3  2 10  4  8  4  9  2  7  7  1  9  2  7 10
    2  2  7  7  6  8  9  4  5  1  7  2  6  2
 4  5 14  7  3  9 13  4  2  2  4 11  5  4  7
    4  5  7  6  8  2  9  9  1  6  4  1  8  9
 5 12 12  3  8  6  2  1  7  3  5  6 11  6  9
    5  2 13 16  7  1  9  4  1  1  7  1  1  2
 6  4  7  7  8  7 12  4  2  6  5  5 11  5 11
    2  4 15 11  1  1  9  2  6  5  7  6  1  5
 7  2 10  6  8  7  1  2  3  6  2  5  8  5  6
    5  6 13 10  1  8  9  3  1  6  7  7  2  6
 8  2 11  6  2  7  1  2  3  6  6 10 11 11  6
    4  2 11 11  4  5 11  2  3  2  1  4  1  2
 9  2  7  1  1  3  1  5  7  2  5  5 11 11 11
    2  6 11  2  1  6  4  9  5  5  4  2  5  9
10 11 12  2  4 13  3  1  2  4  9  5 10  7  5
    4  4  1  6  8  1  6 10  1  1  2  5  1  1
11 11  2  7  8  1  5  4  6  4  7  5 11 11  6
    5  4 16 13  7  4  5  6  6  1  1  4  1  1
12  2 12  6  8  2  7  3  2  7  5  2  8  9  6
    2  4  7 16  7  1 10  9  5  1  1  4  9  1
```

```
13 13 14  8  3 12 13  7  4  3  2  6 10  9  5
    4  4  2 14  8  8  3  6  5  1  1  6  6  2
14  7 10  6  5 10 13  8  3  5  5  9  9 11  6
    5  4 13 14  1  1  6  9  2  1  5  3  1  2
15 10 11  4  3  9  7  6  3  4  6 10  1  7  9
    2  2  2 14  6  1  9  2  1  1  6  7  5  2
16  2  5  2  7  7  2  2  9  2  2  2  6  9  5
    2  2  7  1  1  2  6  2  1  1  1  1  9  6
17 11  4  4  4  9  1  7  8  5  3  5  1 11  5
    6  5  2 12  1  5  9  9  1  5  7  7  6  1
;
```

The actual names of the markers can be used, by creating a data set with the variable NAME containing these names.

```
data map;
   input name $ location;
   datalines;
D22G001   0.50
D22G002   0.79
D22G003   0.88
D22G004   1.02
D22G005   1.24
D22G006   2.20
D22G007   4.27
D22G008   5.85
D22G009   6.70
D22G010   9.36
D22G011  10.87
D22G012  11.67
D22G013  12.66
D22G014  15.89
;
```

Now an analysis using PROC ALLELE can be performed as follows:

```
proc allele data=gaw ndata=map nofreq perms=10000 seed=456;
   var m1-m28;
run;
```

This analysis produces summary statistics of the 14 markers and is using 10,000 permutations to approximate an exact p-value for the HWE test. The allele and genotype frequency output tables are suppressed with the NOFREQ option.

The results from the analysis are shown in Output 2.1.1. Note the names of the markers that are used.

Output 2.1.1 Summary of Microsatellites for the ALLELE Procedure

```
                      The ALLELE Procedure

                         Marker Summary

            Number     Number
                of         of               Hetero-      Allelic
Locus        Indiv    Alleles       PIC    zygosity    Diversity

D22G001         17          9    0.8384      0.9412       0.8547
D22G002         17          8    0.8296      0.8824       0.8478
D22G003         17         11    0.8749      0.9412       0.8858
D22G004         17          9    0.8259      0.9412       0.8443
D22G005         17          8    0.8272      0.8235       0.8460
D22G006         17          8    0.8257      0.8235       0.8443
D22G007         17          7    0.8012      0.9412       0.8253
D22G008         17          5    0.6665      0.6471       0.7163
D22G009         17         11    0.8788      0.8824       0.8893
D22G010         17          7    0.7572      0.8235       0.7820
D22G011         17          8    0.7274      0.8235       0.7509
D22G012         17          5    0.5661      0.6471       0.6142
D22G013         17          7    0.7965      0.8235       0.8201
D22G014         17          6    0.7507      0.8824       0.7837

                         Marker Summary

                  --------------Test for HWE--------------

                        Chi-                Pr >      Prob
      Locus           Square        DF     ChiSq     Exact

      D22G001        32.5172        36    0.6350    0.8581
      D22G002        28.5222        28    0.4370    0.3868
      D22G003        48.2139        55    0.7295    0.7050
      D22G004        24.9692        36    0.9166    0.8361
      D22G005        20.9416        28    0.8278    0.9413
      D22G006        32.0018        28    0.2744    0.1102
      D22G007        19.7625        21    0.5363    0.5745
      D22G008        11.4619        10    0.3227    0.2525
      D22G009        52.1333        55    0.5849    0.3866
      D22G010        14.7227        21    0.8366    0.8624
      D22G011        19.0400        28    0.8969    0.8898
      D22G012        17.3473        10    0.0670    0.5122
      D22G013        38.8062        21    0.0104    0.0390
      D22G014        17.2802        15    0.3024    0.4651
```

Example 2.2: Computing Linkage Disequilibrium Measures for SNP Data

The following data set contains 44 individuals' genotypes at five SNPs.

```
data snps;
   input s1-s10;
   datalines;
2 2 2 1 2 1 1 1 2 2
2 2 2 2 2 1 1 1 2 2
2 2 2 2 2 1 2 1 2 2
2 2 2 2 . . 1 1 2 2
2 2 2 2 1 2 1 2 2 2
2 2 2 2 . . 2 1 2 2
2 2 2 2 2 1 2 1 2 2
2 2 2 2 . . 2 1 2 2
2 2 2 2 1 1 1 1 2 2
2 2 1 1 2 2 2 1 2 2
2 2 2 1 2 2 2 1 2 2
2 2 2 2 1 1 1 1 2 2
2 2 2 1 2 2 2 2 2 2

   ... more lines ...

2 2 2 1 2 2 2 1 2 2
2 2 2 2 2 2 2 1 2 2
2 2 2 2 2 1 1 1 2 2
2 2 2 2 2 2 1 1 2 2
2 2 2 2 2 1 2 1 2 2
2 2 2 2 2 2 2 2 2 2
2 2 2 2 2 2 2 1 2 2
2 2 2 2 2 2 2 1 2 2
;
```

Now an analysis using PROC ALLELE can be performed as follows:

```
proc allele data=snps prefix=SNP nofreq haplo=est
      corrcoeff dprime;
   var s1-s10;
run;
```

This analysis produces summary statistics of the five SNPs as well as the "Linkage Disequilibrium Measures" table, which contains estimated two-locus haplotype frequencies and disequilibrium coefficients, and the linkage disequilibrium measures r and D'. The allele and genotype frequency output tables are suppressed with the NOFREQ option.

The results from the analysis are shown in Output 2.2.1 and Output 2.2.2. Note the names of the markers that are used.

Output 2.2.1 Summary of SNPs for the ALLELE Procedure

```
                              The ALLELE Procedure

                                 Marker Summary

                                                      -------Test for HWE-------
       Number   Number
           of       of           Hetero-    Allelic        Chi-             Pr >
Locus   Indiv  Alleles      PIC  zygosity  Diversity      Square       DF   ChiSq

SNP1       44        1   0.0000    0.0000     0.0000      0.0000        0    .
SNP2       44        2   0.1190    0.0909     0.1271      3.5627        1  0.0591
SNP3       41        2   0.3283    0.4390     0.4140      0.1493        1  0.6992
SNP4       43        2   0.3728    0.4884     0.4957      0.0093        1  0.9231
SNP5       44        1   0.0000    0.0000     0.0000      0.0000        0    .
```

There are two SNPs that have only one allele appearing in the data.

Output 2.2.2 Linkage Disequilibrium Measures for SNPs Using the ALLELE Procedure

```
                         Linkage Disequilibrium Measures

                  Number
                      of                                 LD      Corr  Lewontin's
Locus1  Locus2     Indiv  Haplotype  Frequency        Coeff     Coeff          D'

SNP1    SNP2          44  2-1           0.0682       0.0000       .             .
SNP1    SNP2          44  2-2           0.9318       0.0000       .             .

SNP1    SNP3          41  2-1           0.2927       0.0000       .             .
SNP1    SNP3          41  2-2           0.7073       0.0000       .             .

SNP1    SNP4          43  2-1           0.5465       0.0000       .             .
SNP1    SNP4          43  2-2           0.4535       0.0000       .             .

SNP1    SNP5          44  2-2           1.0000       0.0000       .             .
---------------------------------------------------------------------------------
SNP2    SNP3          41  1-2           0.0732       0.0214    0.1807      1.0000
SNP2    SNP3          41  2-1           0.2927       0.0214    0.1807      1.0000
SNP2    SNP3          41  2-2           0.6341      -0.0214   -0.1807     -1.0000

SNP2    SNP4          43  1-1           0.0331      -0.0050   -0.0398     -0.1322
SNP2    SNP4          43  1-2           0.0367       0.0050    0.0398      0.1322
SNP2    SNP4          43  2-1           0.5134       0.0050    0.0398      0.1322
SNP2    SNP4          43  2-2           0.4168      -0.0050   -0.0398     -0.1322

SNP2    SNP5          44  1-2           0.0682       0.0000       .             .
SNP2    SNP5          44  2-2           0.9318       0.0000       .             .
---------------------------------------------------------------------------------
SNP3    SNP4          40  1-1           0.2221       0.0608    0.2661      0.4382
SNP3    SNP4          40  1-2           0.0779      -0.0608   -0.2661     -0.4382
SNP3    SNP4          40  2-1           0.3154      -0.0608   -0.2661     -0.4382
SNP3    SNP4          40  2-2           0.3846       0.0608    0.2661      0.4382

SNP3    SNP5          41  1-2           0.2927       0.0000       .             .
SNP3    SNP5          41  2-2           0.7073       0.0000       .             .
---------------------------------------------------------------------------------
SNP4    SNP5          43  1-2           0.5465       0.0000       .             .
SNP4    SNP5          43  2-2           0.4535       0.0000       .             .
```

In Output 2.2.2, the values for the linkage disequilibrium measures are missing for several haplotypes; this occurs when there is only one allele at one of the markers contained in the haplotype, and thus the denominators for these measures are zero. Also note that when the markers are biallelic, the gametic disequilibria have the same absolute values for all four possible haplotypes.

References

Botstein, D., White, R. L., Skolnick, M., and Davis, R. W. (1980), “Construction of a Genetic Linkage Map in Man Using Restriction Fragment Length Polymorphisms,” *American Journal of Human Genetics*, 32, 314–331.

Cockerham, C. C. (1969), “Variance of Gene Frequencies,” *Evolution*, 23, 72–84.

Cockerham, C. C. (1973), “Analyses of Gene Frequencies,” *Genetics*, 74, 679–700.

Devlin, B. and Risch, N. (1995), “A Comparison of Linkage Disequilibrium Measures for Fine-Scale Mapping,” *Genomics*, 29, 311–322.

Excoffier, L. and Lischer, H. (2011), *Arlequin version 3.5: An Integrated Software Package for Population Genetics Data Analysis*.
URL http://cmpg.unibe.ch/software/arlequin35/man/Arlequin35.pdf

Guo, S. W. and Thompson, E. A. (1992), “Performing the Exact Test of Hardy-Weinberg Proportion for Multiple Alleles,” *Biometrics*, 48, 361–372.

Hamilton, D. C. and Cole, D. E. (2004), “Standardizing a Composite Measure of Linkage Disequilibrium,” *Annals of Human Genetics*, 68, 234–239.

Morton, N. E., Zhang, W., Taillon-Miller, P., Ennis, S., Kwok, P.-Y., and Collins, A. (2001), “The Optimal Measure of Allelic Association,” in *Proceedings of the National Academy of Sciences of the United States of America*, volume 98, 5217–5221.

Weir, B. S. (1979), “Inferences about Linkage Disequilibrium,” *Biometrics*, 35, 235–254.

Weir, B. S. (1996), *Genetic Data Analysis II*, Sunderland, MA: Sinauer Associates.

Weir, B. S. and Cockerham, C. C. (1979), “Estimation of Linkage Disequilibrium in Randomly Mating Populations,” *Heredity*, 42, 105–111.

Wijsman, E. M., Almasy, L., Amos, C. I., Borecki, I., Falk, C. T., King, T. M., Martinez, M. M., Meyers, D., Neuman, R., Olson, J. M., Rich, S., Spence, M. A., Thomas, D. C., Vieland, V. J., Witte, J. S., and MacCluer, J. W. (2001), “Analysis of Complex Genetic Traits: Applications to Asthma and Simulated Data,” *Genetic Epidemiology*, 21(Suppl. 1), S1–S853.

Wright, S. (1951), “The Genetical Structure of Populations,” *Annals of Eugenics*, 15, 323–354.

Zaykin, D. (2004), “Bounds and Normalization of the Composite Linkage Disequilibrium Coefficient,” *Genetic Epidemiology*, 27, 252–257.

Zaykin, D. V., Zhivotovsky, L. A., and Weir, B. S. (1995), “Exact Tests for Association between Alleles at Arbitrary Numbers of Loci,” *Genetica*, 96, 169–178.

Chapter 3
The BTL Procedure (Experimental)

Contents

Overview: BTL Procedure

The BTL procedure analyzes marker and trait data in order to find and characterize binary trait loci (BTL). Mixed model analysis of variance is used to find a locus or loci associated with a trait, and a maximum likelihood model is used to estimate the recombination and penetrance parameters for a given set of BTL.

The data consist of marker genotype and a binary trait for a set of individuals. Marker genotypes can be of single nucleotide polymorphisms (SNPs), microsatellite data, or any other kind of marker as long as it is a heritable unit that obeys the laws of transmission genetics. However, only biallelic markers can be used to estimate BTL parameters; multiallelic markers can be used to find BTL but not to estimate parameters.

The output of PROC BTL is a “Model Statistics” table that contains a sorted list of the BTL models evaluated by a user-specified model selection criterion (p-value by default). Additionally, the maximum likelihood parameter estimates for a selected BTL model are written to a “Parameter Estimates” table if the PARMEST statement is used. The penetrance parameters can be calculated for a specified set of recombination parameters, or alternatively, a grid search can be performed over a specified range of possible recombination parameters. Finally, a $100(1-\alpha)\%$ confidence interval can be computed for the parameter estimates of a given model.

Getting Started: BTL Procedure

Example

Suppose you have genotyped 20 members of an experimental backcross population at five markers spanning two chromosomes, and you have also recorded the disease resistance status of each subject as resistant or not resistant. You are interested in finding whether there are BTL in the genetic region spanned by your marker set, and if so, where those BTL are and how strongly linked they are to the disease resistance locus. The first step is to input the data, and then to use PROC BTL with the appropriate options to request all single-marker models of the data to be calculated, as in the following program:

```
data MarkerDat;
   input (m1-m5) ($) trait;
   datalines;
A B B A B 1
B A B B A 1
A B A B B 1
A A B B A 1
A B A B B 1
B A A A B 1
A A B A A 1
B B A B A 1
A A B B A 1
A A A B A 1
A B A A B 0
B A B A A 0
B A A A B 0
B B A B B 0
B A A B B 0
B B A A B 0
A B B A A 0
B A B A A 0
A B A A B 0
B A B A B 0
;
```

```
proc btl data=MarkerDat;
   marker m1-m5 /all=1;
   model trait;
run;
```

The results from the analysis are shown in Figure 3.1.

Figure 3.1 Single-Marker Model Statistics

```
                              The BTL Procedure

                              Model Statistics

      Marker                  Chi-      Pr >
      Effect    DF          Square     ChiSq        AIC        AICC         BIC

      M4         1          6.8470    0.0089       29.2        30.7        32.2
      M1         1          4.5129    0.0336       31.5        33.0        34.5
      M5         1          2.9321    0.0868       33.1        34.6        36.1
      M2         1          1.2289    0.2676       34.8        36.3        37.8
      M3         1          1.2289    0.2676       34.8        36.3        37.8
```

Similarly, all two-marker models can be calculated as follows, with the results shown in Figure 3.2.

```
proc btl data=MarkerDat;
   marker m1-m5 /all=2;
   model trait;
run;
```

Figure 3.2 Two-Marker Model Statistics

```
                              The BTL Procedure

                              Model Statistics

      Marker                  Chi-      Pr >
      Effect    DF          Square     ChiSq        AIC        AICC         BIC

      M1*M4      3         11.0214    0.0116       29.0        33.3        34.0
      M4*M5      3         10.0860    0.0178       30.0        34.3        35.0
      M3*M4      3          8.3508    0.0393       31.7        36.0        36.7
      M1*M2      3          7.6224    0.0545       32.4        36.7        37.4
      M2*M4      3          7.1383    0.0676       32.9        37.2        37.9
      M1*M5      3          6.3399    0.0962       33.7        38.0        38.7
      M1*M3      3          5.0764    0.1663       35.0        39.3        40.0
      M3*M5      3          4.4450    0.2172       35.6        39.9        40.6
      M2*M5      3          3.6150    0.3061       36.4        40.7        41.4
      M2*M3      3          1.3136    0.7259       38.7        43.0        43.7
```

Since m1×m4 appears to be the best two-marker effect, you can then estimate the recombination and penetrance parameters for this BTL model. First you have to enter the mapping information for the markers as follows:

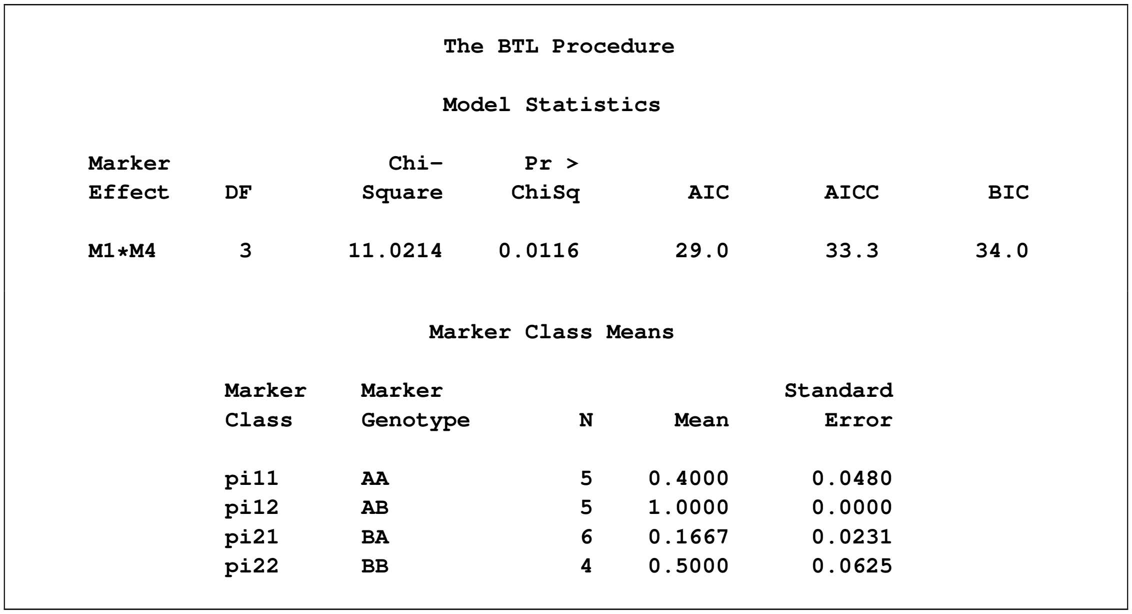
```
data MarkerMap;
   input marker $ chromosome position location;
   datalines;
m1 1 1 0
m2 1 2 4.3
m3 1 3 16
m4 2 1 0
m5 2 2 5.5
;
```

Now you can use the PARMEST statement to request the parameter estimates to be calculated, as in the following code. PROC BTL estimates penetrance values with each recombination parameter *r* set to 0.5.

```
proc btl data=MarkerDat map=MarkerMap;
   marker m1 m4 /group=chromosome;
   model trait;
   parmest cross=b gen=1 r=0.5;
run;
```

Figure 3.3 displays information about the model that includes the two-marker effect m1×m4. The “Parameter Estimates” table shows that penetrance values are not in the valid range (between 0 and 1) for this model with the given values of *r*.

Figure 3.3 Model Statistics, Marker Class Means, and Parameter Estimates for m1*m4 Model



```
                          The BTL Procedure

                          Model Statistics

Marker                 Chi-      Pr >
Effect     DF        Square     ChiSq         AIC        AICC         BIC

M1*M4       3       11.0214    0.0116        29.0        33.3        34.0


                         Marker Class Means

           Marker    Marker                          Standard
           Class     Genotype         N       Mean      Error

           pi11      AA               5     0.4000     0.0480
           pi12      AB               5     1.0000     0.0000
           pi21      BA               6     0.1667     0.0231
           pi22      BB               4     0.5000     0.0625
```

Figure 3.3 *continued*

```
 Parameter Estimates

Parameter    Estimate

r1             0.0000
r2             0.0000
p11            1.6000
p12            4.0000
p21            0.6667
p22            2.0000
theta          0.5000

NOTE: The r and theta
parameters are fixed.
```

Syntax: BTL Procedure

The following statements are available in PROC BTL.

PROC BTL < *options* > ;
  BY *variables* ;
  CLASS *variables* ;
  MARKER *variables* < / *options* > ;
  MODEL *dependent* < = *fixed-effects* > < / *options* > ;
  PARMEST < *options* > ;
  PARMS *(value-list)* . . . < / *options* > ;
  RANDOM *random-effects* < / *options* > ;
  REPEATED < *repeated-effect* > < / *options* > ;
  WEIGHT *variable* ;

Items within angle brackets (< >) are optional. The RANDOM statements can appear multiple times; all other statements can appear only once.

The PROC BTL, MARKER, and MODEL statements are required, and the MODEL statement must appear after the CLASS and/or MARKER statement if either or both statements are included. The RANDOM and REPEATED statements must follow the MODEL statement.

PROC BTL Statement

PROC BTL < *options* > **;**

The PROC BTL statement invokes the procedure. You can specify the following options.

DATA=*SAS-data-set*

names the SAS data set to be used by PROC BTL. The default is the most recently created data set.

MAP=*SAS-data-set*

names the marker map data set to be used. It can include the variables Marker, Location, and Name and a user-designated linkage group variable. The user can specify the name of the numeric linkage group variable in the GROUP= option in the MARKER statement, and this data set must be sorted by this variable. The variable Marker must contain the names of the variables specified in the MARKER statement. If this variable is not in the MAP= data set, the order of the markers in the MARKER statement is used. The Name variable, if present in this data set, is displayed in the “Model Statistics” table when ALL=1 is specified in the MARKER statement. If the Location variable is in the data set, it is used for calculating the recombination parameters between pairs of markers, θ, when the PARMEST statement is given.

For a backcross, A is the default value for the homozygous genotype and B is the default value for the heterozygous genotype. For an F cross, A is the default value for the genotype homozygous in parent 1’s allele, B is the default value for the heterozygous genotype, and C is the default value for the genotype homozygous in parent 2’s allele. If values other than these are used, they need to be specified using the HOMOZYGOUS=, HETEROZYGOUS=, and HOMOZYGOUS2= options in the PARMEST statement.

NOMCMPR

suppresses the display of the “Marker Class Means” table.

NOPARMPR

suppresses the display of the “Parameter Estimates” table.

NOPRINT

suppresses all output.

NOREGPR

suppresses the display of the “Model Statistics” table.

ORDER=DATA

ORDER=FORMATTED

ORDER=FREQ

ORDER=INTERNAL

specifies the sorting order for the levels of all CLASS variables. This ordering determines which parameters in the model correspond to each level in the data.

The default is ORDER=FORMATTED. When the default ORDER=FORMATTED is in effect for numeric variables for which you have supplied no explicit format, the levels are ordered by their internal values.

The following table shows how PROC BTL interprets values of the ORDER= option.

Value of ORDER=	**Levels Sorted By**
DATA	order of appearance in the input data set
FORMATTED	external formatted value, except for numeric variables with no explicit format, which are sorted by their unformatted (internal) value
FREQ	descending frequency count; levels with the most observations come first in the order
INTERNAL	unformatted value

For FORMATTED and INTERNAL, the sort order is machine dependent.

For more information about sorting order, see the chapter on the SORT procedure in the *Base SAS Procedures Guide* and the discussion of BY-group processing in *SAS Language Reference: Concepts*.

BY Statement

BY *variables* **;**

You can specify a BY statement with PROC BTL to obtain separate analyses on observations in groups defined by the BY variables. When a BY statement appears, the procedure expects the input data set to be sorted in the order of the BY variables. The *variables* are one or more variables in the input data set.

If your input data set is not sorted in ascending order, use one of the following alternatives:

- Sort the data by using the SORT procedure with a similar BY statement.
- Specify the BY statement option NOTSORTED or DESCENDING in the BY statement for the BTL procedure. The NOTSORTED option does not mean that the data are unsorted but rather that the data are arranged in groups (according to values of the BY variables) and that these groups are not necessarily in alphabetical or increasing numeric order.
- Create an index on the BY variables by using the DATASETS procedure (in Base SAS software).

Because sorting the data changes the order in which PROC BTL reads observations, the sorting order for the levels of the CLASS variable might be affected if you have specified ORDER=DATA in the PROC BTL statement.

For more information about the BY statement, see *SAS Language Reference: Concepts*. For more information about the DATASETS procedure, see the *Base SAS Procedures Guide*.

CLASS Statement

CLASS *variables* **;**

The CLASS statement names the classification variables to be used in the analysis. If the CLASS statement is used, it must appear before the MODEL statement.

Classification variables can be either character or numeric. By default, class levels are determined from the entire formatted values of the CLASS variables. You can use formats to group values into levels. See the discussion of the FORMAT procedure in the *Base SAS Procedures Guide* and the discussions of the FORMAT statement and SAS formats in *SAS Language Reference: Dictionary*. You can adjust the order of CLASS variable levels by using the ORDER= option in the PROC BTL statement.

You can specify the following option in the CLASS statement after a slash (/):

TRUNCATE

specifies that class levels should be determined using only the first 16 characters of the formatted values of CLASS variables.

MARKER Statement

MARKER *variables < / options >* **;**

The MARKER statement lists the genetics marker variables used in constructing the BTL model. This statement is required, and at least one marker variable must be listed or an error will be generated. The MARKER statement is similar to a class statement, except the interaction effect for all the marker variables listed is implicitly added to the model, as in the following code:

```
proc btl;
   marker m1 m2 m3;
   model trait;
run;
```

The model trait = m1×m2×m3 is fit by PROC BTL. Additionally, if the PARMEST statement is used, these markers are included in the calculated BTL model and the model recombination and penetrance parameters are estimated.

You can specify the following options in the MARKER statement after a slash (/).

ALL=*number*

requests that regression models be calculated for all combinations of *number* markers from the marker statement, where *number* is a positive integer. **NOTE:** If the ALL option is used, the PARMEST statement is ignored if specified. If this option and the ALLUPTO= option are omitted, then only a model including all markers is fit. This option cannot be used if ALLUPTO= is specified.

ALLUPTO=*number*

requests that regression models be calculated for all combinations of *number* and fewer markers from the marker statement; e.g., ALLUPTO=3 calculates all combinations of 3, 2, and 1 markers, where *number* is a positive integer. **NOTE:** If the ALLUPTO option is used, the PARMEST statement is ignored if specified. If this option and the ALL= option are omitted, then only a model including all markers is fit. This option cannot be used if ALL= is specified.

BEST=*number*

requests that only the top *number* models be included in the output, where *number* is a positive integer. By default, all models are displayed. If the GROUP= option is used, *number* specifies the number of models per linkage group to include in the output. The ranking of the models is determined using the model selection criterion selected using the MC= option or the p-value of the likelihood ratio test by default.

GROUP=*variable*

specifies the name of the linkage group variable in the map data set to be used in grouping the models in the "Model Statistics" output if the output contains single-gene models only. The output contains the models sorted by linkage group and by the criterion specified in the MC= option (or the likelihood ratio p-value by default). If the θs are not specified using the THETA option but are calculated using the MAP= data set, then θs between markers from different linkage groups are set to 0.5. **NOTE:** If this variable is specified, markers in the MAP= data set from the same linkage group must be in consecutive observations. Values for this variable must be nonnegative numbers; any negative values are treated as missing.

MC=P

MC=LRCHI

MC=AIC

MC=BIC

MC=AICC

specifies the model selection criterion used to sort models. The default value is the p-value from the likelihood ratio test (MC=P).

MODEL Statement

MODEL *dependent* < = *fixed-effects* > < / *options* > **;**

The MODEL statement names a single dependent variable and the fixed effects. These user-specified fixed effects, along with the interaction effect of all the markers in the MARKER statement (that is implicitly added to the model if the MARKER statement is used), determine the **X** matrix of the mixed model. The specification of effects is the same as in the GLM procedure; however, unlike PROC GLM, you do not specify random effects in the MODEL statement. The MODEL statement is required.

An intercept is included in the fixed-effects model by default. If no fixed effects or markers are specified, only this intercept term is fit. The intercept can be removed by using the NOINT option.

The dependent variable can be of any type for the computation of model statistics; however, it needs to be a binary variable if the PARMEST statement is used with values 0 and 1.

You can specify the following option in the MODEL statement after a slash (/):

NOINT

requests that no intercept be included in the model. An intercept is included by default.

PARMEST Statement

PARMEST < *options* > **;**

The PARMEST statement requests that PROC BTL estimate the recombination and penetrance parameters for a BTL model containing the markers listed in the MARKER statement. The marker class means are calculated from the input data and written to a table. By default a grid search over the range of all possible values of the recombination parameters is performed, and the resulting penetrance values that are in range for each set of rs are displayed. Alternatively, a specific set of r values can be specified using the R option.

NOTE: A grid search over the range of all values of r performed for a model with several markers (> 4) can be lengthy and computationally intensive. The computational time increases as n^k increases, where n is the number of increments of r and k is the number of markers. For this reason, it is recommended that a model with several markers use consecutive grid searches with very few increments (small n) in order to zero in on the correct values of r.

The BTL model that is estimated is specified by the CROSS= option (backcross is the default) and the GEN= option (1 is the default). The θ parameters are calculated from the map data set by using whichever model is specified in the LINKMOD= option (Haldane is the default). Alternatively, a specific set of θ values can be specified using the THETA option. A confidence interval of significance level α can be requested by using the BOOT= option and specifying the number of bootstrap iterations.

You can specify the following options in the PARMEST statement.

ALPHA=*number*

specifies that a confidence level of $100(1-\textit{number}\,)\%$ is to be used in forming bootstrap confidence intervals for the penetrance parameters when the BOOT= option is given. This value of *number* must be between 0 and 1 and is set to 0.05 by default.

BOOT=*number*

requests that confidence intervals be calculated for the penetrance parameters by using *number* iterations of the bootstrap. You must input which recombination parameters (rs) to use in the calculation by using the R option or an error is generated.

CROSS=BACK | B | DH

CROSS=INTER | F

specifies the type of cross for the input data set. The options include BACK or B for a backcross or, equivalently, DH for a doubled-haploid population. The other option is INTER or F for an F intercross. The default is backcross.

GEN=*number*

specifies the generation number of the offspring in the input data set. Valid values include any integer greater than or equal to one. The default is 1.

HETEROZYGOTE=*"heterozygote"*

HE=*"heterozygote"*

specifies the value for the heterozygous genotype used in the input data set. The default value is "B."

HOMOZYGOTE=*"homozygote"*

HO=*"homozygote"*

specifies the value for the homozygous genotype used in the input data set. The default value is "A." If the experimental design is an F cross, then this is the genotype homozygous for the parent 1 allele.

HOMOZYGOTE2=*"homozygote2"*

HO2=*"homozygote2"*

specifies the value for the genotype homozygous in the parent 2 allele used in the input data set. The default value is "C."

LINKMOD=HALDANE | H

LINKMOD=KOSAMBI | K

specifies the model to be used to calculate the marker recombination parameters from the marker location values in the map data set. The options include Haldane and Kosambi. The default value is Haldane.

LINKUNIT=CM | C

LINKUNIT=RECDIST | R

specifies the units used for the marker location variable in the marker data set. The options include centimorgans or recombination distance (kilobases). The default value is centimorgans.

OUTSTAT=*SAS-data-set*

names the SAS data set to be used for the parameter estimates and, when the BOOT= option is specified, confidence intervals.

PMAX=*number*

specifies the highest penetrance value that is considered in range and included in the output. Any real number is a valid value as long as it is greater than PMIN. The PMAX option is ignored if the R option, which precludes a grid search, is used. By default, there is no upper limit for the range of penetrance values included in the output.

PMIN=*number*

specifies the lowest penetrance value that is considered in range and included in the output. Any real number is a valid value as long as it is less than PMAX. The PMIN option is ignored if the R option, which precludes a grid search, is used. By default, there is no lower limit for the range of penetrance values included in the output.

R=*number-list*

specifies the values of r (recombination parameters) used to estimate the penetrance parameters. There is one r for each of the k adjacent marker/BTL pairs, where k is the number of markers in the MARKER statement. A list of values can be given to specify a different r for each pair, or a single value can be specified to be used for all r. If there are fewer than k values specified, the last value given is used for the remaining r. If the R option is used to specify r, the grid search parameters (RSTART, REND, and RINC) are ignored. The R option is required if the BOOT= option is specified. These rs are used to calculate the confidence intervals of the penetrance parameters in the bootstrap calculation. Each r must be a real number greater than or equal to 0 and less than 0.5, and invalid values are replaced by the default value of 0.

REND=_number_

specifies the ending value for each recombination parameter in the grid search. The default value is 0.5. REND must be a real number greater than 0 and less than or equal to 0.5.

RINC=_number_

specifies the increment to be used for the recombination parameter grid search. The default value is 0.1. Any real number greater than 0 and less than or equal to 0.5 is valid.

RSTART=_number_

specifies the starting value for each recombination parameter in the grid search. The default value is 0. RSTART must be a real number from 0 to (but not including) 0.5.

SEED=_number_

specifies the initial seed for the random number generator used for creating the bootstrap samples when the BOOT= option is given. The value for *number* must be an integer; the computer clock time is used if the option is omitted or the integer specified is less than or equal to 0. For more details about seed values, see *SAS Language Reference: Concepts*.

THETA=_number-list_

specifies the values of recombination probabilities between adjacent pairs of markers listed in the MARKER statement. There is one θ for each of the $k - 1$ pairs of adjacent markers, where k is the number of markers specified in the MARKER statement. A list of values can be given to specify a different θ for each pair, or a single value can be specified to be used for all θs. If there are fewer than $k - 1$ values specified, the last value given is used for the remaining θ. **NOTE:** If the MAP= data set is specified and contains the variable Location, the θ values are calculated using these distances and this option is ignored. If locations are not provided in the MAP= data set and this option is omitted, then default values of θ of 0.5 are used. Each θ must be a real number between 0 and 0.5, and invalid values are replaced by the default value of 0.5.

PARMS Statement

PARMS *(value-list) . . . < / options >* **;**

The PARMS statement specifies initial values for the covariance parameters, or it requests a grid search over the range of several values of these parameters.

The *value-list* specification can take any of several forms:

m	a single value
$m_1, m_2, \ldots, m_n$	several values
m to n	a sequence where m equals the starting value, n equals the ending value, and the increment equals 1
m to n by i	a sequence where m equals the starting value, n equals the ending value, and the increment equals i
m_1, m_2 to m_3	mixed values and sequences

You can use the PARMS statement to input known parameters. You can also specify known parameters of **G** by using the GDATA= option in the RANDOM statement.

If you specify more than one set of initial values, PROC BTL performs a grid search of the likelihood surface and uses the best point on the grid for subsequent analysis. Specifying a large number of grid points can result in long computing times. The grid search feature is also useful for exploring the likelihood surface.

The results from the PARMS statement are the values of the parameters on the specified grid (denoted by CovP1–CovP*n*), the residual variance (possibly estimated) for models with a residual variance parameter, and various functions of the likelihood.

You can specify the following options in the PARMS statement after a slash (/).

HOLD=*value-list*

EQCONS=*value-list*

specifies which parameter values PROC BTL should hold to equal the specified values. For example, the statement

```
parms (5) (3) (2) (3) / hold=1,3;
```

constrains the first and third covariance parameters to equal 5 and 2, respectively.

LOWERB=*value-list*

enables you to specify lower boundary constraints on the covariance parameters. The *value-list* specification is a list of numbers or missing values (.) separated by commas. You must list the numbers in the order that PROC BTL uses for the covariance parameters, and each number corresponds to the lower boundary constraint. A missing value instructs PROC BTL to use its default constraint, and if you do not specify numbers for all of the covariance parameters, PROC BTL assumes that the remaining ones are missing.

An example for which this option is useful is when you want to constrain the **G** matrix to be positive definite in order to avoid the more computationally intensive algorithms required when **G** becomes singular. The corresponding code for a random coefficients model is as follows:

```
proc btl;
   class person;
   model y = time;
   random int time / type=fa0(2) sub=person;
   parms / lowerb=1e-4,.,1e-4;
run;
```

Here the FA0(2) structure is used in order to specify a Cholesky root parameterization for the 2 × 2 unstructured blocks in **G**. This parameterization ensures that the **G** matrix is nonnegative definite, and the PARMS statement then ensures that it is positive definite by constraining the two diagonal terms to be greater than or equal to 1E−4.

NOBOUND

requests the removal of boundary constraints on covariance parameters. For example, variance components have a default lower boundary constraint of 0, and the NOBOUND option allows their estimates to be negative.

NOITER

requests that no Newton-Raphson iterations be performed and that PROC BTL use the best value from the grid search to perform inferences. By default, iterations begin at the best value from the PARMS grid search.

NOPROFILE

specifies a different computational method for the residual variance during the grid search. By default, PROC BTL estimates this parameter by using the profile likelihood when appropriate. The NOPROFILE option suppresses the profiling and uses the actual value of the specified variance in the likelihood calculations.

OLS

requests starting values corresponding to the usual general linear model. Specifically, all variances and covariances are set to zero except for the residual variance, which is set equal to its ordinary least-squares (OLS) estimate. This option is useful when the default MIVQUE0 procedure produces poor starting values for the optimization process.

PARMSDATA=*SAS-data-set*

PDATA=*SAS-data-set*

reads in covariance parameter values from a SAS data set. The data set should contain the EST or COVP1–COVP*n* variables.

RATIOS

indicates that ratios with the residual variance are specified instead of the covariance parameters themselves. The default is to use the individual covariance parameters.

UPPERB=*value-list*

enables you to specify upper boundary constraints on the covariance parameters. The *value-list* specification is a list of numbers or missing values (.) separated by commas. You must list the numbers in the order that PROC BTL uses for the covariance parameters, and each number corresponds to the upper boundary constraint. A missing value instructs PROC BTL to use its default constraint, and if you do not specify numbers for all of the covariance parameters, PROC BTL assumes that the remaining ones are missing.

RANDOM Statement

RANDOM *random-effects* < / *options* > **;**

The RANDOM statement defines the random effects constituting the $\boldsymbol{\gamma}$ vector in the mixed model. It can be used to specify traditional variance component models (as in the VARCOMP procedure) and to specify random coefficients. The random effects can be classification or continuous variables, and multiple RANDOM statements are possible.

The purpose of the RANDOM statement is to define the **Z** matrix of the mixed model, the random effects in the $\boldsymbol{\gamma}$ vector, and the structure of **G**. The **Z** matrix is constructed exactly like the **X** matrix for the fixed effects, and the **G** matrix is constructed to correspond with the effects constituting **Z**. The structure of **G** is defined by using the TYPE= option described on page 52.

You can specify INTERCEPT (or INT) as a random effect to indicate the intercept. PROC BTL does not include the intercept in the RANDOM statement by default as it does in the MODEL statement.

You can specify the following options in the RANDOM statement after a slash (/).

GDATA=*SAS-data-set*

requests that the **G** matrix be read in from a SAS data set. This **G** matrix is assumed to be known; therefore, only **R**-side parameters from effects in the REPEATED statement are included in the Newton-Raphson iterations. If no REPEATED statement is specified, then only a residual variance is estimated.

The information in the GDATA= data set can appear in one of two ways. The first is a sparse representation for which you include ROW, COL, and VALUE variables to indicate the row, column, and value of **G**. All unspecified locations are assumed to be 0. The second representation is for dense matrices. In it you include ROW and COL1–COL*n* variables to indicate the row and columns of **G**, which is a symmetric matrix of order *n*. For both representations, you must specify effects in the RANDOM statement that generate a **Z** matrix that contains *n* columns.

If you have more than one RANDOM statement, only one GDATA= option is required in any one of them, and the data set you specify must contain the entire **G** matrix defined by all of the RANDOM statements.

If the GDATA= data set contains variance ratios instead of the variances themselves, then use the RATIOS option.

Known parameters of **G** can also be input using the PARMS statement with the HOLD= option.

GROUP=*effect*

GRP=*effect*

defines an effect specifying heterogeneity in the covariance structure of **G**. All observations having the same level of the group effect have the same covariance parameters. Each new level of the group effect produces a new set of covariance parameters with the same structure as the original group. You should exercise caution in defining the group effect, because strange covariance patterns can result from its misuse. Also, the group effect can greatly increase the number of estimated covariance parameters, which can adversely affect the optimization process.

Continuous variables are permitted as arguments to the GROUP= option. PROC BTL does not sort by the values of the continuous variable; rather, it considers the data to be from a new subject or group whenever the value of the continuous variable changes from the previous observation. Using a continuous variable decreases execution time for models with a large number of subjects or groups and also prevents the production of a large “Class Levels Information” table.

LDATA=*SAS-data-set*

reads the coefficient matrices associated with the TYPE=LIN(*number*) option. The data set must contain the variables PARM, ROW, COL1–COLn, or PARM, ROW, COL, VALUE. The PARM variable denotes which of the *number* coefficient matrices is currently being constructed, and the ROW, COL1–COLn, or ROW, COL, VALUE variables specify the matrix values, as they do with the GDATA= option. Unspecified values of these matrices are set equal to 0.

RATIOS

indicates that ratios with the residual variance are specified in the GDATA= data set instead of the covariance parameters themselves. The default GDATA= data set contains the individual covariance parameters.

SUBJECT=*effect*

SUB=*effect*

identifies the subjects in your mixed model. Complete independence is assumed across subjects; thus, for the RANDOM statement, the SUBJECT= option produces a block-diagonal structure in **G** with identical blocks. The **Z** matrix is modified to accommodate this block-diagonality. In fact, specifying a subject effect is equivalent to nesting all other effects in the RANDOM statement within the subject effect.

Continuous variables are permitted as arguments to the SUBJECT= option. PROC BTL does not sort by the values of the continuous variable; rather, it considers the data to be from a new subject or group whenever the value of the continuous variable changes from the previous observation. Using a continuous variable decreases execution time for models with a large number of subjects or groups.

When you specify the SUBJECT= option and a classification random effect, computations are usually much faster if the levels of the random effect are duplicated within each level of the SUBJECT= effect.

TYPE=*covariance-structure*

specifies the covariance structure of **G**. Although a variety of structures are available, most applications call for either TYPE=VC or TYPE=UN. The TYPE=VC (variance components) option is the default structure, and it models a different variance component for each random effect.

The TYPE=UN (unstructured) option is useful for correlated random coefficient models. For example,

```
random intercept age / type=un subject=person;
```

specifies a random intercept-slope model that has different variances for the intercept and slope and a covariance between them. You can also use TYPE=FA0(2) here to request a **G** estimate that is constrained to be nonnegative definite.

If you are constructing your own columns of **Z** with continuous variables, you can use the TYPE=TOEP(1) structure to group them together to have a common variance component. If you want to have different covariance structures in different parts of **G**, you must use multiple RANDOM statements with different TYPE= options.

REPEATED Statement

REPEATED < *repeated-effect*> < */ options*> **;**

The REPEATED statement is used to specify the **R** matrix in the mixed model. Its syntax is different from that of the REPEATED statement in PROC GLM. If no REPEATED statement is specified, **R** is assumed to be equal to $\sigma^2\mathbf{I}$.

For many repeated measures models, no repeated effect is required in the REPEATED statement. Simply use the SUBJECT= option to define the blocks of **R** and use the TYPE= option to define their covariance structure. In this case, the repeated measures data must be similarly ordered for each subject, and you must indicate all missing response variables with periods in the input data set unless they all fall at the end of a subject's repeated response profile. These requirements are necessary in order to inform PROC BTL of the proper location of the observed repeated responses.

Specifying a repeated effect is useful when you do not want to use periods to indicate missing values in the input data set. The repeated effect must contain only classification variables. Make sure that the levels of the repeated effect are different for each observation within a subject; otherwise, PROC BTL constructs identical rows in **R** corresponding to the observations with the same level. This results in a singular **R** and an infinite likelihood.

Whether you specify a REPEATED effect or not, the rows of **R** for each subject are constructed in the order in which they appear in the input data set.

You can specify the following options in the REPEATED statement after a slash (/).

GROUP=*effect*

GRP=*effect*

defines an effect specifying heterogeneity in the covariance structure of **R**. All observations having the same level of the GROUP effect have the same covariance parameters. Each new level of the GROUP effect produces a new set of covariance parameters with the same structure as the original group. You should exercise caution in properly defining the GROUP effect, because strange covariance patterns can result from its misuse. Also, the GROUP effect can greatly increase the number of estimated covariance parameters, which can adversely affect the optimization process.

Continuous variables are permitted as arguments to the GROUP= option. PROC BTL does not sort by the values of the continuous variable; rather, it considers the data to be from a new subject or group whenever the value of the continuous variable changes from the previous observation. Using a continuous variable decreases execution time for models with a large number of subjects or groups and also prevents the production of a large "Class Levels Information" table.

LDATA=*SAS-data-set*

reads the coefficient matrices associated with the TYPE=LIN(*number*) option. The data set must contain the variables PARM, ROW, COL1–COLn, or PARM, ROW, COL, VALUE. The PARM variable denotes which of the *number* coefficient matrices is currently being constructed, and the ROW, COL1–COLn, or ROW, COL, VALUE variables specify the matrix values, as they do with the RANDOM statement option GDATA=. Unspecified values of these matrices are set equal to 0.

LOCAL

LOCAL=EXP(< effects >)

LOCAL=POM(*POM-data-set* **)**

requests that a diagonal matrix be added to **R**. With just the LOCAL option, this diagonal matrix equals $\sigma^2\mathbf{I}$, and σ^2 becomes an additional variance parameter that PROC BTL profiles out of the likelihood provided that you do not specify the NOPROFILE option in the PROC BTL statement. The LOCAL option is useful if you want to add an observational error to a time series structure (Jones and Boadi-Boateng 1991) or a nugget effect to a spatial structure (Cressie 1991).

The LOCAL=EXP(<*effects*>) option produces exponential local effects, also known as dispersion effects, in a log-linear variance model. These local effects have the form

$$\sigma^2 \text{diag}[\exp(\mathbf{U}‹)]$$

where **U** is the full-rank design matrix corresponding to the effects that you specify, and ‹ are the parameters that PROC BTL estimates. An intercept is not included in **U** because it is accounted for by σ^2. PROC BTL constructs the full-rank **U** in terms of 1s and -1s for classification effects. Be sure to scale continuous effects in **U** sensibly.

The LOCAL=POM(*POM-data-set*) option specifies the power-of-the-mean structure. This structure possesses a variance of the form $\sigma^2|\mathbf{x}_i'\boldsymbol{\beta}^*|^\theta$ for the ith observation, where $\mathbf{x}_i$ is the ith row of $\mathbf{X}$ (the design matrix of the fixed effects), and $\boldsymbol{\beta}^*$ is an estimate of the fixed-effects parameters that you specify in *POM-data-set*.

The SAS data set specified by *POM-data-set* contains the numeric variable Estimate (in previous releases, the variable name was required to be EST), and it has at least as many observations as there are fixed-effects parameters. The first p observations of the Estimate variable in *POM-data-set* are taken to be the elements of $\boldsymbol{\beta}^*$, where p is the number of columns of $\mathbf{X}$. You must order these observations according to the non-full-rank parameterization of the BTL procedure.

LOCALW

specifies that only the local effects and no others be weighted. By default, all effects are weighted. The LOCALW option is used in connection with the WEIGHT statement and the LOCAL option in the REPEATED statement.

NONLOCALW

specifies that only the nonlocal effects and no others be weighted. By default, all effects are weighted. The NONLOCALW option is used in connection with the WEIGHT statement and the LOCAL option in the REPEATED statement.

SUBJECT=*effect*

SUB=*effect*

identifies the subjects in your mixed model. Complete independence is assumed across subjects; therefore, the SUBJECT= option produces a block-diagonal structure in **R** with identical blocks. When the SUBJECT= effect consists entirely of classification variables, the blocks of **R** correspond to observations sharing the same level of that effect. These blocks are sorted according to this effect as well.

Continuous variables are permitted as arguments to the SUBJECT= option. PROC BTL does not sort by the values of the continuous variable; rather, it considers the data to be from a new subject or group whenever the value of the continuous variable changes from the previous observation. Using a continuous variable decreases execution time for models with a large number of subjects or groups.

If you want to model nonzero covariance among all of the observations in your SAS data set, specify SUBJECT=INTERCEPT to treat the data as if they are all from one subject. However, be aware that, in this case, PROC BTL manipulates an **R** matrix with dimensions equal to the number of observations. If no SUBJECT= effect is specified, then every observation is assumed to be from a different subject and **R** is assumed to be diagonal. For this reason, you usually want to use the SUBJECT= option in the REPEATED statement.

TYPE=*covariance-structure*

specifies the covariance structure of the **R** matrix. The SUBJECT= option defines the blocks of **R**, and the TYPE= option specifies the structure of these blocks. The default structure is VC. See PROC MIXED for details about the available covariance structures.

WEIGHT Statement

WEIGHT *variable* **;**

If you do not specify a REPEATED statement, the WEIGHT statement operates exactly like the one in PROC GLM. In this case PROC BTL replaces $\mathbf{X}'\mathbf{X}$ and $\mathbf{Z}'\mathbf{Z}$ with $\mathbf{X}'\mathbf{W}\mathbf{X}$ and $\mathbf{Z}'\mathbf{W}\mathbf{Z}$, where $\mathbf{W}$ is the diagonal weight matrix. If you specify a REPEATED statement, then the WEIGHT statement replaces $\mathbf{R}$ with $\mathbf{LRL}$, where $\mathbf{L}$ is a diagonal matrix with elements $\mathbf{W}^{-1/2}$. Observations with nonpositive or missing weights are not included in the PROC BTL analysis.

Details: BTL Procedure

Statistical Computations

The model specified using the MARKER, MODEL, RANDOM, and REPEATED statements is estimated using mixed model theory, and the resulting model statistics are printed in the "Model Statistics" table. For more details about these calculations, see the "Mixed Models Theory" section in "The MIXED Procedure" chapter in the *SAS/STAT User's Guide*.

If the PARMEST statement is used, a BTL model will be fit to the input data. This section describes the formulation of the BTL model and the procedure for estimating model parameters for the given data set. The section has been adapted from Coffman et al. (2005).

PROC BTL fits the input data to a probability model for multiple binary trait loci (Simonsen 2004). The assumed genetic map contains alternating markers (M_i) and binary trait loci (G_i), with at least one marker associated with each binary trait locus: $M_1 G_1 M_2 G_2 \cdots M_k G_k$. One allele is fixed in backcross populations, so there are $K = 2^k$ unique marker classes or BTL k-locus genotypes. In an F_n population with phase unknown, each locus has three possible genotypes, giving a total of $K = 3^k$ genotypes across the k markers or BTL. The recombination rate between two loci is the probability that a crossover occurs between the loci, ranging from 0 (complete linkage) to 0.5 (no linkage). This value is represented by r_i for the loci G_i and M_i, $i = 1, \ldots, k$, and θ_i for markers M_i and M_{i+1}, so there are $k - 1$ marker recombination parameters. Each penetrance parameter, p_m, is the probability that a binary trait is present for the mth BTL genotype (McIntyre, Coffman, and Doerge 2001). Similarly, π_m denotes the penetrance parameter for the mth marker genotype.

The joint probabilities of the BTL genotypes (G), the marker classes (M), and the trait (Y) can be expressed in matrix form in terms of $\mathbf{r}$, ,, and $\mathbf{p}$, assuming no selection, interference, or mutation, as shown by Simonsen (2004). These probabilities provide a likelihood equation for $\mathbf{r}$, ,, and $\mathbf{p}$. From this likelihood, the maximum likelihood estimate (MLE) for π_m, $\hat{\pi}_m$, is given by the observed binomial proportion of individuals with marker genotype m in whom the trait is present. The invariance property of MLEs (Casella and Berger 1990) can be applied to obtain the MLE of penetrance parameters $\mathbf{p}$ as the product of $\hat{\mathbf{\imath}}$ and a function of the recombination rates $\mathbf{r}$. By entering a known set of $\mathbf{r}$ or performing a grid search over a range of possible values of $\mathbf{r}$, unique estimates of penetrance parameters $\mathbf{p}$ can be computed.

Displayed Output

This section describes the displayed output from PROC BTL. See the section "ODS Table Names" on page 57 for details about how this output interfaces with the Output Delivery System.

Model Statistics

The "Model Statistics" table is displayed by default and contains the following information about the fitted model:

- LinkageGroup, the linkage group of a marker (displayed for single marker effects only if the GROUP= option is given in the MARKER statement)
- MarkerEffect, the marker(s) constituting the fixed marker effect
- Name, the name of the marker (displayed for single marker effects only if a Name variable is contained in the MAP= data set)
- DF, the degrees of freedom for the chi-squared test statistic
- ChiSquare, the test statistic for the likelihood ratio of the full model to the null model
- ProbChiSq, the *p*-value of the chi-squared test statistic
- AIC, Akaike's information criterion (Akaike 1974)
- AICC, a finite-sample corrected version of AIC (Burnham and Anderson 1998)
- BIC, Bayesian information criterion (Schwarz 1978)

Marker Class Means

The "Marker Class Means" table is displayed by default if the PARMEST statement is used and contains the average binary trait for each marker class. It contains the following:

- Parameter, the marker class parameter
- MarkerGenotype, the marker class genotype
- N, the number of individuals in the marker class
- Mean, the MLE of the marker class mean, which is the proportion of individuals in the marker class with trait 1
- StdErr, the standard error of the marker class mean estimate

Parameter Estimates

The “Parameter Estimates” table is displayed by default if the PARMEST statement is used and contains the estimated recombination and penetrance parameters for the estimated BTL model. It contains the model number, the parameter name, the parameter estimate, and the $100(1-\alpha)\%$ confidence interval for the estimate if the BOOT= option is used.

ODS Table Names

PROC BTL assigns a name to each table it creates, and you must use this name to reference the table when using the Output Delivery System (ODS). These names are listed in Table 3.1.

Table 3.1 ODS Tables Created by the BTL Procedure

ODS Table Name	Description	PROC BTL Statement
ModelStatistics	Model Statistics	default
MarkerClassMeans	Marker Class Means	PARMEST
ParmEst	Parameter Estimates	PARMEST

Example: BTL Procedure

Example 3.1: Performing BTL Model Selection and Estimation

In the following data taken from Coffman et al. (2005), a sample of trout from an experimental backcross population were genotyped at markers spanning 38 chromosomes, and the disease resistance of each trout was recorded concomitantly. In a prior analysis, single-marker tests were performed with each marker, and the marker most strongly associated with the trait for each chromosome was kept. This reduced set of data along with the corresponding map data is input in the following code. “A” corresponds to the homozygous genotype, and “B” corresponds to the heterozygous genotype; these are the default values for PROC BTL.

```
data TroutDat;
   input (m1-m17) ($) trait;
   datalines;
A A A B A A B B A A A B B A A B A 0
A B A A B B A B A A A A B B A A B 0
B B A A B B A B A A B B A A A B A 0

   ... more lines ...

B B A B B A B A B B A B A B B B B 1
B B B A A B A B A A A A A A A B A 1
;
```

```
data TroutMap;
   input marker $ location name $ chromosome;
   datalines;
m1 0 agcagc3 1
m2 0 acgatt11 2
m3 0 agcagc5 3
m4 11.6 agcagc6 4
m5 0 accaag22 5
m6 0 agcaca5 6
m7 43 agcagt27 7
m8 9.3 acgagt8 8
m9 52.8 agcact10 9
m10 2.3 agccag11 10
m11 0 accatc3 11
m12 0 accaag8 12
m13 14 accaag13 13
m14 7 acgaag8 14
m15 7 acgatc9 15
m16 0 agcagt26 16
m17 18.4 acccca20 17
;
```

First, the single-marker models are tested as shown in the following code. The ALL=1 option in the MARKER statement requests the single-marker model tests for each marker in the MARKER statement. The GROUP= option indicates that chromosome is the linkage group variable in the map data set.

```
proc btl data=TroutDat map=TroutMap;
   marker m1-m17 /all=1 group=chromosome;
   model trait;
run;
```

The single-marker model results are shown in Output 3.1.1.

Output 3.1.1 Single-Marker Model Statistics

```
                              The BTL Procedure

                               Model Statistics

Linkage  Marker                    Chi-    Pr >
  Group  Effect  Name       DF    Square   ChiSq       AIC      AICC       BIC

      1  M1      agcagc3     1    2.9479  0.0860      48.3      49.1      52.6
      2  M2      acgatt11    1    1.0612  0.3029      50.1      51.0      54.4
      3  M3      agcagc5     1    2.9479  0.0860      48.3      49.1      52.6
      4  M4      agcagc6     1    2.1504  0.1425      49.0      49.9      53.3
      5  M5      accaag22    1    1.2346  0.2665      50.0      50.9      54.3
      6  M6      agcaca5     1    1.6173  0.2035      49.6      50.5      53.9
      7  M7      agcagt27    1    3.5413  0.0599      47.7      48.5      52.0
      8  M8      acgagt8     1    1.2158  0.2702      50.0      50.9      54.3
      9  M9      agcact10    1    3.5413  0.0599      47.7      48.5      52.0
     10  M10     agccag11    1    2.9479  0.0860      48.3      49.1      52.6
     11  M11     accatc3     1    1.2848  0.2570      49.9      50.8      54.2
     12  M12     accaag8     1    7.3678  0.0066      43.8      44.7      48.1
     13  M13     accaag13    1    5.1787  0.0229      46.0      46.9      50.3
     14  M14     acgaag8     1    2.6072  0.1064      48.6      49.5      52.9
     15  M15     acgatc9     1    1.7060  0.1915      49.5      50.4      53.8
     16  M16     agcagt26    1    2.3491  0.1254      48.8      49.7      53.2
     17  M17     acccca20    1    3.5413  0.0599      47.7      48.5      52.0
```

Next, all two-marker models are estimated with the following code. Note that only the five best models are requested using the BEST= option, and the model selection criterion is selected to be the *p*-value from using the MC= option.

```
proc btl data=TroutDat map=TroutMap;
   marker m1-m17 /all=2 best=5 mc=p group=chromosome;
   model trait;
run;
```

The two-marker model results are shown in the Output 3.1.2.

Output 3.1.2 Two-Marker Model Statistics

```
                             The BTL Procedure

                              Model Statistics

     Marker                   Chi-      Pr >
     Effect       DF        Square     ChiSq          AIC        AICC         BIC

     M12*M17       3       24.8422    <.0001         30.4        32.8        37.5
     M12*M14       3       15.1226    0.0017         40.1        42.5        47.2
     M8*M12        3       13.6663    0.0034         41.5        43.9        48.7
     M7*M12        3       12.9359    0.0048         42.3        44.7        49.4
     M9*M12        3       11.6006    0.0089         43.6        46.0        50.8
```

It looks like m12 xm17 is the best two-marker effect. You can try to estimate parameters for a BTL model by using the following code. The parameter estimation is requested using the PARMEST statement. The linkage units of the location variable from the map data set are specified to be CM (centimorgans) by using the LINKUNIT option. The linkage model used to create the inputted linkage map is specified as Haldane by using the LINKMOD= option.

Now that the experimental design has been specified, by default PROC BTL performs a grid search over a range of possible recombination values and displays penetrance estimates that are found to be within the range of valid values. The default grid search for the *r* uses values from 0 to 0.5 in increments of 0.1. In the following code, a finer grid search is requested by specifying the increment to be 0.05 in the RINC= option. Also, the penetrance parameter limits are set to be –0.1 and 1.1 using the options PMIN= and PMAX=, respectively (the default values are 0 and 1). The SAS code follows.

```
proc btl data=TroutDat map=TroutMap;
   marker m12 m17 /group=chromosome;
   model trait;
   parmest r=0 linkmod=H linkunit=cm boot=1000;
run;
```

The average penetrance value for each marker class is shown in the “Marker Class Means” table in Output 3.1.3, and the BTL penetrance estimates found using 0 for the marker-BTL recombination rates are shown in the “Parameter Estimates” table in Output 3.1.4. The confidence limits for the penetrance parameters from using the bootstrap technique are displayed in Output 3.1.4 as well.

Output 3.1.3 Marker Class Means

```
                         The BTL Procedure

                        Marker Class Means

      Marker    Marker                               Standard
      Class     Genotype         N       Mean           Error

      pi11      AA               8     1.0000          0.0000
      pi12      AB               8     0.2500          0.0234
      pi21      BA               6     0.0000          0.0000
      pi22      BB               9     0.3333          0.0247
```

Output 3.1.4 Parameter Estimates

```
                        Parameter Estimates

                                         95% Confidence
            Parameter    Estimate            Limits

            r1             0.0000
            r2             0.0000
            p11            4.0000     4.0000     4.0000
            p12            1.0000     0.0000     2.2857
            p21            0.0000     0.0000     0.0000
            p22            1.3333     0.0000     2.6667
            theta          0.5000

            NOTE: The r and theta parameters are fixed.
```

Suppose your input data were coded differently, and "A" signified heterozygote and "B" signified homozygote. Since this coding is different from the default, you can specify the genotype values by using the HOMOZYGOTE= (or HO=) and HETEROZYGOTE= (or HE=) options.

```
proc btl data=TroutDat map=TroutMap;
   marker m12 m17 /group=chromosome;
   model trait;
   parmest ho="B" he="A" r=0 linkmod=H linkunit=cm boot=1000;
run;
```

In Outputs 3.1.5 and 3.1.6, notice that the marker class mean values are reversed and different penetrance estimates are obtained.

Output 3.1.5 Marker Class Means with Different Genotype Coding

```
                         The BTL Procedure

                        Marker Class Means

         Marker    Marker                              Standard
         Class     Genotype         N       Mean          Error

         pi11      BB               9     0.3333         0.0247
         pi12      BA               6     0.0000         0.0000
         pi21      AB               8     0.2500         0.0234
         pi22      AA               8     1.0000         0.0000
```

Output 3.1.6 Parameter Estimates with Different Genotype Coding

```
                    Parameter Estimates

                                95% Confidence
      Parameter    Estimate         Limits

      r1             0.0000
      r2             0.0000
      p11            1.3333     0.0000     2.6667
      p12            0.0000     0.0000     0.0000
      p21            1.0000     0.0000     2.2857
      p22            4.0000     4.0000     4.0000
      theta          0.5000

      NOTE: The r and theta parameters are fixed.
```

References

Akaike, H. (1974), “A New Look at the Statistical Model Identification,” *IEEE Transactions on Automatic Control*, AC-19, 716–723.

Burnham, K. P. and Anderson, D. R. (1998), *Model Selection and Inference: A Practical Information-Theoretic Approach*, New York: Springer-Verlag.

Casella, G. and Berger, R. (1990), *Statistical Inference*, Pacific Grove, CA: Wadsworth and Brooks/Cole.

Coffman, C. J., Doerge, R. W., Simonsen, K. L., Nichols, K., Duarte, C. W., Wolfinger, R. D., and McIntyre, L. M. (2005), “Model Selection in Binary Trait Locus Mapping,” *Genetics*, 170, 1281–1297.

Cressie, N. (1991), *Statistics for Spatial Data*, New York: John Wiley & Sons.

Jones, R. H. and Boadi-Boateng, F. (1991), “Unequally Spaced Longitudinal Data with AR(1) Serial Correlation,” *Biometrics*, 47, 161–175.

McIntyre, L. M., Coffman, C. J., and Doerge, R. W. (2001), “Detection and Localization of a Single Binary Trait Locus in Experimental Populations,” *Genetical Research*, 78, 79–92.

Schwarz, G. (1978), “Estimating the Dimension of a Model,” *Annals of Statistics*, 6, 461–464.

Simonsen, K. L. (2004), *A Probability Model for the Inheritance of Binary Traits*, Technical Report tr03-04, Purdue University Statistics Department.

Chapter 4
The CASECONTROL Procedure

Contents

Overview: CASECONTROL Procedure

Marker information can be used to help locate the genes that affect susceptibility to a disease. The CASECONTROL procedure is designed for the interpretation of marker data when random samples are available from the populations of unrelated individuals who are either affected or unaffected by the disease. Several tests are available in PROC CASECONTROL that compare marker allele and/or genotype frequencies in the two populations, with frequency differences indicating an association of the marker with the disease. Although such an association can point to the proximity of the marker and disease genes in the genome, it can also reflect population structure, so care is needed in interpreting the results; association does not necessarily imply linkage.

The three chi-square tests available for testing case-control genotypic data are the genotype case-control test, which tests for dominant allele effects on the disease penetrance, and the allele case-control test and linear trend test, which test for additive allele effects on the disease penetrance. Since the allele case-control test requires the assumption of Hardy-Weinberg equilibrium (HWE), it might be desirable to run the ALLELE procedure on the data to perform the HWE test on each marker (see Chapter 2, "The ALLELE Procedure," for more information) prior to applying PROC CASECONTROL.

Getting Started: CASECONTROL Procedure

Example

Here are some sample SNP data on which the three case-control tests can be performed using PROC CASECONTROL:

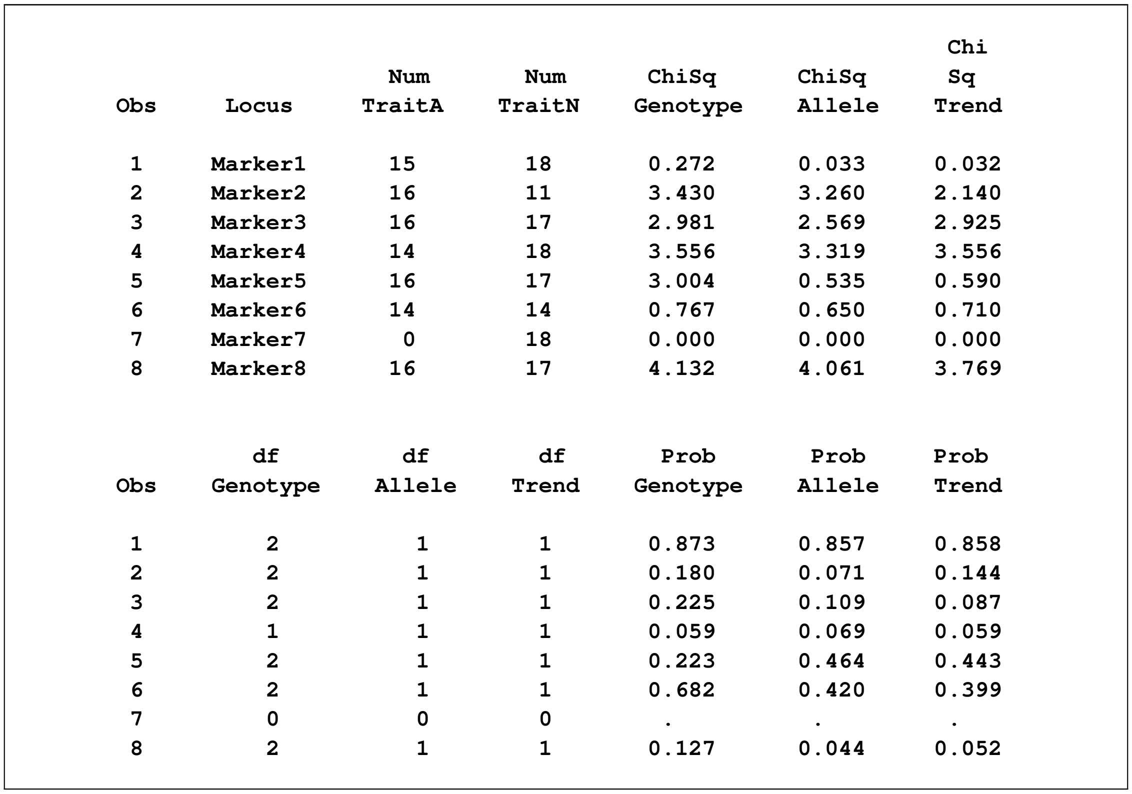
```
data cc;
   input affected $ m1-m16;
   datalines;
 N  1 1 2 2 2 2 2 1 2 1 2 2 1 1 2 2
 N  1 1 1 1 2 2 1 1 2 1 2 1 1 1 1 1
 N  2 1 1 1 2 1 1 1 2 2 1 1 1 1 1 1
 N  2 2 2 1 2 2 1 1 2 2 2 1 1 1 2 2
 N  1 1 1 1 2 2 2 1 1 1 1 1 2 1 . .
 N  2 1 1 1 2 1 1 1 2 1 2 1 1 1 2 1
 N  1 1 1 1 2 2 1 1 2 2 2 2 2 1 2 2
 N  2 2 1 1 2 1 2 1 2 2 2 1 1 1 2 1
 N  2 1 1 1 2 2 2 1 2 1 . . 1 1 2 1
 N  2 1 1 1 2 1 1 1 2 2 1 1 1 1 1 1
 N  2 1 2 2 . . 1 1 2 1 1 1 1 1 1 1
 N  2 2 . . 2 1 1 1 2 1 2 1 1 1 2 1
 N  2 1 . . 2 2 1 1 2 2 1 1 1 1 2 1
 N  2 1 . . 2 2 1 1 2 1 . . 2 1 1 1
 N  2 2 . . 2 2 1 1 . . 2 1 1 1 2 1
 N  1 1 . . 2 2 1 1 1 1 2 1 1 1 2 1
 N  1 1 . . 2 2 1 1 1 1 . . 1 1 2 1
 N  2 1 . . 2 2 1 1 1 1 . . 2 1 2 1
 A  2 1 2 1 2 1 1 1 1 1 2 1 . . 2 1
 A  2 1 2 1 2 2 1 1 2 1 1 1 . . 1 1
 A  2 2 2 1 2 2 1 1 2 2 . . . . 2 1
 A  2 1 2 2 2 1 1 1 2 1 2 1 . . 2 2
 A  . . 2 2 2 1 . . 1 1 2 2 . . 2 1
 A  1 1 1 1 2 1 1 1 2 1 1 1 . . 2 2
 A  2 1 1 1 2 2 1 1 1 1 2 1 . . 2 1
 A  2 1 2 2 2 2 1 1 2 2 . . . . 2 2
 A  2 1 1 1 2 2 1 1 2 1 2 1 . . 1 1
 A  2 1 2 2 2 1 1 1 2 1 2 1 . . 2 2
 A  1 1 1 1 2 2 1 1 2 1 2 1 . . 2 2
 A  2 1 2 1 2 1 1 1 2 1 2 2 . . 2 1
 A  2 2 2 2 1 1 1 1 2 1 2 1 . . 2 2
 A  1 1 1 1 2 1 . . 2 1 2 2 . . 2 2
 A  1 1 2 1 2 1 1 1 2 1 2 1 . . 2 2
 A  2 2 1 1 2 2 1 1 2 1 1 1 . . 2 1
 ;
```

The following SAS code can be used to perform the analysis:

```
proc casecontrol data=cc prefix=Marker;
   var m1-m16;
   trait affected;
run;

proc print heading=h;
 format probgenotype proballele probtrend pvalue5.4;
 format chisqgenotype chisqallele chisqtrend 5.3;
run;
```

All three case-control tests are performed by default. The output data set created by default appears in Figure 4.1.

Figure 4.1 Statistics for Case-Control Tests



```
                                                            Chi
                    Num        Num       ChiSq     ChiSq     Sq
 Obs    Locus      TraitA     TraitN    Genotype   Allele   Trend

  1     Marker1      15         18       0.272     0.033    0.032
  2     Marker2      16         11       3.430     3.260    2.140
  3     Marker3      16         17       2.981     2.569    2.925
  4     Marker4      14         18       3.556     3.319    3.556
  5     Marker5      16         17       3.004     0.535    0.590
  6     Marker6      14         14       0.767     0.650    0.710
  7     Marker7       0         18       0.000     0.000    0.000
  8     Marker8      16         17       4.132     4.061    3.769


           df         df         df       Prob      Prob     Prob
 Obs    Genotype    Allele     Trend    Genotype   Allele   Trend

  1         2          1         1       0.873     0.857    0.858
  2         2          1         1       0.180     0.071    0.144
  3         2          1         1       0.225     0.109    0.087
  4         1          1         1       0.059     0.069    0.059
  5         2          1         1       0.223     0.464    0.443
  6         2          1         1       0.682     0.420    0.399
  7         0          0         0         .         .        .
  8         2          1         1       0.127     0.044    0.052
```

Figure 4.1 displays the statistics for the three tests. The genotype case-control statistic has more degrees of freedom than the other two because it is testing for both dominance genotypic effects and additive allelic effects, while the other statistics are testing for the significant additive effects alone. Using the standard significance level of 0.05, none of the p-values, shown in the last three columns, would be considered significant since they are all above this significance level. Thus, you would conclude that none of the markers show a significant association with the binary trait. The p-values for Marker7 are missing because the genotypes of all the affected individuals are missing at that marker.

Syntax: CASECONTROL Procedure

The following statements are available in PROC CASECONTROL.

> **PROC CASECONTROL** < *options* > **;**
> **BY** *variables* **;**
> **STRATA** *variables* < / *options* > **;**
> **TRAIT** *variable* **;**
> **VAR** *variables* **;**

Items within angle brackets (< >) are optional, and statements following the PROC CASECONTROL statement can appear in any order. The TRAIT and VAR statements are required.

PROC CASECONTROL Statement

> **PROC CASECONTROL** < *options* > **;**

You can specify the following options in the PROC CASECONTROL statement.

ALLELE

requests that the allele case-control test be performed. If none of the three test options (ALLELE, GENOTYPE, or TREND) are specified, then all three tests are performed by default.

ALPHA=*number*

specifies that a confidence level of $100(1-\mathit{number})\%$ is to be used in forming confidence intervals for odds ratios. The value of *number* must be between 0 and 1, and is set to 0.05 by default.

DATA=*SAS-data-set*

names the input SAS data set to be used by PROC CASECONTROL. The default is to use the most recently created data set.

DELIMITER=*'string'*

indicates the string that is used to separate the two alleles that compose the genotypes contained in the variables specified in the VAR statement. This option is ignored if GENOCOL is not specified.

GENOCOL

indicates that columns specified in the VAR statement contain genotypes instead of alleles. When this option is specified, there is one column per marker. The genotypes must consist of the two alleles separated by a delimiter. For a genotype with one missing allele, use a blank space to indicate a missing value; if both alleles are missing, either use a single missing value for the entire genotype or use the delimiter alone.

GENOTYPE

requests that the genotype case-control test be performed. If none of the three test options (ALLELE, GENOTYPE, or TREND) are specified, then all three tests are performed by default.

INDIVIDUAL=*variable*

INDIV=*variable*

specifies the individual ID variable when using the TALL option. This variable can be character or numeric.

MARKER=*variable*

specifies the marker ID variable when using the TALL option. This variable contains the names of the markers that are used in all output and can be character or numeric.

NDATA=*SAS-data-set*

names the input SAS data set containing names, or identifiers, for the markers used in the output. There must be a NAME variable in this data set, which should contain the same number of rows as there are markers in the input data set specified in the DATA= option. When there are fewer rows than there are markers, markers without a name are named using the PREFIX= option. Likewise, if there is no NDATA= data set specified, the PREFIX= option is used. Note that this data set is ignored if the TALL option is specified in the PROC CASECONTROL statement. In that case, the marker variable names are taken from the marker ID variable specified in the MARKER= option.

NULLSNPS=(*variable list* **)**

names the markers to be used in calculating the variance inflation factor for genomic control that is applied to the chi-square statistic(s) from the trend test. Only biallelic markers that are listed are used. Note that if GENOCOL is specified, there should be one variable for each marker listed; otherwise, there should be two variables per marker. By default, if VIF is specified in the PROC CASECONTROL statement, all biallelic markers listed in the VAR statement are used. This option must be specified if both the VIF option and the PERMS= option are used; otherwise the variance inflation factor is not applied. This option is ignored if the VIF option is not specified or if the TALL option is used.

OR

requests that odds ratios based on allele counts for biallelic markers be included in the OUTSTAT= data set, along with (1–α)% confidence limits for the value specified in the ALPHA= option. Odds ratios are not reported for markers with more than two alleles.

OUTSTAT=*SAS-data-set*

names the output SAS data set containing counts for the two trait values, the chi-square statistics, degrees of freedom, and p-values for the tests performed. When this option is omitted, an output data set is created by default and named according to the DATA*n* convention.

PERMS=*number*

indicates that Monte Carlo estimates of exact p-values for the case-control tests should be calculated instead of the p-values from the asymptotic χ^2 distribution. In each of the *number* permutation samples, the trait values are permuted among the individuals in the sample. Large values of *number* (10,000 or more) are usually recommended for accuracy, but long execution times can result, particularly with large data sets. When this option is omitted, no permutations are performed and p-values from the asymptotic χ^2 distribution are reported.

PREFIX=*prefix*

specifies a prefix to use in constructing names for marker variables in all output. For example, if PREFIX=VAR, the names of the variables are VAR1, VAR2, ..., VAR*n*. Note that this option is ignored when the NDATA= option is specified, unless there are fewer names in the NDATA data set than there are markers; it is also ignored if the TALL option is specified, in which case the marker variable names are taken from the marker ID variable specified in the MARKER= option. Otherwise, if this option is omitted, PREFIX=M is the default when variables contain alleles; if GENOCOL is specified, then the names of the variables specified in the VAR statement are used as the marker names.

SEED=*number*

specifies the initial seed for the random number generator used for permuting the data to calculate estimates of exact p-values. This option is ignored if PERMS= is not specified. The value for *number* must be an integer; the computer clock time is used if the option is omitted or an integer less than or equal to 0 is specified. For more details about seed values, see *SAS Language Reference: Concepts*.

TALL

indicates that the input data set is of an alternative format. This tall-skinny format contains the following columns: two containing marker alleles (or one containing marker genotypes if GENOCOL is specified), one for the marker identifier, and one for the individual identifier. The MARKER= and INDIV= options must also be specified in order for this option to be in effect. Note that when this option is used, the DATA= data set must first be sorted by any BY variables, then sorted by the marker ID variable, and then sorted by the individual ID variable.

TREND

requests that the linear trend test for allelic effects be performed. If none of the three test options (ALLELE, GENOTYPE, or TREND) are specified, then all three tests are performed by default.

VIF

specifies that the variance inflation factor λ should be applied to the trend chi-square statistic for genomic control. This adjustment is applied only when the trend test is performed and to markers in the VAR statement that are biallelic.

BY Statement

BY *variables* **;**

You can specify a BY statement with PROC CASECONTROL to obtain separate analyses on observations in groups defined by the BY variables. When a BY statement appears, the procedure expects the input data set to be sorted in the order of the BY variables. The *variables* are one or more variables in the input data set.

If your input data set is not sorted in ascending order, use one of the following alternatives:

- Sort the data by using the SORT procedure with a similar BY statement.
- Specify the BY statement option NOTSORTED or DESCENDING in the BY statement for the CASECONTROL procedure. The NOTSORTED option does not mean that the data are unsorted but rather that the data are arranged in groups (according to values of the BY variables) and that these groups are not necessarily in alphabetical or increasing numeric order.
- Create an index on the BY variables by using the DATASETS procedure (in Base SAS software).

For more information about the BY statement, see *SAS Language Reference: Concepts*. For more information about the DATASETS procedure, see the *Base SAS Procedures Guide*.

STRATA Statement

STRATA *variables* < / *options* > **;**

The STRATA statement names the variables defining strata or representing matched or nested sets of individuals in a case-control study. Each STRATA variable can be either character or numeric, and the formatted values of the STRATA variables determine the levels. Thus, you can also use formats to group values into levels. See the discussion of the FORMAT procedure in the *Base SAS Procedures Guide*. At least one variable must be specified in order to invoke the stratified analysis. See the section “Stratified Analysis” on page 72 for more information.

The following options can be specified after a slash (/):

INFO

displays the “Strata Information” table, which includes the stratum number, levels of the STRATA variables that define the stratum, total number of individuals, and counts for the two trait values that define cases and controls in each stratum. Since the number of strata can be very large, this table is displayed only on request.

MISSING

treats missing values (‘.’, ‘.A’,. . . ,‘.Z’ for numeric variables and blanks for character variables) as valid STRATA variable values.

TRAIT Statement

TRAIT *variable* **;**

The TRAIT statement identifies a binary variable indicating which individuals are cases and which are controls or a binary variable representing a dichotomous trait. This variable can be character or numeric, but it must have only two nonmissing levels.

VAR Statement

VAR *variables* **;**

The VAR statement identifies the variables containing either the marker alleles or, if GENOCOL is specified, the marker genotypes. The following number of variables should be specified in this statement for a data set containing m markers according to whether the options GENOCOL and TALL are used:

- When both GENOCOL and TALL are specified, there should be 1 variable named containing marker genotypes.
- When only TALL is specified, there should be 2 variables named containing marker alleles.
- When only GENOCOL is specified, there should be m variables named, one for each marker containing marker genotypes.
- When neither option is specified, there should be $2m$ variables named, two for each marker containing marker alleles.

All variables specified must be of the same type, either character or numeric.

Details: CASECONTROL Procedure

Statistical Computations

Biallelic Markers

PROC CASECONTROL offers three statistics to test for an association between a biallelic marker and a binary variable, typically affection status of a particular disease. Table 4.1 displays the quantities that are used for the three case-control tests for biallelic markers (Sasieni 1997).

Table 4.1 Genotype Distribution for Case-Control Sample

	Number of M_1 Alleles			
	0	**1**	**2**	**Total**
Case	r_0	r_1	r_2	R
Control	s_0	s_1	s_2	S
Total	n_0	n_1	n_2	N

The three statistical methods for testing a marker for association with a disease locus are Armitage's trend test (1955), the allele case-control test, and the genotype case-control test. The trend test and allele case-control test are most useful when there is an additive allele effect on the disease susceptibility. When Hardy-Weinberg equilibrium (HWE) holds in the combined sample of cases and controls, these statistics are approximately equal and have an asymptotic χ_1^2 distribution. However, if the assumption of HWE in the combined sample is violated, then the variance for the allele case-control statistic is incorrect; only the trend test remains valid under this violation. The statistics for the trend and allele case-control test, respectively, are given by Sasieni (1997) as

$$\begin{aligned} X_T^2 &= \frac{N[N(r_1 + 2r_2) - R(n_1 + 2n_2)]^2}{R(N-R)[N(n_1 + 4n_2) - (n_1 + 2n_2)^2]} \\ X_A^2 &= \frac{2N[2N(r_1 + 2r_2) - 2R(n_1 + 2n_2)]^2}{(2R)2(N-R)[2N(n_1 + 2n_2) - (n_1 + 2n_2)^2]} \end{aligned}$$

Devlin and Roeder (1999) describe a genomic control method that adjusts the trend test statistic for correlation between alleles from members of the same subpopulation. Assuming the variance inflation factor λ is constant across the genome, it can be estimated by $\hat{\lambda} = \max([\text{median}(X_1, \ldots, X_m)/0.675]^2, 1)$, where $X_i = X_T$ for the ith biallelic marker, $i = 1, \ldots, m$ (Devlin and Roeder 1999; Bacanu, Devlin, and Roeder 2000). The adjusted trend statistic, $X_{T_a}^2 = X_T^2/\hat{\lambda}$, is approximately distributed as χ_1^2. This variance correction is made to biallelic markers when the VIF option is specified in the PROC statement. By default, any biallelic markers that are specified in the VAR statement are used in computing $\hat{\lambda}$. Alternatively, the NULLSNPS= option can be used to specify biallelic markers other than those in the VAR statement to be used to calculate $\hat{\lambda}$. This enables markers that are assumed to have no effect on disease susceptibility or to not be in linkage disequilibrium with a disease-susceptibility locus to be used in calculating the inflation factor (Bacanu, Devlin, and Roeder 2000).

If dominance effects of alleles are also suspected to contribute to disease susceptibility, the genotype case-control test can be used. The standard 2×3 contingency table analysis is used to form the χ_2^2 statistic for the genotype case-control test as

$$X_G^2 = \sum_{i=0}^{2} \left[\frac{(Nr_i - Rn_i)^2}{NRn_i} + \frac{(Ns_i - Sn_i)^2}{NSn_i} \right]$$

which tests for both additive and dominance (nonadditive) allelic effects (Nielsen and Weir 1999).

When the OR option is specified in the PROC CASECONTROL statement, odds ratios for biallelic markers are calculated based on the 2×2 table of allele-by-trait counts. Using the values given in Table 4.1 to form the cell counts $a = 2r_2 + r_1$, $b = 2s_0 + s_1$, $c = 2r_0 + r_1$, and $d = 2s_2 + s_1$, the odds ratio can be estimated as $\hat{\theta} = ab/(cd)$. The asymptotic $(1-\alpha)\%$ confidence limits for the estimated odds ratio $\hat{\theta}$ are

$$(\hat{\theta} \cdot \exp(-z\sqrt{v}), \hat{\theta} \cdot \exp(z\sqrt{v}))$$

where

$$v = \frac{1}{a} + \frac{1}{b} + \frac{1}{c} + \frac{1}{d}$$

and z is the $100(1-\alpha/2)$ percentile in the standard normal distribution. If any of the four cell frequencies are zero, the limits are not computed. The order of rows and columns is determined by the formatted values of the alleles and trait. Also note that if there are no heterozygous genotypes, $2v$ is used in place of v in the formula for the confidence limits so that each individual is counted only once. This provides the correct limits when combining the heterozygous genotype with a homozygous genotype to obtain odds ratios for dominant or recessive disease models (see Example 4.3).

Multiallelic Markers

When there are multiple alleles of interest at a marker, the same three tests can be performed, except that Devlin and Roeder's genomic control adjustment is not applied to any markers with more than two alleles. To construct the test statistic for the multiallelic trend test for a marker with k alleles (Slager and Schaid 2001), the $p \times (k-1)$ matrix $\mathbf{X}$ is created such that each element X_{iu} represents the number of times the M_u allele appears in the ith genotype, $i = 1, \ldots, p$ and $u = 1, \ldots, k-1$, where $p = k(k+1)/2$, the number of possible genotypes. Vectors $\mathbf{r}$ and $\mathbf{s}$ of length p contain the genotype counts for the cases and controls, respectively, and $\phi = R/N$, the proportion of cases in the sample. The multiallelic trend test statistic can then be expressed as $\mathbf{U}'[\text{Var}(\mathbf{U})]^{-1}\mathbf{U}$, where the vector $\mathbf{U} = \mathbf{X}'[(1-\phi)\mathbf{r} - \phi\mathbf{s}]$. Var($\mathbf{U}$) is calculated under the assumption of independent (or unrelated) subjects in the sample by using Var($\mathbf{r}$) and Var($\mathbf{s}$). These matrices contain elements $\sigma_{ii} = Rn_i(N-n_i)/N^2$ and $\sigma_{ij} = -Rn_in_j/N^2$, where $i, j = 1, \ldots, p$ (the R is replaced by S for Var($\mathbf{s}$)). This statistic has an asymptotic χ^2_{k-1} distribution.

Another way to test for additive allele effects at the disease or trait locus is the allele case-control test, executed using a contingency table analysis similar to the genotype case-control test described in the preceding section, assuming HWE (Nielsen and Weir 1999). For a marker with k alleles, a $2\times k$ contingency table is formed with one row for cases, one for controls, and a column for each allele. The χ^2_{k-1} statistic is formed by summing $(O-E)^2/E$ over all cells in the table, where O is the observed count for the cell and E is the expected count, the cell's column total multiplied by R/N (or S/N) for a cell in the case (or control) row.

The genotypic case-control test statistic is calculated in a similar manner, with columns now representing the p observed genotype classes instead of alleles. Significance of this test statistic based on the χ^2_{p-1} distribution indicates dominance and/or additive allelic effects on the disease or trait (Nielsen and Weir 1999).

Stratified Analysis

A stratified case-control test can be performed to adjust for categorical covariates, such as gender or treatment; to analyze a sample from a matched or nested case-control design; or to accommodate the analysis of X-linked markers. The generalized Cochran-Mantel-Haenszel (CMH) test statistic given by Agresti (1990) can be used to test whether there is an association between the trait and marker alleles or genotypes in any of the strata, still with the same chi-square distribution and degrees of freedom as the test statistic from the nonstratified analysis. For the allele and genotype tests, which are based on contingency tables, the statistic is formed with the following quantities that use observed cell counts c_{ijh} from the ith row (corresponding to one of the two trait categories), jth out of J columns (corresponding to the jth allele or genotype), and hth stratum:

$$\mathbf{c}_h = (c_{11h}, c_{12h}, \ldots, c_{1,J-1,h})'$$

$$\mathbf{e}_h = (c_{1+h}c_{+1h}, \ldots, c_{1+h}c_{+,J-1,h})'/c_{++h}$$

$$\text{Cov}(c_{1jh}, c_{1j'h}) = \frac{c_{1+h}(c_{++h} - c_{1+h})c_{+jh}(\delta_{jj'}c_{++h} - c_{+j'h})}{c^2_{++h}(c_{++h}-1)}$$

with covariance matrix $\mathbf{V}_h$ of $\mathbf{c}_h$ comprising these covariance terms for all h and $j, j' = 1, \ldots, J-1$ and $\delta_{jj'} = 1$ when $j = j'$ and 0 otherwise. Note that cell counts for $i = 2$ are omitted from the vectors and matrix since they are completely dependent on the cell counts from the first row and column totals. For the stratified trend test, which is based on the Mantel score test of conditional independence (Agresti 1990), a trend test vector $\mathbf{U}_h$ and the covariance matrix $\mathbf{V}_h = c_{++h}\text{Var}(\mathbf{U}_h)/(c_{++h}-1)$ are calculated within each stratum with $\mathbf{U}_h$ and $\text{Var}(\mathbf{U}_h)$ defined as in the previous section for the multiallelic trend test. All three test statistics can then be represented as

$$X^2_M = \mathbf{S}'\mathbf{V}^{-1}\mathbf{S}$$

with a χ^2_{J-1} distribution under the null hypothesis, where J represents the number of genotypes for the genotype test or the number of alleles for the allele and trend tests,

$$\mathbf{S} = \begin{cases} \sum_h (\mathbf{c}_h - \mathbf{e}_h), & \text{genotype and allele tests} \\ \sum_h \mathbf{U}_h, & \text{trend test} \end{cases}$$

and $\mathbf{V} = \sum_h \mathbf{V}_h$.

The Mantel-Haenszel estimate of the common odds ratio across strata (Agresti 1990) for biallelic markers is reported when the STRATA statement is used along with the OR option in the PROC CASECONTROL statement. For a contingency table with two columns representing the two alleles at a marker, the estimate in terms of the observed cell counts is

$$\hat{\theta}_{\mathrm{MH}} = \frac{\sum_h (c_{11h} c_{22h} / c_{++h})}{\sum_h (c_{12h} c_{21h} / c_{++h})}$$

The asymptotic $(1-\alpha)\%$ confidence limits for the estimate of the odds ratio $\hat{\theta}_{\mathrm{MH}}$ are again given by Agresti (1990) as

$$(\hat{\theta}_{\mathrm{MH}} \cdot \exp(-z\sqrt{v}), \hat{\theta}_{\mathrm{MH}} \cdot \exp(z\sqrt{v}))$$

now with

$$\begin{aligned} v &= \frac{\sum_h (c_{11h} + c_{22h})(c_{11h} c_{22h})/c_{++h}^2}{2(\sum_h c_{11h} c_{22h}/c_{++h})^2} \\ &+ \frac{\sum_h [(c_{11h} + c_{22h})(c_{12h} c_{21h}) + (c_{12h} + c_{21h})(c_{11h} c_{22h})]/c_{++h}^2}{2(\sum_h c_{11h} c_{22h}/c_{++h})(\sum_h c_{12h} c_{21h}/c_{++h})} \\ &+ \frac{\sum_h (c_{12h} + c_{21h})(c_{12h} c_{21h})/c_{++h}^2}{2(\sum_h c_{12h} c_{21h}/c_{++h})^2} \end{aligned}$$

Again, if all of the strata contain no heterozygous genotypes, v is replaced by $2v$ in the confidence limits formula.

Permutation Tests

By default, the p-values from the χ^2 distribution with the appropriate degrees of freedom are reported for all three case-control tests. However, if the PERMS= option is specified in the PROC CASECONTROL statement, then Monte Carlo estimates of exact p-values are computed instead using the permutation procedure. For the genotype and trend tests, new samples of individuals are formed by permuting the trait value of the individuals in the sample; permutations for the allele test treat the two marker alleles per individual as separate observations each with the same trait, and the trait value is then permuted across these observations. If there are any STRATA variables, permutations are performed within each stratum. For p permutations, the exact p-value is estimated as the proportion of times the chi-square statistic from one of the p new samples is equal to or exceeds the chi-square statistic from the original sample (Westfall and Young 1993).

Missing Values

An individual's genotype for a marker is considered missing if at least one of the alleles at the marker is missing. Any missing genotypes are excluded from all calculations. However, the individual's nonmissing genotypes at other loci can be used as part of the calculations. If an individual has a missing trait value, then that individual is excluded from all calculations.

When the STRATA statement is used, missing stratum levels are handled in one of two ways: when the MISSING option is specified, missing values are treated as another stratum level; otherwise, individuals with a missing value for any of the STRATA variables are excluded from the analysis.

OUTSTAT= Data Set

The output data set specified in the OUTSTAT= option of the PROC CASECONTROL statement contains the following variables for each marker:

- the BY variables, if any
- Locus
- the counts of genotyped individuals for the two values of the TRAIT variable: NumTrait1 and NumTrait2 , where 1 and 2 are replaced by the values of the TRAIT variable
- the odds ratio AlleleOddsRatio and its confidence limits LowerCL and UpperCL if the OR option is used
- the chi-square statistic for each test performed: ChiSqAllele, ChiSqGenotype, and ChiSqTrend
- the degrees of freedom for each test performed: dfAllele, dfGenotype, and dfTrend
- the *p*-value for each test performed: ProbAllele, ProbGenotype, and ProbTrend

Displayed Output

This section describes the displayed output from PROC CASECONTROL. See the section "ODS Table Names" on page 75 for details about how this output interfaces with the Output Delivery System.

Strata Levels

The "Strata Levels" table is displayed by default when the STRATA statement is used and contains the number of levels and the formatted names of the levels for each STRATA variable.

Strata Information

The "Strata Information" table is displayed when the INFO option is specified in the STRATA statement. This table reports each stratum as defined by a unique combination of levels of the STRATA variables, the total count for each stratum, and the number of cases and controls as defined by the TRAIT variable in the strata.

ODS Table Names

PROC CASECONTROL assigns a name to each table it creates, and you must use this name to reference the table when using the Output Delivery System (ODS). These names are listed in Table 4.2.

Table 4.2 ODS Tables Created by the CASECONTROL Procedure

ODS Table Name	Description	Statement / Option
StrataLevels	Strata levels	STRATA
StrataInfo	Strata information	STRATA / INFO

Examples: CASECONTROL Procedure

Example 4.1: Performing Case-Control Tests on Multiallelic Markers

The following data are taken from GAW9 (Hodge 1995). A sample of 60 founders was taken from 200 nuclear families, 30 affected with a disease and 30 unaffected. Each founder was genotyped at two marker loci.

```
data founders;
   input id disease a1-a4 @@;
   datalines;
4   1 6 4 3 7  17  2 4 7 2 7
39  2 6 8 7 7  41  2 4 4 4 7
46  1 8 4 1 5  50  2 4 2 3 7
54  2 4 8 7 6  56  2 7 4 7 7
62  2 4 1 7 3  69  2 6 8 2 7
79  1 6 6 8 7  80  2 6 4 7 3
83  2 8 4 2 7  85  1 5 6 6 2
95  1 3 2 3 7  101 1 4 6 7 7
106 1 2 1 7 2  107 1 1 2 7 7
115 2 4 2 7 5  116 1 4 1 7 3
120 2 1 6 2 7  123 2 4 4 7 2
130 1 5 2 3 7  133 1 8 6 3 6
134 1 8 4 2 2  139 2 6 4 7 6
142 2 3 6 7 7  151 1 4 6 4 3
152 1 6 7 6 7  153 1 5 1 7 6
154 1 4 6 6 6  168 1 1 4 3 7
178 2 4 1 7 1  187 1 1 8 1 2
189 2 6 4 5 7  190 2 4 4 3 7
195 2 4 4 7 2  207 2 1 6 7 7
216 1 7 4 1 5  222 2 4 2 7 3
225 2 8 7 7 6  234 1 6 4 2 2
244 1 4 4 7 6  249 2 6 8 7 2
263 1 8 2 3 7  267 2 2 2 2 7
276 2 1 6 7 1  284 2 4 8 2 2
286 1 8 8 2 1  289 1 2 6 6 3
```

```
290 1 2 4 5 7  294 2 1 8 6 7
297 2 5 4 7 6  313 1 1 7 7 2
337 1 2 6 7 6  366 2 2 2 7 7
368 2 3 1 7 2  381 1 6 4 5 3
384 1 6 2 2 7  396 1 4 5 7 2
;
```

The multiallelic versions of the association tests are performed since each marker has more than two alleles. The following code invokes the three case-control tests to find out whether there is a significant association between either of the markers and disease status. Note that the same output could be produced by omitting the three tests, ALLELE, GENOTYPE, and TREND, from the PROC CASECONTROL statement.

```
proc casecontrol data=founders genotype allele trend;
   trait disease;
   var a1-a4;
run;

proc print noobs heading=h;
    format ProbAllele ProbGenotype ProbTrend pvalue6.5;
    format ChiSqAllele ChiSqGenotype ChiSqTrend 6.3;
run;
```

An output data set is created by default, and the output from the PRINT procedure is displayed in Output 4.1.1.

Output 4.1.1 Output Data Set from PROC CASECONTROL for Multiallelic Markers

```
              Num       Num      ChiSq      ChiSq      ChiSq        df
 Locus      Trait1    Trait2    Genotype    Allele     Trend     Genotype

  M1          30        30       27.333      4.441     5.039        24
  M2          30        30       18.077      8.772    13.244        15

   df         df        Prob       Prob       Prob
 Allele     Trend    Genotype     Allele     Trend

    7          7       0.2892     0.7278    0.6552
    7          7       0.2586     0.2694    0.0664
```

This analysis finds no significant association between disease status and either of the markers. Suppose, however, that allele 7 of the second marker had been identified by previous studies as an allele of interest for this particular disease, and thus there is concern that its effect is swamped by the other seven alleles. The data set can be modified as follows so that the second marker is considered a biallelic marker with alleles 7 and “not 7.”

```
data marker2;
   set founders;
   if a3 ne 7 then a3=1;
   if a4 ne 7 then a4=1;
   keep id a3 a4 disease;
```

Now all three tests can be performed on the marker in the new data set, as follows:

```
proc casecontrol data=marker2;
   trait disease;
   var a3 a4;
run;

proc print noobs heading=h;
    format ProbAllele ProbGenotype ProbTrend pvalue6.5;
    format ChiSqAllele ChiSqGenotype ChiSqTrend 6.3;
run;
```

PROC CASECONTROL performs all three tests by default since none were specified. The output data set for this analysis is displayed in Output 4.1.2.

Output 4.1.2 Output Data Set from PROC CASECONTROL for a Biallelic Marker

```
               Num       Num      ChiSq      ChiSq     ChiSq        df
    Locus    Trait1    Trait2    Genotype    Allele    Trend     Genotype

     M1        30        30       12.193      6.599    10.103        2

      df        df       Prob        Prob      Prob
    Allele    Trend    Genotype     Allele     Trend

       1         1      0.0023      0.0102    0.0015
```

With just the single allele of interest, there is now a significant association (using a significance level of $\alpha = 0.05$) according to all three case-control tests between the marker (specifically, allele 7) and disease status. Note that the allele and trend tests, both of which are testing for additive allele effects, produce quite different p-values, which could be an indication that HWE does not hold for allele 7. This is in fact the case, which can be checked by running the ALLELE procedure on data set marker2 to test for HWE (see Chapter 2, "The ALLELE Procedure," for more information). The excess of heterozygotes forces X_A^2 to be smaller than X_T^2, and only X_T^2 remains a valid chi-square statistic under the HWE violation.

Example 4.2: Analyzing Data in the Tall-Skinny Format

This example demonstrates how data in the tall-skinny format can be analyzed using PROC CASECONTROL with the options TALL, MARKER=, and INDIV=. Here, the same data that were used in the "Getting Started" example are used, but in this alternative format.

```
data talldata;
   input affected $ id snpname $ allele1 allele2;
   datalines;
 N  1 Marker1 1 1
 N  2 Marker1 1 1
 N  3 Marker1 2 1
 N  4 Marker1 2 2
 N  5 Marker1 1 1
 N  6 Marker1 2 1
 N  7 Marker1 1 1
 N  8 Marker1 2 2
 N  9 Marker1 2 1
 N 10 Marker1 2 1
 N 11 Marker1 2 1
 N 12 Marker1 2 2
 N 13 Marker1 2 1
 N 14 Marker1 2 1
 N 15 Marker1 2 2
 N 16 Marker1 1 1
 N 17 Marker1 1 1
 N 18 Marker1 2 1
 A 19 Marker1 2 1
 A 20 Marker1 2 1
 A 21 Marker1 2 2
 A 22 Marker1 1 2
 A 24 Marker1 1 1
 A 25 Marker1 2 1
 A 26 Marker1 2 1
 A 27 Marker1 2 1
 A 28 Marker1 2 1
 A 29 Marker1 1 1
 A 30 Marker1 2 1
 A 31 Marker1 2 2
 A 32 Marker1 1 1
 A 33 Marker1 1 1
 A 34 Marker1 2 2
 N  1 Marker2 2 2
 N  2 Marker2 1 1
 N  3 Marker2 1 1

   ... more lines ...

 A 29 Marker8 2 2
 A 30 Marker8 1 2
 A 31 Marker8 2 2
 A 32 Marker8 2 2
 A 33 Marker8 2 2
 A 34 Marker8 1 2
;
```

Note how all marker alleles are contained in two columns, and there are identifiers for the markers and individuals sampled. The data set is first sorted by the marker ID, then by the individual ID. One advantage of this data format is that there is no restriction on the number of markers analyzed since, unlike the columns, there is no limit on the number of rows in a SAS data set. The following code can be used to analyze this data set:

```
proc casecontrol data=talldata tall marker=snpname indiv=id;
   var allele1 allele2;
   trait affected;
run;

proc print;
    format ProbAllele ProbGenotype ProbTrend pvalue6.5;
    format ChiSqAllele ChiSqGenotype ChiSqTrend 6.3;
run;
```

Applying this code to the data in this format produces the same output shown in the “Getting Started” example, Figure 4.1.

Example 4.3: Producing Odds Ratios for Various Disease Models

In addition to the chi-square test statistics between a marker and a disease, you might be interested in inferences about the odds ratios based on the table of allele-by-disease counts for each marker. You can use the OR option in the PROC CASECONTROL statement to have the odds ratios from these tables included in the OUTSTAT= data set along with confidence limits based on the level specified in the ALPHA= option (or 0.05 by default).

This data set contains 20 individuals genotyped at five SNPs, as follows.

```
data genotypes;
   input (g1-g5) ($) disease;
   datalines;
 B/B B/A B/A A/A A/A 1
 B/B B/B B/A A/A B/B 0
 A/B B/B B/A B/A B/B 1
 B/B A/B B/A A/A B/B 1
 B/B B/B A/B A/B B/B 0
 A/A B/B A/A B/A B/B 0
 B/B B/B B/A B/A A/B 1
 B/B B/B A/A B/A A/B 1
 B/B B/B A/A A/A B/B 1
 B/B A/A B/B B/A B/B 1
 B/B B/A B/B B/A A/B 0
 B/B B/B A/A A/A B/B 1
 B/B B/A B/B B/B B/B 0
 B/B B/B B/B A/A B/B 1
 B/B B/B B/A B/A B/B 0
 A/A B/B B/B B/B B/B 1
 B/B B/B B/B B/A B/B 1
 B/B B/B B/B B/B B/B 1
 B/B B/B B/A A/A B/A 0
 B/B B/B A/A B/A B/B 1
;
```

An output data set containing the odds ratios and respective confidence limits can be produced with the following code:

```
proc casecontrol data=genotypes genocol or;
   var g1-g5;
   trait disease;
run;


proc print heading=h;
   var Locus NumTrait0 NumTrait1 AlleleOddsRatio LowerCL UpperCL;
run;
```

Note that the GENOCOL option is used since columns contain genotypes, not individual alleles. The columns listed in the VAR statement of PROC PRINT are shown in Output 4.3.1. Since the odds ratios are based on the allele counts, an additive disease model is assumed.

Output 4.3.1 Output Data Set from PROC CASECONTROL Containing Odds Ratios: Additive Model

```
                                        Allele
                        Num      Num     Odds
      Obs    Locus    Trait0   Trait1   Ratio     LowerCL    UpperCL

       1      g1         7        13    1.27778    0.18724    8.72011
       2      g2         7        13    0.91667    0.14597    5.75651
       3      g3         7        13    0.87500    0.23620    3.24146
       4      g4         7        13    0.83333    0.22242    3.12219
       5      g5         7        13    0.91667    0.14597    5.75651
```

What if you want to look at odds ratios for genotypes assuming a dominant or recessive disease model? You can use PROC FORMAT to group together genotypes, such as the heterozygous genotype with one of the homozygous genotypes. In the following code, two formats are created for the genotypes: $DOM_B. for a model where allele *B* is dominant (or *A* is recessive) and $REC_B. for a model where allele *B* acts in a recessive manner.

```
proc format;
   value $dom_B 'A/A'='A/A'
                'B/B'='B/B'
                'A/B'='B/B'
                'B/A'='B/B'
                ;
   value $rec_B 'A/A'='A/A'
                'B/B'='B/B'
                'A/B'='A/A'
                'B/A'='A/A'
                ;
run;

proc casecontrol data=genotypes genocol or;
   var g1-g5;
   format g1-g5 $dom_b.;
   trait disease;
run;

proc print heading=h;
   var Locus NumTrait0 NumTrait1 AlleleOddsRatio LowerCL UpperCL;
run;
```

In this code, the FORMAT statement is used in PROC CASECONTROL to request odds ratios for a disease model where allele *B* is dominant; that is, the genotypes A/B and B/B are grouped into one category. The odds ratios for genotype A/A versus A/B and B/B are now shown in Output 4.3.2. Similarly, a disease model with *B* as the recessive allele could be tested instead using the $REC_B. format in the FORMAT statement.

Output 4.3.2 Output Data Set from PROC CASECONTROL Containing Odds Ratios: Dominance Model

```
                                         Allele
                        Num       Num     Odds
   Obs    Locus      Trait0    Trait1    Ratio     LowerCL     UpperCL

    1      g1           7        13      2.000     0.10574     37.8296
    2      g2           7        13      0.000      .            .
    3      g3           7        13      0.375     0.03326      4.2281
    4      g4           7        13      0.640     0.08798      4.6554
    5      g5           7        13      0.000      .            .
```

References

Agresti, A. (1990), *Categorical Data Analysis*, New York: John Wiley & Sons.

Armitage, P. (1955), "Tests for Linear Trend in Proportions and Frequencies," *Biometrics*, 11, 375–386.

Bacanu, S.-A., Devlin, B., and Roeder, K. (2000), "The Power of Genomic Control," *American Journal of Human Genetics*, 66, 1933–1944.

Devlin, B. and Roeder, K. (1999), "Genomic Control for Association Studies," *Biometrics*, 55, 997–1004.

Hodge, S. E. (1995), "An Oligogenic Disease Displaying Weak Marker Associations: A Summary of Contributions to Problem 1 of GAW9," *Genetic Epidemiology*, 12, 545–554.

Nielsen, D. M. and Weir, B. S. (1999), "A Classical Setting for Associations between Markers and Loci Affecting Quantitative Traits," *Genetic Research*, 74, 271–277.

Sasieni, P. D. (1997), "From Genotypes to Genes: Doubling the Sample Size," *Biometrics*, 53, 1253–1261.

Slager, S. L. and Schaid, D. J. (2001), "Evaluation of Candidate Genes in Case-Control Studies: A Statistical Method to Account for Related Subjects," *American Journal of Human Genetics*, 68, 1457–1462.

Westfall, P. H. and Young, S. S. (1993), *Resampling-Based Multiple Testing: Examples and Methods for* p-*Value Adjustment*, New York: John Wiley & Sons.

Chapter 5
The FAMILY Procedure

Contents

Overview: FAMILY Procedure

Family genotype data, though more difficult to collect, often provide a more effective way of testing markers for association with disease status than case-control data. Case-control data can uncover significant associations between markers and a disease that could be caused by factors other than linkage, such as population structure. Analyzing family data by using the FAMILY procedure ensures that any significant associations found between a marker and disease status are due to linkage between the marker and disease locus. This is accomplished by using the transmission/disequilibrium test (TDT) and several variations of it that can accommodate different types of family data. One type of family consists of parents, at least one heterozygous, and an affected child who have all been genotyped. This family structure is suitable for the original TDT. Families having at least one affected and one unaffected sibling from a sibship that have both been genotyped can be analyzed using the sibling tests: the sib TDT (S-TDT) or the nonparametric sibling disequilibrium test (SDT). Both types of families can be jointly analyzed using the combined versions of the S-TDT and SDT and the reconstruction-combined TDT (RC-TDT). The RC-TDT can additionally accommodate families with no unaffected children and missing parental genotypes in certain situations.

When the trait of interest is quantitative, regression and variance component analyses can be used to test for marker associations (Allison 1997; Fulker et al. 1999; Rabinowitz 1997). These models were extended to accommodate any size nuclear family with or without parental genotypes (Abecasis, Cardon, and Cookson 2000; Monks and Kaplan 2000) and then to general pedigrees (Abecasis, Cookson, and Cardon 2000). The strength of many procedures in SAS/STAT in these areas can be applied to these statistical tests, though some data manipulation is required to form the correct inputs. In order to simplify the data preparation steps, PROC FAMILY can produce an output data set containing the pair of allelic transmission scores at each marker allele. This data set can be used in the MIXED procedure, for example, to test for association and linkage between marker genotypes and a quantitative trait via the method of Abecasis, Cookson, and Cardon (2000).

Getting Started: FAMILY Procedure

Example

The following example demonstrates how you can use PROC FAMILY to perform one of several family-based tests, the TDT. You have collected the following family genotypic data that you input into a SAS data set:

```
data example;
   input ped indiv father mother disease (a1-a4)($);
   datalines;
 1   1  0  0 1 a b a c
 1   2  0  0 1 c c a d
 1 101  1  2 1 a c a d
 1 102  1  2 1 b c a d
 1 103  1  2 1 a c c a
 1 104  1  2 2 b c a a
 2   3  0  0 1 e e f g
 2   4  0  0 1 d e g a
 2 105  3  4 1 e d f a
 2 106  3  4 2 e e g a
 3   5  0  0 1 d a a c
 3   6  0  0 1 e e c a
 3 107  5  6 2 a e a a
 4   7  0  0 1 f b a g
 4   8  0  0 1 c e h g
 4 108  7  8 2 b e a g
 4 109  7  8 1 f c g g
 4 110  7  8 1 b c a g
 4 111  7  8 1 b c a h
 5   9  0  0 1 a f d c
 5  10  0  0 1 h d c h
```

```
   5 112   9 10 2 a d d c
   5 113   9 10 1 f d d c
   6  11   0  0 1 b e c g
   6  12   0  0 1 d f a g
   6 114  11 12 2 b f c a
   6 115  11 12 1 b d g a
   7  13   0  0 1 e d c c
   7  14   0  0 1 e h d a
   7 116  13 14 1 e h c a
   7 117  13 14 2 d e c a
   7 118  13 14 1 d h c d
   7 119  13 14 1 d h c d
;
```

The first column of the data set contains the pedigree ID, followed by an individual ID and the two parental IDs. The fifth column is a variable representing affection status of a disease. The last four columns of this data set contain the two alleles at each of two markers for each individual. Since there are no missing parental genotypes in this data set, the TDT is a reasonable test to perform in order to determine whether either of the two markers is significantly linked to the disease locus whose location you are trying to pinpoint. Furthermore, close inspection of the data reveals that there is only one affected child (which corresponds to a value of "2" for the disease affection variable) per each family. Thus, the TDT is also a valid test for association with the disease locus. To perform the analysis, you would use the following statements:

```
proc family data=example prefix=Marker outstat=stats tdt contcorr;
   id ped indiv father mother;
   trait disease / affected=2;
   var a1-a4;
run;



proc print data=stats;
   format ProbTDT pvalue6.5;
run;
```

This code creates an output data set stats, which contains the chi-square statistic, degrees of freedom, and p-value for testing each marker for linkage and association with the disease locus by using the TDT. The PREFIX= option in the PROC FAMILY statement specifies that the two markers be named Marker1 and Marker2 in the output data set. The CONTCORR option indicates that the continuity correction of 0.5 should be used in calculating the chi-square statistic. The AFFECTED= option of the TRAIT statement specifies which value of the variable disease should be considered "affected." Note that the pedigree ID variable is listed in the ID statement; however, it is not necessary for this data set, since all the individual IDs are unique. The same results would be obtained if this variable were omitted.

Figure 5.1 shows the output data set that is produced.

Figure 5.1 Statistics for the TDT

```
                             ChiSq       df       Prob
 Obs       Locus              TDT       TDT        TDT

  1       Marker1          1.57143       6       0.9546
  2       Marker2          5.79861       5       0.3263
```

Figure 5.1 displays the statistics for the TDT. Since both markers are multiallelic, a joint test of all alleles at each marker is performed by default. The degrees of freedom (in the dfTDT column) indicate that there are seven alleles at Marker1 and six alleles at Marker2, since df$= k - 1$, where k is the number of marker alleles. The ProbTDT column shows that neither of the markers is significantly linked and associated with the disease locus.

Syntax: FAMILY Procedure

The following statements are available in PROC FAMILY.

```
PROC FAMILY < options > ;
    BY variables ;
    ID variables ;
    TRAIT variable < / AFFECTED= value > ;
    VAR variables ;
    XLVAR variables ;
```

Items within angle brackets (< >) are optional, and statements following the PROC FAMILY statement can appear in any order. The ID and the VAR and/or XLVAR statements are required.

PROC FAMILY Statement

PROC FAMILY < *options* > **;**

You can specify the following options in the PROC FAMILY statement.

COMBINE

specifies that the combined versions of the S-TDT and SDT be performed. Thus, families containing parental genotypes can be analyzed under certain conditions by using the TDT; otherwise the specified sibling test is performed. Note that if TDT is also being performed, the TDT is done independently of any other tests. By default, the combined versions are not used.

CONTCORR

CC

specifies that a continuity correction of 0.5 should be used for the TDT, S-TDT, and RC-TDT tests in their asymptotic normal approximations. By default, no correction is used.

DATA=*SAS-data-set*

names the input SAS data set to be used by PROC FAMILY. The default is to use the most recently created data set.

DELIMITER=*'string'*

indicates the string that is used to separate the two alleles that compose the genotypes contained in the variables specified in the VAR statement. This option is ignored if GENOCOL is not specified.

GENOCOL

indicates that columns specified in the VAR statement contain genotypes instead of alleles. When this option is specified, there is one column per marker. The genotypes must consist of the two alleles separated by a delimiter.

MULT=JOINT

MULT=MAX

specifies which multiallelic version of the TDT, S-TDT, SDT, and RC-TDT tests should be performed. The joint version of the multiallelic tests combines the analyses for each allele at a marker into one overall test statistic, with degrees of freedom (df) corresponding to the number of alleles at the marker. The max version of the multiallelic tests determines whether there is at least one allele with a significant test statistic, using the maximum 1 df statistic over all alleles with a multiple testing adjustment made. By default, the joint version of the multiallelic tests is performed. This option has no effect on biallelic markers.

NDATA=*SAS-data-set*

names the input SAS data set containing names, or identifiers, for the markers used in the output. There must be a NAME variable in this data set, which should contain the same number of rows as there are markers in the input data set specified in the DATA= option. When there are fewer rows than there are markers, markers without a name are named using the PREFIX= option. Likewise, if there is no NDATA= data set specified, the PREFIX= option is used. If both the VAR and XLVAR statements are specified, names are first used for the markers in the VAR statement, then for the X-linked markers.

OUTQ=*SAS-data-set*

names the output SAS data set containing all the variables from the input data set in addition to the allelic transmission scores at each marker allele to be used in testing for association and linkage with a quantitative trait. When this option is used, the TRAIT statement is not required.

OUTSTAT=*SAS-data-set*

names the output SAS data set containing the p-values for the tests specified in the PROC FAMILY statement. When this option is omitted, an output data set is created by default and named according to the DATA*n* convention.

PERMS=*number*

indicates that Monte Carlo estimates of exact p-values for the family-based tests should be calculated using permutation samples instead of the p-values from the asymptotic χ^2 distribution. Large values of *number* (10,000 or more) are usually recommended for accuracy, but long execution times can result, particularly with large data sets. When this option is omitted, no permutations are performed and p-values from the asymptotic χ^2 distribution are reported.

PREFIX=*prefix*

specifies a prefix to use in constructing names for marker variables in all output. For example, if PREFIX=VAR, the names of the variables are VAR1, VAR2, . . . , VAR*n*. Note that this option is ignored when the NDATA= option is specified, unless there are fewer names in the NDATA data set than there are markers. If this option is omitted, PREFIX=M is the default when variables contain alleles; if GENOCOL is specified, then the names of the variables specified in the VAR statement are used as the marker names.

RCTDT

requests that the reconstruction-combined TDT (RC-TDT) be performed. If none of the four test options (RCTDT, SDT, STDT, or TDT) are specified, then all four tests are performed by default. Note that error checking is always performed on families with at least one untyped parent in order to determine whether or not reconstruction of parental genotypes can be attempted.

SDT

requests that the SDT, a nonparametric alternative to the S-TDT, be performed. If none of the four test options (RCTDT, SDT, STDT, or TDT) are specified, then all four tests are performed by default. The COMBINE option can be used with this test to indicate that the combined version of the SDT should be performed.

SEED=*number*

specifies the initial seed for the random number generator used for permuting the data to calculate estimates of exact p-values. This option is ignored if PERMS= is not specified. The value for *number* must be an integer; the computer clock time is used if the option is omitted or an integer less than or equal to 0 is specified. For more details about seed values, see *SAS Language Reference: Concepts*.

SHOWALL

indicates that all families and markers should be included in the "Family Summary" table. When this option is omitted, a family is included in the table only for a marker where there is a genotype error according to a Mendelian inconsistency.

STDT

requests that the sibling TDT (S-TDT), which analyzes data from sibships, be performed. If none of the four test options (RCTDT, SDT, STDT, or TDT) are specified, then all four tests are performed by default. The COMBINE option can be used with this test to indicate that the combined version of the S-TDT should be performed.

TDT

requests that the original TDT be performed. If none of the four test options (RCTDT, SDT, STDT, or TDT) are specified, then all four tests are performed by default.

BY Statement

BY *variables* **;**

You can specify a BY statement with PROC FAMILY to obtain separate analyses on observations in groups defined by the BY variables. When a BY statement appears, the procedure expects the input data set to be sorted in the order of the BY variables. The *variables* are one or more variables in the input data set.

If your input data set is not sorted in ascending order, use one of the following alternatives:

- Sort the data by using the SORT procedure with a similar BY statement.
- Specify the BY statement option NOTSORTED or DESCENDING in the BY statement for the FAMILY procedure. The NOTSORTED option does not mean that the data are unsorted but rather that the data are arranged in groups (according to values of the BY variables) and that these groups are not necessarily in alphabetical or increasing numeric order.
- Create an index on the BY variables by using the DATASETS procedure (in Base SAS software).

For more information about the BY statement, see *SAS Language Reference: Concepts*. For more information about the DATASETS procedure, see the *Base SAS Procedures Guide*.

ID Statement

ID *variables* **;**

The ID statement is required and must contain, in the following order, either of these sequences of IDs:

- the pedigree ID, the individual ID, then the two parental ID variables
- the individual ID, then the two parental IDs

Thus if only three variables are specified in the ID statement, it is assumed that the pedigree identifier has been omitted. The pedigree ID is not necessary if all the individual identifiers are unique. The individual and two parental ID variables can be either numeric or character, but all three must be of the same type. The pedigree variable, if specified, can be either numeric or character regardless of the type of the other three identifiers.

TRAIT Statement

TRAIT *variable < / AFFECTED=value >* **;**

The TRAIT statement identifies the trait variable and is required when the OUTQ= option is omitted. This variable must be binary, but it can be either character or numeric. By default, the second value of the TRAIT variable that appears in the input data set is considered to be "affected" for the tests. To specify a different value for "affected," add the /AFFECTED=value option to the TRAIT statement. For a variable with a numeric format, the number that corresponds to "affected" should be specified (AFFECTED=1); if the variable has a character format, the level that corresponds to "affected" should be specified in quotes (AFFECTED="a").

VAR Statement

VAR *variables* **;**

The VAR statement identifies the variables containing either the marker alleles or, if GENOCOL is specified, the marker genotypes. By default, the VAR statement should contain $2m$ variable names, where m is the number of markers in the data set. Note that variables containing alleles for the same marker should be listed consecutively. When GENOCOL is specified, there should be one variable per marker.

XLVAR Statement

XLVAR *variables< / SEX=variable >* **;**

The XLVAR statement identifies the variables containing either the X-linked marker alleles or, if GENOCOL is specified, genotypes. By default, the XLVAR statement should contain $2m$ variable names, where m is the number of X-linked markers in the data set. Note that variables containing alleles for the same marker should be listed consecutively. The second allele variable for males in the data set must be nonmissing but is ignored since males have only one allele at markers on the X-chromosome. When GENOCOL is specified, there should be one variable per marker. When X-linked markers are analyzed, there must be a SEX variable in the data set indicating whether individuals are male (1 or "M") or female (2 or "F"). If this variable is named something other than SEX, the /SEX=variable option must be added to the XLVAR statement in order to indicate the name of the variable containing individuals' sex. See "X-Linked Version of Tests" on page 93 for more information about X-linked tests.

Details: FAMILY Procedure

Statistical Computations

For all tests, it is assumed that the marker has two alleles, M_1 and M_2. Extensions to multiallelic markers are made by performing the tests on each allele in turn, with the current allele being considered to be M_1

and all other alleles considered to be M_2. When the CONTCORR option is specified in the PROC FAMILY statement, the z score statistics of all versions of the TDT, S-TDT, and RC-TDT can be continuity corrected by subtracting 0.5 from the absolute value of the numerator. The two-sided p-value for each z score using the normal distribution is equivalent to using the p-value from the χ_1^2 distribution for the square of the z score, and this chi-square form of the statistic is reported in the output data set.

TDT

The TDT (Spielman, McGinnis, and Ewens 1993) is implemented using a normal approximation. This test includes families where both parents have been genotyped for the marker and at least one is heterozygous. If only one parent has been genotyped, that parent is heterozygous, and the affected child is not homozygous and does not have the same genotype as the typed parent, then the TDT can be applied to this family as well (Curtis and Sham 1995). The TDT tests for equality between the proportion of times a heterozygous parent transmits the M_1 allele to an affected child and the proportion of times a heterozygous parent transmits the M_2 allele to an affected child. The normal approximation to the binomial is used to form the z score statistic

$$Z = \frac{b - \frac{b+c}{2}}{\sqrt{\frac{b+c}{4}}}$$

where b is the number of M_1 alleles in affected children from heterozygous parents and c is the number of M_2 alleles in affected children from heterozygous parents.

Two extensions to a multiallelic TDT are available. The first, which is performed by default or when MULT=JOINT is specified in the PROC FAMILY statement, combines the TDT for each of k alleles at a marker into one statistic as follows (Spielman and Ewens 1996):

$$T_J = \frac{k-1}{k} \sum_{v=1}^{k} Z_v^2$$

where Z_v is simply the Z defined in the preceding paragraph, with allele M_v treated as M_1 and all other alleles as M_2 for each $v = 1, \ldots, k$. T_J and the continuity-corrected form T'_J have an asymptotic χ^2_{k-1} distribution, and the corresponding p-value is reported.

Alternatively, if the MULT=MAX option is specified, either z_m or z'_m (when the CONTCORR option is specified) is used, where $z_m = \max_{1 \le v \le k} |Z_v|$. The equivalent one degree of freedom chi-square statistic is reported, and a Bonferroni correction is applied to its p-value.

NOTE: The TDT is a valid test of linkage and association only when the data consist of unrelated nuclear families and each family contains only one affected child. Otherwise, it is a valid test of linkage only.

S-TDT

The z score procedure given by Spielman and Ewens (1998) is used to calculate p-values for the S-TDT. This test can be applied to families where there are at least one affected sibling and one unaffected sibling, and not all siblings have the same genotype. The z score, whose two-sided p-value is approximated using the normal distribution, is calculated as $z = (Y - A)/\sqrt{V}$. Y represents the total observed number of M_1 alleles in the affected siblings. For t total siblings in the family, a affected and u unaffected, and r that are M_1/M_1 and s that are M_1/M_2, summing over families gives

$$A = \sum (2r + s)a/t$$

and

$$V = \sum au[4r(t-r-s) + s(t-s)]/[t^2(t-1)]$$

as the expected value and variance of Y, respectively.

When the COMBINE option is specified in the PROC FAMILY statement, the S-TDT and TDT are combined as follows: the TDT is applied to all alleles within a family that meet the requirements described in the preceding section. The S-TDT is then applied to the remaining alleles within a family that meet its requirements described in the preceding paragraph. Using the notation already given for these tests, the z score for the combined test can then be written as

$$Z = \frac{(Y+b) - (A + \frac{b+c}{2})}{\sqrt{V + \frac{b+c}{4}}}$$

For multiallelic markers, the same extensions can be made to the S-TDT and combined S-TDT that were made to the TDT (Monks, Kaplan, and Weir 1998); that is, either a joint test over all alleles (using T_{mcomb}) or the maximum z score of all the alleles with the p-value being Bonferroni-corrected.

NOTE: The S-TDT is a valid test of linkage and association only when the data consist of unrelated nuclear families and each family contains only one affected and one unaffected sibling. Otherwise, it is a valid test of linkage only.

SDT

The SDT (Horvath and Laird 1998) is a sign test used on discordant sibling pairs. As with the S-TDT, one affected sibling and one unaffected sibling are required to be in each family, but unlike the S-TDT, the SDT remains a valid test of linkage and association when the sibship is larger.

The notation from the S-TDT is used, except now the quantities a, u, r, s, and Y are defined for each sibship/family, so, for example, there are a affected siblings in the family and u unaffected siblings in the family. Treating each allele M_v in turn as M_1 and all other alleles as M_2, $v = 1, \ldots, k$, define for each family in the data the average number of v alleles among affected siblings and unaffected siblings respectively as

$$m_v^a = Y/a$$

$$m_v^u = [(2r+s) - Y]/u$$

Then $d_v = m_v^a - m_v^u$ for each family, and summing over families gives $S_v = \sum \text{sgn}(d_v)$, where $\text{sgn}(d_v) = 1$ for $d_v > 0$, 0 for $d_v = 0$, and -1 for $d_v < 0$. The joint multiallelic SDT statistic (mSDT) is then defined by Czika and Berry (2002) as $T = \mathbf{S}'\mathbf{W}^-\mathbf{S}$, where $\mathbf{S}' = (S_1, \ldots, S_k)'$ and $W_{vw} = \sum \text{sgn}(d_v)\text{sgn}(d_w)$, $v, w = 1, \ldots, k$, and $\mathbf{W}^-$ is the Moore-Penrose generalized inverse of $\mathbf{W}$. T has an asymptotic $\chi^2_{k'}$ distribution, where $k' = \text{rank}(\mathbf{W})$, and this distribution is used to obtain p-values for the SDT (Czika and Berry 2002). When there are only two alleles at the marker, this joint multiallelic version of the SDT reduces to the biallelic version of the SDT.

This sibship test is also combined with the TDT when the COMBINE option in the PROC FAMILY statement is specified, creating a test that can potentially use more of the data (Horvath and Laird 1998; Curtis, Miller, and Sham 1999). In order to maintain the test's validity as a test of association in families with more than one affected and one unaffected sibling, a nonparametric multiallelic TDT is used, which is in the same $\mathbf{S}'\mathbf{W}^-\mathbf{S}$

form as the SDT. This test statistic for the joint test also has an asymptotic $\chi^2_{k'}$ distribution (Czika and Berry 2002), and the corresponding p-value is reported.

When the MULT=MAX option is specified in the PROC FAMILY statement, then the SDT chi-square statistic is simply $\max_{1 \le v \le k}(S_v^2 W_{vv}^{-1})$ and has one degree of freedom. This applies to the SDT when used alone or combined with the TDT. As with the other tests, a Bonferroni correction is made to the p-value.

RC-TDT

The RC-TDT (Knapp 1999a) takes the combined S-TDT a step further by reconstructing missing parental genotypes when possible in order to use more families. The RC-TDT can be applied to families with at least one affected child that meet one of the following conditions:

- Both parents are typed with at least one heterozygous for M_1.
- One parent is typed, the other can be reconstructed, and at least one parent is heterozygous for M_1.
- Both parents' genotypes are missing but can be reconstructed, and at least one parent is heterozygous for M_1.
- At least one parental genotype is missing and cannot be reconstructed, but the conditions for the S-TDT are met.
- One parental genotype is missing and cannot be reconstructed, the other parent is heterozygous for M_1, and at least one affected child is heterozygous for M_1 and an allele not in the typed parent (Knapp 1999b).

Reconstruction of parental genotypes is attempted only when there are no genotyping errors in the family for the marker being tested. As with the S-TDT, a z score is created using the statistic Y, but Knapp (1999a) calculates a different expected value e and variance v of Y, which takes into account the bias created by the genotype reconstruction, to form the z score over all families:

$$Z = (Y - e)/\sqrt{v}$$

For multiallelic markers, the same extensions can be made to the RC-TDT that were made to the TDT and S-TDT–that is, either a joint test over all alleles, or the maximum z score of all the alleles with the p-value being Bonferroni-corrected.

NOTE: The RC-TDT is a valid test of linkage and association only when the data consist of unrelated nuclear families and each family contains only one affected and one unaffected sibling. Otherwise, it is a valid test of linkage only.

X-Linked Version of Tests

For markers from the X-chromosome that are specified in the XLVAR statement, the preceding tests are not applicable since females have two alleles at such markers and males have only one. Horvath, Laird, and Knapp (2000) presented X-linked versions of the TDT, S-TDT, combined S-TDT, and RC-TDT that accommodate these markers. For the X-TDT, the only difference in calculating the values b and c is that for X-linked markers, transmissions only from heterozygous *mothers*, instead of heterozygous parents, are used. Note that even though the paternal genotype is not directly used, it must be nonmissing except for when including transmissions to sons in the family, or for daughters who are heterozygous but with a different genotype than their mother (not possible for a biallelic marker).

For the XS-TDT, each sibship is divided into two subsibships so that female sibs and male sibs are analyzed separately. The statistic is then constructed treating the subsibships independently. For female sibs, the parameters A and V are the same as those defined for the S-TDT. For male sibs, the X-linked expected value and variance of the number of M_v alleles in affected siblings is calculated across male subsibships as

$$A = \sum ac/t$$

and

$$V = \sum auc(t-c)/[t^2(t-1)]$$

where c is the number of M_v alleles among all males in a subsibship. The X-linked version of the combined S-TDT is calculated analogously to the combined S-TDT for autosomal markers by using the X-linked versions of the TDT and S-TDT.

The X-linked RC-TDT can be divided into four situations:

- Both parents are typed and the X-TDT can be applied.
- Only the maternal genotype is missing.
- Only the paternal genotype is missing.
- Both parental genotypes are missing.

(Note that the first situation also includes the preceding exception when the maternal genotype is nonmissing). Horvath, Laird, and Knapp (2000) show, as with the original RC-TDT, expected values and variances of the number of M_1 alleles in affected children when reconstructing parental genotypes in each of the last three situations listed. Using these values, the XRC-TDT can be formed identically to the statistic for the RC-TDT shown in the preceding section.

Permutation Tests

By default, p-values from the asymptotic χ^2 distribution with appropriate degrees of freedom are reported for all tests. However, if the PERMS= option is specified in the PROC FAMILY statement, then Monte Carlo estimates of exact p-values are calculated using the permutation procedure for the TDT, S-TDT, SDT, and combined S-TDT and SDT. When the TDT is being performed, including when it is performed in the combined tests, new samples are formed by permuting the alleles that are transmitted to the offspring from the parents and those that are not transmitted (Kaplan, Martin, and Weir 1997). Each affected child in a nuclear family is assigned a genotype comprising one allele from each parent, with each allele being randomly selected from the pair possessed by an individual parent. When the sibling tests are used and the parental information is ignored, the permutation procedure involves randomly permuting the affection status of siblings within each sibship (Spielman and Ewens 1998; Monks, Kaplan, and Weir 1998). For each test, the corresponding test statistic is calculated for the original sample as well as each of the permuted samples. The approximation to the exact p-value is then calculated as the number of times the test statistic from a permuted sample exceeds the test statistic from the original sample.

Creating Allelic Transmission Scores

Abecasis, Cookson, and Cardon (2000) define a pair of orthogonal allelic transmission scores, b and w, the expected genotype and deviate, respectively, for each individual at each marker. To create these scores, the genotype in terms of allele M_v must first be defined as $g_{jv} = m_{jv} - 1$ for individual j, where m_{jv} represents the number of M_v alleles that the genotype comprises. For any founder j, an individual whose parents are not observed, in the sample, these scores are defined as $b_{jv} = g_{jv}$ and $w_{jv} = 0$. Otherwise, let M_j and F_j be the respective indices of the mother and father of individual j. Then for any nonfounder, assuming scores for an individual's ancestors are calculated before his or her own,

$$b_{jv} = \begin{cases} (b_{M_j v} + b_{F_j v})/2, & b_{M_j v} \text{ and } b_{F_j v} \text{ are nonmissing} \\ \sum_{k \in S_j} g_{kv}/|S_j|, & \text{otherwise} \end{cases}$$

where $S_j = \{k : M_k = M_j, F_k = F_j, \text{ and } k \text{ genotyped}\}$, and then $w_{jv} = g_{jv} - b_{jv}$. These scores are calculated for all alleles at the markers specified in the VAR statement and are included in the OUTQ= data set.

Missing Values

An individual's genotype for a marker is considered missing if at least one of the alleles at the marker is missing. Any missing genotypes are excluded from all calculations. However, the individual's nonmissing genotypes at other loci can be used as part of the calculations. If a child has a missing trait value, then that individual is excluded from the statistical tests; allelic transmission scores can still be calculated for such children. Missing trait values of individuals used only as parents do not affect the analysis. See the following section for information about missing values in the ID variables.

DATA= Data Set

The DATA= data set has columns representing markers, ID variables, and a trait, and rows representing the individuals. There must be one binary trait variable listed in the TRAIT statement; the three ID variables consisting of the individual's ID and the two parental IDs, all of the same type, must be listed in the ID statement, and optionally the pedigree ID if the individual identifiers are not unique. Note that only individuals with both parents appearing in the data, even if all the parents' genotypes are missing, can be used as affected children or in sib pairs for analysis. However, if the individual is used only as a parent, then that individual's parents need not appear in the data. An individual's parents must occur in the data set before the individual does, and full siblings must be in consecutive observations. If a pedigree ID variable is specified in the ID statement, any individual with a missing value for that variable is excluded from the analysis, as a parent and as a child. There are two columns for each marker, representing the two alleles at that marker carried by the individual. These two columns must be listed consecutively in the VAR statement. These marker variables must all be of the same type, but can be either character or numeric variables.

OUTQ= Data Set

The OUTQ= data set contains all the variables from the input data set, as well as the variables B_*marker_allele* and W_*marker_allele* for each allele at the loci specified in the VAR statement containing the allelic transmission scores.

OUTSTAT= Data Set

When the TRAIT statement is specified, the OUTSTAT= data set is created and contains the following variables:

- the BY variables, if any
- Locus
- X_Linked when there is at least one marker specified in the XLVAR statement. This variable contains an "X" for X-linked markers and is blank for markers from the VAR statement.
- the chi-square statistics for each test performed: ChiSqTDT, ChiSqSTDT, ChiSqSDT, and ChiSqRCTDT
- the degrees of freedom for each test performed: dfTDT, dfSTDT, dfSDT, and dfRCTDT
- the p-values for each test performed: ProbTDT, ProbSTDT, ProbSDT, and ProbRCTDT

Displayed Output

This section describes the displayed output from PROC FAMILY. See the section "ODS Table Names" on page 97 for details about how this output interfaces with the Output Delivery System.

Family Summary

The "Family Summary" table lists information about the nuclear families, including the pedigree ID (if listed in the ID statement) and the two parental IDs, and then the following information for each marker locus:

- number of typed parents
- number of affected and unaffected children with nonmissing genotypes (when TRAIT statement is used)
- number of children with nonmissing genotypes (when TRAIT statement is omitted)
- error code

Note that when SHOWALL is specified in the PROC FAMILY statement, all families and all markers are displayed in the table. Otherwise, only families with a Mendelian genotype error and the marker at which they have the error are included in the table. The error code is an integer that represents a particular type of genotype error that is described in the "Description of Error Codes" table.

Description of Error Codes

The "Description of Error Codes" table provides descriptions for the error codes listed in the "Family Summary" table. The descriptions of the family genotype errors all refer to Mendelian inconsistencies in the child(ren)'s genotypes with the parental genotypes. Error codes 1 through 5 can occur when neither of the parental genotypes for that marker is available (the sibship is the family unit). Codes 6 and 7 can occur for families with exactly one parent genotyped, and error code 8 can occur for families with both parents genotyped.

Family X-Linked Summary

The "Family X-linked Summary" table lists information about the nuclear families at each of the X-linked markers, including the pedigree ID (if listed in the ID statement) and the father and mother IDs, and then the following information for each marker locus:

- which, if either, of the parents is typed
- number of affected and unaffected sons with nonmissing genotypes
- number of affected and unaffected daughters with nonmissing genotypes
- error code

Note that when SHOWALL is specified in the PROC FAMILY statement, all families and all X-linked markers are displayed in the table. Otherwise, only families with a Mendelian genotype error and the marker at which they have the error are included in the table. The error code is an integer that represents a particular type of genotype error that is described in the "Description of X-linked Error Codes" table.

Description of X-Linked Error Codes

The "Description of X-linked Error Codes" table provides descriptions for the error codes listed in the "Family X-linked Summary" table. The descriptions of the family genotype errors all refer to Mendelian inconsistencies in the child(ren)'s genotypes with the parental genotypes. Error codes 1 through 4 can occur when neither of the parental genotypes for that marker is available (the sibship is the family unit); codes 5 and 6 can occur for families with only the maternal genotype missing; codes 7 and 8 can occur for families with only the paternal genotype missing; and error code 9 can occur in families with both parents genotyped. If both parents have the same value for the SEX variable, an error code of 10 is reported.

ODS Table Names

PROC FAMILY assigns a name to each table it creates, and you must use this name to reference the table when using the Output Delivery System (ODS). These names are listed in Table 5.1.

Table 5.1 ODS Tables Created by the FAMILY Procedure

ODS Table Name	Description	PROC FAMILY Option	Statement
FamilySummary	Family summary	SHOWALL or at least one family with a genotype error	VAR

ODS Table Name	Description	PROC FAMILY Option	Statement
ErrorCodeDesc	Description of error codes	SHOWALL or at least one family with a genotype error	VAR
FamilyXLSummary	Family X-linked summary	SHOWALL or at least one family with a genotype error	XLVAR
XLErrorCodeDesc	Description of X-linked error codes	SHOWALL or at least one family with a genotype error	XLVAR

Examples: FAMILY Procedure

Example 5.1: Performing Tests with Missing Parental Data

The following data are from GAW9 (Hodge 1995) and contain 20 nuclear families that are genotyped at two markers. The data have been modified so that each mother's genotype is missing.

```
data gaw;
   input ped id f_id m_id sex disease m11 m12 m21 m22;
   datalines;
 1    1    0    0 1 1  7  8  7  2
 1    2    0    0 2 1  .  .  .  .
 1  401    1    2 1 1  7  2  7  6
 1  402    1    2 1 1  8  2  7  6
 1  403    1    2 1 1  7  2  2  7
 1  404    1    2 2 2  8  2  7  7
 2    3    0    0 1 1  4  4  1  3
 2    4    0    0 2 1  .  .  .  .
 2  405    3    4 2 1  4  6  1  7
 2  406    3    4 2 2  4  4  3  7
 3    5    0    0 1 1  6  7  7  2
 3    6    0    0 2 1  .  .  .  .
 3  407    5    6 2 2  7  4  7  7
 4    7    0    0 1 1  1  8  7  3
 4    8    0    0 2 1  .  .  .  .
 4  408    7    8 2 2  8  4  7  3
 4  409    7    8 1 1  1  2  3  3
 4  410    7    8 2 1  8  2  7  3
 4  411    7    8 1 1  8  2  7  5
 5    9    0    0 1 1  7  1  6  2
 5   10    0    0 2 1  .  .  .  .
 5  412    9   10 2 2  7  6  6  2
 5  413    9   10 1 1  1  6  6  2
 6   11    0    0 1 1  8  4  2  3
 6   12    0    0 2 1  .  .  .  .
 6  414   11   12 1 2  8  1  2  7
 6  415   11   12 1 1  8  6  3  7
```

```
   7   13    0     0 1 1  4  6  2  2
   7   14    0     0 2 1  .  .  .  .
   7  416   13    14 1 1  4  5  2  7
   7  417   13    14 2 2  6  4  2  7
   7  418   13    14 2 1  6  5  2  6
   7  419   13    14 1 1  6  5  2  6
   8   15    0     0 1 1  6  8  2  7
   8   16    0     0 2 1  .  .  .  .
   8  420   15    16 2 1  6  2  7  7
   8  421   15    16 2 1  8  6  2  7
   8  422   15    16 2 2  6  6  7  7
   8  423   15    16 2 1  6  6  7  7
   9   17    0     0 1 2  4  7  2  7
   9   18    0     0 2 1  .  .  .  .
   9  424   17    18 2 2  4  5  7  2
   9  425   17    18 2 1  7  4  2  7
   9  426   17    18 1 1  4  5  2  2
  10   19    0     0 1 1  6  4  2  7
  10   20    0     0 2 1  .  .  .  .
  10  427   19    20 2 2  4  4  7  2
  11   21    0     0 1 1  4  7  7  7
  11   22    0     0 2 1  .  .  .  .
  11  428   21    22 1 1  7  6  7  2
  11  429   21    22 2 2  7  4  7  2
  11  430   21    22 2 1  7  6  7  3
  12   23    0     0 1 1  7  6  7  5
  12   24    0     0 2 1  .  .  .  .
  12  431   23    24 1 2  6  4  7  7
  13   25    0     0 1 1  4  1  2  8
  13   26    0     0 2 1  .  .  .  .
  13  432   25    26 1 1  4  8  2  6
  13  433   25    26 1 2  1  8  8  6
  13  434   25    26 1 1  1  4  2  6
  14   27    0     0 1 1  7  6  3  2
  14   28    0     0 2 1  .  .  .  .
  14  435   27    28 1 1  6  2  3  3
  14  436   27    28 1 1  7  4  3  7
  14  437   27    28 1 1  6  2  2  7
  14  438   27    28 1 1  7  4  2  7
  14  439   27    28 2 2  6  2  2  7
  14  440   27    28 1 1  6  4  3  7
  15   29    0     0 1 1  2  4  7  4
  15   30    0     0 2 1  .  .  .  .
  15  441   29    30 1 1  4  2  7  7
  15  442   29    30 2 2  4  8  4  7
  15  443   29    30 2 1  4  2  7  5
  15  444   29    30 2 1  4  2  7  5
  15  445   29    30 1 1  2  8  7  5
  ;
```

Since there are missing parental data, the original TDT might not be the best test to perform on this data set. The following analysis uses the S-TDT, SDT, and RC-TDT to test markers for linkage with the disease locus.

```
proc family data=gaw prefix=Marker sdt stdt rctdt;
   id id f_id m_id;
   var m11 m12 m21 m22;
   trait disease / affected=2;
run;


proc print;
   format Probsdt probstdt probrctdt pvalue5.4;
run;
```

The output data set, which is created by default, is displayed in Output 5.1.1.

Output 5.1.1 Output Data Set from PROC FAMILY

```
                   ChiSq     ChiSq     ChiSq    df    df    df     Prob    Prob   Prob
Obs    Locus        STDT       SDT     RCTDT  STDT   SDT  RCTDT    STDT     SDT  RCTDT

 1    Marker1     5.6179    4.0083    4.7398    6     7     6     0.467   0.779  0.578
 2    Marker2    12.6191   10.7500   11.9388    7     8     7     0.082   0.216  0.103
```

Since only one parent is missing genotype information in each nuclear family, the TDT might be applicable to some of the families. The COMBINE option can be specified, as in the following code, to use the TDT in the appropriate families, and the S-TDT or SDT for all other families. This option does not apply to the RC-TDT, so that test is omitted from this analysis.

```
proc family data=gaw prefix=Marker tdt sdt stdt combine;
   id id f_id m_id;
   var m11 m12 m21 m22;
   trait disease / affected=2;
run;


proc print;
   format Probsdt probstdt probtdt pvalue5.4;
run;
```

The output data set is displayed in Output 5.1.2.

Output 5.1.2 Output Data Set from PROC FAMILY Using COMBINE Option

```
                   ChiSq     ChiSq     ChiSq    df    df    df    Prob    Prob    Prob
Obs    Locus        TDT       STDT       SDT   TDT  STDT   SDT     TDT    STDT     SDT

 1    Marker1   4.44444    6.3692    4.2380    5     6     7    0.487   0.383   0.752
 2    Marker2   2.00000   11.6489   10.7500    3     7     8    0.572   0.113   0.216
```

Note that the test statistics for the TDT and the S-TDT and SDT are not the same; this implies that not all families meet the requirements for the TDT. In this case, the S-TDT, SDT, and RC-TDT use more of the data than the TDT alone. However, since there is only one affected child in each nuclear family, the TDT is a valid test of association; since there is at least one occasion when there is more than one unaffected child in a nuclear family, the S-TDT and RC-TDT are not valid for testing for association of the marker with the disease locus (the SDT is always a valid test of association when the data consist of unrelated nuclear families). Both of these considerations, the amount of information that can be used and the validity for testing association, should be taken into account in deciding which test(s) to perform.

Another type of analysis can be performed using the MULT=MAX option in the PROC FAMILY statement. This option indicates that instead of doing a joint test over all the alleles at each marker, you want to perform a test to see if any of the alleles at a marker are significantly linked with the disease locus. This analysis is invoked with the following code, using only the SDT and RC-TDT:

```
proc family data=gaw prefix=Marker sdt rctdt combine mult=max;
   id id f_id m_id;
   var m11 m12 m21 m22;
   trait disease / affected=2;
run;


proc print;
   format Probsdt Probrctdt pvalue6.5;
run;
```

The output data set produced by this code is displayed in Output 5.1.3.

Output 5.1.3 Output Data Set from PROC FAMILY Using MULT=MAX Option

```
                         ChiSq      ChiSq      df      df        Prob      Prob
  Obs      Locus          SDT       RCTDT      SDT    RCTDT       SDT     RCTDT

   1      Marker1       2.66667    2.90050      1       1       0.7173    0.6199
   2      Marker2       3.57143    3.86422      1       1       0.4703    0.3946
```

The chi-square statistics for the tests always have one degree of freedom when the MULT=MAX option is used. Note, however, that the p-values are not the corresponding right-tailed probabilities for a χ^2_1 statistic; this is because the p-values are Bonferroni-corrected in order to account for taking the maximum of several chi-square statistics.

Example 5.2: Checking for Genotyping Errors

This example demonstrates the different kinds of family genotype errors (that is, Mendelian inconsistencies within a nuclear family) that can be detected by PROC FAMILY, and the output that displays this information. Here is a sample data set that contains genotype errors:

```
data ped_samp;
   input id p1 p2 a1 a2 dis;
   datalines;
  1   0   0 1 1 0
  2   0   0 2 3 0
  3   1   2 1 2 0
  4   1   2 4 5 1
101   0   0 . . 0
102   0   0 2 3 0
103 101 102 4 5 1
104 101 102 2 4 1
201   0   0 . . 0
202   0   0 1 4 0
203 201 202 1 5 1
204 201 202 1 6 0
205 201 202 1 7 1
301   0   0 . . 0
302   0   0 . . 0
303 301 302 1 2 1
304 301 302 1 3 0
305 301 302 1 4 0
401   0   0 . . 0
402   0   0 . . 0
403 401 402 1 1 1
404 401 402 2 2 1
405 401 402 3 3 0
501   0   0 . . 0
502   0   0 . . 0
503 501 502 1 1 0
504 501 502 2 2 0
505 501 502 1 3 1
601   0   0 . . 0
602   0   0 . . 0
603 601 602 1 1 1
604 601 602 1 4 0
605 601 602 2 3 0
701   0   0 . . 0
702   0   0 . . 0
703 701 702 1 2 0
704 701 702 2 3 1
705 701 702 1 4 0
707 701 702 2 5 1
801   0   0 1 3 0
802   0   0 . . 0
804 801 802 1 4 1
805 801 802 3 2 1
;
```

In addition to the usual output data set that is created, the SHOWALL option, used in the following code, requests that all families be included in the “Family Summary” table. Since there are families with genotype errors, this table would have been created by default, but only the families in error would be displayed in it.

```
proc family data=ped_samp showall;
   id id p1 p2;
   trait dis;
   var a1 a2;
run;

proc print;
run;
```

The “Family Summary” table shown in Output 5.2.1 includes an error code, which is explained in the “Description of Error Codes” table in Output 5.2.2. The statistics shown in Output 5.2.3 are based only on the last family since all the other families have some sort of genotype error and thus are excluded from the analyses. The analysis would need to be performed again after genotyping errors have been corrected.

Output 5.2.1 Summary of Family/Marker Information

```
                          The FAMILY Procedure

                            Family Summary

                                  Number       Typed
                                      of      Children
                                   Typed    ------------     Error
 Parent1      Parent2      Locus  Parents    Aff     Unaff     Code

 1            2            M1           2      1         1        8
 101          102          M1           1      2         0        6
 201          202          M1           1      2         1        7
 301          302          M1           0      1         2        5
 401          402          M1           0      2         1        4
 501          502          M1           0      1         2        3
 601          602          M1           0      1         2        2
 701          702          M1           0      2         2        1
 801          802          M1           1      2         0        0
```

Output 5.2.2 Description of Error Codes

```
                       Description of Error Codes

 Code    Description

    0    No errors
    1    More than 4 alleles
    2    1 homozygous genotype and more than 3 alleles
    3    2 homozygous genotypes and more than 2 alleles
    4    More than 2 homozygous genotypes
    5    An allele occurs in more than 2 heterozygous genotypes
    6    At least one genotype does not contain a parental allele
    7    More than 2 alleles from missing parent
    8    At least one genotype incompatible with parental genotypes
```

Output 5.2.3 Output Data Set from PROC FAMILY

```
                    Chi          Chi
              Chi    Sq   Chi     Sq   df  df   df    df  Prob Prob Prob  Prob
Obs Locus  SqTDT   STDT SqSDT  RCTDT  TDT STDT SDT RCTDT  TDT STDT  SDT RCTDT

  1   M1      0      0     0      0    1    0   0     1    1    .    .     1
```

Example 5.3: Using Allelic Transmission Scores for Association Tests

Abecasis, Cookson, and Cardon (2000) show how the allelic transmission scores, which are included in the OUTQ= data set, can be used to form various family-based tests for both discrete and quantitative traits. For example, the statistic for the Rabinowitz TDT for quantitative traits (1997) can be calculated using the deviates w and weights based on the quantitative trait of interest. The following data set and SAS code demonstrate how this test statistic can be computed from these quantities.

```
data fam_q;
   input ped ind father mother qtrt a1-a10;
   datalines;
 1  1  0  0  30.79   1  1  1  1  2  2  2  2  1  2
 2  1  0  0  15.80   1  1  1  1  2  2  2  2  2  2
 2  2  0  0  23.98   1  1  1  1  2  2  1  2  2  2
 2  3  1  2  22.73   1  1  1  1  2  2  2  2  2  2
 3  1  0  0  18.60   1  2  1  2  2  2  1  2  1  2
 3  2  0  0  18.80   1  1  1  1  2  2  2  2  1  2
 3  3  1  2  25.63   1  2  1  1  2  2  1  2  1  2
 4  1  0  0  17.40   1  1  1  1  2  2  1  2  2  2
 4  2  0  0  28.35   1  2  1  2  2  2  1  2  1  2
 4  3  1  2  18.61   1  2  1  2  2  2  2  2  1  2
 5  1  0  0  19.83   1  1  1  1  2  2  2  2  2  2
 5  2  0  0  24.09   1  1  1  1  1  1  2  1  2  2
 5  3  1  2  22.40   1  1  1  1  1  2  2  2  2  2
 6  1  0  0  28.46   1  1  1  1  2  2  2  2  2  2
 6  2  0  0  27.72   1  2  1  2  1  2  2  2  1  2
 6  3  1  2  13.76   1  1  1  1  2  2  2  2  2  2
 7  1  0  0  16.08   1  2  1  2  2  2  1  2  1  2
 7  2  0  0  30.79   1  1  1  2  2  2  1  2  1  2
 7  3  1  2  16.23   1  2  2  2  2  2  1  2  1  2
 8  1  0  0  25.03   1  2  1  2  2  2  1  2  1  2
 9  1  0  0  28.74   1  1  1  1  2  1  2  2  2  2
10  1  0  0  23.02   1  2  1  2  2  2  1  2  1  2
10  2  0  0  26.35   1  2  1  2  2  2  1  2  1  2
10  3  1  2  19.01   1  1  1  2  2  2  2  2  2  2
11  1  0  0  21.52   1  1  1  1  2  2  2  1  2  1
11  2  0  0  22.14   1  1  1  1  2  2  2  1  1  2
12  1  0  0  12.33   1  1  1  1  2  2  2  2  2  2
12  2  0  0   8.66   1  1  1  1  2  2  2  2  2  2
12  3  1  2  11.88   1  1  1  1  2  2  2  2  2  2
13  1  0  0  15.65   2  2  1  1  2  2  1  2  1  1
13  2  0  0  15.14   1  2  1  2  2  2  1  2  1  2
14  1  0  0  25.32   1  1  1  1  2  2  2  2  2  1
14  2  0  0  23.38   1  2  1  2  2  2  1  2  2  2
15  1  0  0  24.31   1  1  1  1  2  2  2  2  2  2
15  2  0  0  29.97   2  1  1  1  2  2  2  2  2  2
15  3  1  2  22.76   1  1  1  1  2  2  2  2  2  2
;
```

This data set contains five biallelic markers and a quantitative trait along with the pedigree identifiers for trios consisting of genotyped parents and a single offspring. Note in the following code for PROC FAMILY that there is no TRAIT statement since there is no dichotomous trait, but that the OUTQ= option is used in the

PROC FAMILY statement to identify a data set containing the allelic transmission scores. The deviates that are used for creating the Rabinowitz test statistic are contained in the variables that begin with "W_ ". PROC MEANS is used to obtain the sample mean of the quantitative trait qtrt among the offspring. A test statistic for each of the five markers is then calculated using the formulas given in Rabinowitz (1997); a general form of these formulas that uses the deviates w is shown in Abecasis, Cookson, and Cardon (2000).

```
proc family data=fam_q outq=w(drop=a1-a2 b:);
   var a1-a10;
   id ped ind father mother;
run;

proc means data=w noprint;
   var qtrt;
   output out=stats(keep=qbar) mean=qbar;
   where ind > 2;
run;

data _null_;
   set stats;
   call symput('qbar',trim(left(qbar)));
run;

data rab_test;
   set w end=last;
   where ind > 2;
   array w{10} w:;
   array num{5};
   array var{5};
   array t{5};
   array pvalt{5};
   a = qtrt - %sysevalf(&qbar);
   do i=1 to 5;
    aw = w{2*i-1} * a;
    num{i} + aw;
    var{i} + (aw*aw);
    if last then do;
     t{i}=num{i}/sqrt(var{i});
     pvalt{i}=2*(1-probnorm(abs(t{i})));
     if i=5 then output;
    end;
   end;
   keep t1-t5 pvalt1-pvalt5;
run;


proc print data=rab_test noobs;
   title 'Test Statistics and P-Values for 5 Markers';
run;
```

The data set containing the test statistic and corresponding p-value for each marker is displayed in Output 5.3.1. From this output, you can conclude that there are no markers significantly linked and associated with the QTL for this quantitative trait.

Output 5.3.1 Rabinowitz Test Statistics

```
                 Test Statistics and P-Values for 5 Markers

    t1       t2    t3     t4      t5    pvalt1  pvalt2  pvalt3  pvalt4  pvalt5

-0.53461 0.72887  1 0.17060 0.95693 0.59292 0.46608 0.31731 0.86454 0.33860
```

References

Abecasis, G. R., Cardon, L. R., and Cookson, W. O. C. (2000), “A General Test of Association for Quantitative Traits in Nuclear Families,” *American Journal of Human Genetics*, 66, 279–292.

Abecasis, G. R., Cookson, W. O. C., and Cardon, L. R. (2000), “Pedigree Tests of Transmission Disequilibrium,” *European Journal of Human Genetics*, 8, 545–551.

Allison, D. B. (1997), “Transmission-Disequilibrium Tests for Quantitative Traits,” *American Journal of Human Genetics*, 60, 676–690.

Curtis, D., Miller, M. B., and Sham, P. C. (1999), “Combining the Sibling Disequilibrium Test and Transmission/Disequilibrium Test for Multiallelic Markers,” *American Journal of Human Genetics*, 64, 1785–1786.

Curtis, D. and Sham, P. C. (1995), “A Note on the Application of the Transmission Disequilibrium Test When a Parent Is Missing,” *American Journal of Human Genetics*, 56, 811–812.

Czika, W. and Berry, J. J. (2002), “Using All Alleles in the Multiallelic Versions of the SDT and Combined SDT/TDT,” *American Journal of Human Genetics*, 71, 1235–1236.

Fulker, D. W., Cherny, S. S., Sham, P. C., and Hewitt, J. K. (1999), “Combined Linkage and Association Sib-Pair Analysis for Quantitative Traits,” *American Journal of Human Genetics*, 64, 259–267.

Hodge, S. E. (1995), “An Oligogenic Disease Displaying Weak Marker Associations: A Summary of Contributions to Problem 1 of GAW9,” *Genetic Epidemiology*, 12, 545–554.

Horvath, S. and Laird, N. M. (1998), “A Discordant-Sibship Test for Disequilibrium and Linkage: No Need for Parental Data,” *American Journal of Human Genetics*, 63, 1886–1897.

Horvath, S., Laird, N. M., and Knapp, M. (2000), “The Transmission/Disequilibrium Test and Parental-Genotype Reconstruction for X-Chromosomal Markers,” *American Journal of Human Genetics*, 66, 1161–1167.

Kaplan, N. L., Martin, E. R., and Weir, B. S. (1997), “Power Studies for the Transmission/Disequilibrium Tests with Multiple Alleles,” *American Journal of Human Genetics*, 60, 691–702.

Knapp, M. (1999a), “The Transmission/Disequilibrium Test and Parental-Genotype Reconstruction: The Reconstruction-Combined Transmission/Disequilibrium Test,” *American Journal of Human Genetics*, 64, 861–870.

Knapp, M. (1999b), “Using Exact *P* Values to Compare the Power between the Reconstruction-Combined Transmission/Disequilibrium Test and the Sib Transmission/Disequilibrium Test,” *American Journal of Human Genetics*, 65, 1208–1210.

Monks, S. A. and Kaplan, N. L. (2000), “Removing the Sampling Restrictions from Family-Based Tests of Association for a Quantitative-Trait Locus,” *American Journal of Human Genetics*, 66, 576–592.

Monks, S. A., Kaplan, N. L., and Weir, B. S. (1998), “A Comparative Study of Sibship Tests of Linkage and/or Association,” *American Journal of Human Genetics*, 63, 1507–1516.

Rabinowitz, D. (1997), “A Transmission Disequilibrium Test for Quantitative Trait Loci,” *Human Heredity*, 47, 342–350.

Spielman, R. S. and Ewens, W. J. (1996), “The TDT and Other Family-Based Tests for Linkage Disequilibrium and Association,” *American Journal of Human Genetics*, 59, 983–989.

Spielman, R. S. and Ewens, W. J. (1998), “A Sibship Test for Linkage in the Presence of Association: The Sib Transmission/Disequilibrium Test,” *American Journal of Human Genetics*, 62, 450–458.

Spielman, R. S., McGinnis, R. E., and Ewens, W. J. (1993), “Transmission Test for Linkage Disequilibrium: The Insulin Gene Region and Insulin-Dependent Diabetes Mellitus (IDDM),” *American Journal of Human Genetics*, 52, 506–516.

Chapter 6

The GENESELECT Procedure (Experimental)

Contents

Overview: GENESELECT Procedure

The GENESELECT procedure identifies influential genetic and environmental variables and their interactions by fitting a model to predict a trait and then evaluating the influence that the predictor variables and their interactions have on the model.

The GENESELECT procedure does the following:

- allows qualitative and quantitative variables
- allows a large number of predictor variables
- minimizes bias from missing values
- outputs an influence measure for single variables and interactions
- outputs an influence measure of a variable for each observation
- outputs predictions, including probabilities for qualitative traits
- outputs a matrix of similarities between predictor variables

The procedure can handle a large number of variables because only one variable enters the model at a time. The procedure usually stops before using all the available variables.

If two variables are highly correlated, the model might use only one of the variables. The other variable would therefore not appear influential, although it might actually be the biologically relevant variable. The GENESELECT procedure estimates how similar the model would be if a variable in the model were replaced by another, and then outputs the set of estimates as a similarity matrix. The matrix includes variables not in the model. Thus, when the model identifies one variable, the similarity matrix identifies the set of similar variables, one of which might be biologically important. Note that similar variables need not be correlated.

If two variables are highly correlated and the model uses both of them, the credit for influence is shared between them. Neither of them gets the credit it would if one of them were replaced by the other. To correctly estimate the influence of a variable, avoid using similar variables in the model.

A predictive model is likely to find fewer influential variables than a set of association tests would. For example, the GENESELECT procedure is likely to find fewer influential variables than the CASECONTROL procedure. The reason is simple: once a predictive model is sufficiently fit, other variables are not needed, not even ones that are associated with the trait.

The GENESELECT procedure minimizes bias from missing values in two ways. When entering a variable into the fit of the model, the procedure uses all observations with nonmissing values on the variable, and ignores all observations missing that variable value. When applying the model for prediction, the procedure replaces the value by the distribution of nonmissing values among all observations used during the fit.

The GENESELECT procedure fits a model by *recursively partitioning* the data. A *partitioning* procedure is one that searches for an optimal partition of the data. The partitioning or *splitting* rule is defined in terms of the values of a single variable. Optimality depends on the distribution of the trait into the partition segments. The more similar the trait values are within the segments, the better the partition. Decision trees are the most common type of recursive partitioning model. The GENESELECT procedure fits a boosted series of decision trees by default.

A *recursive partitioning* procedure partitions the data into subsets and then partitions each of the subsets, and so on. In the terminology of the tree metaphor, the partition subsets are *nodes*, the original data set is the *root* node, and the final, unpartitioned subsets are *terminal nodes* or *leaves*. Nodes that are not terminal nodes are sometimes called *internal* nodes. The subsets of a single partition are commonly called *child* nodes, thereby mixing the metaphor with genealogy, which also provides the terms *descendant* and *ancestor* nodes. A *branch* of a node consists of a child node and its descendants.

Syntax: GENESELECT Procedure

The following statements are available in PROC GENESELECT.

PROC GENESELECT < *options* > ;
  FREQ *variable* ;
  PERFORMANCE < *options* > ;
  TRAIT *variable* < / *options* > ;
  VAR *variables* < / *options* > ;

  IMPORTANCE < *options* > ;
  PARTIALDEP < *options* > ;
  SAVE < *options* > ;
  SCORE < *options* > ;

The FREQ, PERFORMANCE, TRAIT, and VAR statements describe the data and performance options. They must appear before the IMPORTANCE, PARTIALDEP, SAVE, and SCORE statements.

Table 6.1 summarizes what each statement (other than the PROC statement) in the GENESELECT procedure enables you to do.

Table 6.1 Statements in the GENESELECT Procedure

Statement	Description
FREQ	specify a frequency variable
PERFORMANCE	specify memory size and where to locate data
TRAIT	specify the response variable
VAR	specify input variables with common options
IMPORTANCE	output estimates of variable influence
PARTIALDEP	output partial dependency for plots and interaction detection
SAVE	output data sets containing model results
SCORE	output model predictions

PROC GENESELECT Statement

PROC GENESELECT < *options* > ;

The PROC GENESELECT statement starts the GENESELECT procedure. Either the DATA= option or the INMODEL= option must appear, but not both. The DATA= option must appear in order to fit a model. The INMODEL= option specifies a previously saved model. Table 6.2 summarizes the options available in the PROC GENESELECT statement.

Table 6.2 PROC GENESELECT Statement Options

Option	Description	Default
DATA=	data set	

Option	Description	Default
INMODEL=	data set containing model information	
LEAFFRACTION=	LEAFSIZE as fraction of data	0.001
LEAFSIZE=	minimum number of observations in a branch	
MAXDEPTH=	maximum depth of a tree	6
MAXSURROGATES=	maximum number of surrogates rules in a node	10
MINCATSIZE=	observations needed for each category	5
MODELTYPE=	type of model to fit	TREEBOOST
SEED=	seed for pseudo-random number generator	8976153
SPLITSIZE=	minimum number of observations to split a node	10

DATA=*SAS-data-set*

specifies data. Either the DATA= or INMODEL= option must be specified, but not both.

INMODEL=*SAS-data-set*

names a data set created from the SAVE MODEL= option. When using the INMODEL= option, the VAR, TRAIT, and FREQ statements are prohibited, as are the DATA= option and other model parameters.

ITERATIONS=*n*

specifies the number of terms in a boosted series of trees. For quantitative and binary traits, the number of iterations equals the number of trees. For a qualitative, nonbinary trait, a separate tree is created for each trait category in each iteration, resulting in nJ trees, where J is the number of trait values and n is an integer from 1 to 1000. The default value of n is 50.

LEAFFRACTION=*p*

specifies the smallest number of observations a new branch can have, expressed as the proportion of the number N of available observations in the DATA= data set specified in the PROC statement. N can be less than the total number of observations in the data set because observations with a missing trait or nonpositive value of the variable specified in the FREQ statement are excluded from N. The LEAFSIZE= option specifies the same quantity as an absolute number. The procedure uses the larger of the two. P can be any number from zero through one. The default value equals 0.001.

LEAFSIZE=*n*

specifies the smallest number of observations a new branch can have. The LEAFFRACTION= option specifies the same quantity as a proportion of the original data. The procedure uses the larger of the two. The default value equals the number N of available observations in the DATA= data set specified in the PROC statement, divided by 1,000, or 5, if 5 is larger, or 5,000, if 5,000 is smaller. N can be less than the total number of observations in the data set because observations with a missing trait or nonpositive value of the variable specified in the FREQ statement are excluded from N. The LEAFSIZE= option does not use the values of the variable in the FREQ statement to adjust the count of observations in the leaf.

MAXDEPTH=*n* | MAX

specifies the maximum depth of a node that the PROC statement creates automatically. The *depth* of a node equals the number of splitting rules needed to define the node. The root node has a depth of zero. The children of the root have a depth of one, and so on.

The MAXDEPTH=MAX option specifies $n = 50$, the largest possible value of n. The smallest acceptable value of n is 0. Specify MAXDEPTH=0 to avoid creating a model. The default value of n depends on the MODELTYPE= option. The default value of n is six for MODELTYPE=TREE, and two for MODELTYPE= TREEBOOST.

MAXSURROGATES | MAXSURRS=*n*

specifies the number of surrogate rules sought for each internal node. A surrogate rule is one that emulates the splitting rule. The measure of agreement between a surrogate and the splitting rule is used to compute the interchangeability of two variables. The GENESELECT procedure does not use surrogate rules to predict observations with missing values. The default value of n is 10.

MINCATSIZE=*n*

specifies the minimum number of observations that a given qualitative variable value must have in order to use the value in a split search. The GENESELECT procedure handles qualitative values that appear in fewer than n observations in the same way it handles missing values. The default value of n is 5.

MODELTYPE=*type*

specifies the type of model to create. Table 6.3 summarizes the types available.

Table 6.3 Types of Models

Type	Description
TREEBOOST	Boosted series of trees
TREE	Decision tree

SEED=*n*

specifies the seed for generating random numbers. The value specified for n must be a nonnegative integer. Set n to 0 to use the internal default.

SPLITSIZE=*n*

specifies the requisite number of observations a node must have in order for the procedure to consider splitting it. By default, n is twice the value of the LEAFSIZE= option. For the LEAFFRACTION=, LEAFSIZE=, MINCATSIZE=, and SPLITSIZE= options in the PROC statement, and for the NODESIZE= option in the PERFORMANCE statement, the procedure counts the number of observations in a node without adjusting the number with the values of the variable specified in the FREQ statement.

FREQ Statement

FREQ *variable* **;**

The FREQ statement names a variable that contains the frequency of occurrence of each observation. The GENESELECT procedure treats an observation as if it appears n times, where n is the value of the FREQ variable for the observation. The value of n can be fractional to indicate a partial observation. If the value of n is close to zero, negative, or missing, the observation is ignored. When the FREQ statement is not specified, each observation is assigned a frequency of one.

PERFORMANCE Statement

PERFORMANCE < *options* > **;**

The PERFORMANCE statement specifies options affecting the speed of computations with little or no impact on the results.

WORKDATALOCATION=*RAM | DISK | SOURCE*

specifies the location where a working copy of the data will be stored. The RAM location requests that the working copy be stored in memory if enough memory is available for it and still allows for minimal calculations in one pass of the data.

The DISK location requests that the working copy be stored in a disk utility file. Storing the copy on disk can free a considerable amount of memory for calculations, possibly speeding up the program.

The SOURCE location requests that the data be read multiple times instead of being copied to memory or a disk utility file. SOURCE is slower than DISK because the DISK copy is converted to encodings directly usable in the calculations. The SOURCE location is preferable only when the data do not fit in RAM or in a disk utility file.

MEMSIZE=*m* **< B | K | M | G >**

specifies the maximum amount of memory to allocate for the computations and the working copy of the data if the data are stored in memory. The optional suffix, *B, K, M*, or *G*, specifies bytes, kilobytes, megabytes, or gigabytes, respectively. Without a suffix, m specifies the number of bytes. M can be fractional.

The default value depends on the computer and can considerably prolong the execution time if SAS cannot distinguish physical memory from virtual memory.

The SAS MEMSIZE system option sets an upper limit to the number of bytes.

TRAIT Statement

TRAIT *variable* < */ option* > **;**

The TRAIT statement names the variable that the model tries to predict.

VARTYPE=QUALITATIVE | QUANTITATIVE

specifies whether the trait is qualitative or quantitative. The default is QUANTITATIVE for a numeric variable and QUALITATIVE for a character variable.

VAR Statement

VAR *variables* < */ options* > **;**

The VAR statement names variables with common options. The VAR statement can be repeated.

VARTYPE=QUALITATIVE | QUANTITATIVE

specifies whether the variables are qualitative or quantitative. The default is QUANTITATIVE for a numeric variable and QUALITATIVE for a character variable.

IMPORTANCE Statement

IMPORTANCE < */ options* > **;**

The IMPORTANCE statement implements an observation-based approach to evaluate the importance of a variable or of a pair of variables to the predictions of the model.

For each observation, the value of the variable or pair of variables being evaluated is rendered uninformative. The IMPORTANCE statement outputs the prediction once using the actual value and a second time using the uninformative value. The difference between the two predictions shows the dependence of the prediction on the variable or pair of variables being evaluated. The differences for all the observations can be plotted against the actual variable value or observation number to explore where the dependence is stronger or weaker.

The observation-based importance differs from the split-based importance computed in the IMPORTANCE= option of the SAVE statement. The latter importance is based on the contribution a variable makes in reducing the residual sum of squares.

The IMPORTANCE statement can be repeated.

DATA=*SAS-data-set*

names the input data set. If the DATA= option is absent, the procedure uses the training data.

N2WAY=*m n*

requests to evaluate the best m variables paired with the best n variables, where the term "best" here refers to the split-based variable importance rankings computed in the IMPORTANCE= option of the SAVE statement. If n is missing, then n is set to m. The default value of m and n is 0. When the procedure evaluates a pair of variables, it also evaluates the two variables individually and outputs results as if the variables were specified in the VAR= option.

NVARS=*n*

requests to evaluate the best n variables as ranked by the split-based variable importance computed in the IMPORTANCE= option of the SAVE statement. If the N2WAY=, NVARS=, and VAR= options are absent, then the procedure assumes NVARS=5.

OUT=*SAS-data-set*

names the output data set to contain the scored data. If the OUT= option is absent, the procedure creates a data set name using the DATA*n* convention.

The OUT= data set in the IMPORTANCE has the same variables as the OUT= data set in the SCORE data set, plus one or two more, _INPUT1_ and _INPUT2_, that contain the name of a variable whose values were treated as uninformative when making the predictions. If _INPUT1_ is blank, then _INPUT2_ is blank and the predictions are the same as in the OUT= data set of the SCORE statement.

The OUT= data set becomes very large if many variables are being evaluated. The number of observations in the OUT= data set equals the number of variables and pairs of variables being evaluated plus one times the number of observations in the data set. Specify OUT=_NULL_ to avoid creating a scored data set.

OUTFIT=*SAS-data-set*

names the output data set to contain the fit statistics. The number of observations in the OUTFIT= data set equals the number of variables and pairs of variables being evaluated plus one.

The OUTFIT= data set in the IMPORTANCE has the same variables as the OUTFIT= data set in the SCORE data set, plus one or two more, _INPUT1_ and _INPUT2_, that contain the name of a variable whose values were treated as uninformative when computating the statistics. If _INPUT1_ is blank, then _INPUT2_ is blank and the statistics are the same as in the OUTFIT= data set of the SCORE statement.

VAR=*(varlist)*

specifies variables and pairs of variables to evaluate. *Varlist* is a list of variable names optionally containing asterisks to indicate a pair of variables. Variables on the left or right of an asterisk may be grouped within square brackets. Brackets may not be nested. Parentheses must enclose the list, *varlist*.

When a procedure evaluates a pair of variables, it also evaluates the two variables individually and outputs the results. For example, the following *varlist* would specify variables A, B, C, D, E, and pairs of variables, B-C, D-E, D-C, and E-C:
A B*C [D E] * [E C]

PARTIALDEP Statement

PARTIALDEP *< DATA= > < OUT= > VAR=(varlist)* **;**

The PARTIALDEP statement outputs a partial dependency function for plots and interaction detection.

DATA=*SAS-data-set*

names the input data set. If the DATA= option is absent, the procedure uses the data. Only the variables listed in the VAR= option are used.

OUT=*SAS-data-set*

names the output data set to contain the partial dependency function. If the OUT= option is absent, the GENESELECT procedure creates a data set name by using the DATA*n* convention. The output variables are the same as output with the OUT= option to the SCORE statement.

VAR=*(varlist)*

specifies the variable or pair of variables for which to compute the partial dependency. Parentheses must enclose the list, *varlist*. The PARTIALDEP statement requires at least one variable and at most two variables. The procedure ignores the value of all other variables.

SAVE Statement

SAVE < *options* > **;**

The SAVE statement outputs model information into SAS data sets.

DISSIMILARITY=*SAS-data-set*

names the output data set to contain a dissimilarity statistic for pairs of input variables. The data set has the type DISTANCE and is suitable for input to the DATA= option of the CLUSTER procedure. The data set includes an ID variable, _VAR_. The dissimilarity matrix equals one minus the similarity matrix output in the SIMILARITY= option. Similarity relies on surrogate rules. Use the MAXSURROGATES= option in the PROC statement to create surrogate rules when the model is fit.

IMPORTANCE=*SAS-data-set*

names the output data set to contain the split-based variable importance.

MODEL=*SAS-data-set*

names the output data set to encode the information necessary for use with the INMODEL= option in a subsequent invocation of the GENESELECT procedure.

SIMILARITY=*SAS-data-set*

names the output data set to contain a similarity statistic for pairs of input variables. The data set contains a variable for every input variable used in a primary splitting rule, and an additional identification variable, _VAR_, whose value is the name of an input variable. Similarity relies on surrogate rules. Use the MAXSURROGATES= option in the PROC statement to create surrogate rules when the model is fit. The similarity matrix equals one minus the dissimilarity matrix that is created by using the DISSIMILARITY= option. The DISSIMILARITY= option creates a DISTANCE matrix suitable for input to the CLUSTER procedure.

SCORE Statement

SCORE < *options* > **;**

The SCORE statement reads a data set containing the input variables used in the model and then outputs a data set containing the original variables plus new variables to contain predictions, residuals, decisions, and leaf assignments. The SCORE statement can be repeated.

DATA=*SAS-data-set*

names the input data set. If the DATA= option is absent, the procedure uses the data.

PREDICTION | NOPREDICTION

indicates whether prediction variables, such as P_*, should be generated. The default is PREDICTION, requesting prediction variables.

OUT=*SAS-data-set*

names the output data set to contain the scored data. If the OUT= option is absent, the procedure creates a data set name by using the DATA*n* convention. Specify OUT=_NULL_ to avoid creating a scored data set.

OUTFIT=*SAS-data-set*

names the output data set to contain the fit statistics.

ROLE=TRAIN | VALID | TEST | SCORE

specifies the role of the input data set and determines the fit statistics to compute. For ROLE=TRAIN, VALID, or TEST, observations without a trait value are ignored.

Chapter 7
The HAPLOTYPE Procedure

Contents

Overview: HAPLOTYPE Procedure

A *haplotype* is a combination of alleles at multiple loci on a single chromosome. A pair of haplotypes constitutes the multilocus genotype. Haplotype information has to be inferred because data are usually collected at the genotypic, not haplotype pair, level. For homozygous markers, there is no problem. If one locus has alleles A and a, and a second locus has alleles B and b, the observed genotype $AABB$ must contain two haplotypes of type AB; genotype $AaBB$ must contain haplotypes AB and aB, and so on. Haplotypes and their frequencies can be obtained directly. When both loci are heterozygous, however, there is ambiguity; a variety of combinations of haplotypes can generate the genotype, and it is not possible to determine directly which two haplotypes constitute any individual genotype. For example, the genotype $AaBb$ might be of type

AB/ab with haplotypes AB and ab, or of type Ab/aB with haplotypes Ab and aB. The HAPLOTYPE procedure uses the expectation-maximization (EM) algorithm to generate maximum likelihood estimates of haplotype frequencies given a multilocus sample of genetic marker genotypes under the assumption of Hardy-Weinberg equilibrium (HWE). These estimates can then be used to assign the probability that each individual possesses a particular haplotype pair. A Bayesian approach for haplotype frequency estimation is also implemented in PROC HAPLOTYPE.

Estimation of haplotype frequencies is important for several applications in genetic data analysis. One application is determining whether there is linkage disequilibrium (LD), or association, between loci. PROC HAPLOTYPE performs a likelihood ratio test to test the hypothesis of no LD between marker loci. Another application is association testing of disease susceptibility. Since sites that affect disease status are embedded in haplotypes, it has been postulated that the power of case-control studies might be increased by testing for haplotype rather than allele or genotype associations. One reason is that haplotypes might include two or more causative sites whose combined effect is measurable, particularly if they show synergistic interaction. Another reason is that fewer tests need to be performed, although if there are a large number of haplotypes, this advantage is offset by the increased degrees of freedom of each test. PROC HAPLOTYPE can use case-control data to calculate test statistics for the hypothesis of no association between alleles composing the haplotypes and disease status; such tests are carried out across all haplotypes at the loci specified, or for individual haplotypes.

Getting Started: HAPLOTYPE Procedure

Example

Assume you have a random sample with 25 individuals genotyped at four markers. You want to infer the gametic phases of the genotypes and estimate their frequencies. There are eight columns of data, with the first two columns containing the pair of alleles at the first marker, the next two columns containing the pair of alleles for the second marker, and so on. Each row represents an individual. The data can be read into a SAS data set as follows:

```
data markers;
   input (m1-m8) ($);
   datalines;
B  B  A  B  B  B  A  A
A  A  B  B  A  B  A  B
B  B  A  A  B  B  B  B
A  B  A  B  A  B  A  B
A  A  A  B  A  B  B  B
B  B  A  A  A  B  A  B
A  B  B  B  A  B  A  A
A  B  A  A  A  A  A  A
B  B  A  A  A  A  A  B
A  B  A  B  A  B  B  B
A  B  A  B  A  B  A  A
B  B  A  B  A  B  A  A
A  B  A  A  A  B  A  B
A  B  B  B  B  B  A  B
```

```
   A  A  A  B  A  A  A  B
   B  B  A  B  A  B  A  B
   A  B  B  B  A  A  A  B
   B  B  B  B  A  A  A  A
   A  B  A  A  A  B  A  A
   A  B  A  A  A  B  A  B
   B  B  A  A  A  A  A  B
   A  A  A  B  A  A  A  B
   A  B  A  A  A  A  B  B
   A  A  A  A  A  A  A  A
   A  B  B  B  A  A  A  A
;
```

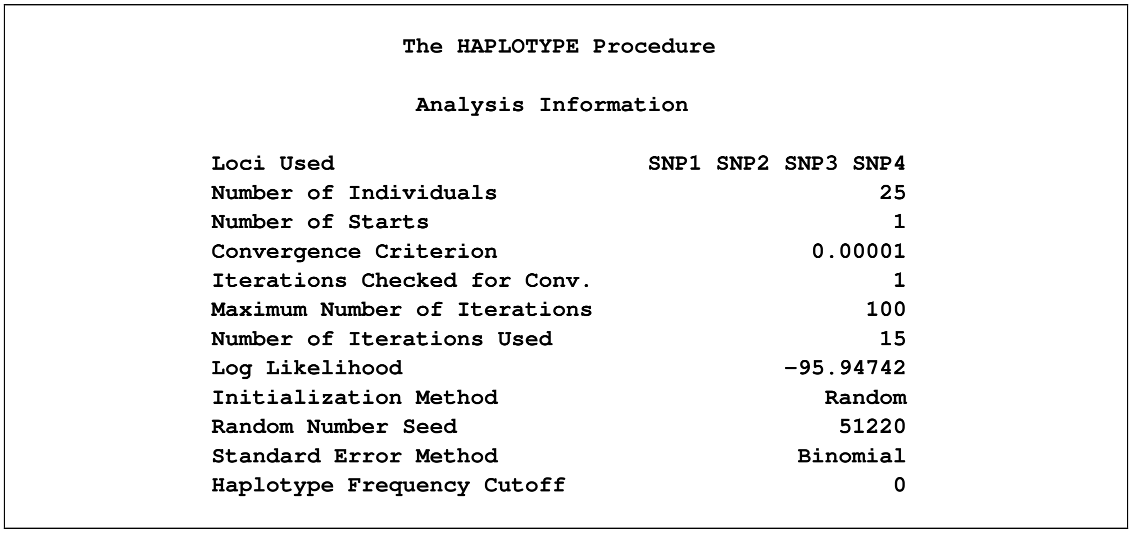
You can now use PROC HAPLOTYPE to infer the possible haplotypes and estimate the four-locus haplotype frequencies in this sample. The following statements perform these calculations:

```
proc haplotype data=markers out=hapout init=random prefix=SNP seed=51220;
   var m1-m8;
run;

proc print data=hapout noobs round;
run;
```

This analysis uses the EM algorithm to estimate the haplotype frequencies from the sample. The standard errors and a confidence interval are estimated, by default, under a binomial assumption for each haplotype frequency estimate. A more precise estimate of the standard error can be obtained through the jackknife process by specifying the option SE=JACKKNIFE in the PROC HAPLOTYPE statement, but this takes considerably more computations (see the "Methods of Estimating Standard Error" on page 131 section for more information). The option INIT=RANDOM indicates that initial haplotype frequencies are randomly generated, using a random seed created by the system clock since the SEED= option is omitted. The default confidence level 0.95 is used, since the ALPHA= option of the PROC HAPLOTYPE statement was omitted. Also by default, the convergence criterion of 0.00001 must be satisfied for one iteration, and the maximum number of iterations is set to 100. The PREFIX= option requests that the four markers, indicated by the eight allele variables in the VAR statement, be named SNP1–SNP4.

The results from the procedure are shown in Figures 7.1 through 7.3.

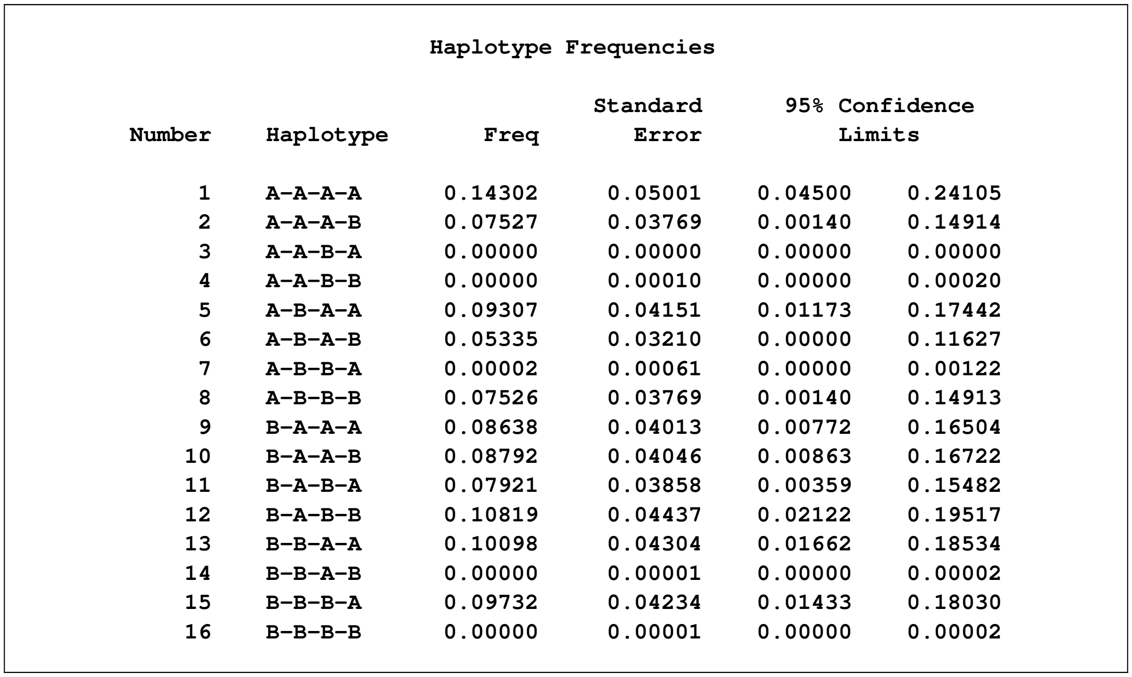
Figure 7.1 Analysis Information for the HAPLOTYPE Procedure



```
                     The HAPLOTYPE Procedure

                      Analysis Information

         Loci Used                          SNP1 SNP2 SNP3 SNP4
         Number of Individuals                               25
         Number of Starts                                     1
         Convergence Criterion                          0.00001
         Iterations Checked for Conv.                         1
         Maximum Number of Iterations                       100
         Number of Iterations Used                           15
         Log Likelihood                               -95.94742
         Initialization Method                           Random
         Random Number Seed                               51220
         Standard Error Method                         Binomial
         Haplotype Frequency Cutoff                           0
```

Figure 7.1 displays a table with information about several of the settings used to perform the HAPLOTYPE procedure as well as information about the EM algorithm. Note that you can obtain from this table the random seed that was generated by the system clock if you need to replicate this analysis.

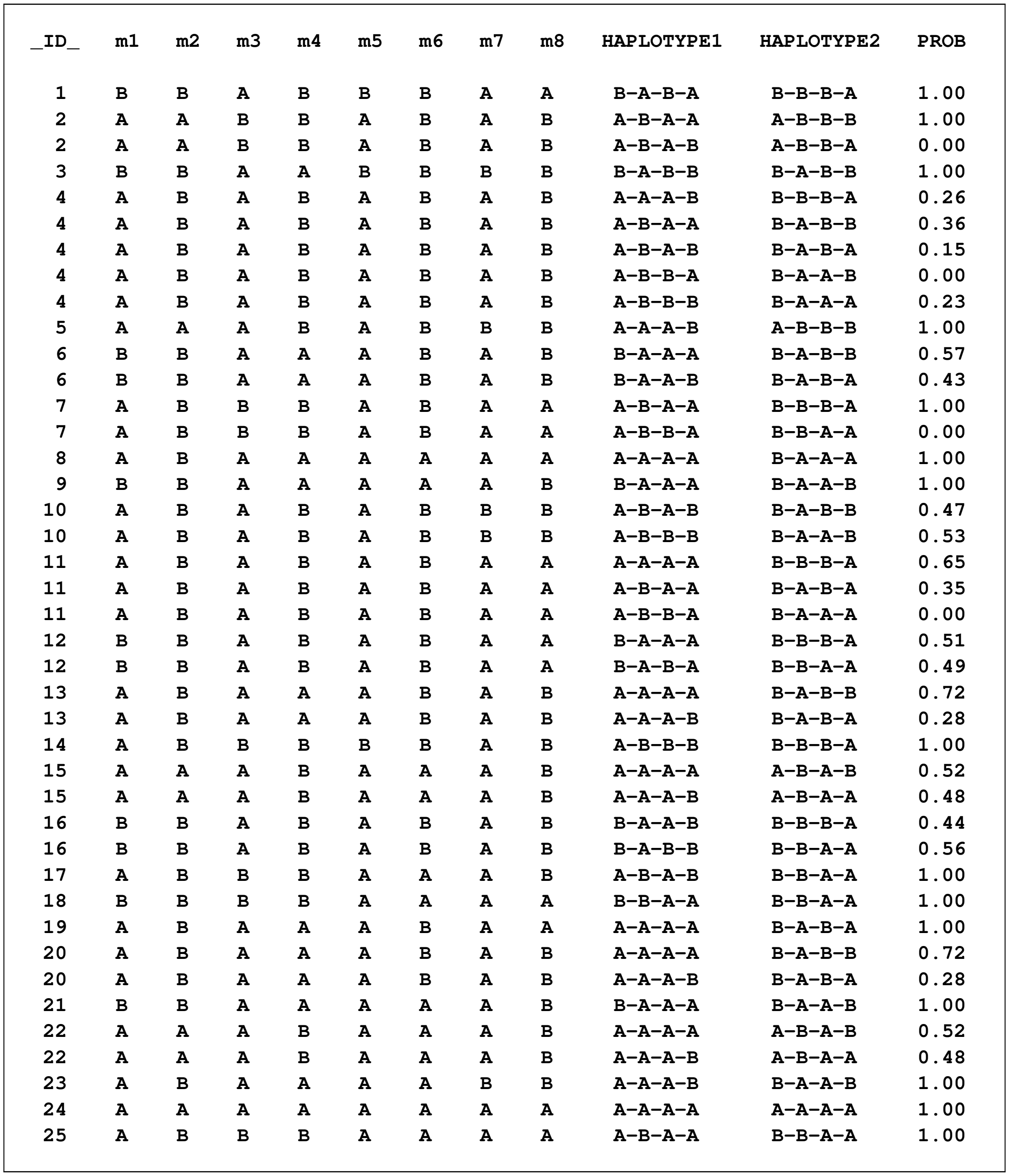
Figure 7.2 Haplotype Frequencies from the HAPLOTYPE Procedure



```
                      Haplotype Frequencies

                                      Standard     95% Confidence
   Number    Haplotype        Freq       Error         Limits

        1    A-A-A-A       0.14302     0.05001    0.04500    0.24105
        2    A-A-A-B       0.07527     0.03769    0.00140    0.14914
        3    A-A-B-A       0.00000     0.00000    0.00000    0.00000
        4    A-A-B-B       0.00000     0.00010    0.00000    0.00020
        5    A-B-A-A       0.09307     0.04151    0.01173    0.17442
        6    A-B-A-B       0.05335     0.03210    0.00000    0.11627
        7    A-B-B-A       0.00002     0.00061    0.00000    0.00122
        8    A-B-B-B       0.07526     0.03769    0.00140    0.14913
        9    B-A-A-A       0.08638     0.04013    0.00772    0.16504
       10    B-A-A-B       0.08792     0.04046    0.00863    0.16722
       11    B-A-B-A       0.07921     0.03858    0.00359    0.15482
       12    B-A-B-B       0.10819     0.04437    0.02122    0.19517
       13    B-B-A-A       0.10098     0.04304    0.01662    0.18534
       14    B-B-A-B       0.00000     0.00001    0.00000    0.00002
       15    B-B-B-A       0.09732     0.04234    0.01433    0.18030
       16    B-B-B-B       0.00000     0.00001    0.00000    0.00002
```

Figure 7.2 displays the possible haplotypes in the sample and their estimated frequencies with standard errors and the lower and upper limits of the 95% confidence interval.

Figure 7.3 Output Data Set from the HAPLOTYPE Procedure



```
_ID_   m1    m2    m3    m4    m5    m6    m7    m8    HAPLOTYPE1    HAPLOTYPE2    PROB

  1    B     B     A     B     B     B     A     A      B-A-B-A       B-B-B-A      1.00
  2    A     A     B     B     A     B     A     B      A-B-A-A       A-B-B-B      1.00
  2    A     A     B     B     A     B     A     B      A-B-A-B       A-B-B-A      0.00
  3    B     B     A     A     B     B     B     B      B-A-B-B       B-A-B-B      1.00
  4    A     B     A     B     A     B     A     B      A-A-A-B       B-B-B-A      0.26
  4    A     B     A     B     A     B     A     B      A-B-A-A       B-A-B-B      0.36
  4    A     B     A     B     A     B     A     B      A-B-A-B       B-A-B-A      0.15
  4    A     B     A     B     A     B     A     B      A-B-B-A       B-A-A-B      0.00
  4    A     B     A     B     A     B     A     B      A-B-B-B       B-A-A-A      0.23
  5    A     A     A     B     A     B     B     B      A-A-A-B       A-B-B-B      1.00
  6    B     B     A     A     A     B     A     B      B-A-A-A       B-A-B-B      0.57
  6    B     B     A     A     A     B     A     B      B-A-A-B       B-A-B-A      0.43
  7    A     B     B     B     A     B     A     A      A-B-A-A       B-B-B-A      1.00
  7    A     B     B     B     A     B     A     A      A-B-B-A       B-B-A-A      0.00
  8    A     B     A     A     A     A     A     A      A-A-A-A       B-A-A-A      1.00
  9    B     B     A     A     A     A     A     B      B-A-A-A       B-A-A-B      1.00
 10    A     B     A     B     A     B     B     B      A-B-A-B       B-A-B-B      0.47
 10    A     B     A     B     A     B     B     B      A-B-B-B       B-A-A-B      0.53
 11    A     B     A     B     A     B     A     A      A-A-A-A       B-B-B-A      0.65
 11    A     B     A     B     A     B     A     A      A-B-A-A       B-A-B-A      0.35
 11    A     B     A     B     A     B     A     A      A-B-B-A       B-A-A-A      0.00
 12    B     B     A     B     A     B     A     A      B-A-A-A       B-B-B-A      0.51
 12    B     B     A     B     A     B     A     A      B-A-B-A       B-B-A-A      0.49
 13    A     B     A     A     A     B     A     B      A-A-A-A       B-A-B-B      0.72
 13    A     B     A     A     A     B     A     B      A-A-A-B       B-A-B-A      0.28
 14    A     B     B     B     B     B     A     B      A-B-B-B       B-B-B-A      1.00
 15    A     A     A     B     A     A     A     B      A-A-A-A       A-B-A-B      0.52
 15    A     A     A     B     A     A     A     B      A-A-A-B       A-B-A-A      0.48
 16    B     B     A     B     A     B     A     B      B-A-A-B       B-B-B-A      0.44
 16    B     B     A     B     A     B     A     B      B-A-B-B       B-B-A-A      0.56
 17    A     B     B     B     A     A     A     B      A-B-A-B       B-B-A-A      1.00
 18    B     B     B     B     A     A     A     A      B-B-A-A       B-B-A-A      1.00
 19    A     B     A     A     A     B     A     A      A-A-A-A       B-A-B-A      1.00
 20    A     B     A     A     A     B     A     B      A-A-A-A       B-A-B-B      0.72
 20    A     B     A     A     A     B     A     B      A-A-A-B       B-A-B-A      0.28
 21    B     B     A     A     A     A     A     B      B-A-A-A       B-A-A-B      1.00
 22    A     A     A     B     A     A     A     B      A-A-A-A       A-B-A-B      0.52
 22    A     A     A     B     A     A     A     B      A-A-A-B       A-B-A-A      0.48
 23    A     B     A     A     A     A     B     B      A-A-A-B       B-A-A-B      1.00
 24    A     A     A     A     A     A     A     A      A-A-A-A       A-A-A-A      1.00
 25    A     B     B     B     A     A     A     A      A-B-A-A       B-B-A-A      1.00
```

Figure 7.3 displays each individual's genotype with each of the possible haplotype pairs that the genotype can comprise, and the probability that the genotype can be resolved into each of the possible haplotype pairs.

Syntax: HAPLOTYPE Procedure

The following statements are available in PROC HAPLOTYPE.

PROC HAPLOTYPE < *options* > ;
  BY *variables* ;
  ID *variables* ;
  TRAIT *variable* < / *options* > ;
  VAR *variables* ;

Items within angle brackets (< >) are optional, and statements following the PROC HAPLOTYPE statement can appear in any order. Only the VAR statement is required.

PROC HAPLOTYPE Statement

PROC HAPLOTYPE < *options* > ;

You can specify the following options in the PROC HAPLOTYPE statement.

ALPHA=*number*

specifies that a confidence level of $100(1-\textit{number}\ )\%$ is to be used in forming the confidence intervals for estimates of haplotype frequencies. The value of *number* must be between 0 and 1, inclusive, and 0.05 is used as the default value if it is not specified.

BURNIN=*number* (Experimental)

indicates that *number* iterations are discarded as burn-in when EST=BAYESIAN is specified. The value of *number* cannot be greater than the value specified in the TOTALRUN= option and must be greater than 0. The default is $\min(5000, t/2)$, where t is the number of total runs.

CONV=*number*

specifies the convergence criterion for iterations of the EM algorithm, where $0 < \textit{number} \leq 1$. The iteration process is stopped when the ratio of the change in the log likelihoods to the former log likelihood is less than or equal to *number* for the number of consecutive iterations specified in the NLAG= option (or 1 by default), or after the number of iterations specified in the MAXITER= option has been performed. The default value is 0.00001.

CUTOFF=*number*

specifies a lower bound on a haplotype's estimated frequency in order for that haplotype to be included in the “Haplotype Frequencies” table. The value of *number* must be between 0 and 1, inclusive. By default, all possible haplotypes from the sample are included in the table.

DATA=*SAS-data-set*

names the input SAS data set to be used by PROC HAPLOTYPE. The default is to use the most recently created data set.

DELIMITER=*'string'*

indicates the string that is used to separate the two alleles that compose the genotypes contained in the variables specified in the VAR statement. This option is ignored if GENOCOL is not specified.

EST=BAYESIAN

EST=EM

EST=STEPEM

indicates the method to be used for estimating haplotype frequencies. By default or when EST=EM is specified, the EM algorithm is used. When EST=STEPEM, the stepwise EM algorithm is used to calculate estimates of haplotype frequencies. When EST=BAYESIAN, a Bayesian method is used for estimating haplotype frequencies.

GENOCOL

indicates that columns specified in the VAR statement contain genotypes instead of alleles. When this option is specified, there is one column per marker. The genotypes must consist of the two alleles separated by a delimiter. For a genotype with one missing allele, use a blank space to indicate a missing value; if both alleles are missing, either use a single missing value for the entire genotype or use the delimiter alone.

INDIVIDUAL=*variable*

INDIV=*variable*

specifies the individual ID variable when using the TALL option. This variable can be character or numeric.

INIT=LINKEQ

INIT=RANDOM

INIT=UNIFORM

indicates the method of initializing haplotype frequencies to be used in the EM algorithm. INIT=LINKEQ initializes haplotype frequencies assuming linkage equilibrium by calculating the product of the frequencies of the alleles that compose the haplotype. INIT=RANDOM initializes haplotype frequencies with random values from a Uniform(0,1) distribution, and INIT=UNIFORM assigns equal frequency to all haplotypes. By default, INIT=LINKEQ.

INTERVAL=*number* (Experimental)

indicates that the non-burn-in iterations of the Bayesian estimation method when EST=BAYESIAN is specified are thinned by only recording the result from every *number* iterations. The value of *number* must be greater than 0, and the default is 1 (every iteration is used).

ITPRINT

requests that the “Iteration History” table be displayed. This option is ignored if the NOPRINT option is specified.

LD

requests that haplotype frequencies be calculated under the assumption of no LD, in addition to being calculated using the EM algorithm. When this option is specified, the “Test for Allelic Associations” table is displayed, which contains statistics for the likelihood ratio test for allelic associations. This option is ignored if the NOPRINT option is specified.

MARKER=*variable*

specifies the marker ID variable when using the TALL option. This variable contains the names of the markers that are used in all output and can be character or numeric.

MAXITER=*number*

specifies the maximum number of iterations to be used in the EM algorithm. The number must be a nonnegative integer. Iterations are carried out until convergence is reached according to the convergence criterion or until *number* iterations have been performed. The default is MAXITER=100.

NDATA=*SAS-data-set*

names the input SAS data set containing names, or identifiers, for the markers used in the output. There must be a NAME variable in this data set, which should contain the same number of rows as there are markers in the input data set specified in the DATA= option. When there are fewer rows than there are markers, markers without a name are named using the PREFIX= option. Likewise, if there is no NDATA= data set specified, the PREFIX= option is used. Note that this data set is ignored if the TALL option is specified in the PROC HAPLOTYPE statement. In that case, the marker variable names are taken from the marker ID variable specified in the MARKER= option.

NLAG=*number*

specifies the number of consecutive iterations that must meet the convergence criterion specified in the CONV= option (0.00001 by default) for the iteration process of the EM algorithm to stop. The number must be a positive integer. If this option is omitted, one iteration must satisfy the convergence criterion by default.

NOPRINT

suppresses the display of the “Analysis Information,” “Iteration History,” “Haplotype Frequencies,” and “Test for Allelic Associations” tables. Either the OUT= option, the TRAIT statement, or both must be used with the NOPRINT option.

NSTART=*number*

specifies the number of different starts used for the EM algorithm. When this option is specified, PROC HAPLOTYPE starts the iterations with different random initial values *number* -2 times as well as once with uniform frequencies for all the haplotypes and once using haplotype frequencies assuming linkage equilibrium (independence). Results on the analysis, using the initial values that produce the best log likelihood, are then reported. The number must be a positive integer. If this option is omitted or NSTART=1, only one start with initial frequencies generated according to the INIT= option is used.

OUT=*SAS-data-set*

names the output SAS data set containing the probabilities of each genotype being resolved into all of the possible haplotype pairs.

OUTCUT=*number*

specifies a lower bound on a haplotype pair’s estimated probability given the individual’s genotype in order for that haplotype pair to be included in the OUT= data set. The value of *number* must be between 0 and 1, inclusive. By default, *number* $= 0.00001$. In order to be able to view all possible haplotype pairs for an individual’s genotype, OUTCUT=0 can be specified.

OUTID

indicates that the variable _ID_ created by PROC HAPLOTYPE should be included in the OUT= data set in addition to the variable(s) listed in the ID statement. When the ID statement is omitted, this variable is automatically included. This option is ignored when the TALL option is used.

PREFIX=*prefix*

specifies a prefix to use in constructing names for marker variables in all output. For example, if PREFIX=VAR, the names of the variables are VAR1, VAR2, ..., VAR*n*. Note that this option is ignored when the NDATA= option is specified, unless there are fewer names in the NDATA data set than there are markers; it is also ignored if the TALL option is specified, in which case the marker variable names are taken from the marker ID variable specified in the MARKER= option. Otherwise, if this option is omitted, PREFIX=M is the default when variables contain alleles; if GENOCOL is specified, then the names of the variables specified in the VAR statement are used as the marker names.

SE=BINOMIAL

SE=JACKKNIFE

specifies the standard error estimation method when the EM or stepwise EM algorithm is used for estimating haplotype frequencies. There are two methods available: the BINOMIAL option, which gives a standard error estimator from a binomial distribution and is the default method, and the JACKKNIFE option, which requests that the jackknife procedure be used to estimate the standard error.

SEED=*number*

specifies the initial seed for the random number generator used for creating the initial haplotype frequencies when INIT=RANDOM and/or to permute the data when the PERMUTATION= option of the TRAIT statement is specified. The value for *number* must be an integer; the computer clock time is used if the option is omitted or an integer less than or equal to 0 is specified. For more details about seed values, see *SAS Language Reference: Concepts*.

STEPTRIM=*number*

indicates the cutoff to be used for the stepwise EM algorithm when trimming the haplotype table, where $0 < number < 1$. This option is implemented only when EST=STEPEM is specified. By default, this number is set to $\min(0.001, 1/(2n))$, where n is the number of individuals in the data set.

TALL

indicates that the input data set is of an alternative format. This format contains the following columns: two containing marker alleles (or one containing marker genotypes if GENOCOL is specified), one for the marker identifier, and one for the individual identifier. The MARKER= and INDIV= options must also be specified in order for this option to be in effect. Note that when this option is used, the DATA= data set must first be sorted by any BY variables, then sorted by the marker ID variable, and then sorted by the individual ID variable.

THETA=*number* (Experimental)

requests that *number* be used as the scaled mutation rate θ when EST=BAYESIAN instead of the default, which is $\theta = 1/\sum_{i=1}^{2n-1} 1/i$ for a sample of n individuals. This value must be positive.

TOTALRUN=*number* (Experimental)

TOT=*number*

indicates the total number of iterations to use when EST=BAYESIAN, including the burn-in. The value of *number* must be greater than 0, and the default is 10,000.

BY Statement

BY *variables* **;**

You can specify a BY statement with PROC HAPLOTYPE to obtain separate analyses on observations in groups defined by the BY variables. When a BY statement appears, the procedure expects the input data set to be sorted in the order of the BY variables. The *variables* are one or more variables in the input data set.

If your input data set is not sorted in ascending order, use one of the following alternatives:

- Sort the data by using the SORT procedure with a similar BY statement.
- Specify the BY statement option NOTSORTED or DESCENDING in the BY statement for the HAPLOTYPE procedure. The NOTSORTED option does not mean that the data are unsorted but rather that the data are arranged in groups (according to values of the BY variables) and that these groups are not necessarily in alphabetical or increasing numeric order.
- Create an index on the BY variables by using the DATASETS procedure (in Base SAS software).

For more information about the BY statement, see *SAS Language Reference: Concepts*. For more information about the DATASETS procedure, see the *Base SAS Procedures Guide*.

ID Statement

ID *variables* **;**

The ID statement identifies the variable(s) from the DATA= data set to be included in the OUT= data set. When this statement is omitted, PROC HAPLOTYPE creates in the OUT= data set the variable _ID_ that contains a unique numeric identifier for each individual. When the TALL option is used, this statement is ignored, and the INDIVIDUAL variable is automatically included in the OUT= data set along with the trait variable if the TRAIT statement is specified.

TRAIT Statement

TRAIT *variable < / options >* **;**

The TRAIT statement identifies the binary variable that indicates which individuals are cases and which are controls, or that represents a dichotomous trait. This variable can be character or numeric, but it must have only two nonmissing levels. When this statement is used and the original or stepwise EM algorithm is implemented to estimate haplotype frequencies, the "Test for Marker-Trait Association" table is included in the output.

There are two options you can specify in the TRAIT statement:

PERMS=*number*

PERMUTATION=*number*

specifies the number of permutations to be used to calculate the empirical p-value of the haplotype case-control tests. This number must be a positive integer. By default, no permutations are used and the p-value is calculated using the chi-square test statistic. Note that this option can greatly increase the computation time.

TESTALL

specifies that each individual haplotype should be tested for association with the TRAIT variable. When this option is included in the TRAIT statement, the "Tests for Haplotype-Trait Association" table is included in the output.

VAR Statement

VAR *variables* **;**

The VAR statement identifies the variables containing either the marker alleles or, if GENOCOL is specified, the marker genotypes. The following number of variables should be specified in this statement for a data set containing m markers according to whether the options GENOCOL and TALL are used:

- When both GENOCOL and TALL are specified, there should be one variable named containing marker genotypes.
- When only TALL is specified, there should be two variables named containing marker alleles.
- When only GENOCOL is specified, there should be m variables named, one for each marker containing marker genotypes.
- When neither option is specified, there should be $2m$ variables named, two for each marker containing marker alleles.

All variables specified must be of the same type, either character or numeric.

Details: HAPLOTYPE Procedure

Statistical Computations

The EM Algorithm

The EM algorithm (Excoffier and Slatkin 1995; Hawley and Kidd 1995; Long, Williams, and Urbanek 1995) iteratively furnishes the maximum likelihood estimates (MLEs) of m-locus haplotype frequencies, for any integer $m > 1$, when a direct solution for the MLE is not readily feasible. The EM algorithm assumes HWE; it has been argued (Fallin and Schork 2000) that positive increases in the Hardy-Weinberg disequilibrium coefficient (toward excess heterozygosity) can increase the error of the EM estimates, but negative increases (toward excess homozygosity) do not demonstrate a similar increase in the error. The iterations start with assigning initial values to the haplotype frequencies. When the INIT=RANDOM option is included in the PROC HAPLOTYPE statement, uniformly distributed random values are assigned to all haplotype frequencies; when INIT=UNIFORM, each haplotype is given an initial frequency of $1/h$, where h is the number of possible haplotypes in the sample. Otherwise, the product of the frequencies of the alleles that constitute the haplotype is used as the initial frequency for the haplotype. Different starting values can lead to different solutions since a maximum that is found could be a local maximum and not the global maximum. You can try different starting values for the EM algorithm by specifying a number greater than 1 in the NSTART= option to get better estimates. The expectation and maximization steps (E step and M step, respectively) are then carried out until the convergence criterion is met or the number of iterations exceeds the number specified in the MAXITER= option of the PROC HAPLOTYPE statement.

For a sample of n individuals, suppose the ith individual has genotype G_i. The probability of this genotype in the population is P_i, so the log likelihood is

$$\log L = \sum_{i=1}^{n} \log P_i$$

which is calculated after each iteration's E step of the EM algorithm, described in the following paragraphs.

Let h_j be the jth possible haplotype, and let f_j be its frequency in the population. For genotype G_i, the set H_i is the collection of pairs of haplotypes, h_j and its "complement" h_j^{ci}, that constitute that genotype. The haplotype frequencies f_j used in the E step for iteration 0 of the EM algorithm are given by the initial values; all subsequent iterations use the haplotype frequencies calculated by the M step of the previous iteration. The E step sets the genotype frequencies to be products of these frequencies:

$$P_i = \sum_{j \in H_i} f_j f_j^{ci}$$

When G_i has m heterozygous loci, there are 2^{m-1} terms in this sum. The number of times haplotype h_j occurs in the sum is written as m_{ij}, which is 2 if G_i is completely homozygous, and either 1 or 0 otherwise.

The M step sets new haplotype frequencies from the genotype frequencies:

$$f_j = \frac{1}{2n} \sum_{i=1}^{n} \frac{m_{ij} f_j f_j^{ci}}{P_i}$$

The EM algorithm increases the likelihood after each iteration, and multiple starting points can generally lead to the global maximum.

When the option EST=STEPEM is specified in the PROC HAPLOTYPE statement, a stepwise version of the EM algorithm is performed. A common difficulty in haplotype analysis is that the number of possible haplotypes grows exponentially with the number of loci, as does the computation time, which makes the EM algorithm infeasible for a large number of loci. However, the most common haplotypes can still be estimated by trimming the haplotype table according to a given cutoff (Clayton 2002). The two-locus haplotype frequencies are first estimated, and those below the cutoff are discarded from the table. The remaining haplotypes are expanded to the next locus by forming all possible three-locus haplotypes, and the EM algorithm is then invoked for this haplotype table. The trimming and expanding process is performed repeatedly, adding one locus at a time, until all loci are considered.

After the EM or stepwise EM algorithm has arrived at the MLEs of the haplotype frequencies, each individual i's probability of having a particular haplotype pair (h_j, h_j^{ci}) given the individual's genotype G_i is calculated as

$$\Pr\{h_j, h_j^{ci} | G_i\} = \frac{f_j f_j^{ci}}{P_i}$$

for each $j \in H_i$. These probabilities are displayed in the OUT= data set.

Methods of Estimating Standard Error

Typically, an estimate of the variance of a haplotype frequency is obtained by inverting the estimated information matrix from the distribution of genotype frequencies. However, it often turns out that in a large multilocus system, a certain proportion of haplotypes have ML frequencies equal or close to zero, which makes the sample information matrix nearly singular (Excoffier and Slatkin 1995). Therefore, two approximation methods are used to estimate the variances, as proposed by Hawley and Kidd (1995).

The binomial method estimates the standard error by calculating the square root of the binomial variance, as if the haplotype frequencies are obtained by direct counting:

$$\mathrm{Var}_{\mathrm{B}}(f_j) = \frac{f_j(1-f_j)}{2n-1}$$

The jackknife method is a simulation-based method that can be used to estimate the standard errors of haplotype frequencies. Each individual is in turn removed from the sample, and all the haplotype frequencies are recalculated from this “delete-1” sample. Let $T_{n-1,i}$ be the haplotype frequency estimator from the ith “delete-1” sample; then the jackknife variance estimator has the following formula:

$$\mathrm{Var}_{\mathrm{J}}(f_j) = \frac{n-1}{n} \sum_{i=1}^{n} \left(T_{n-1,i} - \frac{1}{n} \sum_{j=1}^{n} T_{n-1,j} \right)^2$$

and the square root of this variance estimate is the estimate of standard error. The jackknife is less dependent on the model assumptions; however, it requires computing the statistic n times.

Confidence intervals with confidence level $1-\alpha$ for the haplotype frequency estimates from the final iteration are then calculated using the following formula:

$$f_j \pm z_{1-\alpha/2} \sqrt{\mathrm{Var}(f_j)}$$

where $z_{1-\alpha/2}$ is the value from the standard normal distribution that has a right-tail probability of $\alpha/2$.

Testing for Allelic Associations

When the LD option is specified in the PROC HAPLOTYPE statement, haplotype frequencies are calculated using the EM algorithm as well as by assuming no allelic associations among loci—that is, no LD. Under the null hypothesis of no LD, haplotype frequencies are simply the product of the individual allele frequencies. The log likelihood under the null hypothesis, $\log L_0$, is calculated based on these haplotype frequencies with degrees of freedom $\mathrm{df}_0 = \sum_{i=1}^{m}(k_i - 1)$, where m is the number of loci and k_i is the number of alleles for the ith locus (Zhao, Curtis, and Sham 2000). Under the alternative hypothesis, the log likelihood, $\log L_1$, is calculated from the EM estimates of the haplotype frequencies with degrees of freedom $\mathrm{df}_1 =$ number of haplotypes $- 1$. A likelihood ratio test is used to test this hypothesis as follows:

$$2(\log L_1 - \log L_0) \sim \chi^2_\nu$$

where $\nu = \mathrm{df}_1 - \mathrm{df}_0$ is the difference between the number of degrees of freedom under the null hypothesis and the alternative.

Testing for Trait Associations

When the TRAIT statement is included in PROC HAPLOTYPE, case-control tests are performed to test for association between the dichotomous trait (often, an indicator of individuals with or without a disease) and the marker loci by using haplotypes. In addition to an omnibus test that is performed across all haplotypes, when the TESTALL option is specified in the TRAIT statement, a test for association between each individual haplotype and the trait is performed. Note that the individual haplotype tests should be performed only if the omnibus test statistic is significant.

Chi-Square Tests The test performed over all haplotypes is based on the log likelihoods: under the null hypothesis, the log likelihood over all the individuals in the sample, regardless of the value of their trait variable, is calculated as described in "The EM Algorithm" on page 129; the log likelihood is also calculated separately for the two sets of individuals within the sample as determined by the trait value under the alternative hypothesis of marker-trait association. A likelihood ratio test (LRT) statistic can then be formed as follows:

$$X^2 = 2(\log L_1 + \log L_2 - \log L_0)$$

where $\log L_0$, $\log L_1$, and $\log L_2$ are the log likelihoods under the null hypothesis, for individuals with the first trait value, and for individuals with the second trait value, respectively (Zhao, Curtis, and Sham 2000). Defining degrees of freedom for each log likelihood similarly, this statistic has an asymptotic chi-square distribution with $(\mathrm{df}_1 + \mathrm{df}_2 - \mathrm{df}_0)$ degrees of freedom.

An association between individual haplotypes and the trait can also be tested. To do so, a contingency table (Table 7.1) is formed where $T = 2n = t_1 + t_2 = h_1 + h_2$, the total number of haplotypes in the sample, "Hap 1" refers to the current haplotype being tested, "Hap 2" refers to all other haplotypes, and c_{ij} is the pseudo-observed count of individuals with trait i and haplotype j (note that these counts are not necessarily integers since haplotypes are not actually observed; they are calculated based on the estimated haplotype frequencies). The column totals h_j are not calculated in the usual fashion, by summing the cells in each column; rather, h_1 and h_2 are calculated as $T \times f_j$ and $T - T \times f_j$, respectively, where f_j is the estimated frequency of "Hap 1" in the overall sample.

Table 7.1 Haplotype-Trait Counts

	Hap 1	Hap 2	Total
Trait 1	c_{11}	c_{12}	t_1
Trait 2	c_{21}	c_{22}	t_2
Total	h_1	h_2	T

The usual contingency table chi-square test statistic has a 1 df chi-square distribution:

$$\sum_{i=1,2} \sum_{j=1,2} \frac{(c_{ij} - t_i h_j / T)^2}{t_i h_j / T}$$

Permutation Tests Since the assumption of a chi-square distribution in the preceding section might not hold, estimates of exact p-values via Monte Carlo methods are recommended. New samples are formed by randomly permuting the trait values, and either of the chi-square test statistics shown in the previous section can be calculated for each of these samples. The number of new samples created is determined by the number given in the PERMS= option of the TRAIT statement. The exact p-value approximation is then calculated as m/p, where m is the number of samples with a test statistic greater than or equal to the test statistic in the actual sample and p is the total number of permutation samples. This method is used to obtain empirical p-values for both the overall and the individual haplotype tests (Zhao, Curtis, and Sham 2000; Fallin et al. 2001).

Bayesian Estimation of Haplotype Frequencies

The Bayesian algorithm for haplotype reconstruction incorporates coalescent theory in a Markov chain Monte Carlo (MCMC) technique (Stephens, Smith, and Donnelly 2001; Lin et al. 2002). The algorithm starts with a random phase assignment for each multilocus genotype and then uses a Gibbs sampler to assign a haplotype pair to a randomly picked phase-unknown genotype. The algorithm implemented in PROC HAPLOTYPE is from Lin et al. (2002), which has several variations from that of Stephens, Smith, and Donnelly (2001).

Initially, gamete pairs are randomly assigned to each genotype, and the assignment set is denoted as $H = (H_1, \ldots, H_n)$. An individual i is then randomly picked, and its two current haplotypes are removed from H. The remaining assignment set is denoted H_{-i}. Let Y be the positions where the diplotypic sequence of individual i is ambiguous, and let $h_j(Y)$ be the partial sequence of the jth haplotype at Y. From H_{-i}, a list of partial haplotypes $h(Y) = [h_1(Y), \ldots, h_m(Y)]$ is made, with corresponding counts $[r_1, \ldots, r_m]$ sampled from H_{-i}.

The next step is to reassign a haplotype pair to individual i and add to H_{-i}. The probability vector, $p = (p_1, \ldots, p_m)$, of sampling each partial haplotype $h_j(Y)$ from $h(Y)$ is calculated as follows: let $d_i(Y)$ be the partial genotype of individual i at Y. For $j = 1, \ldots, m$, check whether $d_i(Y)$ can comprise $h_j(Y)$ plus a complementary haplotype $h'_j(Y)$ (note that $h'_j(Y)$ can bear missing alleles if $d_i(Y)$ is incomplete). If not, set $p_j = 0$; if so, with all $h_k(Y)$ in $h(Y)$ that are compatible with $h'_j(Y)$, set $p_j = \sum_k [r_j r_k + (r_j + r_k)\theta/M]$, where $\theta = 1/\sum_{i=1}^{2n-1} 1/i$ by default (or this value can be set in the THETA= option), n is the number of individuals in the sample, and M is the number of distinct haplotypes possible in the population. If no such $h_k(Y)$ can be found, set $p_j = r_j(\theta/M)$.

With probability $2^k(\theta/M)^2/[\sum_j p_j + 2^k(\theta/M)^2]$, k being the number of ambiguous sites in individual i, randomly reconstruct phases for individual i. Missing alleles are assigned proportional to allele frequencies at each site. Otherwise, with probability $p_j / \sum_{j'} p_{j'}$, assign $h_j(Y)$ to individual i and the corresponding complementary haplotype. The new assignment is then added back to H_{-i}.

These steps are repeated t times, where t is the value specified in the TOTALRUN= option. The first b times are discarded as burn-in when BURNIN=b. The results are then thinned by recording every rth assignment specified by the INTERVAL=r option so that $(t - b)/r$ iterations are used for the estimates.

The probability of each individual i having a particular haplotype pair (h_j, h_j^{ci}) given the individual's genotype G_i for each $j \in H_i$ is given in the OUT= data set as the proportion of iterations after burn-in that are recorded which have that particular haplotype pair assigned to the individual.

Missing Values

An individual's m-locus genotype is considered to be partially missing if any, but not all, of the alleles are missing. Genotypes with all missing alleles are dropped from calculations for haplotype frequencies, although these individuals can still be used as described in the following paragraph. Also, if there are any markers with all missing values in a BY group (or the entire data set if there is no BY statement), no calculations are performed for that BY group. Partially missing genotypes are used in the EM algorithm and the jackknife procedure. In calculating the allele frequencies, missing alleles are dropped and the frequency of an allele u at a marker is obtained as the number of u alleles in the data divided by the total number of nonmissing alleles at the marker in the data. In the E step of the EM algorithm, the frequency of a partially missing genotype is updated for every possible genotype. In the M step, haplotypes resulting from a missing genotype can bear some missing alleles. Such a haplotype is not considered as a new haplotype, but rather all existing haplotypes that have alleles identical to the nonmissing alleles of this haplotype are updated. Dealing with missing genotypes involves looping through all possible genotypes in the E step and all possible haplotypes in the M step. The stepwise EM algorithm performs a series of two-step processes involving EM estimation followed by trimming the set of haplotypes. Thus, in the EM estimation step, missing values are handled as described for the EM algorithm. Depending on the input data set, missing genotypes can increase the computation time substantially for either estimation method.

When the TRAIT statement is specified, any observation with a missing trait value is dropped from calculations used in the tests for marker-trait association and haplotype-trait associations. However, observations with missing trait values are included in calculating the frequencies shown in the "Haplotype Frequencies" table, which are then used in the OUT= data set. The combined frequencies listed in the "Tests for Haplotype-Trait Association" table might therefore be different from these frequencies in this situation. Also, if an individual is missing all alleles but has a nonmissing trait value, the individual is included in the permutations of the trait value when PERMS= is specified in the TRAIT statement.

OUT= Data Set

The OUT= data set contains the following variables: the BY variables (if any), _ID_ that identifies the individual and/or any variables listed in the ID statement, the pair of alleles at each marker analyzed, HAPLOTYPE1 and HAPLOTYPE2 that contain the pair of haplotypes that each genotype can comprise, and PROB containing the probability of each individual's genotype being resolved into that haplotype pair. Note that when GENOCOL or the option TALL is specified, the pair of alleles at a marker are contained in a single column separated by the delimiter '/' or the character given in the DELIMITER= option.

Displayed Output

This section describes the displayed output from PROC HAPLOTYPE. See "ODS Table Names" on page 137 for details about how this output interfaces with the Output Delivery System.

Analysis Information

The “Analysis Information” table lists information about the following settings used in PROC HAPLOTYPE for all of the estimation methods:

- Loci Used, the loci used to form haplotypes
- Number of Individuals
- Random Number Seed, the value specified in the SEED= option or generated by the system clock
- Haplotype Frequency Cutoff, the value specified in the CUTOFF= option or the default (0)

When EST=EM or EST=STEPEM is specified in the PROC HAPLOTYPE statement, the following information is also included in the table:

- Number of Starts, the value specified in the NSTART= option or the default (1)
- Convergence Criterion, the value specified in the CONV= option or the default (0.00001)
- Iterations Checked for Conv., the value specified in the NLAG= option or the default (1)
- Maximum Number of Iterations, the value specified in the MAXITER= option or the default (100)
- Number of Iterations Used, as determined by the CONV= or MAXITER= option
- Log Likelihood, from the last iteration performed
- Initialization Method, the method specified in the INIT= option or “Linkage Equilibrium” by default
- Standard Error Method, the method specified in the SE= option or “Binomial” by default

If EST=BAYESIAN is specified in the PROC HAPLOTYPE statement, then these rows are included in the table:

- Scaled Mutation Rate, the θ parameter used in the algorithm
- Recorded Iterations, the number of iterations of the algorithm actually recorded, which is (total runs $-$ burn-in)$/$interval

Iteration History

The “Iteration History” table displays the log likelihood and the ratio of change for each iteration of the EM algorithm.

Haplotype Frequencies

The “Haplotype Frequencies” table lists all the possible m-locus haplotypes in the sample (where $2m$ variables are specified in the VAR statement), with an estimate of the haplotype frequency, the standard error of the frequency, and the lower and upper limits of the confidence interval for the frequency based on the confidence level determined by the ALPHA= option of the PROC HAPLOTYPE statement (0.95 by default). When the LD option is specified in the PROC HAPLOTYPE statement and EST=EM or STEPEM, haplotype frequency estimates are calculated both under the null hypothesis of no allelic association by taking the product of allele frequencies, and under the alternative, which allows for associations, using the EM algorithm.

Test for Allelic Associations

The “Test for Allelic Associations” table displays the degrees of freedom and log likelihood calculated using the EM algorithm for the null hypothesis of no association and the alternative hypothesis of associations between markers. The chi-square statistic and its p-value are also shown for the test of these hypotheses.

Test for Marker-Trait Association

The “Test for Marker-Trait Association” table displays the number of observations, degrees of freedom, and log likelihood for both trait values as well as the combined sample when EST=EM or STEPEM. The chi-square test statistic and its corresponding p-value from performing the case-control test, testing the hypothesis of no association between the trait and the marker loci used in PROC HAPLOTYPE, are also given. When the PERMS= option is included in the TRAIT statement, estimates of exact p-values are provided as well.

Tests for Haplotype-Trait Association

The “Tests for Haplotype-Trait Association” table displays statistics from case-control tests performed on each individual haplotype when the TESTALL option is included in the TRAIT statement and EST=EM or STEPEM. A significant p-value indicates that there is an association between the haplotype and the trait. When the PERMS= option is also given in the TRAIT statement, estimates of exact p-values are provided as well.

ODS Table Names

PROC HAPLOTYPE assigns a name to each table it creates, and you must use this name to reference the table when using the Output Delivery System (ODS). These names are listed in Table 7.2.

Table 7.2 ODS Tables Created by the HAPLOTYPE Procedure

ODS Table Name	Description	Statement or Option
AnalysisInfo	Analysis information	default
IterationHistory	Iteration history	ITPRINT EST=EM or STEPEM
ConvergenceStatus	Convergence status	EST=EM or STEPEM
HaplotypeFreq	Haplotype frequencies	default
LDTest	Test for allelic associations	LD and EST=EM or STEPEM
CCTest	Test for marker-trait association	EST=EM or STEPEM TRAIT statement
HapTraitTest	Tests for haplotype-trait association	EST=EM or STEPEM TRAIT / TESTALL

Examples: HAPLOTYPE Procedure

Example 7.1: Estimating Three-Locus Haplotype Frequencies

Here is an example of 227 individuals genotyped at three markers, data that were created based on genotype frequency tables from the Lab of Statistical Genetics at Rockefeller University (2001). Note that when reading in the data, there are four individuals' genotypes per line, except for the last line of the DATA step, which contains three individuals' genotypes. The SAS data set that is created using the following code contains one individual per row with six columns representing the two alleles at each of three marker loci.

```
data ehdata;
   input m1-m6 @@;
   datalines;
1 1 1 1 1 3 1 1 1 1 1 3 1 1 1 1 1 3 1 1 1 1 1 3
1 1 1 1 1 3 1 1 1 1 1 3 1 1 1 1 1 3 1 1 1 1 1 3
1 1 1 1 1 3 1 1 1 1 1 3 1 1 1 1 1 3 1 1 1 1 1 3
1 1 1 1 2 3 1 1 1 1 2 3 1 1 1 1 2 3 1 1 1 1 3 3
1 1 1 1 3 3 1 1 1 1 3 3 1 1 1 1 3 3 1 1 1 1 3 3
1 1 1 1 3 3 1 1 1 1 3 3 1 1 1 1 3 3 1 1 1 1 3 3
1 1 1 2 1 1 1 1 1 2 1 1 1 1 1 2 1 2 1 1 1 2 1 2
1 1 1 2 1 2 1 1 1 2 1 2 1 1 1 2 1 2 1 1 1 2 1 2
1 1 1 2 1 2 1 1 1 2 1 2 1 1 1 2 1 2 1 1 1 2 1 2
1 1 1 2 1 2 1 1 1 2 1 2 1 1 1 2 2 2 1 1 1 2 2 2
1 1 1 2 1 3 1 1 1 2 1 3 1 1 1 2 2 3 1 1 1 2 2 3
1 1 1 2 2 3 1 1 1 2 3 3 1 1 1 2 3 3 1 1 1 2 3 3
1 1 1 2 3 3 1 1 2 2 1 1 1 1 2 2 1 1 1 1 2 2 1 1
1 1 2 2 1 1 1 1 2 2 1 1 1 1 2 2 1 2 1 1 2 2 1 2
1 1 2 2 1 2 1 1 2 2 1 2 1 1 2 2 2 2 1 1 2 2 2 2
1 1 2 2 2 2 1 1 2 2 2 2 1 1 2 2 1 3 1 1 2 2 1 3
1 1 2 2 3 3 1 1 2 2 3 3 1 1 2 2 3 3 1 1 2 2 3 3
1 1 2 2 3 3 1 1 2 2 3 3 1 1 2 2 3 3 1 1 2 2 3 3
1 1 2 2 3 3 1 1 2 2 3 3 1 1 2 2 3 3 1 1 2 2 3 3
1 1 2 2 3 3 1 1 2 2 3 3 1 1 2 2 3 3 1 1 2 2 3 3
1 1 2 2 3 3 1 1 2 2 3 3 1 2 1 1 1 1 1 2 1 1 1 1
1 2 1 1 1 1 1 2 1 1 1 1 1 2 1 1 1 1 1 2 1 1 1 1
1 2 1 1 1 1 1 2 1 1 1 2 1 2 1 1 1 2 1 2 1 1 1 3
1 2 1 1 1 3 1 2 1 1 2 3 1 2 1 1 2 3 1 2 1 1 2 3
1 2 1 1 2 3 1 2 1 1 2 3 1 2 1 1 2 3 1 2 1 1 3 3
1 2 1 1 3 3 1 2 1 1 3 3 1 2 1 2 1 1 1 2 1 2 1 1
1 2 1 2 1 1 1 2 1 2 1 1 1 2 1 2 1 1 1 2 1 2 1 1
1 2 1 2 1 1 1 2 1 2 1 1 1 2 1 2 1 1 1 2 1 2 1 3
1 2 1 2 1 3 1 2 1 2 1 3 1 2 1 2 2 3 1 2 1 2 2 3
1 2 1 2 2 3 1 2 1 2 2 3 1 2 1 2 3 3 1 2 1 2 3 3
1 2 2 2 1 1 1 2 2 2 1 1 1 2 2 2 1 1 1 2 2 2 1 1
1 2 2 2 1 1 1 2 2 2 1 1 1 2 2 2 1 1 1 2 2 2 1 1
1 2 2 2 1 1 1 2 2 2 1 1 1 2 2 2 1 2 1 2 2 2 1 2
1 2 2 2 1 2 1 2 2 2 1 3 1 2 2 2 1 3 1 2 2 2 2 3
1 2 2 2 2 3 1 2 2 2 2 3 1 2 2 2 3 3 1 2 2 2 3 3
1 2 2 2 3 3 1 2 2 2 3 3 1 2 2 2 3 3 1 2 2 2 3 3
1 2 2 2 3 3 1 2 2 2 3 3 2 2 1 1 1 1 2 2 1 1 1 2
2 2 1 1 1 2 2 2 1 1 2 2 2 2 1 1 2 2 2 2 1 1 2 2
2 2 1 1 2 2 2 2 1 1 2 2 2 2 1 1 2 2 2 2 1 1 2 2
2 2 1 1 2 2 2 2 1 1 2 2 2 2 1 1 2 2 2 2 1 1 1 3
2 2 1 1 1 3 2 2 1 1 2 3 2 2 1 1 2 3 2 2 1 1 2 3
2 2 1 1 2 3 2 2 1 1 2 3 2 2 1 1 2 3 2 2 1 1 2 3
2 2 1 1 2 3 2 2 1 1 2 3 2 2 1 1 3 3 2 2 1 1 3 3
2 2 1 1 3 3 2 2 1 1 3 3 2 2 1 2 1 1 2 2 1 2 1 1
2 2 1 2 1 1 2 2 1 2 1 1 2 2 1 2 1 1 2 2 1 2 1 2
2 2 1 2 1 2 2 2 1 2 1 2 2 2 1 2 2 2 2 2 1 2 2 2
2 2 1 2 2 2 2 2 1 2 2 2 2 2 1 2 1 3 2 2 1 2 1 3
2 2 1 2 1 3 2 2 1 2 1 3 2 2 1 2 1 3 2 2 1 2 1 3
2 2 1 2 2 3 2 2 1 2 2 3 2 2 1 2 2 3 2 2 1 2 2 3
2 2 1 2 2 3 2 2 1 2 2 3 2 2 1 2 2 3 2 2 1 2 3 3
```

```
2 2 1 2 3 3 2 2 1 2 3 3 2 2 2 2 1 1 2 2 2 2 1 1
2 2 2 2 1 1 2 2 2 2 1 1 2 2 2 2 1 1 2 2 2 2 1 1
2 2 2 2 1 1 2 2 2 2 1 1 2 2 2 2 1 1 2 2 2 2 1 2
2 2 2 2 1 2 2 2 2 2 1 2 2 2 2 2 2 3 2 2 2 2 2 3
2 2 2 2 2 3 2 2 2 2 3 3 2 2 2 2 3 3 2 2 2 2 3 3
2 2 2 2 3 3 2 2 2 2 3 3 2 2 2 2 3 3 2 2 2 2 3 3
2 2 2 2 3 3 2 2 2 2 3 3 2 2 2 2 3 3
;
```

The haplotype frequencies can be estimated using the EM algorithm and their standard errors estimated using the jackknife method by implementing the following code:

```
proc haplotype data=ehdata se=jackknife maxiter=20 itprint nlag=4;
   var m1-m6;
run;
```

This produces the ODS output shown in Outputs 7.1.1 through 7.1.4.

Output 7.1.1 Analysis Information for the HAPLOTYPE Procedure

```
                        The HAPLOTYPE Procedure

                         Analysis Information

        Loci Used                                      M1 M2 M3
        Number of Individuals                               227
        Number of Starts                                      1
        Convergence Criterion                           0.00001
        Iterations Checked for Conv.                          4
        Maximum Number of Iterations                         20
        Number of Iterations Used                            11
        Log Likelihood                               -934.97918
        Initialization Method                Linkage Equilibrium
        Standard Error Method                         Jackknife
        Haplotype Frequency Cutoff                            0
```

Output 7.1.1 displays information about several of the settings used to perform the HAPLOTYPE procedure on the ehdata data set. Note that although the MAXITER= option was set to 20 iterations, convergence according to the criterion of 0.00001 was reached for four consecutive iterations prior to the 20th iteration, at which point the estimation process stopped. To obtain more precise frequency estimates, a lower convergence criterion can be used.

Output 7.1.2 Iteration History for the HAPLOTYPE Procedure

```
                         Iteration History

                                          Ratio
                     Iter        LogLike    Changed

                        0     -953.89697
                        1     -937.92181    0.01675
                        2     -935.91870    0.00214
                        3     -935.35775    0.00060
                        4     -935.13050    0.00024
                        5     -935.03710    0.00010
                        6     -935.00051    0.00004
                        7     -934.98679    0.00001
                        8     -934.98180    0.00001
                        9     -934.98002    0.00000
                       10     -934.97940    0.00000
                       11     -934.97918    0.00000
```

Output 7.1.3 Convergence Status for the HAPLOTYPE Procedure

```
         Algorithm converged.
```

Because the ITPRINT option was specified in the PROC HAPLOTYPE statement, the iteration history of the EM algorithm is included in the ODS output. Output 7.1.2 contains the table displaying this information. By default, the “Convergence Status” table is displayed (Output 7.1.3), which consists of only one line indicating whether convergence was met.

Output 7.1.4 Haplotype Frequencies from the HAPLOTYPE Procedure

```
                           Haplotype Frequencies

                                          Standard       95% Confidence
         Number      Haplotype      Freq      Error            Limits

              1      1-1-1       0.09170    0.01505    0.06221    0.12119
              2      1-1-2       0.02080    0.00952    0.00214    0.03946
              3      1-1-3       0.11509    0.01766    0.08048    0.14971
              4      1-2-1       0.07904    0.01696    0.04580    0.11228
              5      1-2-2       0.06768    0.01546    0.03738    0.09799
              6      1-2-3       0.12788    0.02094    0.08685    0.16891
              7      2-1-1       0.05521    0.01227    0.03115    0.07926
              8      2-1-2       0.11700    0.01782    0.08207    0.15193
              9      2-1-3       0.07376    0.01495    0.04446    0.10307
             10      2-2-1       0.11766    0.01831    0.08177    0.15355
             11      2-2-2       0.03020    0.00899    0.01257    0.04782
             12      2-2-3       0.10397    0.01833    0.06805    0.13989
```

Output 7.1.4 displays the 12 possible three-locus haplotypes in the data and their estimated haplotype frequencies, standard errors, and bounds for the 95% confidence intervals for the estimates.

To see how the CUTOFF= option affects the "Haplotype Frequencies" table, suppose you want to view only the haplotypes with an estimated frequency of at least 0.10. The following code creates such a table:

```
proc haplotype data=ehdata se=jackknife cutoff=0.10 nlag=4;
  var m1-m6;
run;
```

Now the "Haplotype Frequencies" table is displayed as in Output 7.1.5.

Output 7.1.5 Haplotype Frequencies from the HAPLOTYPE Procedure Using the CUTOFF= Option

```
                          The HAPLOTYPE Procedure

                            Haplotype Frequencies

                                           Standard       95% Confidence
      Number     Haplotype        Freq        Error            Limits

           1     1-1-3         0.11509      0.01766     0.08048     0.14971
           2     1-2-3         0.12788      0.02094     0.08685     0.16891
           3     2-1-2         0.11700      0.01782     0.08207     0.15193
           4     2-2-1         0.11766      0.01831     0.08177     0.15355
           5     2-2-3         0.10397      0.01833     0.06805     0.13989
```

Output 7.1.5 displays only the five 3-locus haplotypes with estimated frequencies of at least 0.10. This option is especially useful for keeping the "Haplotype Frequencies" table to a manageable size when many marker loci or loci with several alleles are used and when many of the haplotypes have estimated frequencies very near zero. Using CUTOFF=1 suppresses the "Haplotype Frequencies" table.

Example 7.2: Using Multiple Runs of the EM Algorithm

Continuing the example from the section "Getting Started: HAPLOTYPE Procedure" on page 120, suppose you are concerned that the likelihood reached a local and not a global maximum. You can request that PROC HAPLOTYPE use several different sets of initial haplotype frequencies to ensure that you find a global maximum of the likelihood. The following code invokes the EM algorithm with five different sets of initial values, including the set used in the "Getting Started" example:

```
proc haplotype data=markers prefix=SNP init=random seed=51220
               nstart=5;
   var m1-m8;
run;
```

The NSTART=5 option requests that the EM algorithm be run three times with randomly generated initial frequencies, including once using the seed 51220 that was previously used, once using uniform initial frequencies, and once using haplotype frequencies given by the product of the allele frequencies. The two tables in Output 7.2.1 are from the run that produced the best log likelihood:

Output 7.2.1 Output from PROC HAPLOTYPE

```
                          The HAPLOTYPE Procedure

                           Analysis Information

              Loci Used                        SNP1 SNP2 SNP3 SNP4
              Number of Individuals                             25
              Number of Starts                                   5
              Convergence Criterion                        0.00001
              Iterations Checked for Conv.                       1
              Maximum Number of Iterations                     100
              Number of Iterations Used                         19
              Log Likelihood                             -95.94742
              Initialization Method                         Random
              Random Number Seed                         499887544
              Standard Error Method                       Binomial
              Haplotype Frequency Cutoff                         0


                           Haplotype Frequencies

                                      Standard      95% Confidence
        Number     Haplotype     Freq      Error            Limits

             1     A-A-A-A    0.14324    0.05005    0.04515    0.24133
             2     A-A-A-B    0.07507    0.03764    0.00129    0.14885
             3     A-A-B-A    0.00000    0.00001    0.00000    0.00001
             4     A-A-B-B    0.00000    0.00010    0.00000    0.00019
             5     A-B-A-A    0.09295    0.04148    0.01165    0.17425
             6     A-B-A-B    0.05349    0.03214    0.00000    0.11649
             7     A-B-B-A    0.00001    0.00052    0.00000    0.00103
             8     A-B-B-B    0.07523    0.03768    0.00138    0.14909
             9     B-A-A-A    0.08644    0.04014    0.00776    0.16512
            10     B-A-A-B    0.08784    0.04044    0.00859    0.16710
            11     B-A-B-A    0.07904    0.03854    0.00350    0.15459
            12     B-A-B-B    0.10836    0.04441    0.02133    0.19540
            13     B-B-A-A    0.10097    0.04304    0.01661    0.18533
            14     B-B-A-B    0.00000    0.00000    0.00000    0.00000
            15     B-B-B-A    0.09735    0.04235    0.01435    0.18035
            16     B-B-B-B    0.00000    0.00000    0.00000    0.00000
```

Example 7.3: Testing for Linkage Disequilibrium

Again looking at the data from the Lab of Statistical Genetics at Rockefeller University (2001), if you request the test for linkage disequilibrium by specifying the LD option in the PROC HAPLOTYPE statement as follows, the “Test for Allelic Associations” table containing the test statistics is included in the output.

```
proc haplotype data=ehdata ld;
   var m1-m6;
run;
```

The “Haplotype Frequencies” table (Output 7.3.1) now contains an extra column of the haplotype frequencies under the null hypothesis.

Output 7.3.1 Haplotype Frequencies under the Null and Alternative Hypotheses

```
                              The HAPLOTYPE Procedure

                               Haplotype Frequencies

                                                 Standard      95% Confidence
 Number      Haplotype     H0 Freq     H1 Freq      Error           Limits

      1      1-1-1         0.08172     0.09124    0.01353     0.06472     0.11775
      2      1-1-2         0.05605     0.02124    0.00677     0.00796     0.03452
      3      1-1-3         0.10006     0.11501    0.01499     0.08563     0.14439
      4      1-2-1         0.09084     0.07952    0.01271     0.05461     0.10443
      5      1-2-2         0.06231     0.06726    0.01177     0.04419     0.09032
      6      1-2-3         0.11122     0.12794    0.01569     0.09718     0.15870
      7      2-1-1         0.08100     0.05540    0.01075     0.03433     0.07647
      8      2-1-2         0.05556     0.11690    0.01510     0.08732     0.14649
      9      2-1-3         0.09918     0.07378    0.01228     0.04971     0.09785
     10      2-2-1         0.09005     0.11746    0.01513     0.08781     0.14711
     11      2-2-2         0.06176     0.03028    0.00805     0.01450     0.04606
     12      2-2-3         0.11025     0.10398    0.01434     0.07587     0.13209
```

Note that since the INIT= option was omitted from the PROC HAPLOTYPE statement, the initial haplotype frequencies used in the EM algorithm are identical to the frequencies that appear in the H0 FREQ column in Output 7.3.1. The frequencies in the H1 FREQ column are those calculated from the final iteration of the EM algorithm, and these frequencies’ standard errors and confidence limits are included in the table as well.

Output 7.3.2 displays the log likelihood under the null hypothesis assuming independence among all the loci and the alternative, which allows for associations between markers. The empirical chi-square test statistic of the likelihood ratio test is calculated as $X^2 = 2[-934.98180 - (-953.89697)] = 37.8303$ with degrees of freedom $\nu = 11 - 4 = 7$ that gives a p-value < 0.0001. The test indicates significant linkage disequilibrium among the three loci, as shown in the online documentation from the Lab of Statistical Genetics at Rockefeller University (2001).

Output 7.3.2 Testing for Linkage Disequilibrium Using the LD Option

```
                          Test for Allelic Associations

                                                                 Chi-      Pr >
   Hypothesis                        DF        LogLike         Square     ChiSq

   H0: No Association                 4     -953.89697
   H1: Allelic Associations          11     -934.98180        37.8303    <.0001
```

Example 7.4: Testing for Marker-Trait Associations

To demonstrate how the TRAIT statement can be used, a subset of data from GAW12 (Wijsman et al. 2001) is read into a SAS data set as follows:

```
data gaw;
   input status $ a1-a24;
   datalines;
U 8 4 4 4 2 7 3 2 1 4 10 2  6 6 1 2 1 1 7  7  8 7 8  8
U 5 9 3 5 3 4 2 3 4 3 14 10 3 6 7 7 1 4 5  12 3 3 1  2
A 8 2 5 1 6 3 3 5 3 4 5  3  3 1 5 3 3 4 7  7  7 3 7  7
U 7 8 5 3 8 4 5 3 3 4 13 8  1 3 4 5 4 4 10 7  1 2 2  2
U 9 2 2 5 7 6 9 3 2 4 3  2  5 2 1 2 2 4 5  7  4 3 1  12
U 2 7 1 4 6 7 8 4 4 3 10 5  5 2 4 3 3 1 8  11 2 3 7  7
U 7 7 6 6 1 4 9 5 3 1 14 6  5 3 1 3 3 1 12 1  3 7 7  7
U 4 4 3 7 3 2 8 9 3 1 9  10 6 4 5 3 1 4 10 8  8 5 8  2
A 8 9 6 5 6 4 3 4 4 1 9  1  7 7 2 5 4 1 1  1  5 1 10 2
U 9 5 6 1 2 6 3 3 3 2 8  7  1 5 3 8 1 3 1  8  3 5 1  4
U 8 1 1 5 8 6 3 3 4 3 1  10 3 1 2 3 4 4 5  10 4 5 7  9
A 7 2 3 4 1 3 2 3 3 3 7  1  7 7 2 3 3 4 5  1  5 5 7  9
U 9 3 1 1 2 3 9 8 3 1 13 13 7 1 2 2 3 4 10 3  1 1 10 1
U 2 9 6 1 3 4 3 2 4 3 2  1  4 3 8 1 4 3 9  5  4 2 1  10
U 2 1 1 4 4 7 5 8 3 4 10 13 5 4 4 4 4 3 12 2  3 7 2  12
U 7 7 6 6 3 3 9 3 4 3 14 14 2 1 2 2 1 4 9  1  5 8 4  10
U 1 3 6 5 5 4 9 4 3 4 13 1  2 3 1 2 1 3 1  3  5 3 2  1
U 9 2 6 6 3 4 3 4 2 4 14 9  5 2 4 4 1 1 12 7  5 5 11 7
U 3 3 5 5 8 4 6 5 4 3 2  13 7 1 1 2 3 2 10 7  3 4 7  10
U 4 3 4 5 7 7 8 8 3 3 8  13 3 4 3 2 4 1 1  12 1 3 10 7
U 3 8 1 1 3 8 8 3 4 4 13 12 1 4 5 7 1 4 1  8  3 2 3  3
U 7 8 5 7 7 3 3 3 4 3 14 5  5 1 8 5 4 4 12 12 5 5 10 10
A 7 2 5 4 1 3 3 9 4 3 13 9  2 3 6 5 4 4 1  10 5 2 1  10
U 7 2 4 5 6 1 1 2 4 4 10 8  4 5 5 4 1 1 6  9  2 7 2  12
U 3 3 4 2 7 3 8 3 4 4 14 12 3 2 5 4 3 3 9  3  2 1 12 12
A 2 3 4 1 4 3 3 3 4 4 6  14 1 1 2 2 1 3 3  1  2 8 2  7
U 5 9 3 1 7 4 3 4 2 4 9  8  5 7 3 1 1 3 9  9  2 5 1  9
U 8 5 6 5 3 7 4 4 4 3 10 9  7 5 2 8 4 1 7  8  2 7 12 1
U 9 8 5 5 7 3 6 5 1 3 13 5  2 2 8 7 3 3 9  12 1 3 4  1
A 7 8 5 2 3 5 3 9 3 3 12 5  1 1 1 2 1 4 7  2  5 3 6  1
A 5 4 1 1 3 7 4 5 3 3 14 13 7 3 3 1 4 3 1  8  3 3 2  9
U 8 9 3 2 7 3 8 9 4 1 1  12 5 4 4 6 3 4 2  7  5 2 3  10
A 9 2 3 5 3 3 2 3 2 3 14 13 6 1 3 1 4 3 3  2  3 1 1  7
A 2 5 7 5 6 7 9 4 3 4 14 13 5 1 2 3 4 4 2  10 3 1 12 12
U 7 2 3 1 1 3 4 4 3 4 2  8  5 3 4 6 3 3 10 12 8 3 2  1
A 7 5 1 5 3 3 9 2 3 3 10 6  1 7 2 4 4 4 10 9  1 8 7  3
U 3 2 5 5 4 3 3 5 1 3 1  1  5 2 1 2 3 3 10 3  3 3 10 4
A 3 2 5 5 8 5 3 7 4 3 2  14 5 5 3 3 3 4 11 1  6 2 1  10
A 2 7 5 5 3 2 9 4 3 3 1  7  7 5 4 7 4 1 12 7  2 3 12 9
A 5 7 2 3 7 3 3 3 3 4 9  2  4 1 2 7 1 4 6  1  2 1 7  7
U 7 4 3 4 5 3 3 8 3 3 2  8  4 6 7 7 4 1 3  1  2 4 12 1
U 7 8 5 4 4 7 9 9 4 3 5  13 7 1 4 4 4 4 9  8  8 3 3  10
U 2 8 4 5 3 7 3 4 3 3 8  14 6 4 6 2 3 4 7  1  3 3 3  10
U 6 8 1 3 6 7 5 4 3 4 1  12 3 7 8 4 3 4 12 12 4 7 12 6
A 8 7 3 1 3 6 4 4 3 3 4  10 6 5 8 1 1 4 1  10 2 2 5  2
U 2 8 6 6 4 8 4 3 4 3 9  1  1 1 2 3 4 4 2  6  2 3 9  7
```

```
U 9 8 4 3 7 3 8 4 4 3 8  8  6 6 4 5 3 4 5  5  1 8 10 1
U 9 3 5 1 8 6 5 3 3 2 13 2  3 5 8 2 1 3 1  10 3 3 10 12
U 2 9 1 6 7 4 9 9 4 1 8  1  3 2 5 8 4 4 3  1  3 3 12 7
U 8 8 6 2 3 2 2 4 3 4 6  12 3 1 7 2 4 4 5  9  2 3 1  10
;
```

This data set contains 12 markers. Suppose you are interested in testing three of the marker loci at a time for association with the trait (status in this case: “A” for affected or “U” for unaffected with a particular disease) over all of their haplotypes. That is, assuming the markers are numbered in the order in which they appear on the chromosome, haplotypes at marker loci 1 through 3 are analyzed, then haplotypes at marker loci 4 through 6 are analyzed, and so on. These tests can be performed in addition to, or in place of, single-marker case-control tests (see Chapter 4 for more information). In order to reduce the amount of SAS code needed for this analysis, a SAS macro can be used as follows:

```
%macro hap_trait;
 %do firsta=1 %to 19 %by 6;
  %let lasta=%eval(&firsta+5);
  %let firstm=%eval((&firsta+1)/2);
  %let lastm=%eval(&lasta/2);

  proc haplotype data=gaw noprint;
     var a&firsta-a&lasta;
     trait status;
  run;

 %end;
%mend;
%hap_trait
```

Since the NOPRINT option is specified, this code produces only the “Test for Marker-Trait Association” table each of the four times PROC HAPLOTYPE is invoked.

Output 7.4.1 Testing for Marker-Trait Associations Using Haplotypes

```
                            The HAPLOTYPE Procedure

                        Test for Marker-Trait Association

  Trait    Trait          Num                              Chi-      Pr >
 Number    Value          Obs       DF       LogLike      Square     ChiSq

      1    U               36      156    -245.18487
      2    A               14       68     -69.90500
           Combined        50      181    -355.16139     80.1430    0.0005
```

Output 7.4.1 *continued*

```
                         The HAPLOTYPE Procedure

                   Test for Marker-Trait Association

 Trait    Trait          Num                              Chi-       Pr >
Number    Value          Obs        DF        LogLike    Square     ChiSq

     1    U               36       140     -236.78471
     2    A               14        62      -78.22280
          Combined        50       162     -349.30084    68.5867    0.0033

                         The HAPLOTYPE Procedure

                   Test for Marker-Trait Association

 Trait    Trait          Num                              Chi-       Pr >
Number    Value          Obs        DF        LogLike    Square     ChiSq

     1    U               36       119     -242.53993
     2    A               14        56      -68.34854
          Combined        50       139     -348.95917    76.1414    0.0001

                         The HAPLOTYPE Procedure

                   Test for Marker-Trait Association

 Trait    Trait          Num                              Chi-       Pr >
Number    Value          Obs        DF        LogLike    Square     ChiSq

     1    U               36       180     -268.92245
     2    A               14        75      -85.15400
          Combined        50       233     -395.70275    83.2526    <.0001
```

Output 7.4.1 displays the four tables that are created by this macro. The first corresponds to testing the three-locus haplotypes at the first three marker loci with the TRAIT variable, the second corresponds to the second set of three markers, and so on. From the LRTs that are performed and summarized in the output, it can be concluded that out of the four sets of marker loci tested, the haplotypes at markers 10, 11, and 12 show the most significant association with the trait variable status. The chi-square statistic for testing the haplotypes at these markers for association with disease status is calculated as $83.2526 = 2(-268.92245 - 85.15400 + 395.70275)$ with degrees of freedom $22 = 180 + 75 - 233$, which has a p-value < 0.0001.

Suppose you want to further explore the association between these three markers and the trait. You can also perform tests of association between each individual haplotype at these marker loci and disease status by using the following code:

```
ods output haplotype.haptraittest=outhap;
proc haplotype data=gaw noprint seed=123;
   var a19-a24;
   trait status / testall perms=100;
run;
```

```
proc print data=outhap(obs=20) noobs;
   title 'The HAPLOTYPE Procedure';
   title2 ' ';
   title3 'Tests for Haplotype-Trait Association';
run;
```

The TESTALL option indicates that a test for trait association should be performed on each haplotype by using a chi-square test statistic, which is performed by default. In addition, since the PERMS=100 option is included, an empirical *p*-value is calculated. Because of the number of alleles at each marker in this example, this option increases the computation time substantially, even with this small number of permutations.

Output 7.4.2 Using the TESTALL Option on Markers 10-12

```
                         The HAPLOTYPE Procedure

                  Tests for Haplotype-Trait Association

                                           Combined                 Prob    Prob
Number  Haplotype  Trait1Freq  Trait2Freq    Freq         ChiSq    ChiSq   Exact

     1   1-1-2      0.00000     0.03571    0.00000            0   1.0000  1.0000
     2   1-1-7      0.00000     0.00000    0.01000       1.0101   0.3149  0.3900
     3   1-1-10     0.00000     0.00000    0.01950       1.9883   0.1585  0.3700
     4   1-2-1      0.00000     0.01786    0.03000       2.3686   0.1238  0.1800
     5   1-2-2      0.00000     0.05357    0.01000       6.0967   0.0135  0.0600
     6   1-2-3      0.00000     0.00000    0.00000     0.001666   0.9674  0.6100
     7   1-2-5      0.00000     0.00000    0.01000       1.0101   0.3149  0.2200
     8   1-2-7      0.00000     0.05357    0.00000            0   1.0000  1.0000
     9   1-2-10     0.00000     0.01786    0.00000            0   1.0000  1.0000
    10   1-2-12     0.00000     0.00000    0.00000            0   1.0000  1.0000
    11   1-3-1      0.00694     0.00000    0.00000            0   1.0000  1.0000
    12   1-3-2      0.00000     0.01786    0.01000       0.9019   0.3423  0.4600
    13   1-3-3      0.02777     0.00000    0.02000       0.7934   0.3731  0.7300
    14   1-3-4      0.00000     0.00000    0.00000            0   1.0000  1.0000
    15   1-3-7      0.04167     0.00000    0.02045       2.2035   0.1377  0.1300
    16   1-3-9      0.00000     0.01786    0.00000            0   1.0000  1.0000
    17   1-3-10     0.00000     0.00000    0.00000    7.8011E-8   0.9998  0.9700
    18   1-3-12     0.01389     0.00000    0.01006       0.3905   0.5320  0.8600
    19   1-4-1      0.01389     0.00000    0.00000            0   1.0000  1.0000
    20   1-4-12     0.00000     0.00000    0.00000            0   1.0000  1.0000
```

Output 7.4.2 displays the table “Test for Haplotype-Trait Association” as a SAS data set by using the ODS system in order to show only the first 20 rows. The table contains haplotypes at markers 10, 11, and 12 and their estimated frequencies among individuals with the first trait value, individuals with the second trait value, and all individuals. The chi-square statistic testing whether the frequencies between the two trait groups are significantly different is also shown, along with its 1 df *p*-value. Note that none of the haplotypes shown here have an association with disease status significant at the 0.05 level according to the approximations of exact *p*-values.

Example 7.5: Creating a Data Set for a Regression Model

Another approach to testing haplotypes for association with a phenotype uses a regression model, which can be more powerful than the omnibus chi-square test performed in PROC HAPLOTYPE (Schaid et al. 2002; Zaykin et al. 2002). The output data set produced by PROC HAPLOTYPE can easily be transformed into one that can be used by one of the regression procedures offered by SAS/STAT. This approach can be used for quantitative traits as well as binary or ordinal traits.

Here is an example data set that can be analyzed using PROC HAPLOTYPE:

```
data alleles;
   input (a1-a6) ($) disease;
   datalines;
A  a  B  B  c  C  1
A  A  B  b  c  C  1
a  A  B  b  c  c  0
A  A  B  B  c  C  1
A  A  b  B  c  C  1
A  A  B  b  C  c  0
A  a  b  B  C  c  1
A  A  b  B  C  c  1
A  a  B  B  c  c  1
a  a  B  b  c  c  0
A  A  B  B  C  C  1
A  A  B  B  c  c  1
a  A  b  b  c  c  0
A  A  B  B  c  c  1
A  A  b  b  c  c  0
A  A  b  B  c  C  0
A  A  B  b  c  C  1
A  a  b  B  c  c  1
A  a  B  B  c  C  1
A  A  b  b  C  C  0
A  A  B  B  C  C  1
A  A  b  B  C  c  1
A  A  b  B  c  C  1
a  A  B  b  C  c  0
A  a  B  B  C  C  0
A  A  B  B  C  c  1
A  A  B  b  C  c  0
A  A  B  B  c  C  1
a  A  B  b  C  C  1
A  a  B  b  C  c  1
A  A  B  b  c  C  1
A  a  B  B  c  c  1
A  A  B  b  C  c  1
a  A  B  b  C  c  1
A  A  B  b  C  C  1
A  a  B  B  C  C  1
a  A  B  b  C  c  0
a  A  b  B  C  C  0
A  A  B  b  c  C  1
```

```
a  A  B  b  c  c  0
A  A  B  B  C  C  0
A  A  B  B  c  c  1
A  a  B  B  C  c  1
;
```

The following code creates an output data set containing individuals' probabilities of having particular haplotype pairs, with the ID statement and OUTID option indicating that this data set includes the disease variable from the input data set and a unique identifier for each individual assigned by PROC HAPLOTYPE, respectively. An omnibus test for association between the three markers and disease status is also performed.

```
proc haplotype data=alleles out=out outid;
   var a1-a6;
   trait disease;
   id disease;
run;
```

This code executes the omnibus marker-trait association test whose p-value is given by the chi-square distribution.

Output 7.5.1 Testing for an Overall Marker-Trait Association

```
                             The HAPLOTYPE Procedure

                      Tests for Haplotype-Trait Association

                             The HAPLOTYPE Procedure

                        Test for Marker-Trait Association

 Trait    Trait                Num                                Chi-      Pr >
Number    Value                Obs        DF        LogLike     Square     ChiSq

     1    1                     29         7      -68.11558
     2    0                     14         7      -37.28544
          Combined              43         7     -115.48338    20.1647    0.0052
```

Output 7.5.1 shows that there is a significant overall association between the markers and the trait, disease status. However, the more powerful score test for regression can be implemented by using the following code to perform a test for additive effects of the marker haplotypes.

```
data out1;
   set out;
   haplotype=tranwrd(haplotype1,'-','_');
run;

data out2;
   set out;
   haplotype=tranwrd(haplotype2,'-','_');
run;

data outnew;
   set out1 out2;
run;
```

```
proc sort data=outnew;
   by haplotype;
run;

data outnew2;
   set outnew;
   lagh=lag(haplotype);
   if haplotype ne lagh then num+1;
   hapname=compress("H"||num,' ');
run;

proc sort data=outnew2;
   by _id_ haplotype;
run;

data outt;
   set outnew2;
   by _id_ haplotype;
   if first.haplotype then totprob=prob/2;
   else totprob+prob/2;
   if last.haplotype;
run;

proc transpose data=outt out=outreg(drop=_NAME_) ;
   id hapname;
   idlabel haplotype;
   var totprob;
   by _id_ disease;
run;

data htr;
   set outreg;
   array h{8};
   do i=1 to 8;
    if h{i}=. then h{i}=0;
   end;
   keep _id_ disease h1-h8;
run;

proc print data=htr noobs round label;
run;

proc logistic data=htr descending;
   model disease = h1-h8 / selection=stepwise;
run;
```

This SAS code produces a data set htr from the output data set of PROC HAPLOTYPE that contains the variables needed to be able to perform a regression analysis. There is now one column for each possible haplotype in the sample, with each column containing the haplotype's frequency, or probability, within an individual.

The data set shown in Output 7.5.2 can now be used in one of the regression procedures offered by SAS/STAT.

In this example, since the trait is binary, the LOGISTIC procedure can be used to perform a regression on the variable disease. The REG procedure could be used in a similar manner to analyze a quantitative trait.

Output 7.5.2 Regression Data Set

```
                            The HAPLOTYPE Procedure

                      Tests for Haplotype-Trait Association

Individual
    ID      disease  A_B_C  A_B_c  a_B_C  a_B_c  A_b_C  A_b_c  a_b_c  a_b_C

     1         1      0.29   0.21   0.21   0.29   0.00   0.00   0.00   0.00
     2         1      0.27   0.23   0.00   0.00   0.23   0.27   0.00   0.00
     3         0      0.00   0.27   0.00   0.23   0.00   0.23   0.27   0.00
     4         1      0.50   0.50   0.00   0.00   0.00   0.00   0.00   0.00
     5         1      0.27   0.23   0.00   0.00   0.23   0.27   0.00   0.00
     6         0      0.27   0.23   0.00   0.00   0.23   0.27   0.00   0.00
     7         1      0.22   0.00   0.13   0.15   0.15   0.13   0.22   0.00
     8         1      0.27   0.23   0.00   0.00   0.23   0.27   0.00   0.00
     9         1      0.00   0.50   0.00   0.50   0.00   0.00   0.00   0.00
    10         0      0.00   0.00   0.00   0.50   0.00   0.00   0.50   0.00
    11         1      1.00   0.00   0.00   0.00   0.00   0.00   0.00   0.00
    12         1      0.00   1.00   0.00   0.00   0.00   0.00   0.00   0.00
    13         0      0.00   0.00   0.00   0.00   0.00   0.50   0.50   0.00
    14         1      0.00   1.00   0.00   0.00   0.00   0.00   0.00   0.00
    15         0      0.00   0.00   0.00   0.00   0.00   1.00   0.00   0.00
    16         0      0.27   0.23   0.00   0.00   0.23   0.27   0.00   0.00
    17         1      0.27   0.23   0.00   0.00   0.23   0.27   0.00   0.00
    18         1      0.00   0.27   0.00   0.23   0.00   0.23   0.27   0.00
    19         1      0.29   0.21   0.21   0.29   0.00   0.00   0.00   0.00
    20         0      0.00   0.00   0.00   0.00   1.00   0.00   0.00   0.00
    21         1      1.00   0.00   0.00   0.00   0.00   0.00   0.00   0.00
    22         1      0.27   0.23   0.00   0.00   0.23   0.27   0.00   0.00
    23         1      0.27   0.23   0.00   0.00   0.23   0.27   0.00   0.00
    24         0      0.22   0.00   0.13   0.15   0.15   0.13   0.22   0.00
    25         0      0.50   0.00   0.50   0.00   0.00   0.00   0.00   0.00
    26         1      0.50   0.50   0.00   0.00   0.00   0.00   0.00   0.00
    27         0      0.27   0.23   0.00   0.00   0.23   0.27   0.00   0.00
    28         1      0.50   0.50   0.00   0.00   0.00   0.00   0.00   0.00
    29         1      0.01   0.00   0.49   0.00   0.49   0.00   0.00   0.01
    30         1      0.22   0.00   0.13   0.15   0.15   0.13   0.22   0.00
    31         1      0.27   0.23   0.00   0.00   0.23   0.27   0.00   0.00
    32         1      0.00   0.50   0.00   0.50   0.00   0.00   0.00   0.00
    33         1      0.27   0.23   0.00   0.00   0.23   0.27   0.00   0.00
    34         1      0.22   0.00   0.13   0.15   0.15   0.13   0.22   0.00
    35         1      0.50   0.00   0.00   0.00   0.50   0.00   0.00   0.00
    36         1      0.50   0.00   0.50   0.00   0.00   0.00   0.00   0.00
    37         0      0.22   0.00   0.13   0.15   0.15   0.13   0.22   0.00
    38         0      0.01   0.00   0.49   0.00   0.49   0.00   0.00   0.01
    39         1      0.27   0.23   0.00   0.00   0.23   0.27   0.00   0.00
    40         0      0.00   0.27   0.00   0.23   0.00   0.23   0.27   0.00
    41         0      1.00   0.00   0.00   0.00   0.00   0.00   0.00   0.00
    42         1      0.00   1.00   0.00   0.00   0.00   0.00   0.00   0.00
    43         1      0.29   0.21   0.21   0.29   0.00   0.00   0.00   0.00
```

Output 7.5.3 shows two of the tables produced by PROC LOGISTIC. The first one displays the test of the global null hypothesis, ¸ = **0**. You can see that the score test indicates a significant association between the haplotypes at the three markers and disease status. In particular, the second table shows that as a result of the stepwise selection, the haplotype H8 (a-b-c) has a statistically significant effect on disease status. This is an example of how a regression analysis can be used to detect association in a similar manner to the LRT implemented by PROC HAPLOTYPE.

Output 7.5.3 PROC LOGISTIC Output

```
                         The HAPLOTYPE Procedure

                  Tests for Haplotype-Trait Association

                          The LOGISTIC Procedure

                 Testing Global Null Hypothesis: BETA=0

         Test                 Chi-Square       DF     Pr > ChiSq

         Likelihood Ratio         6.1962        1         0.0128
         Score                    6.3995        1         0.0114
         Wald                     4.9675        1         0.0258


                Analysis of Maximum Likelihood Estimates

                                    Standard          Wald
   Parameter    DF     Estimate        Error    Chi-Square    Pr > ChiSq

   Intercept     1       1.1986       0.4058        8.7224        0.0031
   H8            1      -6.3249       2.8378        4.9675        0.0258
```

Example 7.6: Using the Tall-Skinny Data Format

This example uses the data from the example Testing for Marker-Trait Associations, with the data now in the tall-skinny format. When this format is used, BY groups can be created in order to estimate haplotype frequencies in nonoverlapping windows of marker loci instead of using a macro as shown in the other example; here four sets of three loci are examined, but in general, loci with the same value of the BY variable are included in the same analysis, so sets of varying sizes can be used as well. The following DATA step is used to input the data.

```
data gaw_tall;
   input hap_win markername $ id status $ allele1 allele2;
   datalines;
 1        marker1       1      U          8          4
 1        marker1       2      U          5          9
 1        marker1       3      A          8          2
 1        marker1       4      U          7          8
 1        marker1       5      U          9          2
 1        marker1       6      U          2          7
 1        marker1       7      U          7          7
 1        marker1       8      U           4          4

   ... more lines ...

 4        marker12     43      U           3         10
 4        marker12     44      U          12          6
 4        marker12     45      A           5          2
 4        marker12     46      U           9          7
 4        marker12     47      U          10          1
 4        marker12     48      U          10         12
 4        marker12     49      U          12          7
 4        marker12     50      U           1         10
;
```

Using the options TALL, MARKER=, and INDIV=, along with the BY statement to indicate the BY variable representing haplotype windows, the same analysis shown in Testing for Marker-Trait Associations can be performed on the 50 individuals typed at 12 markers by using the following code, where sets of three loci at a time are tested for an association with the trait.

```
proc haplotype data=gaw_tall tall marker=markername indiv=id noprint;
   var allele1 allele2;
   by hap_win;
   trait status;
 run;
```

This produces the same numerical results as those shown in Output 7.4.1.

References

Clayton, D. (2002), "Choosing a Set of Haplotype Tagging SNPs from a Larger Set of Diallelic Loci," `http://www-gene.cimr.cam.ac.uk/clayton/software/stata/htSNP/htsnp.pdf`.

Excoffier, L. and Slatkin, M. (1995), "Maximum-Likelihood Estimation of Molecular Haplotype Frequencies in a Diploid Population," *Molecular Biology and Evolution*, 12, 921–927.

Fallin, D., Cohen, A., Essioux, L., Chumakov, I., Blumenfeld, M., Cohen, D., and Schork, N. J. (2001), "Genetic Analysis of Case/Control Data Using Estimated Haplotype Frequencies: Application to APOE Locus Variation and Alzheimer's Disease," *Genome Research*, 11, 143–151.

Fallin, D. and Schork, N. J. (2000), "Accuracy of Haplotype Frequency Estimation for Biallelic Loci, via the Expectation-Maximization Algorithm for Unphased Diploid Genotype Data," *American Journal of Human Genetics*, 67, 947–959.

Hawley, M. E. and Kidd, K. K. (1995), "HAPLO: A Program Using the EM Algorithm to Estimate the Frequencies of Multi-site Haplotypes," *Journal of Heredity*, 86, 409–411.

Lab of Statistical Genetics at Rockefeller University (2001), *User's Guide to the EH Program*, New York: Lab of Statistical Genetics at Rockefeller University.
URL `http://linkage.rockefeller.edu/ott/eh.htm`

Lin, S., Cutler, D. J., Zwick, M. E., and Chakravarti, A. (2002), “Haplotype Inference in Random Population Samples,” *American Journal of Human Genetics*, 71, 1129–1137.

Long, J. C., Williams, R. C., and Urbanek, M. (1995), “An E-M Algorithm and Testing Strategy for Multiple-Locus Haplotypes,” *American Journal of Human Genetics*, 56, 799–810.

Schaid, D. J., Rowland, C. M., Tines, D. E., Jacobson, R. M., and Poland, G. A. (2002), “Score Tests for Association between Traits and Haplotypes When Linkage Phase Is Ambiguous,” *American Journal of Human Genetics*, 70, 425–434.

Stephens, M., Smith, N. J., and Donnelly, P. (2001), “A New Statistical Method for Haplotype Reconstruction from Population Data,” *American Journal of Human Genetics*, 68, 978–989.

Wijsman, E. M., Almasy, L., Amos, C. I., Borecki, I., Falk, C. T., King, T. M., Martinez, M. M., Meyers, D., Neuman, R., Olson, J. M., Rich, S., Spence, M. A., Thomas, D. C., Vieland, V. J., Witte, J. S., and MacCluer, J. W. (2001), “Analysis of Complex Genetic Traits: Applications to Asthma and Simulated Data,” *Genetic Epidemiology*, 21(Suppl. 1), S1–S853.

Zaykin, D. V., Westfall, P. H., Young, S. S., Karnoub, M. A., Wagner, M. J., and Ehm, M. G. (2002), “Testing Association of Statistically Inferred Haplotypes with Discrete and Continuous Traits in Samples of Unrelated Individuals,” *Human Heredity*, 53, 79–91.

Zhao, J. H., Curtis, D., and Sham, P. C. (2000), “Model-Free Analysis and Permutation Tests for Allelic Associations,” *Human Heredity*, 50, 133–139.

Chapter 8
The HTSNP Procedure (Experimental)

Contents

Overview: HTSNP Procedure

Single nucleotide polymorphism (SNP) is the most abundant form of genetic variation and accounts for about 90% of human DNA polymorphism. There is roughly one SNP per 1 kilobase in the human genome. Studies of human haplotype variations that use SNPs over large genomic regions suggest the presence of discrete blocks with limited haplotype diversity punctuated by recombination hot spots. The intrablock linkage disequilibrium (LD) decreases only gradually with distance, while the interblock LD decays much more rapidly. Within each block, because of high LD, some allele(s) might always be coexistent with a particular allele at another locus such that (1) little haplotype diversity exists in the block, and (2) not all SNPs will be essential in characterizing the haplotype structure in the block. Therefore, the most common haplotypes could usually be captured by a small subset of SNPs, termed *haplotype tagging SNPs (htSNPs)* by Johnson et al. (2001).

The selection of such a SNP subset that distinguishes all haplotypes, however, is known as the *minimum test set* problem and is NP-complete. The search space of choosing k SNPs out of M is $\binom{M}{k} = \frac{M!}{k!(M-k)!}$, for which enumerating all possible k-SNP combinations becomes impractical even for moderate numbers of M and k. The HTSNP procedure implements some heuristic algorithms for fast identification of an optimal subset of SNPs without mining through all possible combinations. An exhaustive search algorithm throughout the $\binom{M}{k}$ search space is also provided in PROC HTSNP.

Getting Started: HTSNP Procedure

Example

The following haplotypes from markers at the *CTLA4* locus (Johnson et al. 2001) can be read into a SAS data set as follows:

```
data ctla4;
   input (m1-m12)($) freq;
   datalines;
C T A A G C C A C C A G 0.333
T T A G G C C G C T G G 0.224
T C A G G C C G C T G G 0.058
T T A A G C C G C T G G 0.020
C T A A G T C A C C A G 0.080
C T A G G T C A C C A G 0.017
C T A G G C C A C C A G 0.045
T T A G G C C A C C A G 0.018
C T G G A C T A T C G A 0.086
C T G G A C C A T C G A 0.054
C T G G A C C A C C G A 0.021
;
```

You can now use PROC HTSNP to search a subset of markers that explains most of the haplotype diversity in this sample. The following statements perform the search:

```
proc htsnp data=ctla4 size=5 method=im
           cutoff=0.05 seed=244 conv=0.99;
   var m1-m12;
   freq freq;
run;
```

The iterative maximization algorithm is selected as the search method with the METHOD=IM option. The SIZE=5 option indicates that only subsets containing exactly five SNPs are considered in the search. All haplotypes in the data set with a frequency below 0.05 are excluded from the search process because the CUTOFF=0.05 option was specified. The search continues until the convergence criterion of 0.99 is met as specified in the CONV= option. The iterative maximization algorithm randomly selects an initial set of markers, so using different seeds can produce different results.

The results from the procedure are shown in Figures 8.1 and 8.2.

Figure 8.1 Marker Summary for PROC HTSNP

```
                         The HTSNP Procedure

                           Marker Summary

          Locus     Allele       Frequency    Diversity

          m1        C               0.6653       0.4454
                    T               0.3347       0.4454

          m2        C               0.0607       0.1140
                    T               0.9393       0.1140

          m3        A               0.8316       0.2801
                    G               0.1684       0.2801

          m4        A               0.4529       0.4956
                    G               0.5471       0.4956

          m5        A               0.1684       0.2801
                    G               0.8316       0.2801

          m6        C               0.8985       0.1823
                    T               0.1015       0.1823

          m7        C               0.9100       0.1637
                    T               0.0900       0.1637

          m8        A               0.6841       0.4322
                    G               0.3159       0.4322

          m9        C               0.8536       0.2500
                    T               0.1464       0.2500

          m10       C               0.6841       0.4322
                    T               0.3159       0.4322

          m11       A               0.5157       0.4995
                    G               0.4843       0.4995

          m12       A               0.1684       0.2801
                    G               0.8316       0.2801
```

Figure 8.1 displays the summary of the marker loci for this sample. This includes the frequency of each allele and the gene diversity at each marker.

Figure 8.2 htSNP Evaluation

```
                           htSNP Evaluation

    Rank     HTSNP1     HTSNP2     HTSNP3     HTSNP4     HTSNP5         PDE

       1     m2         m3         m6         m7         m8          1.0000
```

Figure 8.2 displays the ODS table containing the set of five SNPs that were selected as the htSNPs; these five markers correspond to those selected by Johnson et al. (2001).

Syntax: HTSNP Procedure

The following statements are available in PROC HTSNP.

PROC HTSNP < *options* > ;
  BY *variables* ;
  FREQ *variable* ;
  VAR *variables* ;

Items within angle brackets (< >) are optional, and statements following the PROC HTSNP statement can appear in any order. Only the VAR statement is required.

PROC HTSNP Statement

PROC HTSNP < *options* > ;

You can specify the following options in the PROC HTSNP statement.

BEST=*number*

specifies the number of the best selections displayed in the “htSNP Evaluation” table during an exhaustive or simulated annealing search process when METHOD=EX or SA, respectively, is specified. The *number* must be a positive integer. By default, only one best selection is reported. Note that sets of SNPs with the same value of the criterion measure as the last displayed set(s) are not necessarily all shown since *number* indicates the number of sets actually displayed. If *number* is greater than the number of possible sets when METHOD=EX or greater than the number of sets examined when METHOD=SA, there are fewer than *number* sets displayed.

CONV=*number*

specifies the convergence criterion for search of htSNPs, where $0 < number \leq 1$. The search process is stopped when the haplotype criterion is greater than or equal to *number* specified in the CONV= option. The default value is 0.90. When METHOD=SA or METHOD=EX is specified, the CONV= option is ignored and the searching continues until the annealing schedule is finished or the whole search space is traversed.

CRITERION=PDE | RSQH

CRIT=PDE | RSQH

indicates the criterion to use for evaluating candidate sets of htSNPs. By default or when CRITERION=PDE is specified, the proportion of diversity explained (PDE) is used (Clayton 2002). When CRITERION=RSQH, Stram et al.’s R_h^2 is used to measure haplotype richness (2003). See the section “Evaluating Sets of htSNPs” on page 162 for more information about these measures.

CUTOFF=*number*

specifies a lower bound on a haplotype's frequency in order for that haplotype to be included in the search process for sets of htSNPs. The value of *number* must be between 0 and 1. By default, all haplotypes from the sample are included in the search process.

DATA=*SAS-data-set*

names the input SAS data set to be used by PROC HTSNP. The default is to use the most recently created data set.

MAXSIZE=*number*

specifies the maximum number of markers to be included in the subset for incremental search by default or when METHOD=INCR is specified. The number must be a positive integer that is less than or equal to the number of markers specified in the VAR statement. Searching is carried out until convergence is reached according to the convergence criterion, or until *number* of markers have been included in the subset.

METHOD=INCR | INCREMENTAL

METHOD=DECR | DECREMENTAL

METHOD=EX | EXHAUSTIVE

METHOD=IM | ITERMAX

METHOD=SA | SIMANNEAL

indicates the method used for core marker set selection. By default or when METHOD=INCR is specified, the incremental search algorithm is used. When METHOD=DECR, the decremental algorithm is used. When METHOD=EX, the exhaustive search algorithm is used. When METHOD=IM, the iterative maximization algorithm is used. When METHOD=SA, the simulated annealing search algorithm is used. See the section "Search Algorithms" on page 162 for more information about these methods.

NOSUMMARY

NOSUMM

suppresses the display of the "Marker Summary" table.

SCHEDULE=*number*

specifies the number of reconfigurations used in each annealing step when METHOD=SA. The value for *number* must be a positive integer. The default value is 100 × (number of variables specified in the VAR statement).

SEED=*number*

specifies the initial seed for the random number generator used for the sampling of markers. The value for *number* must be an integer; the computer clock time is used if the option is omitted or an integer less than or equal to 0 is specified. For more details about seed values, see *SAS Language Reference: Concepts*.

SIZE=*number*

specifies the size of the subset of markers to select. The value for *number* must be a positive integer that is less than or equal to the number of markers specified in the VAR statement. The SIZE= option must be specified for an exhaustive search, iterative maximization search, and simulated annealing search.

STEP=*number*

specifies the steps used for simulated annealing search when METHOD=SA. The value for *number* must be a positive integer. The default value is 1.

TEMPERATURE=*number*

T=*number*

specifies the temperature used for the simulated annealing search when METHOD=SA is specified. The value for *number* must be a positive number. The default value is 1.

TFACTOR=*number*

specifies the factor by which the temperature is reduced for each annealing step during simulated annealing search when METHOD=SA. The value for *number* must satisfy $0 < number < 1$. The default value is 0.90.

BY Statement

BY *variables* **;**

You can specify a BY statement with PROC HTSNP to obtain separate analyses on observations in groups defined by the BY variables. When a BY statement appears, the procedure expects the input data set to be sorted in the order of the BY variables. The *variables* are one or more variables in the input data set.

If your input data set is not sorted in ascending order, use one of the following alternatives:

- Sort the data by using the SORT procedure with a similar BY statement.
- Specify the BY statement option NOTSORTED or DESCENDING in the BY statement for the HTSNP procedure. The NOTSORTED option does not mean that the data are unsorted but rather that the data are arranged in groups (according to values of the BY variables) and that these groups are not necessarily in alphabetical or increasing numeric order.
- Create an index on the BY variables by using the DATASETS procedure (in Base SAS software).

For more information about the BY statement, see *SAS Language Reference: Concepts*. For more information about the DATASETS procedure, see the *Base SAS Procedures Guide*.

FREQ Statement

FREQ *variable* **;**

The FREQ statement identifies the variable that indicates the frequency for each haplotype. If there is no FREQ statement, the frequency of each distinct haplotype is calculated by dividing its count by the total haplotype count. When a frequency value is missing or negative, the corresponding haplotype is ignored.

VAR Statement

VAR *variables* **;**

The VAR statement identifies the variables, one for each marker, containing the marker alleles that construct the haplotypes. Two or more variables must be specified.

Details: HTSNP Procedure

Statistical Computations

Diversity

Let $f_1, \ldots, f_n$ represent the proportional frequencies of the n unique M-locus haplotypes in the input data set. The locus or allelic diversity $D_1, \ldots, D_M$ for the M individual loci and the overall haplotype diversity D can be calculated as

$$\begin{aligned} D_m &= \sum_{i=1}^{n} \sum_{j=1}^{n} f_i f_j \, \mathrm{I}(h_{im} \neq h_{jm}) \\ D &= \sum_{m=1}^{M} D_m \end{aligned}$$

where h_{im} is the allele of the ith haplotype observed at the mth locus and the indicator function I() equals 1 when true and 0 otherwise (Clayton 2002).

Based on a selected subset of k SNPs, the n observed haplotypes can be partitioned into T distinct groups. Let $\mathcal{T}_t$ represent the set of haplotypes in group $t = 1, \ldots, T$, where each set contains all haplotypes that have identical alleles at the k selected loci. The residual diversity is calculated by Clayton (2002) by summing the within-group diversity over the T groups, again both for the individual loci and over all haplotypes:

$$\begin{aligned} R_m &= \sum_{t=1}^{T} \sum_{i \in \mathcal{T}_t} \sum_{j \in \mathcal{T}_t} f_i f_j \, \mathrm{I}(h_{im} \neq h_{jm}) \\ R &= \sum_{m=1}^{M} R_m \end{aligned}$$

where $m = 1, \ldots, M$. Note that $R_m = 0$ if locus m is one of the k selected SNPs.

Evaluating Sets of htSNPs

One of two criteria for finding the optimal set of htSNPs can be selected with the CRITERION= option. Using the diversity measures previously defined, the proportion of diversity explained (PDE) by a candidate SNP set can be calculated to evaluate the goodness of the set (Clayton 2002):

$$\mathrm{PDE} = 1 - \frac{R}{D}$$

The higher (that is, closer to 1) the value of PDE is, the better the set of htSNPs is for explaining the diversity among the haplotypes.

Alternatively, the approach of Stram et al. (2003) is implemented when CRITERION=RSQH. For these computations, define $\delta_h(\mathcal{H}_i)$ to be the actual number of copies of haplotype h that an individual with the M-locus haplotype pair $\mathcal{H}_i$ (usually unknown) and genotype G_i carries. Assuming Hardy-Weinberg equilibrium, this can be estimated as

$$E[\delta_h(\mathcal{H}_i)|G_i] = \frac{\sum_{j \in H_i} \delta_h(h_j, h_j^{ci}) f_j f_j^{ci}}{\sum_{j \in H_i} f_j f_j^{ci}}$$

where H_i is the set of haplotype pairs, h_j and its complement h_j^{ci}, compatible with genotype G_i. Then R_h^2 can be defined as follows for each haplotype h:

$$R_h^2 = \frac{\mathrm{Var}\{E[\delta_h(\mathcal{H}_i)|G_i]\}}{2f_h(1-f_h)} = \frac{\sum_i \{[E(\delta_h(\mathcal{H}_i)|G_i)]^2 \Pr(G_i)\} - 4f_h^2}{2f_h(1-f_h)}$$

with G_i representing each possible k-locus genotype at the selected SNPs and $\Pr(G_i) = \sum_{j \in H_i} f_j f_j^{ci}$. The set of k SNPs with the highest (that is, closest to 1) value of $\min_h R_h^2$ is selected as the best set of htSNPs, for it optimizes the predictability of the common haplotypes (Stram et al. 2003).

Search Algorithms

Incremental Search

The incremental search algorithm starts with finding a first marker that has maximum locus richness and then goes through the remaining markers to find the next one that brings the greatest increase in the criterion measure, PDE or R_h^2. The selected markers are kept and the search process is continued using the remaining ones, one marker being added at a time, until a convergence criterion is met.

Decremental Search

The decremental search operates in an opposite manner from the incremental search. Starting with all M markers, one marker that causes the smallest loss in the criterion measure is excluded each time and the rest of the markers are kept. The exclusion process is continued until the criterion measure falls below a predefined criterion; the last set with the measure above the criterion is reported.

Iterative Maximization Search

The iterative maximization search (Gouesnard et al. 2001) is a fast algorithm for choosing an optimal k-subset from M accessions. The algorithm starts from a random selection of k markers for which all the core collections of size $k-1$ are tested. The subset with the highest criterion measure is retained. Among the other $M-k$ markers, one that brings the greatest increase in the goodness criterion is selected and a new k-locus set is obtained. Exclusion and inclusion of one marker in the new k-locus set are repeated until convergence. Each iteration needs to evaluate the criterion measure k times for $k-1$ markers and $M-k$ times for k markers.

Simulated Annealing Search

Simulation annealing (Kirkpatrick, Gelatt, and Vecchi 1983) has been adopted in many combinatorial optimization problems. The global optimum could be approximated with simulated annealing by using a proper annealing schedule. Starting from a selection of k markers (the selection could be random or obtained from a previously mentioned algorithm), one marker is randomly swapped with another from the unselected markers. The change of haplotype goodness is evaluated using an energy function for the marker exchange. Acceptance of the exchange is judged with the Metropolis criterion (Metropolis et al. 1953), and

$$\Pr\{\text{new point is accepted}\} = \begin{cases} 1, & \Delta \le 0 \\ \exp(-\Delta/T), & \Delta > 0 \end{cases}$$

where Δ is the change of energy function and T is the annealing temperature.

Exhaustive Search

An exhaustive search of k markers from M involves traversal of all $\binom{M}{k}$ possible selections once and only once. The traversal is implemented in lexicographical order (Nijenhuis and Wilf 1978). Let $S_i = (s_1, s_2, \ldots, s_k)$ denote a selection i, where $1 \le s_{ij} \le M$ is the index of the jth element in selection i. Lexicographical traversal of all k subsets then starts with $(1, 2, \ldots, k-1, k)$, $(1, 2, \ldots, k-1, k+1)$, and ends with $(M-k+1, M-k+2, \ldots, M-1, M)$.

Missing Values

An M-locus haplotype is considered to be partially missing if any, but not all, of the alleles are missing. A haplotype that is missing all the alleles is dropped from any analysis.

Displayed Output

This section describes the displayed output from PROC HTSNP. See the “ODS Table Names” on page 164 section for details about how this output interfaces with the Output Delivery System.

Marker Summary

The “Marker Summary” table lists the following information for each marker allele:

- Locus, the name of the marker locus

- Allele, the allele
- Frequency, the frequency of the allele
- Diversity, the gene diversity of the marker

HTSNP Evaluation

The “htSNP Evaluation” table displays the best set(s) of htSNPs according to the criterion specified in the CRITERION= option.

ODS Table Names

PROC HTSNP assigns a name to each table it creates, and you must use this name to reference the table when using the Output Delivery System (ODS). These names are listed in Table 8.1.

Table 8.1 ODS Tables Created by the HTSNP Procedure

ODS Table Name	Description	Statement or Option
MarkerSummary	Marker Summary	default
HTSNPEvaluation	htSNP Evaluation	default

Example: HTSNP Procedure

Example 8.1: Using the HAPLOTYPE and HTSNP Procedures Together

Before using PROC HTSNP, you might need to run PROC HAPLOTYPE (see Chapter 7, “The HAPLOTYPE Procedure,” for more details) if you have data with unknown phase in order to estimate the haplotype frequencies. This example demonstrates how output from PROC HAPLOTYPE can be manipulated to be in the appropriate form for an input data set for PROC HTSNP.

The following data set contains 150 individuals with genotypes at 13 SNPs that were simulated to mimic the frequencies of SNPs in the *CASP8* gene (Johnson et al. 2001).

```
data casp8;
   input id (m1-m13) ($);
   datalines;
1 T/T T/T A/G G/G C/G A/G A/G G/C C/C G/G A/A A/G A/C
2 G/T T/T A/G T/G C/G G/G G/G C/C C/C G/G A/A A/G C/C
3 T/T C/T G/A G/G C/C G/A G/G C/G C/C G/G A/A A/A C/A
4 T/T T/C G/G G/G C/C G/G G/G C/C C/C G/G A/A A/A C/C

   ... more lines ...
```

```
150 T/T T/T A/A G/G C/C A/A A/G G/G C/C G/G A/A A/A A/A
;
```

The following code can be used to first estimate haplotype frequencies by using the EM algorithm, and then to identify the haplotype tag SNPs.

```
ods output haplotypefreq=freqout(keep=haplotype freq);
proc haplotype data=casp8 genocol cutoff=0.0075;
   var m1-m13;
run;

data hapfreq;
   set freqout;
   array m{13} $ 1;
   do i = 1 to 13;
    m{i} = substr(haplotype, 2*i-1, 1);
   end;
   drop haplotype i;
run;

proc htsnp data=hapfreq size=4 method=sa best=5 cutoff=0.05
           seed=123 nosumm;
   var m1-m13;
   freq freq;
run;
```

The ODS statement is used to create a data set from the “Haplotype Frequencies” ODS table, which is displayed in its table form in Output 8.1.1.

Output 8.1.1 ODS Table Containing Haplotype Frequencies

```
                              The HAPLOTYPE Procedure

                                Haplotype Frequencies

                                                     Standard       95% Confidence
Number    Haplotype                        Freq         Error           Limits

     1    G-T-A-T-C-G-G-C-C-G-A-A-C     0.01988       0.00807     0.00406     0.03570
     2    T-C-G-G-C-G-G-C-C-G-A-A-C     0.09173       0.01669     0.05902     0.12445
     3    T-T-A-G-C-A-A-G-C-G-A-A-A     0.16666       0.02155     0.12442     0.20890
     4    T-T-A-G-C-A-G-G-C-G-A-A-A     0.05667       0.01337     0.03046     0.08287
     5    T-T-A-G-C-A-G-G-C-G-G-A-A     0.03663       0.01086     0.01534     0.05793
     6    T-T-G-G-C-A-G-G-C-G-A-A-A     0.01579       0.00721     0.00166     0.02992
     7    T-T-G-G-C-G-G-C-C-G-A-A-C     0.40576       0.02840     0.35011     0.46142
     8    T-T-G-G-C-G-G-G-T-C-A-A-A     0.02667       0.00932     0.00841     0.04493
     9    T-T-G-G-C-G-G-G-T-G-A-A-A     0.00861       0.00534     0.00000     0.01908
    10    T-T-G-G-G-G-G-C-C-G-A-G-C     0.16250       0.02133     0.12069     0.20432
```

With this table in the form of a SAS data set, the preceding DATA step code can be used to convert it to an input data set for PROC HTSNP, using the estimated frequencies from PROC HAPLOTYPE as the FREQ variable. In this example, the simulated annealing search method is specified for finding the best sets of size four. The “htSNP Evaluation” table that is created by PROC HTSNP is displayed in Output 8.1.2 to show the best five sets of SNPs that were selected.

Output 8.1.2 Candidate Sets of htSNPs from PROC HTSNP

```
                    The HTSNP Procedure

                      htSNP Evaluation

Rank     HTSNP1     HTSNP2     HTSNP3     HTSNP4          PDE

   1     m2         m5         m7         m13          1.0000
   1     m2         m7         m8         m12          1.0000
   1     m2         m5         m7         m8           1.0000
   1     m2         m7         m12        m13          1.0000
   1     m2         m5         m6         m7           1.0000
```

Note that the last selection shown in Output 8.1.2 matches the set of htSNPs found by Johnson et al. (2001).

References

Clayton, D. (2002), "Choosing a Set of Haplotype Tagging SNPs from a Larger Set of Diallelic Loci," `http://www-gene.cimr.cam.ac.uk/clayton/software/stata/htSNP/htsnp.pdf`.

Gouesnard, B., Bataillon, T. M., Decoux, G., Rozale, C., Schoen, D. J., and David, J. L. (2001), "MSTRAT: An Algorithm for Building Germ Plasm Core Collections by Maximizing Allelic or Phenotypic Richness," *Journal of Heredity*, 92, 93–94.

Johnson, G. C. L., Esposito, L., Barratt, B. J., Smith, A. N., Heward, J., Di Genova, G., Ueda, H., Cordell, H. J., Eaves, I. A., Dudbridge, F., Twells, R. C. J., Payne, F., Hughes, W., Nutland, S., Stevens, H., Carr, P., Tuomilehto-Wolf, E., Tuomilehto, J., Gough, S. C. L., Clayton, D. G., and Todd, J. A. (2001), "Haplotype Tagging for the Identification of Common Disease Genes," *Nature Genetics*, 29, 233–237.

Kirkpatrick, S., Gelatt, C. D., Jr., and Vecchi, M. P. (1983), "Optimization by Simulated Annealing," *Science*, 220, 671–680.

Metropolis, N., Rosenbluth, A. W., Rosenbluth, M. N., Teller, A. H., and Teller, E. (1953), "Equation of State Calculations by Fast Computing Machines," *Journal of Chemical Physics*, 21, 1087–1092.

Nijenhuis, A. and Wilf, H. S. (1978), *Combinatorial Algorithms*, New York: Academic Press.

Stram, D. O., Haiman, C. A., Hirschhorn, D. A., Kolonel, L. N., Henderson, B. E., and Pike, M. C. (2003), "Choosing Haplotype-Tagging SNPs Based on Unphased Genotype Data Using a Preliminary Sample of Unrelated Subjects with an Example from the Multiethnic Cohort Study," *Human Heredity*, 55, 27–36.

Chapter 9
The PSMOOTH Procedure

Contents

Overview: PSMOOTH Procedure

In the search for complex disease genes, linkage and/or association tests are often performed on markers from a genome-wide scan or SNPs from a finely scaled map. This means hundreds or even thousands of hypotheses are being simultaneously tested. Plotting the negative log p-values of all the marker tests will reveal many peaks that indicate significant test results, some of which are false positives. In order to reduce the number of false positives or improve power, smoothing methods can be applied that take into account p-values from neighboring, and possibly correlated, markers. That is, the peak length can be used to indicate significance in addition to the peak height. The PSMOOTH procedure offers smoothing methods that implement Simes' method (1986), Fisher's method (1932), and/or the truncated product method (TPM) (Zaykin et al. 2002) for multiple hypothesis testing. These methods modify the p-value from each marker test by using a function of its original p-value and the p-values of the tests on the nearest markers. Since the number of hypothesis tests being performed is not reduced, adjustments to correct the smoothed p-values for multiple testing are available as well.

PROC PSMOOTH can take any data set containing any number of columns of p-values as an input data set, including the output data sets from the CASECONTROL and FAMILY procedures (see Chapter 4 and Chapter 5 for more information).

Getting Started: PSMOOTH Procedure

Example

Suppose you want to test the 16 markers represented in the following data for association with a disease by using the genotype case-control and trend tests in PROC CASECONTROL. You are concerned about the multiple hypothesis testing issue, and so you also want to run PROC PSMOOTH on the output data set from PROC CASECONTROL in order to eliminate the number of false positives found using the individual p-values from the marker-trait association tests.

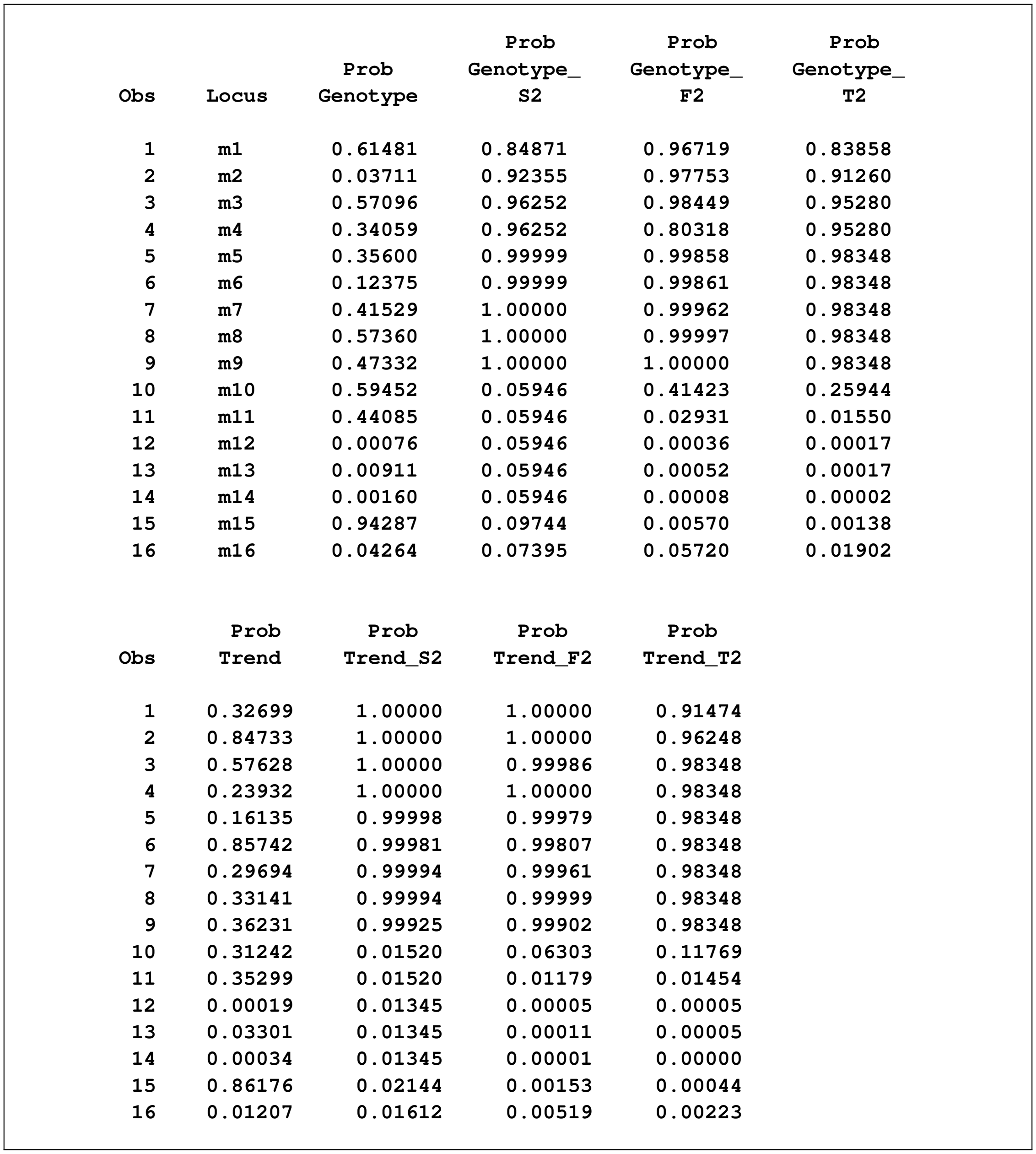
```
data in;
   input affected (m1-m16) ($);
   datalines;
1 1/2 2/2 2/2 2/2 1/1 2/2 1/2 1/2 1/1 1/2 1/2 2/2 2/2 2/2 2/2 1/2
1 1/2 1/1 1/2 1/2 1/1 1/1 1/2 1/1 1/2 1/2 1/1 2/2 1/1 1/2 1/1 1/2
1 1/1 2/2 1/2 1/2 1/1 1/2 1/1 1/2 1/2 2/2 2/2 1/2 1/2 1/2 2/2 1/2
1 1/1 1/2 2/2 1/2 1/2 1/1 1/2 1/2 1/2 1/1 1/1 1/2 2/2 1/2 1/1 1/1
1 1/2 1/1 1/1 1/2 2/2 1/1 1/1 1/2 1/1 2/2 1/2 2/2 2/2 2/2 1/2 1/1
1 1/2 1/1 1/2 2/2 2/2 1/1 1/1 1/2 1/2 1/2 2/2 2/2 1/1 2/2 2/2 1/1

   ... more lines ...

0 2/2 1/1 1/2 1/1 1/2 1/2 1/2 2/2 1/1 1/2 1/1 1/1 1/1 2/2 1/1 1/2
;
```

Note that the columns marker1-marker16 contain genotypes at each of the markers, so the GENOCOL option must be used in PROC CASECONTROL as follows to correctly read in the data.

```
proc casecontrol data=in outstat=cc_tests genotype trend genocol;
   trait affected;
   var m1-m16;
run;

proc psmooth data=cc_tests simes fisher tpm bw=2 adjust=sidak
             out=adj_p;
   var ProbGenotype ProbTrend;
   id Locus;
run;

proc print data=adj_p heading=h;
run;
```

This code modifies the p-values contained in the output data set from PROC CASECONTROL, first by smoothing the p-values by using Simes' method, Fisher's method, and the TPM with a bandwidth of 2, then by applying Šidák's multiple testing adjustment to the smoothed p-values.

Figure 9.1 PROC PSMOOTH Output Data Set



```
                                        Prob         Prob         Prob
                          Prob       Genotype_    Genotype_    Genotype_
   Obs     Locus        Genotype         S2           F2           T2

     1      m1           0.61481      0.84871      0.96719      0.83858
     2      m2           0.03711      0.92355      0.97753      0.91260
     3      m3           0.57096      0.96252      0.98449      0.95280
     4      m4           0.34059      0.96252      0.80318      0.95280
     5      m5           0.35600      0.99999      0.99858      0.98348
     6      m6           0.12375      0.99999      0.99861      0.98348
     7      m7           0.41529      1.00000      0.99962      0.98348
     8      m8           0.57360      1.00000      0.99997      0.98348
     9      m9           0.47332      1.00000      1.00000      0.98348
    10      m10          0.59452      0.05946      0.41423      0.25944
    11      m11          0.44085      0.05946      0.02931      0.01550
    12      m12          0.00076      0.05946      0.00036      0.00017
    13      m13          0.00911      0.05946      0.00052      0.00017
    14      m14          0.00160      0.05946      0.00008      0.00002
    15      m15          0.94287      0.09744      0.00570      0.00138
    16      m16          0.04264      0.07395      0.05720      0.01902


              Prob        Prob         Prob         Prob
   Obs       Trend      Trend_S2     Trend_F2     Trend_T2

     1      0.32699     1.00000      1.00000      0.91474
     2      0.84733     1.00000      1.00000      0.96248
     3      0.57628     1.00000      0.99986      0.98348
     4      0.23932     1.00000      1.00000      0.98348
     5      0.16135     0.99998      0.99979      0.98348
     6      0.85742     0.99981      0.99807      0.98348
     7      0.29694     0.99994      0.99961      0.98348
     8      0.33141     0.99994      0.99999      0.98348
     9      0.36231     0.99925      0.99902      0.98348
    10      0.31242     0.01520      0.06303      0.11769
    11      0.35299     0.01520      0.01179      0.01454
    12      0.00019     0.01345      0.00005      0.00005
    13      0.03301     0.01345      0.00011      0.00005
    14      0.00034     0.01345      0.00001      0.00000
    15      0.86176     0.02144      0.00153      0.00044
    16      0.01207     0.01612      0.00519      0.00223
```

Figure 9.1 displays the original and modified *p*-values.

Syntax: PSMOOTH Procedure

The following statements are available in PROC PSMOOTH.

PROC PSMOOTH < *options* > **;**
  BY *variables* **;**
  ID *variables* **;**
  VAR *variables* **;**

Items within angle brackets (< >) are optional, and statements following the PROC PSMOOTH statement can appear in any order. The VAR statement is required.

PROC PSMOOTH Statement

PROC PSMOOTH < *options* > **;**

You can specify the following options in the PROC PSMOOTH statement.

ADJUST=NONE

ADJUST=BON | BONFERRONI

ADJUST=FDR

ADJUST=SIDAK

indicates which adjustment for multiple testing to apply to the set(s) of p-values in the output data set. This adjustment is applied after any smoothing has occurred. ADJUST=NONE is the default.

BANDWIDTH=*number-list*

BW=*number-list*

gives the values for the bandwidths to use in combining p-values. A bandwidth of w indicates that w p-values on each side of the original p-value are included in the combining method to create a sliding window of size $2w + 1$. The number list can contain any combination of the following forms, with the forms separated by commas:

$w_1, w_2, \ldots, w_n$	a list of several values
w_1 to w_2	a sequence where w_1 is the starting value, w_2 is the ending value, and the increment is 1.
w_1 to w_2 by i	a sequence where w_1 is the starting value, w_2 is the ending value, and the increment is i.

All numbers in the number list must be integers, and any negative numbers are ignored. An example of a valid number list is

```
bandwidth = 1,2, 5 to 15 by 5, 18
```

which would perform the combining of p-values by using bandwidths 1, 2, 5, 10, 15, and 18, which create sliding windows of size 3, 5, 11, 21, 31, and 37, respectively.

DATA=*SAS-data-set*

names the input SAS data set to be used by PROC PSMOOTH. The default is to use the most recently created data set.

FISHER

requests that Fisher's method for combining p-values from multiple hypotheses be applied to the original p-values.

NEGLOG

requests that all p-values, original and combined, be transformed to their negative log (base e) in the output data set; that is, for each p-value, $-\log(p\text{-value})$ is reported in the OUT= data set. This option is useful for graphing purposes.

NEGLOG10

requests that all p-values, original and combined, be transformed to their negative log (base 10) in the output data set; that is, for each p-value, $-\log_{10}(p\text{-value})$ is reported in the OUT= data set. This option is useful for graphing purposes.

OUT=*SAS-data-set*

names the output SAS data set containing the original p-values and the new combined p-values. When this option is omitted, an output data set is created by default and named according to the DATA*n* convention.

SIMES

requests that Simes' method for combining p-values from multiple hypotheses be applied to the original p-values.

TAU=*number*

indicates the value of τ to be used in the TPM. The significance level for the tests can be used as the value for *number*, although this is not the only possibility. The value of *number* must be greater than 0 and less than or equal to 1. By default, *number* is set to 0.05. This option is ignored if the TPM option is not specified.

TPM

requests that the TPM for combining p-values from multiple hypotheses be applied to the original p-values.

BY Statement

BY *variables* **;**

You can specify a BY statement with PROC PSMOOTH to obtain separate analyses on observations in groups defined by the BY variables. When a BY statement appears, the procedure expects the input data set to be sorted in the order of the BY variables. The *variables* are one or more variables in the input data set.

If your input data set is not sorted in ascending order, use one of the following alternatives:

- Sort the data by using the SORT procedure with a similar BY statement.

- Specify the BY statement option NOTSORTED or DESCENDING in the BY statement for the PSMOOTH procedure. The NOTSORTED option does not mean that the data are unsorted but rather that the data are arranged in groups (according to values of the BY variables) and that these groups are not necessarily in alphabetical or increasing numeric order.
- Create an index on the BY variables by using the DATASETS procedure (in Base SAS software).

For more information about the BY statement, see *SAS Language Reference: Concepts*. For more information about the DATASETS procedure, see the *Base SAS Procedures Guide*.

ID Statement

ID *variables* **;**

The ID statement identifies the variables from the DATA= data set that should be included in the OUT= data set.

VAR Statement

VAR *variables* **;**

The VAR statement identifies the variables containing the original p-values on which the combining methods should be performed.

Details: PSMOOTH Procedure

Statistical Computations

Methods of Smoothing p-Values

PROC PSMOOTH offers three methods of combining p-values over specified sizes of sliding windows. For each value w listed in the BANDWIDTH= option of the PROC PSMOOTH statement, a sliding window of size $2w + 1$ is used; that is, the p-values for each set of $2w + 1$ consecutive markers are considered in turn, for each value w. The approach described by Zaykin et al. (2002) is implemented, where the original p-value at the center of the sliding window is replaced by a function of the original p-value and the p-values from the w nearest markers on each side to create a new sequence of p-values. Note that for markers less than w from the beginning or end of the data set (or BY group if any variables are specified in the BY statement), the number of hypotheses tested, L, is adjusted accordingly. The three methods of combining p-values from multiple hypotheses are Simes' method, Fisher's method, and the TPM, described in the following three sections. Plotting the new p-values versus the original p-values reveals the smoothing effect this technique has.

Simes' Method

Simes' method of combining p-values (1986) is performed as follows when the SIMES option is specified in the PROC PSMOOTH statement: let p_j be the original p-value at the center of the current sliding window, which contains $p_{j-w}, \ldots, p_{j+w}$. From these $L = 2w + 1$ p-values, the ordered p-values, $p_{(1)}, \ldots, p_{(L)}$ are formed. Then the new value for p_j is $\min_{1 \le i \le L}(Lp_{(i)}/i)$.

This method controls the Type I error rate even when hypotheses are positively correlated (Sarkar and Chang 1997), which is expected for nearby markers. Thus if dependencies are suspected among tests that are performed, this method is recommended due to its conservativeness.

Fisher's Method

When the FISHER option is issued in the PROC PSMOOTH statement, Fisher's method of combining p-values (1932) is applied by replacing the p-value at the center of the current sliding window p_j with the p-value of the statistic t, where

$$t = -2 \sum_{i=j-w}^{j+w} \log(p_i)$$

which has a χ^2_{2L} distribution under the null hypothesis of all $L = 2w + 1$ hypotheses being true.

CAUTION: t has a χ^2 distribution only under the assumption that the tests performed are mutually independent. When this assumption is violated, the probability of Type I error can exceed the significance level α.

TPM

The TPM is a variation of Fisher's method that leads to a different alternative hypothesis when τ, the value specified in the TAU= option, is less than 1 (Zaykin et al. 2002). With the TPM, rejection of the null hypothesis implies that there is at least one false null hypothesis among those with p-values $\le \tau$. To calculate a combined p-value by using the TPM for the p-value at the center of the sliding window, p_j, the quantity u must first be calculated as

$$u = \prod_{i=j-w}^{j+w} p_i^{I(p_i \le \tau)}$$

Then the formula for the new value for the p-value at the center of the sliding window of L markers is

$$\sum_{k=1}^{L} \binom{L}{k} (1-\tau)^{L-k} \left(u \sum_{s=0}^{k-1} \frac{(k \log \tau - \log u)^s}{s!} I(u \le \tau^k) + \tau^k I(u > \tau^k) \right)$$

When TAU=1 is specified, the TPM and Fisher's method are equivalent and the previous formula simplifies to

$$u \sum_{s=0}^{L-1} \frac{(-\log u)^s}{s!}$$

Multiple Testing Adjustments for p-Values

While the smoothing methods take into account the p-values from neighboring markers, the number of hypothesis tests performed does not change. Therefore, the Bonferroni, false discovery rate (FDR), and Šidák methods are offered by PROC PSMOOTH to adjust the smoothed p-values for multiple testing. The number of tests performed, R, is the number of valid observations in the current BY group if any variables are specified in the BY statement, or the number of valid observations in the entire data set if there are no variables specified in the BY statement. Note that these adjustments are not applied to the original column(s) of p-values; if you would like to adjust the original p-values for multiple testing, you must include a bandwidth of 0 in the BANDWIDTH= option of the PROC PSMOOTH statement along with one of the smoothing methods (SIMES, FISHER, or TPM).

For R tests, the p-value p_i results in an adjusted p-value of s_i according to these methods:

Bonferroni adjustment: $s_i = \min(Rp_i, 1.0), i = 1, \ldots, R$

Šidák adjustment (Šidák 1967): $s_i = 1 - (1 - p_i)^R, i = 1, \ldots, R$

FDR adjustment (Benjamini and Hochberg 1995):

$$\begin{aligned} s_{(R)} &= p_{(R)} \\ s_{(R-1)} &= \min\left(s_{(R)}, [R/(R-1)]p_{(R-1)}\right) \\ s_{(R-2)} &= \min\left(s_{(R-1)}, [R/(R-2)]p_{(R-2)}\right) \\ &\vdots \end{aligned}$$

where the R p-values have been ordered as $p_{(1)} \leq p_{(2)} \leq \cdots \leq p_{(R)}$. The Bonferroni and Šidák methods are conservative for controlling the family-wise error rate; however, often in the association mapping of a complex trait, it is desirable to control the FDR instead (Sabatti, Service, and Freimer 2003).

Missing Values

Missing values in a sliding window, even at the center of the window, are simply ignored, and the number of hypotheses L is reduced accordingly. Thus the smoothing methods can be applied to any window that contains at least one nonmissing value. Any p-values in the input data set that fall outside the interval [0,1] are treated as missing.

OUT= Data Set

The output data set specified in the OUT= option of the PROC PSMOOTH statement contains any BY variables and ID variables. Then for each variable in the VAR statement, the original column is included along with a column for each method and bandwidth specified in the PROC PSMOOTH statement. These variable names are formed by adding the suffixes "_Sw", "_Fw", and "_Tw" for Simes' method, Fisher's method, and the TPM, respectively, and a bandwidth of size w. For example, if the options BANDWIDTH=1,4 and SIMES, FISHER, and TPM are all specified in the PROC PSMOOTH statement, and RawP is the variable specified in the VAR statement, the OUT= data set includes RawP, RawP_S1, RawP_F1, RawP_T1, RawP_S4, RawP_F4, and RawP_T4. If the NEGLOG or NEGLOG10 option is specified in the PROC

PSMOOTH statement, then these columns all contain the negative logs (base *e* or base 10, respectively) of the *p*-values.

Example: PSMOOTH Procedure

Example 9.1: Displaying Plot of PROC PSMOOTH Output Data Set

Data other than the output data sets from the CASECONTROL and FAMILY procedures can be used in PROC PSMOOTH; here is an example of how to use *p*-values from another source, read into a SAS data set by using the following DATA step.

```
data tests;
   input Marker Pvalue @@;
   datalines;
 1  0.72841     2   0.40271
 3  0.32147     4   0.91616
 5  0.27377     6   0.48943
 7  0.40131     8   0.25555
 9  0.57585    10   0.20925
11  0.01531    12   0.23306
13  0.69397    14   0.33040
15  0.97265    16   0.53639
17  0.88397    18   0.03188
19  0.13570    20   0.79138
21  0.99467    22   0.37831
23  0.86459    24   0.97092
25  0.19372    26   0.85339
27  0.32078    28   0.31806
29  0.00655    30   0.82401
31  0.65339    32   0.36115
33  0.92704    34   0.49558
35  0.64842    36   0.43606
37  0.67060    38   0.87520
39  0.78006    40   0.27252
41  0.28561    42   0.80495
43  0.98159    44   0.97030
45  0.53831    46   0.78712
47  0.88493    48   0.36260
49  0.53310    50   0.65709
51  0.26527    52   0.46860
53  0.55465    54   0.54956
55  0.44477    56   0.04933
57  0.12016    58   0.76181
59  0.80158    60   0.18244
61  0.01382    62   0.15100
63  0.04713    64   0.52655
65  0.59368    66   0.94420
67  0.60104    68   0.32848
69  0.90195    70   0.21374
```

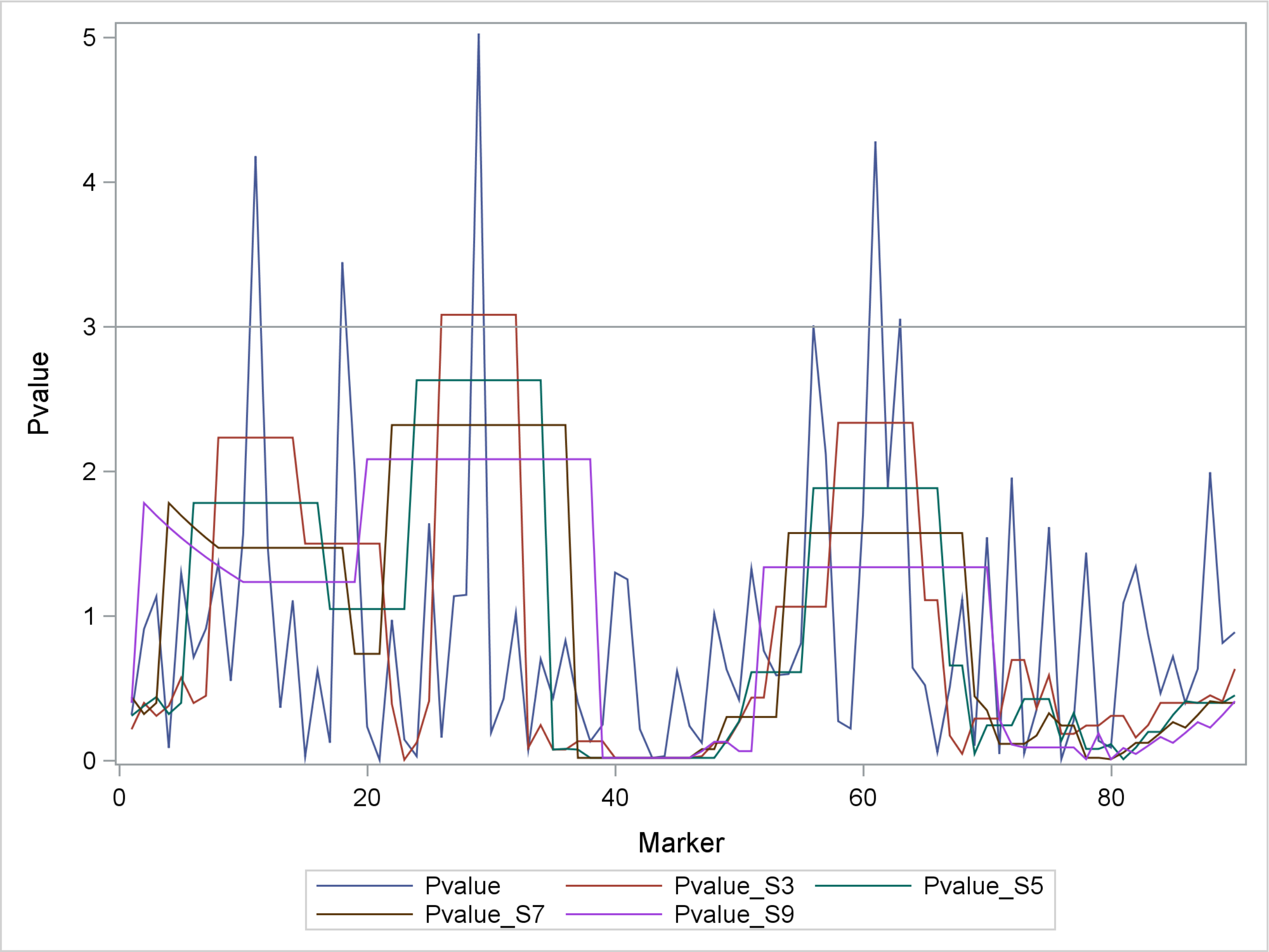
```
71  0.95471    72   0.14145
73  0.95215    74   0.70330
75  0.19921    76   0.99086
77  0.75736    78   0.23761
79  0.87260    80   0.91472
81  0.33650    82   0.26160
83  0.41948    84   0.62817
85  0.48721    86   0.67093
87  0.53089    88   0.13623
89  0.44344    90   0.41172
;
```

The following code applies Simes' method for multiple hypothesis testing in order to adjust the *p*-values.

```
proc psmooth data=tests out=pnew simes bandwidth=3 to 9 by 2 neglog;
   var Pvalue;
   id Marker;
run;

proc sgplot data=pnew;
   series x=Marker y=Pvalue    / lineattrs=(pattern=solid);
   series x=Marker y=Pvalue_S3 / lineattrs=(pattern=solid);
   series x=Marker y=Pvalue_S5 / lineattrs=(pattern=solid);
   series x=Marker y=Pvalue_S7 / lineattrs=(pattern=solid);
   series x=Marker y=Pvalue_S9 / lineattrs=(pattern=solid);
   refline 3.0 / axis=y;
   discretelegend;
run;
```

The NEGLOG option is used in the PROC PSMOOTH statement to facilitate plotting the *p*-values by using the GPLOT procedure of SAS/GRAPH. The plot in Output 9.1.1 demonstrates the effect of the different window sizes that are implemented.

Output 9.1.1 Line Plot of Negative Log p-Values



Note how the plots become progressively smoother as the window size increases. Points above the horizontal reference line represent significant p-values at the 0.05 level. While six of the markers have significant p-values before adjustment, only the method that uses a bandwidth of 3 finds any significant markers, all in the 26–32 region. This can be an indication that the other five markers are significant only by chance; that is, they might be false positives.

References

Benjamini, Y. and Hochberg, Y. (1995), "Controlling the False Discovery Rate: A Practical and Powerful Approach to Multiple Testing," *Journal of the Royal Statistical Society, Series B*, 57, 289–300.

Fisher, R. A. (1932), *Statistical Methods for Research Workers*, 4th Edition, London: Oliver & Boyd.

Sabatti, C., Service, S., and Freimer, N. (2003), "False Discovery Rate in Linkage and Association Genome Screens for Complex Disorders," *Genetics*, 164, 829–833.

Sarkar, S. K. and Chang, C.-K. (1997), "The Simes Method for Multiple Hypothesis Testing with Positively Dependent Test Statistics," *Journal of the American Statistical Association*, 92, 1601–1608.

Šidák, Z. (1967), "Rectangular Confidence Regions for the Means of Multivariate Normal Distributions," *Journal of the American Statistical Association*, 62, 626–633.

Simes, R. J. (1986), "An Improved Bonferroni Procedure for Multiple Tests of Significance," *Biometrika*, 73, 751–754.

Zaykin, D. V., Zhivotovsky, L. A., Westfall, P. H., and Weir, B. S. (2002), "Truncated Product Method for Combining *P*-Values," *Genetic Epidemiology*, 22, 170–185.

Chapter 10
The TPLOT Macro

Contents

Overview: TPLOT Macro

The %TPLOT macro creates a triangular plot that graphically displays genetic marker test results. The plot has colors and shapes representing p-value ranges for tests of the following quantities: linkage disequilibrium between pairs of markers, Hardy-Weinberg equilibrium (HWE) for individual markers, and associations between markers and a dichotomous trait (such as disease status). This is a convenient way of combining information contained in output data sets from two separate SAS/Genetics procedures and summarizing it in an easily interpretable plot. Thus, insights can be gleaned by simply studying a plot rather than having to search through many rows of data or writing code to attempt to summarize the results.

The %TPLOT macro is a part of the SAS Autocall library, and it is automatically available for use in your SAS program provided that the SAS system option MAUTOSOURCE is in effect. For more information about autocall libraries, see *SAS Macro Language: Reference, Version 8* (2000).

Syntax: TPLOT Macro

The %TPLOT macro has the following form:

%TPLOT *(SAS-data-set, SAS-data-set, variable < ,ALPHA=number>)* **;**

The first argument, *SAS-data-set*, specifies the name of the SAS data set that is the output data set from the ALLELE procedure (see Chapter 2), containing the linkage disequilibrium test and HWE test p-values. A user-created data set can be used instead, but it is required to contain the variables Locus1 and Locus2 and a variable ProbChi containing the p-values from the disequilibrium tests. The order in which the Locus1 and Locus2 variables are sorted is the order in which the values are displayed on the vertical and horizontal axes, respectively.

The second argument, *SAS-data-set*, specifies the name of the SAS data set that contains the p-values for the marker-trait association tests. This data set can be the output data set from the CASECONTROL procedure, the FAMILY procedure, or the PSMOOTH procedure, or it can be created by the user. A user-created data set must contain a Locus variable for the values on the axes and a variable containing p-values that is specified in the third argument, discussed in the following paragraph. The Locus variable must be in the same sorted order as the Locus1 variable in the data set named in the first argument.

The third argument, *variable*, names the variable that contains the marker-trait association p-values in the SAS data set that is specified in the second argument.

The first three arguments are required. The following option can be used with the %TPLOT macro. The option must follow the three required arguments.

ALPHA=*number*

specifies the significance level for the marker-trait association test. This level is used as a cut-off for the p-value range corresponding to the symbol shape on the plot. This number must be between 0 and 1. The default is ALPHA=0.05.

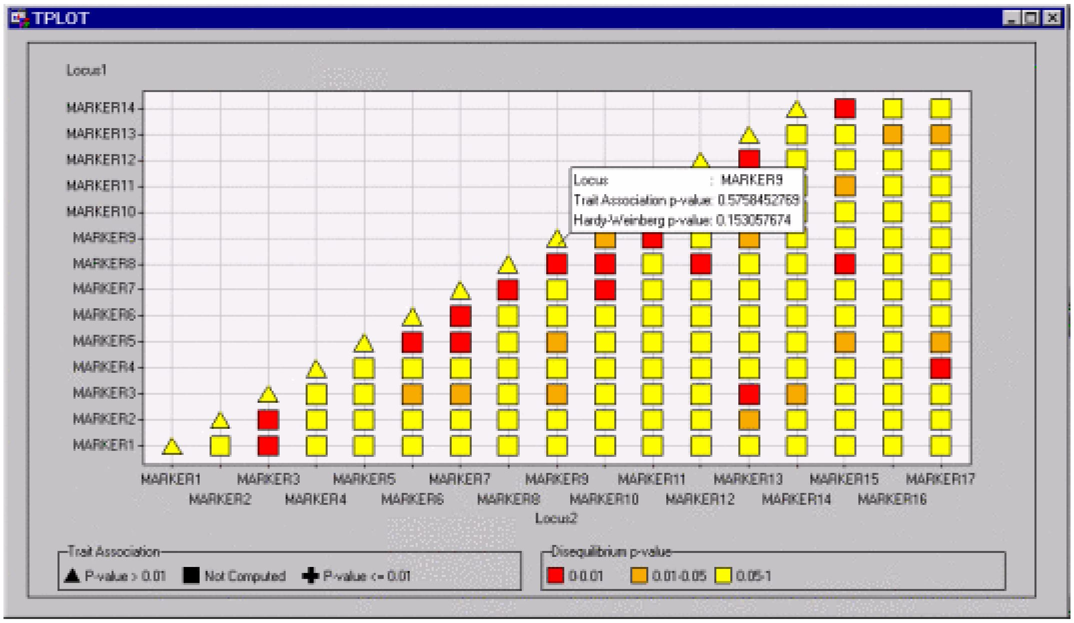
Results

Plot

Running the %TPLOT macro creates a window displaying a graphical representation of the marker test results.

Figure 10.1 shows an example of the TPLOT results window.

Figure 10.1 Results Window for TPLOT Macro



Locus : MARKER9
Trait Association p-value: 0.5758452769
Hardy-Weinberg p-value: 0.153057674

This plot contains a grid of points with symbols that represent the p-values for various marker tests. Colors and shapes of the data points are used to symbolize p-value ranges. The **Show Info About Points** button on the toolbar enables the p-values to be displayed. If you hold down the left mouse button while positioning the mouse pointer on any point in the plot, the pop-up window appears for off-diagonal points, displaying the two markers being tested for linkage disequilibrium and the p-value of the test; the window displays the marker and its p-values for the HWE test and marker-trait association test for points on the diagonal, as shown in Figure 10.1.

Disequilibrium Tests

The p-values from the linkage disequilibrium tests between all pairs of markers (or all markers within a certain range of each other) are represented by the color of the squares on the off-diagonal of the plot. For the points on the diagonal, the results from the Hardy-Weinberg equilibrium test are displayed instead of the linkage disequilibrium tests since the same marker locus is on the horizontal and vertical axes.

The three ranges of p-values that correspond to different colored symbols in the plot are as follows:

Red	[0, 0.01]
Orange	(0.01, 0.05]
Yellow	(0.05, 1]

The disequilibrium test p-values that are plotted can be provided by the output data set from PROC ALLELE, or by a user-created data set meeting the requirements described in the "Syntax: TPLOT Macro" section.

Marker-Trait Association Tests

Points on the diagonal also display p-values from marker-trait association tests, using the shape of the symbol to correspond to two categories of p-values, significant and not significant. The significance level is set to 0.05 by default, but can be modified using the ALPHA= option in the %TPLOT macro. Thus, for a significance level of α, the following shapes represent the following ranges:

Plus	✚	$[0, \alpha]$
Triangle	▲	$(\alpha, 1]$

Note that the square shape ■ of the off-diagonal points does not represent a marker-trait association p-value since there are two different marker loci represented on the horizontal and vertical axes. These p-values can be provided by the output data set of PROC CASECONTROL, PROC FAMILY, or PROC PSMOOTH. Alternatively, a user-created data set that meets the conditions described in the "Syntax: TPLOT Macro" section can be used.

Menu Bar

The results window contains the following drop-down lists:

File	**Close**	closes the results window.
	Print Setup	opens the printer setup utility.
	Print	prints the plot as it is currently shown.
	Exit	exits the current SAS session.

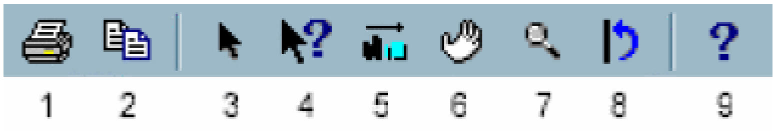
Edit	**Copy**	copies the plot to the clipboard.
Format	**Rescale Axes**	when selected, changes the scale of the axes to fit the entire plot in the window.

You can also access these menus by right-clicking anywhere in the TPLOT results window.

Toolbar

A toolbar (Figure 10.2) is displayed at the top of the TPLOT results window. Use the toolbar to display information about points on the plot or to modify the plot's appearance. Tool tips are displayed when you place your mouse pointer over an icon in the toolbar.

Figure 10.2 Toolbar for the %TPLOT Results Window



Tool icons from left to right are as follows:

1. **Print** - prints the plot.
2. **Copy** - copies the plot to the clipboard.
3. **Select a Node or Point** - activates a point on the plot.
4. **Show Info About Points** - displays a text box with information about the selected point.
5. **Scroll Data** - scrolls across data points within the plot. Use this tool when the plot is not able to display all of the points in a single frame.
6. **Move Graph** - moves the plot within the window.
7. **Zoom In/Out** - increases or decreases the size of the plot.
8. **Reset** - returns the plot to its default settings.
9. **What's This?** - displays the Help for the results window.

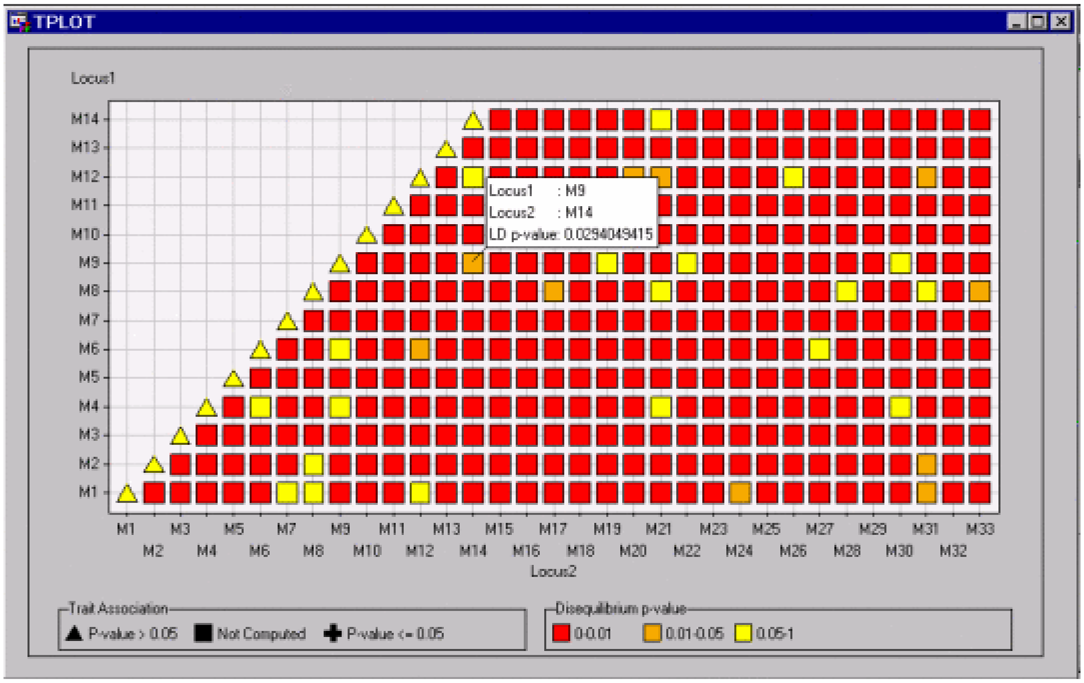
Example: TPLOT Macro

The following is an example of the code that can be used to create the triangular plot of *p*-values for the data set pop22. This data set is in the proper form for a PROC ALLELE input data set, containing columns of alleles for 150 markers.

```
proc allele data=pop22 outstat=ldstats noprint maxdist=150;
   var a1-a300;
run;

proc casecontrol data=pop22 outstat=assocstats genotype;
   trait affected;
   var a1-a300;
run;

proc psmooth data=assocstats out=sm_assocstats bw=5 simes;
   id Locus;
   var ProbGenotype;
run;

%tplot(ldstats, sm_assocstats, ProbGenotype_S5);
```

Note that the output data set from PROC CASECONTROL can be used in place of the output data set from PROC PSMOOTH if you want to use unadjusted p-values. This code creates the plot in the TPLOT window shown in Figure 10.3.

Figure 10.3 Results Window for TPLOT Macro



Locus1 : M9
Locus2 : M14
LD p-value: 0.0294049415

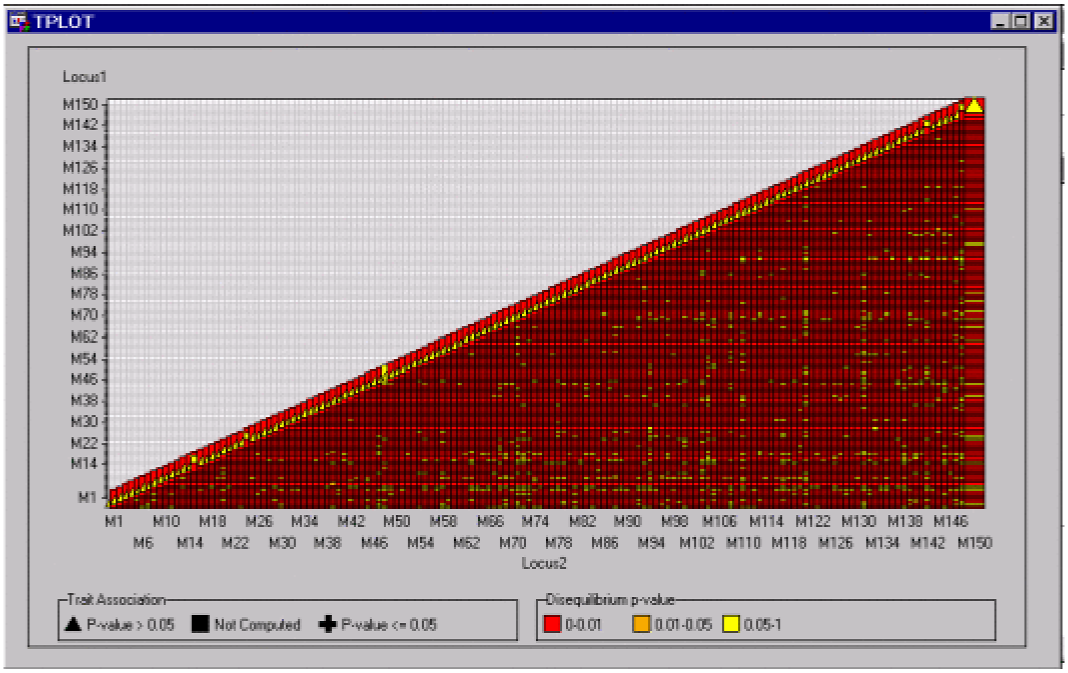
Trait Association
▲ P-value > 0.05 ■ Not Computed ✚ P-value <= 0.05

Disequilibrium p-value
🟥 0-0.01 🟧 0.01-0.05 🟨 0.05-1

Figure 10.3 displays the bottom-left corner of the plot. The pop-up window is displayed by selecting **Show Info About Points** from the toolbar and holding the mouse pointer over the point shown. The orange color of this point indicates that the p-value for testing that there is no linkage disequilibrium between M9 and M18 is between 0.01 and 0.05. The pop-up window provides the exact value of this p-value.

Other parts of the plot can be viewed by selecting **Scroll Data** from the toolbar. Alternatively, the entire plot can be viewed in the window by selecting **Format**→ **Rescale Axes** from the menu bar. This creates the view of the plot shown in Figure 10.4.

Figure 10.4 Results Window for TPLOT Macro



The view shown in Figure 10.4 displays all the data points at once.

Subject Index

Syntax Index