SAS/STAT® 14.2 User's Guide

The MULTTEST Procedure

This document is an individual chapter from *SAS/STAT® 14.2 User's Guide*.

The correct bibliographic citation for this manual is as follows: SAS Institute Inc. 2016. *SAS/STAT® 14.2 User's Guide*. Cary, NC: SAS Institute Inc.

SAS/STAT® 14.2 User's Guide

Copyright © 2016, SAS Institute Inc., Cary, NC, USA

All Rights Reserved. Produced in the United States of America.

For a hard-copy book: No part of this publication may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means, electronic, mechanical, photocopying, or otherwise, without the prior written permission of the publisher, SAS Institute Inc.

For a web download or e-book: Your use of this publication shall be governed by the terms established by the vendor at the time you acquire this publication.

The scanning, uploading, and distribution of this book via the Internet or any other means without the permission of the publisher is illegal and punishable by law. Please purchase only authorized electronic editions and do not participate in or encourage electronic piracy of copyrighted materials. Your support of others' rights is appreciated.

U.S. Government License Rights; Restricted Rights: The Software and its documentation is commercial computer software developed at private expense and is provided with RESTRICTED RIGHTS to the United States Government. Use, duplication, or disclosure of the Software by the United States Government is subject to the license terms of this Agreement pursuant to, as applicable, FAR 12.212, DFAR 227.7202-1(a), DFAR 227.7202-3(a), and DFAR 227.7202-4, and, to the extent required under U.S. federal law, the minimum restricted rights as set out in FAR 52.227-19 (DEC 2007). If FAR 52.227-19 is applicable, this provision serves as notice under clause (c) thereof and no other notice is required to be affixed to the Software or documentation. The Government's rights in Software and documentation shall be only those set forth in this Agreement.

SAS Institute Inc., SAS Campus Drive, Cary, NC 27513-2414

November 2016

SAS® and all other SAS Institute Inc. product or service names are registered trademarks or trademarks of SAS Institute Inc. in the USA and other countries. ® indicates USA registration.

Other brand and product names are trademarks of their respective companies.

SAS software may be provided with certain third-party software, including but not limited to open-source software, which is licensed under its applicable third-party software license agreement. For license information about third-party software distributed with SAS software, refer to http://support.sas.com/thirdpartylicenses.

Chapter 80
The MULTTEST Procedure

Contents

Overview: MULTTEST Procedure **6412**
Getting Started: MULTTEST Procedure **6413**
Drug Example 6413
Syntax: MULTTEST Procedure **6416**
PROC MULTTEST Statement 6417
BY Statement 6428
CLASS Statement 6429
CONTRAST Statement 6429
FREQ Statement 6430
ID Statement 6431
STRATA Statement 6431
TEST Statement 6431
Details: MULTTEST Procedure **6434**
Statistical Tests 6434
p-Value Adjustments 6441
Missing Values 6449
Computational Resources 6450
Output Data Sets 6450
Displayed Output 6452
ODS Table Names 6453
ODS Graphics 6454
Examples: MULTTEST Procedure **6455**
Example 80.1: Cochran-Armitage Test with Permutation Resampling 6455
Example 80.2: Freeman-Tukey and *t* Tests with Bootstrap Resampling 6458
Example 80.3: Peto Mortality-Prevalence Test 6462
Example 80.4: Fisher Test with Permutation Resampling 6465
Example 80.5: Inputting Raw *p*-Values 6469
Example 80.6: Adaptive Adjustments and ODS Graphics 6471
References **6479**

Overview: MULTTEST Procedure

The MULTTEST procedure addresses the multiple testing problem. This problem arises when you perform many hypothesis tests on the same data set. Carrying out multiple tests is often reasonable because of the cost of obtaining data, the discovery of new aspects of the data, and the many alternative statistical methods. However, a disadvantage of multiple testing is the greatly increased probability of declaring false significances.

For example, suppose you carry out 10 hypothesis tests at the 5% level, and you assume that the distributions of the *p*-values from these tests are uniform and independent. Then, the probability of declaring a particular test significant under its null hypothesis is 0.05, but the probability of declaring at least 1 of the 10 tests significant is 0.401. If you perform 20 hypothesis tests, the latter probability increases to 0.642. These high chances illustrate the danger of multiple testing.

PROC MULTTEST approaches the multiple testing problem by adjusting the *p*-values from a family of hypothesis tests. An adjusted *p*-value is defined as the smallest significance level for which the given hypothesis would be rejected, when the entire family of tests is considered. The decision rule is to reject the null hypothesis when the adjusted *p*-value is less than α. For most methods, this decision rule controls the *familywise error rate* at or below the α level. However, the *false discovery rate* controlling procedures control the false discovery rate at or below the α level.

PROC MULTTEST provides the following *p*-value adjustments:

- Bonferroni
- Šidák
- step-down methods
- Hochberg
- Hommel
- Fisher and Stouffer combination
- bootstrap
- permutation
- adaptive methods
- false discovery rate
- positive FDR

The Bonferroni and Šidák adjustments are simple functions of the raw *p*-values. They are computationally quick, but they can be too conservative. Step-down methods remove some conservativeness, as do the step-up methods of Hochberg (1988), and the adaptive methods. The bootstrap and permutation adjustments resample the data with and without replacement, respectively, to approximate the distribution of the minimum *p*-value of all tests. This distribution is then used to adjust the individual raw *p*-values. The bootstrap and permutation methods are computationally intensive but appealing in that, unlike the other methods, correlations and distributional characteristics are incorporated into the adjustments (Westfall and Young 1989; Westfall et al. 1999).

PROC MULTTEST handles data arising from a multivariate one-way ANOVA model, possibly stratified, with continuous and discrete response variables; it can also accept raw *p*-values as input data. You can perform a *t* test for the mean for continuous data with or without a homogeneity assumption, and the following statistical tests for discrete data:

- Cochran-Armitage linear trend test
- Freeman-Tukey double arcsine test
- Peto mortality-prevalence (log-rank) test
- Fisher exact test

The Cochran-Armitage and Peto tests have exact versions that use permutation distributions and asymptotic versions that use an optional continuity correction. Also, with the exception of the Fisher exact test, you can use a stratification variable to construct Mantel-Haenszel-type tests. All of the previously mentioned tests can be one- or two-sided.

As in the GLM procedure, you can specify linear contrasts that compare means or proportions of the treated groups. The output contains summary statistics and regular and multiplicity-adjusted *p*-values. You can create output data sets containing raw and adjusted *p*-values, test statistics and other intermediate calculations, permutation distributions, and resampling information.

The MULTTEST procedure uses ODS Graphics to create graphs as part of its output. For general information about ODS Graphics, see Chapter 21, "Statistical Graphics Using ODS."

The GLIMMIX, GLM, MIXED, and LIFETEST procedures, and other procedures that implement the ESTIMATE, LSMEANS, LSMESTIMATE, and SLICE statements, also adjust their results for multiple tests. For more information, see the documentation for these procedures and statements, and Westfall et al. (1999).

Getting Started: MULTTEST Procedure

Drug Example

Suppose you conduct a small study to test the effect of a drug on 15 subjects. You randomly divide the subjects into three balanced groups receiving 0 mg, 1 mg, and 2 mg of the drug, respectively. You carry out the experiment and record the presence or absence of 10 side effects for each subject. Your data set is as follows:

```
data Drug;
   input Dose$ SideEff1-SideEff10;
   datalines;
0MG  0   0   1   0   0   1   0   0   0   0
0MG  0   0   0   0   0   0   0   0   0   1
0MG  0   0   0   0   0   0   0   0   1   0
0MG  0   0   0   0   0   0   0   0   0   0
0MG  0   1   0   0   0   0   0   0   0   0
1MG  1   0   0   1   0   1   0   0   1   0
1MG  0   0   0   1   1   0   0   1   0   1
1MG  0   1   0   0   0   0   1   0   0   0
1MG  0   0   1   0   0   0   0   0   0   1
1MG  1   0   1   0   0   0   0   1   0   0
2MG  0   1   1   1   0   1   1   1   0   1
2MG  1   1   1   1   1   1   0   1   1   0
2MG  1   0   0   1   0   1   1   0   1   0
2MG  0   1   1   1   1   0   1   1   1   1
2MG  1   0   1   0   1   1   1   0   0   1
;
```

The increasing incidence of 1s for higher dosages in the preceding data set provides an initial visual indication that the drug has an effect. To explore this statistically, you perform an analysis in which the possibility of side effects increases linearly with drug level. You can analyze the data for each side effect separately, but you are concerned that, with so many tests, there might be a high probability of incorrectly declaring some drug effects significant. You want to correct for this multiplicity problem in a way that accounts for the discreteness of the data and for the correlations between observations on the same unit.

PROC MULTTEST addresses these concerns by processing all of the data simultaneously and adjusting the *p*-values. The following statements perform a typical analysis:

```
ods graphics on;
proc multtest bootstrap nsample=20000 seed=41287 notables
              plots=PByTest(vref=0.05 0.1);
   class Dose;
   test ca(SideEff1-SideEff10);
   contrast 'Trend' 0 1 2;
run;
ods graphics off;
```

This analysis uses the BOOTSTRAP option to adjust the *p*-values. The NSAMPLE= option requests 20,000 samples for the bootstrap analysis, and the starting seed for the random number generator is 41287. The NOTABLES option suppresses the display of summary statistics for each side effect and drug level combination. The PLOTS= option displays a visual summary of the unadjusted and adjusted *p*-values against each test, and the VREF= option adds reference lines to the display.

The CLASS statement is used to specify the grouping variable, Dose. The **ca(sideeff1-sideeff10)** specification in the TEST statement requests a Cochran-Armitage linear trend test for all 10 characteristics. The CONTRAST statement gives the coefficients for the linear trend test.

The "Model Information" table in Figure 80.1 describes the statistical tests performed by PROC MULTTEST. For this example, PROC MULTTEST carries out a two-tailed Cochran-Armitage linear trend test with no continuity correction or strata adjustment. This test is performed on the raw data and on 20,000 bootstrap samples.

Figure 80.1 Output Summary for the MULTTEST Procedure

The Multtest Procedure

Model Information	
Test for discrete variables	Cochran-Armitage
Z-score approximation used	Everywhere
Continuity correction	0
Tails for discrete tests	Two-tailed
Strata weights	None
P-value adjustment	Bootstrap
Number of resamples	20000
Seed	41287

The "Contrast Coefficients" table in Figure 80.2 displays the coefficients for the Cochran-Armitage test. They are 0, 1, and 2, as specified in the CONTRAST statement.

Figure 80.2 Coefficients Used in the MULTTEST Procedure

Contrast Coefficients			
	Dose		
Contrast	**0MG**	**1MG**	**2MG**
Trend	0	1	2

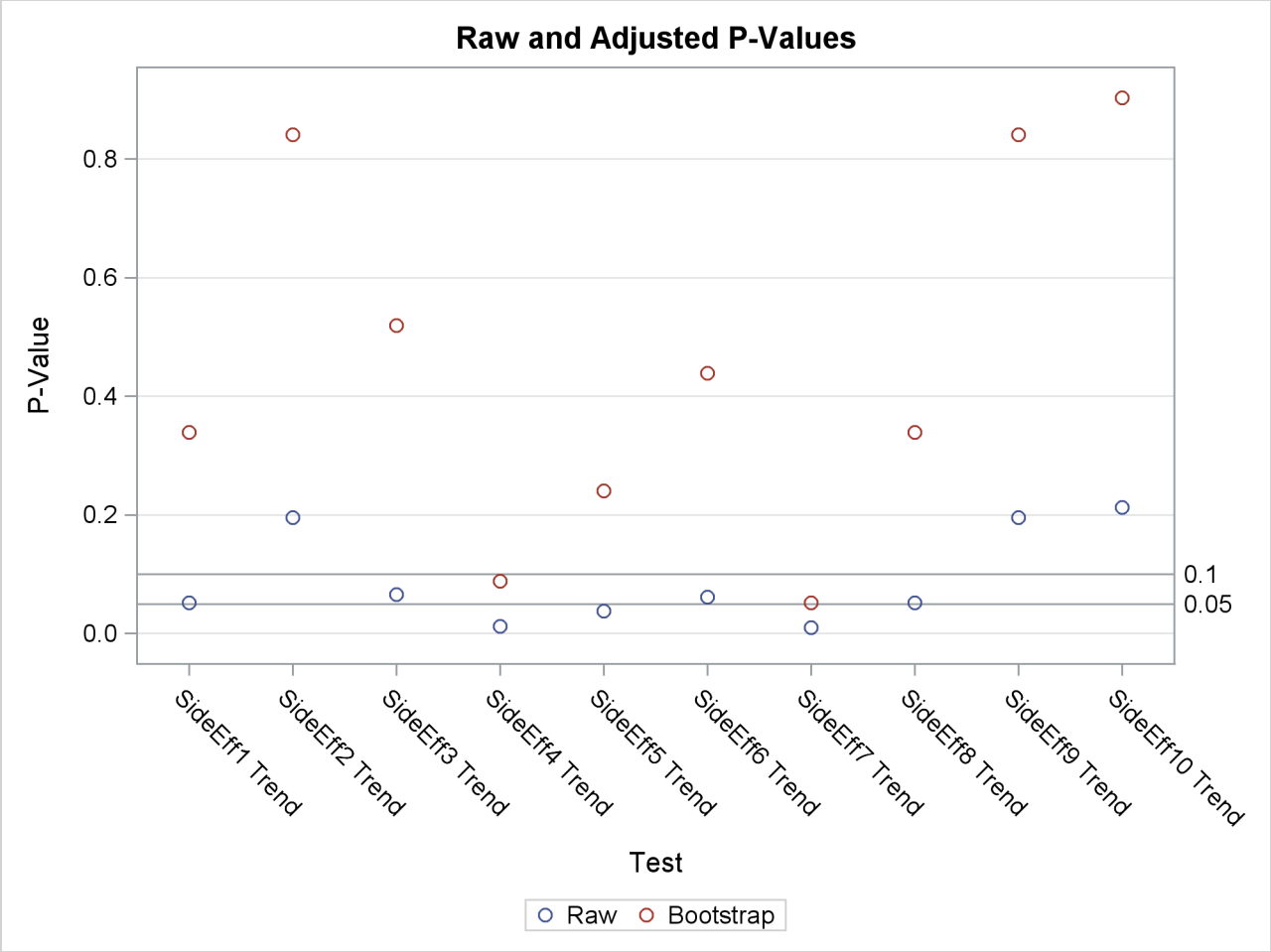
The "p-Values" table in Figure 80.3 lists the *p*-values for the drug example. The Raw column lists the *p*-values for the Cochran-Armitage test on the original data, and the Bootstrap column provides the bootstrap adjustment of the raw *p*-values.

Note that the raw *p*-values lead you to reject the null hypothesis of no linear trend for 3 of the 10 characteristics at the 5% level and 7 of the 10 characteristics at the 10% level. The bootstrap *p*-values, however, lead to this conclusion for 0 of the 10 characteristics at the 5% level and only 2 of the 10 characteristics at the 10% level; you can also see this in Figure 80.4.

Figure 80.3 Summary of *p*-Values for the MULTTEST Procedure

p-Values			
Variable	**Contrast**	**Raw**	**Bootstrap**
SideEff1	Trend	0.0519	0.3388
SideEff2	Trend	0.1949	0.8403
SideEff3	Trend	0.0662	0.5190
SideEff4	Trend	0.0126	0.0884
SideEff5	Trend	0.0382	0.2408
SideEff6	Trend	0.0614	0.4383
SideEff7	Trend	0.0095	0.0514
SideEff8	Trend	0.0519	0.3388
SideEff9	Trend	0.1949	0.8403
SideEff10	Trend	0.2123	0.9030

Figure 80.4 Adjusted p-Values



The bootstrap adjustment gives the probability of observing a p-value as extreme as each given p-value, considering all 10 tests simultaneously. This adjustment incorporates the correlation of the raw p-values, the discreteness of the data, and the multiple testing problem. Failure to account for these issues can certainly lead to misleading inferences for these data.

Syntax: MULTTEST Procedure

The following statements are available in the MULTTEST procedure:

PROC MULTTEST < *options* > ;
  BY *variables* ;
  CLASS *variable* ;
  CONTRAST '*label*' *values* ;
  FREQ *variable* ;
  ID *variables* ;
  STRATA *variable* ;
  TEST *name* **(***variables* < / *options* >**)** ;

Statements that follow the PROC MULTTEST statement can appear in any order. The CLASS and TEST statements are required unless the INPVALUES= option is specified in the PROC MULTTEST statement.

The following sections describe the PROC MULTTEST statement and then describe the other statements in alphabetical order.

PROC MULTTEST Statement

PROC MULTTEST < *options* > **;**

The PROC MULTTEST statement invokes the MULTTEST procedure. It also specifies the *p*-value adjustments. Table 80.1 summarizes the *options* available in the PROC MULTTEST statement. These *options* are described in alphabetical order following the table.

Table 80.1 PROC MULTTEST Statement Options by Function

Option	Description
FWE-Controlling *p*-Value Adjustments	
ADAPTIVEHOLM	Computes the adaptive step-down Bonferroni adjustment
ADAPTIVEHOCHBERG	Computes the adaptive step-up Bonferroni adjustment
BONFERRONI	Computes the Bonferroni adjustment
BOOTSTRAP	Computes the bootstrap min-*p* adjustment
FISHER_C	Computes Fisher's combination adjustment
HOCHBERG	Computes the step-up Bonferroni adjustment
HOMMEL	Computes Hommel's adjustment
HOLM	Computes the step-down Bonferroni adjustment
PERMUTATION	Computes the permutation min-*p* adjustment
SIDAK	Computes Šidák's adjustment
STEPBON	Computes the step-down Bonferroni adjustment
STEPBOOT	Computes the step-down bootstrap adjustment
STEPPERM	Computes the step-down permutation adjustment
STEPSID	Computes the step-down Šidák adjustment
STOUFFER	Computes the Stouffer-Liptak combination adjustment
FDR-Controlling *p*-Value Adjustments	
ADAPTIVEFDR	Computes the adaptive linear step-up adjustment
DEPENDENTFDR	Computes the linear step-up adjustment under dependence
FDR	Computes the linear step-up adjustment
FDRBOOT	Computes the linear step-up bootstrap min-*p* adjustment
FDRPERM	Computes the linear step-up permutation min-*p* adjustment
PFDR	Computes the positive FDR adjustment
Input/Output Data Sets	
DATA=	Names the input data set
INPVALUES=	Names the input data set of raw *p*-values
OUT=	Names the output data set
OUTPERM=	Names the output permutation data set
OUTSAMP=	Names the output resample data set

Table 80.1 *continued*

Option	Description
Displayed Output Options	
NOPRINT	Suppresses all tables
NOTABLES	Suppresses variable tables
NOZEROS	Suppresses zero tables for CLASS variables
NOPVALUE	Suppresses the "p-Values" table
PLOTS	Requests ODS Graphics
Resampling Options	
CENTER	Mean-centers continuous variables before resampling
NOCENTER	Does not mean-center continuous variables before resampling
NSAMPLE=	Specifies the number of resamples
RANUNI	Specifies a different random number generator
SEED=	Specifies the seed for resampling
CLASS Variable Options	
NOZEROS	Suppresses zero tables for CLASS variables
ORDER=	Specifies CLASS variable order
Computational Options	
EPSILON=	Specifies the comparison value
NTRUENULL=	Specifies the estimation method for the number of true nulls
PTRUENULL=	Specifies the estimation method for the proportion of true nulls

You can specify the following *options* in the PROC MULTTEST statement.

ADAPTIVEHOCHBERG

AHOC

requests adjusted *p*-values by using the Hochberg and Benjamini (1990) adaptive step-up Bonferroni method. See the section "Adaptive Adjustments" on page 6446 for more details.

ADAPTIVEHOLM

AHOLM

requests adjusted *p*-values by using the Hochberg and Benjamini (1990) adaptive step-down Bonferroni method. See the section "Adaptive Adjustments" on page 6446 for more details.

ADAPTIVEFDR< (UNRESTRICT) >

AFDR< (UNRESTRICT) >

requests adjusted *p*-values by using the Benjamini and Hochberg (2000) adaptive linear step-up method (AFDR). The UNRESTRICT option estimates the AFDR as defined in Benjamini and Hochberg (2000), which allows the adjustment to reduce the raw *p*-value. By default, the AFDR is constrained to be greater than or equal to the raw *p*-value. See the section "Adaptive False Discovery Rate" on page 6448 for more details.

BONFERRONI

BON

specifies that the Bonferroni adjustments (number of tests × p-value) be computed for each test. These adjustments can be extremely conservative and should be viewed with caution. When exact tests are specified via the PERMUTATION= option in the TEST statement, the actual permutation distributions are used, resulting in a much less conservative version of this procedure (Westfall and Wolfinger 1997). See the section "Bonferroni" on page 6443 for more details.

BOOTSTRAP

BOOT

specifies that the p-values be adjusted by using the bootstrap method to resample vectors (Westfall and Young 1993). Resampling is performed with replacement and independently within levels of the STRATA variable. Continuous variables are mean-centered by default prior to resampling; specify the NOCENTER option to change this. See the section "Bootstrap" on page 6443 for more details. The BOOTSTRAP option is not allowed with the Peto test.

If the PERMUTATION= suboption is used with the CA test in the TEST statement, the exact permutation distribution is recomputed for each bootstrap sample. **CAUTION:** This can be very time-consuming. It is preferable to use permutation resampling when permutation base tests are used.

CENTER

requests that continuous variables be mean-centered prior to resampling. The default action is to mean-center for bootstrap resampling and not to mean-center for permutation resampling.

DATA=*SAS-data-set*

names the input SAS data set to be used by PROC MULTTEST. The default is to use the most recently created data set. The DATA= and INPVALUES= options cannot both be specified.

DEPENDENTFDR

DFDR

requests adjusted p-values by using the method of Benjamini and Yekutieli (2001). See the section "Dependent False Discovery Rate" on page 6447 for more details.

EPSILON=*number*

specifies the amount by which two p-values must differ to be declared unequal. The value *number* must be between 0 and 1; the default value is 1000 times the machine epsilon, which is approximately 1E–12. For SAS 9.1 and earlier releases the default value was 1E–8. See Westfall and Young (1993, pp. 165–166) for more information.

FDR

LSU

requests adjusted p-values by using the linear step-up method of Benjamini and Hochberg (1995). These p-values do not control the familywise error rate, but they do control the false discovery rate in some cases. See the section "False Discovery Rate Controlling Adjustments" on page 6446 for more details.

FDRBOOT<(*β*)>

A bootstrap-resampling false discovery rate controlling method due to Yekutieli and Benjamini (1999). This method uses the same resampling algorithm as the BOOTSTRAP option. Every resample is saved in order to compute a quantile of the resampled p-values; therefore, this method can use a lot

of memory. The parameter β designates that a $100(1-\beta)$ quantile is used in the computations for determining the adjustments; by default, $\beta = 0.05$. See the section "False Discovery Rate Resampling Adjustments" on page 6447 for details.

FDRPERM< (*β*) >

A permutation-resampling false discovery rate controlling method due to Yekutieli and Benjamini (1999). This method uses the same resampling algorithm as the PERMUTATION option. Every resample is saved in order to compute a quantile of the resampled *p*-values; therefore, this method can use a lot of memory. The parameter β designates that a $100(1-\beta)$ quantile is used in the computations for determining the adjustments; by default, $\beta = 0.05$. See the section "False Discovery Rate Resampling Adjustments" on page 6447 for details.

FISHER_C

FIC

requests adjusted *p*-values by using Fisher's combination method. See the section "Fisher Combination" on page 6445 for more details.

HOCHBERG

HOC

requests adjusted *p*-values by using the step-up Bonferroni method due to Hochberg (1988). See the section "Hochberg" on page 6445 for more details.

HOMMEL

HOM

requests adjusted *p*-values by using the method of Hommel (1988). See the section "Hommel" on page 6445 for more details.

HOLM

is an alias for the STEPBON adjustment.

INPVALUES< (*pvalue-name*) >=*SAS-data-set*

names an input SAS data set that includes a variable containing raw *p*-values. The MULTTEST procedure adjusts the collection of raw *p*-values for multiplicity. Resampling-based adjustments are not permitted with this type of data input. The CLASS, CONTRAST, FREQ, STRATA, and TEST statements are ignored when an INPVALUES= data set is specified. The INPVALUES= and DATA= options cannot both be specified. The *pvalue-name* enables you to specify the name of the *p*-value column from your data set. By default, *pvalue-name*='raw_p'. The INPVALUES= data set can contain variables in addition to the raw *p*-values variable; see Example 80.5 for an example.

LIPTAK

is an alias for the STOUFFER adjustment.

NOCENTER

requests that continuous variables not be mean-centered prior to resampling. The default action is to mean-center for bootstrap resampling and not to mean-center for permutation resampling.

NOPRINT

suppresses the normal display of results. Note that this option temporarily disables the Output Delivery System (ODS); see Chapter 20, "Using the Output Delivery System," for more information.

NOPVALUE
suppresses the display of the "p-Values" table of raw and adjusted p-values. This option is most useful when you are adjusting many tests and need to create only an OUT= data set or display graphics.

NOTABLES
suppresses display of the "Discrete Variable Tabulations" and "Continuous Variable Tabulations" tables.

NOZEROS
suppresses display of tables having zero occurrences for all CLASS levels.

NSAMPLE=*number*

N=*number*
specifies the number of resamples for use with the resampling methods. The value *number* must be a positive integer; by default, 20,000 resamples are used. Large values of *number* (20,000 or more) are usually recommended for accuracy, but long execution times can result, particularly with large data sets.

NTRUENULL=*keyword* | *value*

M0=*keyword* | *value*
Controls the method used to estimate the number of true NULL hypotheses (m_0) for the adaptive methods. This option is ignored unless one of the adaptive methods is specified. By default, PROC MULTTEST uses the DECREASESLOPE method for the ADAPTIVEHOLM and ADAPTIVEHOCHBERG adjustments, and the LOWESTSLOPE method for ADAPTIVEFDR adjustment. For the PFDR adjustment, the SPLINE method is attempted first. If the estimate is nonpositive or if the slope of the spline at the last λ is greater than 0.1 times the range of the fitted spline values, then the BOOTSTRAP method is used.

You can specify a positive integer as the *value*, or you can specify one of the *keywords* in the following list. Alternatively, you can specify the proportion of true NULL hypotheses by using the PTRUENULL= option. Suppose you have m tests with ordered p-values $p_{(1)} \leq \ldots \leq p_{(m)}$, and define $q_{(i)} = 1 - p_{(i)}$.

BOOTSTRAP< (*bootstrap-options***) >**
uses the bootstrap method of Storey and Tibshirani (2003). Compute the proportion of true null hypotheses $\hat{\pi}_0(\lambda) = \frac{m - N(\lambda) + f}{(1-\lambda)m}$ for $\lambda \in L = \{0, 0.05, \ldots, 0.95\}$, where $N(\lambda)$ is the number of p-values less than or equal to λ, and $f = 1$ for the finite-sample case; otherwise $f = 0$. For each λ, bootstrap on the p-values to form B bootstrap versions $\hat{\pi}_0^b(\lambda), b = 1, \ldots, B$, and choose the λ that yields the minimum $\widehat{\mathrm{MSE}}(\lambda) = \frac{1}{B}\sum_{b=1}^{B}(\hat{\pi}_0^b(\lambda) - \min_{\lambda' \in L}\hat{\pi}_0(\lambda'))^2$. The available *bootstrap-options* are as follows:

FINITE
the finite-sample case of the PFDR option, described on page 6424.

NBOOT=*B*
bootstrap resamples of the raw p-values for the λ computations. NBOOT=10,000 by default; *B* must be a positive integer.

NLAMBDA=*n*

"optimal" λ is the value in $\{0, \frac{1}{n}, \ldots, \frac{n-1}{n}\}$ that minimizes the MSE. NLAMBD=20 by default; *n* must be an integer greater than 1.

DECREASESLOPE

Schweder and Spjøtvoll (1982) as modified by Hochberg and Benjamini (1990). Let b_i be the slope of the least squares line fit to $\{q_{(m)}, \ldots, q_{(m-i+1)}\}$ and through the origin, for $i = 1, \ldots, m$. Find the first $i = m-1, m-2, \ldots, 1$ such that $b_i < b_{i+1}$. Then $\hat{m}_0 = \text{ceil}(\frac{1}{b_{i+1}} - 1)$.

KSTEST<(*β*)>

uses the Kolmogorov-Smirnov uniformity test method of Turkheimer, Smith, and Schmidt (2001). Let $k_{min} = 1, k_{max} = m$, and the Kolmogorov-Smirnov statistic $D = \max(q_{(i)} - i/(m+1)(\sqrt{k} + 0.12 + 0.11/\sqrt{k}))$. If D is greater than the upper-tail probability (Press et al. 1992), then $k_{max} = k, k = \text{floor}((k_{min} + k)/2)$; otherwise, let $k_{min} = k, k = \text{floor}((k + k_{max})/2)$. Repeat until $k = k_{min}$. Next compute the slope b of the weighted least squares regression line on the k smallest $q_{(i)}$ by using weights $w_i = i(k - i + 1)/((k+1)^2(k+2))$. Then $\hat{m}_0 = \text{ceil}(\frac{1}{b} - 1)$.

LEASTSQUARES

uses a linear least squares method to search for the correct cutpoint. For each $i = 0, \ldots, m$ compute the SSE of the least squares line through the origin fitting $\{q_{(m)}, \ldots, q_{(m-i+1)}\}$, let b_i be the slope of this line, and add the SSE of the unconstrained least squares line through the rest of the qs. For $i = 0$ compute the SSE for the unconstrained line. The argument i that minimizes the SSE is the cutpoint: if $i = 0$ then $\hat{m}_0 = 0$; if $i = m$ then $\hat{m}_0 = m$; otherwise $\hat{m}_0 = \text{ceil}(\frac{1}{b_i} - 1)$.

LOWESTSLOPE

uses the lowest slope method of Benjamini and Hochberg (2000). Find the first $i = 1, \ldots, m$ such that $b_i = q_{(i)}/(m - i + 1)$ decreases. Then $\hat{m}_0 = \text{floor}(\min(\frac{1}{b_i} + 1, m))$.

MEANDIFF

uses the mean of differences method of Hsueh, Chen, and Kodell (2003). Let $\bar{d}_i = \frac{q_{(m-i+1)}}{i}$ and estimate $\hat{m}_0^i = \frac{1}{\bar{d}_i} - 1$. Start from $i = m$ and proceed downward until the first time $\hat{m}_0^{i-1} \geq \hat{m}_0^i$ occurs.

SPLINE<(*spline-options*)>

uses the cubic spline method of Storey and Tibshirani (2003). For each $\lambda \in \{0, \frac{1}{n}, \frac{2}{n}, \ldots, \frac{n-1}{n}\}$ compute $\hat{\pi}_0(\lambda) = \frac{\#\{p_i > \lambda\}}{m(1-\lambda)}$. Let $\widehat{f}(\lambda)$ be the natural cubic spline with 3 degrees of freedom of $\hat{\pi}_0(\lambda)$ versus λ. Estimate $\hat{\pi}_0$ by taking the spline value at the last λ: $\hat{\pi}_0 = \hat{\pi}_0(\frac{n-1}{n})$, so that $\hat{m}_0 = m\hat{\pi}_0$. The available *spline-options* are as follows:

DF=*df*

sets the degrees of freedom of the spline, where *df* is a nonnegative integer. The default is DF=3.

DFCONV=*number*

specifies the absolute change in spline degrees of freedom value for concluding convergence. If $|df_i - df_{i+1}| <$ *number* (or if the SPCONV= criterion is satisfied), then convergence is declared. *number* must be between 0 and 1; by default, *number* is 1000 times the square root of machine epsilon, which is about 1E–5.

FINITE
computations for the finite-sample case of the PFDR option, described on page 6424.

MAXITER=*n*
specifies the maximum number of golden-search iterations used to find a spline with DF=df degrees of freedom. By default, MAXITER=100; *number* must be a nonnegative integer.

NLAMBDA=*n*
$\hat{\pi}_0(\lambda)$ for $\lambda \in \{0, \frac{1}{n}, \frac{2}{n}, \ldots, \frac{n-1}{n}\}$ for the spline fit. By default, NLAMBDA=20; *number* must be an integer greater than 1.

SPCONV=*number*
specifies the absolute change in smoothing parameter value for concluding convergence of the spline. If $|sp_i - sp_{i+1}| <$ *number* (or if the DFCONV= criterion is satisfied), then convergence is declared. By default, *number* equals the square root of the machine epsilon, which is about 1E–8.

In all cases $\hat{m}_0$ is constrained to lie between 0 and m; if the computed $\hat{m}_0 = 0$, then the adaptive adjustments do not produce results. If you specify $\hat{m}_0 > m$, then it is reduced to m. Values of $\hat{m}_0$ are displayed in the "Estimated Number of True Null Hypotheses" table.

ORDER=DATA | FORMATTED | FREQ | INTERNAL
specifies the sort order for the levels of the classification variables (which are specified in the CLASS statement). This option applies to the levels for all classification variables, except when you use the (default) ORDER=FORMATTED option with numeric classification variables that have no explicit format. In that case, the levels of such variables are ordered by their internal value.

The ORDER= option can take the following values:

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

By default, ORDER=FORMATTED. For ORDER=FORMATTED and ORDER=INTERNAL, the sort order is machine-dependent. For more information about sort 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*.

OUT=*SAS-data-set*
names the output SAS data set containing variable names, contrast names, intermediate calculations, and all associated p-values. See "OUT= Data Set" on page 6450 for more information.

OUTPERM=*SAS-data-set*
names the output SAS data set containing entire permutation distributions (upper-tail probabilities) for all tests when the PERMUTATION= option is specified. See "OUTPERM= Data Set" on page 6451 for more information. **CAUTION:** This data set can be very large.

OUTSAMP=*SAS-data-set*
names the output SAS data set containing information from the resampled data sets when resampling is performed. See "OUTSAMP= Data Set" on page 6451 for more information. **CAUTION:** This data set can be very large.

PDATA=*SAS-data-set*
is an alias for the INPVALUES= option.

PERMUTATION
PERM
computes adjusted p-values in identical fashion as the BOOTSTRAP option, with the exception that PROC MULTTEST resamples without replacement rather than with replacement. Resampling is performed independently within levels of the STRATA variable. Continuous variables are not mean-centered prior to resampling; specify the CENTER to change this. See the section "Bootstrap" on page 6443 for more details. The PERMUTATION option is not allowed with the Peto test.

PFDR<(*options***)>**
computes the "q-values" $\hat{q}_\lambda(p_i)$ of Storey (2002) and Storey, Taylor, and Siegmund (2004). PROC MULTTEST treats these "q-values" as adjusted p-values. The computations depend on selecting a parameter λ and an estimation method for the false discovery rate; see the section "Positive False Discovery Rate" on page 6448 for computational details. The available *options* for choosing the method are as follows:

FINITE
estimates the false discovery rate with $\widehat{\text{pFDR}}$ or $\widehat{\text{FDR}}$ for the finite-sample case with independent null p-values.

POSITIVE
estimates the false discovery rate with $\widehat{\text{pFDR}}$ instead of the default $\widehat{\text{FDR}}$.

UNRESTRICT
estimates the false discovery rate as defined in Storey (2002), which allows the adjustment to reduce the raw p-value. By default, the PFDR is constrained to be greater than or equal to the raw p-value.

The available *options* for controlling the λ search are the *bootstrap-options* (page 6421), the *spline-options* (page 6422), and the following *options*:

LAMBDA=*number*
specifies a $\lambda \in [0, 1)$ and does not perform the bootstrap or spline searches for an "optimal" λ.

MAXLAMBDA=*number*
stops the NLAMBDA= search sequence for the bootstrap and spline searches when this *number* is reached. The *number* must be in $[0, 1]$. This option is ignored if the LAMBDA= option is specified.

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

PLOTS<(*global-plot-options***)>=(***plot-request***<(***options***)>< . . . *plot-request*<(*options*)>>)**

controls the plots produced through ODS Graphics. If you specify only one *plot-request*, you can omit the parentheses. For example, the following statements are valid specifications of the PLOTS= option:

```
plots = all
plots = (rawprob adjusted)
plots(sigonly) = (rawprob adjusted(unpack))
```

ODS Graphics must be enabled before plots can be requested. For example:

```
ods graphics on;
proc multtest plots=adjusted inpvalues=a pfdr;
run;
ods graphics off;
```

For more information about enabling and disabling ODS Graphics, see the section "Enabling and Disabling ODS Graphics" on page 607 in Chapter 21, "Statistical Graphics Using ODS."

By default, no graphs are created; you must specify the PLOTS= option to make graphs. You need at least two tests to produce a graph. If you are not using an INPVALUES= data set, then each test is given a name constructed as "variable-name contrast-label". If you specify a MEAN test in the TEST statement, the *t*-test names are prefixed with "Mean:". See Example 80.6 for examples of the ODS graphical displays.

The following *global-plot-options* are available:

UNPACKPANELS | UNPACK

suppresses paneling. By default, the plots produced with the ADJUSTED and RAWPROB options are grouped in a single display, called a *panel*. Specify UNPACK to display each plot separately.

SIGONLY<=*number***>**

displays only those tests with adjusted p-values $\leq$ *number*, where $0 \leq$ *number* ≤ 1. By default, *number* = 0.05.

The following *plot-requests* are available:

ADJUSTED<(UNPACK)>

displays a 2×2 panel of adjusted p-value plots similar to those Storey and Tibshirani (2003) developed for use with the PFDR p-value adjustment method. The plots of the adjusted p-values by the raw p-values and the adjusted p-values by their rank show the effect of the adjustments. The plot of the proportion of adjusted p-values $\leq$ each adjusted p-value and the plot of the expected number of false positives (the proportion significant multiplied by the adjusted p-value) versus the proportion significant show the effect of choosing different significance levels. The UNPACK option unpanels the display.

ALL
produces all appropriate plots. You can specify other options with ALL; for example, to display all plots and unpack the RAWPROB plots you can specify **plots=(all rawprob(unpack))**.

LAMBDA
displays plots of the MSE and the estimated number of true nulls against the λ parameter when the NTRUENULL=SPLINE or NTRUENULL=BOOTSTRAP option is in effect.

MANHATTAN< (*options*) >
displays the Manhattan plot (a plot of $-\log_{10}$ of the adjusted *p*-values versus the tests). You can specify the following *options*:

GROUP=*variable*
specifies a variable to group the adjusted *p*-values in the display.

LABEL < =OBS >
labels the observations that have adjusted *p*-values that are less than the value specified in the VREF= option. By default, labels are created as follows: if an INPVALUES= data set and an ID statement are specified, then the observations are labeled with the ID values; if a DATA= data set is specified, then the observations are labeled with their constructed test name; otherwise, the observation or test number is displayed.

NOTESTNAME
displays the number of the test instead of the test name on the X-axis, which is useful when you have many tests.

UNPACK
suppresses paneling. By default, Manhattan plots are created for each requested *p*-value adjustment, and the results are grouped in a single display, called a *panel*. Specify UNPACK to display each plot separately.

VREF=*number* | NONE
displays a reference line at $-\log_{10}(\textit{number})$. The *number* must be between 0 and 1. By default, a reference line at $-\log_{10}(0.05)$ is displayed; it can be suppressed by specifying VREF=0 or VREF=NONE. If the LABEL option is also specified, then observations above this line are labeled with their ID variables, their observation number, their test name, or their test number.

NONE
suppresses all plots.

PBYTEST< (*options*) >
displays the adjusted *p*-values for each test. The available options are as follows:

NOTESTNAME
displays the number of the test instead of the test name on the axis, which is useful when you have many tests.

VREF=

number-list displays reference lines at the p-values specified in the *number-list*. The values in the *number-list* must be between 0 and 1; otherwise they are ignored. You can specify a single value or a list of values; for example, `vref=0.1 0 to 0.05 by 0.01` displays reference lines at each of the values {0.01, 0.02, 0.03, 0.04, 0.05, and 0.1}.

RAWPROB< (UNPACK) >

displays a uniform probability plot of 1 minus the raw p-values (Schweder and Spjøtvoll 1982) along with a histogram. If m_0 is the number of true null hypotheses among the m tests, the points on the left side of the plot should be approximately linear with slope $\frac{1}{m_0+1}$. This graphic is displayed when an adaptive p-value adjustment method is requested in order to see if the NTRUENULL= estimate is appropriate. The UNPACK option unpanels the display.

PTRUENULL=*keyword* | *value*

PI0=*keyword* | *value*

is alias for the NTRUENULL= option, except that you can specify the proportion of true null hypotheses as a *value* between 0 and 1, instead of specifying the number of true null hypotheses. The available *keywords* are also the NTRUENULL= options described on page 6421.

RANUNI

requests the random number generator used in releases prior to SAS 9.2. Beginning with SAS 9.2, the random number generator is the Mersenne Twister, which has better performance when bootstrapping. Changes in the bootstrap- or permutation-adjusted p-values from prior releases are due to unimportant sampling differences.

SEED=*number*

S=*number*

specifies the initial seed for the random number generator used for resampling. The value for *number* must be an integer. If you do not specify a seed, or if you specify a value less than or equal to zero, then PROC MULTTEST uses the time of day from the computer's clock to generate an initial seed. For more details about seed values, see *SAS Language Reference: Concepts*.

SIDAK

SID

computes the Šidák adjustment for each test. These adjustments take the form

$$1 - (1 - p)^m$$

where p is the raw p-value and m is the number of tests. These are slightly less conservative than the Bonferroni adjustments, but they still should be viewed with caution. When exact tests are specified via the PERMUTATION= option in the TEST statement, the actual permutation distributions are used, resulting in a much less conservative version of this procedure (Westfall and Wolfinger 1997). See the section "Šidák" on page 6443 for more details.

STEPBON

HOLM

requests adjusted p-values by using the step-down Bonferroni method of Holm (1979). See the section "Step-Down Methods" on page 6444 for more details.

STEPBOOT
requests that adjusted *p*-values be computed by using bootstrap resampling as described under the BOOTSTRAP option, but in step-down fashion. See the section "Step-Down Methods" on page 6444 for more details.

STEPPERM
requests that adjusted *p*-values be computed by using permutation resampling as described under the PERMUTATION option, but in step-down fashion. See the section "Step-Down Methods" on page 6444 for more details.

STEPSID
requests adjusted *p*-values by using the Šidák method as described in the SIDAK option, but in step-down fashion. See the section "Step-Down Methods" on page 6444 for more details.

STOUFFER
LIPTAK
requests adjusted *p*-values by using the Stouffer-Liptak combination method. See the section "Stouffer-Liptak Combination" on page 6445 for more details.

BY Statement

BY *variables* **;**

You can specify a BY statement with PROC MULTTEST to obtain separate analyses of observations in groups that are defined by the BY variables. When a BY statement appears, the procedure expects the input data set to be sorted in order of the BY variables. If you specify more than one BY statement, only the last one specified is used.

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 NOTSORTED or DESCENDING option in the BY statement for the MULTTEST 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).

You can specify one or more *variables* in the input data set on the BY statement.

Since sorting the data changes the order in which PROC MULTTEST reads observations, this can affect the sort order for the levels of the CLASS variable if you have specified ORDER=DATA in the PROC MULTTEST statement. This, in turn, affects specifications in the CONTRAST statements.

For more information about BY-group processing, see the discussion in *SAS Language Reference: Concepts*. For more information about the DATASETS procedure, see the discussion in the *Base SAS Procedures Guide*.

CLASS Statement

CLASS *variable* < / **TRUNCATE** > **;**

The CLASS statement is required unless the INPVALUES= option is specified. The CLASS statement specifies a single variable (character or numeric) used to identify the groups for the analysis. For example, if the variable Treatment defines different levels of a treatment that you want to compare, then you would specify the following statements:

```
class Treatment;
```

The CLASS variable can be either character or numeric. By default, class levels are determined from the entire set of formatted values of the CLASS variable. The order of the class levels used by PROC MULTTEST corresponds to the order of their formatted values; this order can be changed with the ORDER= option in the PROC MULTTEST statement.

NOTE: Prior to SAS 9, class levels were determined by using no more than the first 16 characters of the formatted values. To revert to this previous behavior you can specify the TRUNCATE option in the CLASS statement.

In any case, 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 Formats and Informats: Reference*. You can adjust the order of CLASS variable levels with the ORDER= option in the PROC MULTTEST statement. You need to be aware of the order when using the CONTRAST statement, and you should check the "Contrast Coefficients" table to verify that it is suitable.

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

TRUNCATE
specifies that class levels should be determined by using only up to the first 16 characters of the formatted values of CLASS variables. When formatted values are longer than 16 characters, you can use this option to revert to the levels as determined in releases prior to SAS 9.

CONTRAST Statement

CONTRAST '*label*' *values* **;**

This statement is used to identify tests between the levels of the CLASS variable; in particular, it is used to specify the coefficients for the trend tests. The *label* is a string naming the contrast; it contains a maximum of 21 characters. The *values* are scoring coefficients across the CLASS variable levels.

You can specify multiple CONTRAST statements, thereby specifying multiple contrasts for each variable. Multiplicity adjustments are computed for all contrasts and all variables simultaneously. The coefficients are applied to the ordered CLASS variables; this order can be changed with the ORDER= option in the PROC MULTTEST statement. For example, consider a four-group experiment with CLASS variable levels A1, A2, B1, and B2 denoting two levels of two treatments. The following statements produce three linear trend tests for each variable identified in the TEST statement. PROC MULTTEST computes the multiplicity adjustments over the entire collection of tests, which is three times the number of variables.

```
contrast 'a vs b'    -1 -1  1  1;
contrast 'a linear'  -1  1  0  0;
contrast 'b linear'   0  0 -1  1;
```

As another example, consider an animal carcinogenicity experiment with dose levels 0, 4, 8, 16, and 50. You can specify a trend test with the indicated scoring coefficients by using the following statement:

```
contrast 'arithmetic trend' 0 4 8 16 50;
```

Multiplicity-adjusted *p*-values are then computed over the collection of variables identified in the TEST statement. See Lagakos and Louis (1985) for guidelines on the selection of contrast-scoring values.

When a Fisher test is specified in the TEST statement, the CONTRAST statement coefficients are used to group the CLASS variable's levels. Groups with a –1 contrast coefficient are combined and compared with groups with a 1 contrast coefficient for each test, and groups with a 0 coefficient are not included in the contrast. For example, the following statements compute Fisher exact tests for (a) control versus the combined treatment groups, (b) control versus the first treatment group, and (c) control versus the third treatment group:

```
contrast 'c vs all' 1 -1 -1 -1;
contrast 'c vs t1'  1 -1  0  0;
contrast 'c vs t3'  1  0  0 -1;
```

Multiplicity adjustments are then computed over the entire collection of tests and variables. Only –1, 1, and 0 are acceptable CONTRAST coefficients when the Fisher test is specified; PROC MULTTEST ignores the CONTRAST statement if any other coefficients appear.

If you specify the FISHER test and no CONTRAST statements, then all contrasts of control versus treatment are automatically generated, with the first level of the CLASS variable deemed to be the control. In this case, the control level is assigned the value 1 in each contrast and the other treatment levels are assigned –1. You should therefore use the LOWERTAILED option to test for higher success rates in the treatment groups.

For tests other than FISHER, CONTRAST values are $0, 1, 2, \ldots$ by default. If you specify the CA or PETO test with the PERMUTATION= option, then your CONTRAST coefficients must be integer valued.

For *t* tests for the mean of continuous data (and for the FT tests), the contrast coefficients are centered to have mean $= 0$. The resulting centered scoring coefficients are then applied to the sample means (or to the double-arcsine-transformed proportions in the case of the FT tests).

FREQ Statement

FREQ *variable* **;**

The FREQ statement names a variable that provides frequencies for each observation in the DATA= data set. Specifically, if *n* is the value of the FREQ variable for a given observation, then that observation is used *n* times.

If the value of the FREQ variable is missing or is less than 1, the observation is not used in the analysis. If the value is not an integer, only the integer portion is used.

ID Statement

ID *variables* **;**

The ID statement names one or more variables for identifying observations in the output and in the plots. The statement requires an INPVALUES= data set. All ID variables are displayed in the "pValues" table. The ID variables are used as the X axis for the plots requested by the PLOTS=PBYTEST and PLOTS=MANHATTAN options in the PROC MULTTEST statement; they are also used to label points on the Manhattan plots. This option has no effect on the OUT= data set.

STRATA Statement

STRATA *variable* **;**

The STRATA statement identifies a single variable to use as a stratification variable in the analysis. This yields tests similar to those discussed in Mantel and Haenszel (1959) and Hoel and Walburg (1972) for binary data and pooled-means tests for continuous data. For example, when you test for prevalence in a carcinogenicity study, it is common to stratify on intervals of the time of death; the first level of the stratification variable might represent weeks 0–52, the second might represent weeks 53–80, and so on. In multicenter clinical studies, each level of the stratification variable might represent a particular center.

The following option is available in the STRATA statement after a slash (/):

WEIGHT=*keyword*

specifies the type of strata weighting to use when computing the Freeman-Tukey and *t* tests. Valid *keywords* are SAMPLESIZE, HARMONIC, and EQUAL. SAMPLESIZE requests weights proportional to the within-stratum sample sizes, and is the default method even if the WEIGHT= option is not specified. HARMONIC sets up weights equal to the harmonic mean of the nonmissing within-stratum CLASS sizes, and is similar to a Type 2 analysis in PROC GLM. EQUAL specifies equal weights, and is similar to a Type 3 analysis in PROC GLM.

TEST Statement

TEST *name* **(***variables* < */ options* >**) ;**

The TEST statement is required unless the INPVALUES= option is specified. The TEST statement identifies statistical tests to be performed and the discrete and continuous variables to be tested. Table 80.2 summarizes the names and options available in the TEST statement.

Table 80.2 TEST Statement Names and Options

Option	Description
TEST Names	
CA	Requests the Cochran-Armitage linear trend tests for group comparisons
FISHER	Requests Fisher exact tests
FT	Requests *Z*-score CA tests based upon the Freeman-Tukey double arcsine transformation
MEAN	Requests the *t* test for the mean
PETO	Requests the Peto mortality-prevalence test
TEST Options	
BINOMIAL	Uses the binomial variance estimate for CA and Peto tests
CONTINUITY=	Specifies a continuity correction
DDFM=	Specifies whether to use homogeneous or heterogeneous variances
LOWERTAILED	Makes all tests lower-tailed
PERMUTATION=	Computes *p*-values for the CA and Peto tests by using exact permutation distributions
TIME=	Identifies the Peto test variable containing the age at death
UPPERTAILED	Makes all tests upper-tailed

The following tests are permitted as *name* in the TEST statement.

CA

requests the Cochran-Armitage linear trend tests for group comparisons. The test variables should take the value 0 for a failure and 1 for a success. PERMUTATION= option can be used to request an exact permutation test; otherwise, a *Z*-score approximation is used. The CONTINUITY= option can be used to specify a continuity correction for the *Z*-score approximation.

FISHER

requests Fisher exact tests for comparing two treatment groups. The test variables should take the value 0 for a failure and 1 for a success.

FT

requests *Z*-score CA tests based upon the Freeman-Tukey double arcsine transformation of the frequencies. The test variables should take the value 0 for a failure and 1 for a success.

MEAN

requests the *t* test for the mean. The test variables can take on any numeric values.

PETO

requests the Peto mortality-prevalence test. The test variables should take the value 0 for a nonoccurrence, 1 for an incidental occurrence, and 2 for a fatal occurrence. The TIME= option should be used with the Peto test to specify an integer-valued variable giving the age at death. The CONTINUITY= option can be used to specify a continuity correction for the test.

If the value of a TEST variable is invalid, the observation is not used in the analysis. You can specify two tests only if one of them is MEAN. For example, the following statement is valid:

```
test ca(d1-d2) mean(c1-c2);
```

But specifying both CA and FT, as shown in the following statement, is invalid:

```
test ca(d1-d2) ft(d1-d2);
```

You can specify the following *options* in the TEST statement (some apply to only one test).

BINOMIAL
uses the binomial variance estimate for CA and Peto tests in their asymptotic normal approximations. The default is to use the hypergeometric variance.

CONTINUITY=*number*

C=*number*
specifies *number* as a particular continuity correction for the Z-score approximation in the CA and Peto tests. The default is 0.

LOWERTAILED

LOWER
is used to make all tests lower-tailed. All tests are two-tailed by default.

PERMUTATION=*number*

PERM=*number*
computes p-values for the CA and Peto tests by using exact permutation distributions when marginal success or failure totals within a stratum are *number* or less. You can specify *number* as a nonnegative integer. For totals greater than *number* (or when the PERMUTATION= option is omitted), PROC MULTTEST uses standard normal approximations with a continuity correction chosen to approximate the permutation distribution. PROC MULTTEST computes the appropriate convolution distributions when you use the STRATA statement along with the PERMUTATION= option.

DDFM= POOLED | SATTERTHWAITE
specifies whether the MEAN test uses a homogeneity assumption (DDFM=POOLED, the default) or deals with heterogeneous variances (DDFM=SATTERTHWAITE). See "t Test for the Mean" on page 6440 for more information.

TIME=*variable*
identifies the Peto test variable containing the age at death, which must be integer valued. If the TIME= option is omitted, all ages are assumed to equal 1.

UPPERTAILED

UPPER
is used to make all tests upper-tailed. All tests are two-tailed by default.

Details: MULTTEST Procedure

Statistical Tests

The following section discusses the statistical tests performed in the MULTTEST procedure. For continuous data, a *t* test for the mean (MEAN) is available. For discrete variables, available tests are the Cochran-Armitage linear trend test (CA), the Freeman-Tukey double arcsine test (FT), the Peto mortality-prevalence test (PETO), and the Fisher exact test (FISHER).

Throughout this section, the discrete and continuous variables are denoted by S_{vgsr} and X_{vgsr}, respectively, where v is the variable, g is the treatment group, s is the stratum, and r is the replication. Let m_{vgs} denote the sample size for a binary variable v within group g and stratum s. A plus sign (+) subscript denotes summation over an index. Note that the tests are invariant to the location and scale of the contrast coefficients t_g.

Cochran-Armitage Linear Trend Test

The Cochran-Armitage linear trend test (Cochran 1954; Armitage 1955; Agresti 2002) is implemented by using a *Z*-score approximation, an exact permutation distribution, or a combination of both.

Z-Score Approximation

The pooled probability estimate for variable v and stratum s is

$$p_{vs} = \frac{S_{v+s+}}{m_{v+s}}$$

The expected value (under constant within-stratum treatment probabilities) for variable v, group g, and stratum s is

$$E_{vgs} = m_{vgs}\, p_{vs}$$

Letting t_g denote the contrast trend coefficients specified by the CONTRAST statement, the test statistic for variable v has numerator

$$N_v = \sum_s \sum_g t_g (S_{vgs+} - E_{vgs})$$

The binomial variance estimate for this statistic is

$$V_v = \sum_s p_{vs}(1 - p_{vs}) \sum_g m_{vgs}(t_g - \bar{t}_{vs})^2$$

where

$$\bar{t}_{vs} = \sum_g \frac{m_{vgs} t_g}{m_{v+s}}$$

The hypergeometric variance estimate (the default) is

$$V_v = \sum_s \{m_{v+s}/(m_{v+s} - 1)\} p_{vs}(1 - p_{vs}) \sum_g m_{vgs}(t_g - \bar{t}_{vs})^2$$

For any strata s with $m_{v+s} \leq 1$, the contribution to the variance is taken to be zero.

PROC MULTTEST computes the Z-score statistic

$$Z_v = \frac{N_v}{\sqrt{V_v}}$$

The p-value for this statistic comes from the standard normal distribution. Whenever a 0 is computed for the denominator, the p-value is set to 1. This p-value approximates the probability obtained from the exact permutation distribution, discussed in the following text.

The Z-score statistic can be continuity-corrected to better approximate the permutation distribution. With continuity correction c, the upper-tailed p-value is computed from

$$Z_v = \frac{N_v - c}{\sqrt{V_v}}$$

For two-tailed, noncontinuity-corrected tests, PROC MULTTEST reports the p-value as $2\min(p, 1-p)$, where p is the upper-tailed p-value. The same formula holds for the continuity-corrected test, with the exception that when the noncontinuity-corrected Z and the continuity-corrected Z have opposite signs, the two-tailed p-value is 1.

When the PERMUTATION= option is specified and no STRATA variable is specified, PROC MULTTEST uses a continuity correction selected to optimally approximate the upper-tail probability of permutation distributions with smaller marginal totals (Westfall and Lin 1988). Otherwise, the continuity correction is specified by the CONTINUITY= option in the TEST statement.

The CA Z-score statistic is the Hoel-Walburg (Mantel-Haenszel) statistic reported by Dinse (1985).

Exact Permutation Test

When you use the PERMUTATION= option for CA in the TEST statement, PROC MULTTEST computes the exact permutation distribution of the trend score

$$T_v = \sum_s \sum_g t_g S_{vgs+}$$

where the contrast trend coefficients t_g must be integer valued. The observed value of this trend is compared to the permutation distribution to obtain the p-value

$$p_v = \Pr(X \geq \text{ observed } T_v)$$

where X is a random variable from the permutation distribution and where upper-tailed tests are requested. This probability can be viewed as a binomial probability, where the within-stratum probabilities are constant and where the probability is conditional with respect to the marginal totals S_{v+s+}. It also can be considered a rerandomization probability.

Because the computations can be quite time-consuming with large data sets, specifying the PERMUTATION=*number* option in the TEST statement limits the situations where PROC MULTTEST computes the exact permutation distribution. When marginal total success or total failure frequencies exceed *number* for a particular stratum, the permutation distribution is approximated by a continuity-corrected normal distribution. You should be cautious when using the PERMUTATION= option in conjunction with bootstrap resampling because the permutation distribution is recomputed for each bootstrap sample. This recomputation is not necessary with permutation resampling.

The permutation distribution is computed in two steps:

1. The permutation distributions of the trend scores are computed within each stratum.
2. The distributions are convolved to obtain the distribution of the total trend.

As long as the total success or failure frequency does not exceed *number* for any stratum, the computed distributions are exact. In other words, if $S_{v+s+} \leq$ *number* or $(m_{v+s} - S_{v+s+}) \leq$ *number* for all s, then the permutation trend distribution for variable v is computed exactly.

In step 1, the distribution of the within-stratum trend

$$\sum_g t_g S_{vgs+}$$

is computed by using the multivariate hypergeometric distribution of the S_{vgs+}, provided *number* is not exceeded. This distribution can be written as

$$\Pr(S_{v1s+}, S_{v2s+}, \ldots, S_{vGs+}) = \prod_{g=1}^{G} \frac{\begin{pmatrix} m_{vgs} \\ S_{vgs+} \end{pmatrix}}{\begin{pmatrix} m_{v+s} \\ S_{v+s+} \end{pmatrix}}$$

The distribution of the within-stratum trend is then computed by summing these probabilities over appropriate configurations. For further information about this technique, see Bickis and Krewski (1986) and Westfall and Lin (1988). In step 2, the exact convolution distribution is obtained for the trend statistic summed over all strata having totals that meet the threshold criterion. This distribution is obtained by applying the fast Fourier transform to the exact within-stratum distributions. A description of this general method can be found in Pagano and Tritchler (1983) and Good (1987).

The convolution distribution of the overall trend is then computed by convolving the exact distribution with the distribution of the continuity-corrected standard normal approximation. To be more specific, let S_1 denote the subset of stratum indices that satisfy the threshold criterion, and let S_2 denote the subset of indices that do not satisfy the criterion. Let T_{v1} denote the combined trend statistic from the set S_1, which has an exact distribution obtained from Fourier analysis as previously outlined, and let T_{v1} denote the combined trend statistic from the set S_2. Then the distribution of the overall trend $T_v = T_{v1} + T_{v2}$ is obtained by convolving the analytic distribution of T_{v1} with the continuity-corrected normal approximation for T_{v2}. Using the notation from the section "Z-Score Approximation" on page 6434, this convolution can be written as

$$\begin{aligned} \Pr(T_{v1} + T_{v2} \geq u) &= \sum_{u1} \Pr(T_{v1} + T_{v2} \geq u \mid T_{v1} = u1) \Pr(T_{v1} = u1) \\ &\approx \sum_{u1} \Pr(Z \geq z) \Pr(T_{v1} = u1) \end{aligned}$$

where Z is a standard normal random variable, and

$$z = \frac{1}{\sqrt{V_v}} \left(u - u1 - \sum_{S_2} p_{vs} \sum_g t_g m_{vgs} - c \right)$$

In this expression, the summation of s in V_v is over S_2, and c is the continuity correction discussed under the Z-score approximation.

When a two-tailed test is requested, the expected trend is computed

$$E_v = \sum_s \sum_g t_g E_{vgs}$$

The two-tailed p-value is reported as the permutation tail probability for the observed trend T_v plus the permutation tail probability for $2E_v - T_v$, the reflected trend.

Freeman-Tukey Double Arcsine Test

For this test, the contrast trend coefficients $t_1, \ldots, t_G$ are centered to the values $c_1, \ldots, c_G$, where $c_g = t_g - \bar{t}$, $\bar{t} = \sum_g t_g / G$, and G is the number of groups. The numerator of this test statistic is

$$N_v = \sum_s w_{vs} \sum_g c_g f(S_{vgs+}, m_{vgs})$$

where the weights w_{vs} take on three different types of values depending upon your specification of the WEIGHT= option in the STRATA statement. The default value is the within-strata sample size m_{v+s}, ensuring comparability with the ordinary CA trend statistic. WEIGHT=HARMONIC sets w_{vs} equal to the harmonic mean

$$\left[\left(\sum_g \frac{1}{m_{vgs}} \right) / G^* \right]^{-1}$$

where G^* is the number of nonmissing groups and the summation is over only the nonmissing elements. The harmonic means analysis places more weight on the smaller sample sizes than does the default sample size method, and is similar to a Type 2 analysis in PROC GLM. WEIGHT=EQUAL sets $w_{vs} = 1$ for all v and s, and is similar to a Type 3 analysis in PROC GLM.

The function $f(r, n)$ is the double arcsine transformation:

$$f(r, n) = \arcsin\left(\sqrt{\frac{r}{n+1}} \right) + \arcsin\left(\sqrt{\frac{r+1}{n+1}} \right)$$

The variance estimate is

$$V_v = \sum_s w_{vs}^2 \sum_g \frac{c_g^2}{m_{vgs} + \frac{1}{2}}$$

The test statistic is

$$Z_v = \frac{N_v}{\sqrt{V_v}}$$

The Freeman-Tukey transformation and its variance are described by Freeman and Tukey (1950) and Miller (1978). Since its variance is not weighted by the pooled probabilities, as is the CA test, the FT test can be more useful than the CA test for tests involving only a subset of the groups.

Peto Mortality-Prevalence Trend Test

The Peto test is a modified Cochran-Armitage procedure incorporating mortality and prevalence information. The Peto test is computed like two Cochran-Armitage Z-score approximations, one for prevalence and one for mortality (Peto et al. 1980). It represents a special case in PROC MULTTEST because the data structure requirements are different, and the resampling methods used for adjusting p-values are not valid. The TIME= option variable is required to specify "death" times or, more generally, times of occurrence. In addition, the test variables must assume one of the following three values:

- 0 = no occurrence
- 1 = incidental occurrence
- 2 = fatal occurrence

Use the TIME= option variable to define the mortality strata, and use the STRATA statement variable to define the prevalence strata.

In the following notation, the subscript v represents the variable, g represents the treatment group, s represents the stratum, and t represents the time. Recall that a plus sign $(+)$ in a subscript location denotes summation over that subscript.

Let S^P_{vgs} be the number of incidental occurrences, and let m^P_{vgs} be the total sample size for variable v in group g, stratum s, excluding fatal tumors.

Let S^F_{vgt} be the number of fatal occurrences in time period t, and let m^F_{vgt} be the number of patients alive at the end of time $t-1$.

The pooled probability estimates are given by

$$\begin{aligned} p^P_{vs} &= \frac{S^P_{v+s}}{m^P_{v+s}} \\ p^F_{vt} &= \frac{S^F_{v+t}}{m^F_{v+t}} \end{aligned}$$

The expected values are

$$\begin{aligned} E^P_{vgs} &= m^P_{vgs} p^P_{vs} \\ E^F_{vgt} &= m^F_{vgt} p^F_{vt} \end{aligned}$$

Let t_g denote a contrast trend coefficient, and define the numerator terms as follows:

$$\begin{aligned} N^P_v &= \sum_s \sum_g t_g \left(S^P_{vgs} - E^P_{vgs} \right) \\ N^F_v &= \sum_t \sum_g t_g \left(S^F_{vgt} - E^F_{vgt} \right) \end{aligned}$$

Define the denominator variance terms by using the binomial variance:

$$V_v^P = \sum_s p_{vs}^P \left(1 - p_{vs}^P\right) \left[\left(\sum_g m_{vgs}^P {t_g}^2 \right) - \frac{1}{m_{v+s}^P} \left(\sum_g m_{vgs}^P t_g \right)^2 \right]$$

$$V_v^F = \sum_s p_{vt}^F \left(1 - p_{vt}^F\right) \left[\left(\sum_g m_{vgt}^F {t_g}^2 \right) - \frac{1}{m_{v+t}^F} \left(\sum_g m_{vgt}^F t_g \right)^2 \right]$$

The hypergeometric variances (the default) are calculated by weighting the within-strata variances as discussed in the section "Z-Score Approximation" on page 6434.

The Peto statistic is computed as

$$Z_v = \frac{N_v^P + N_v^F - c}{\sqrt{V_v^P + V_v^F}}$$

where c is a continuity correction. The p-value is determined from the standard normal distribution unless the PERMUTATION=*number* option is used. When you use the PERMUTATION= option for PETO in the TEST statement, PROC MULTTEST computes the "discrete approximation" permutation distribution described by Mantel (1980) and Soper and Tonkonoh (1993). Specifically, the permutation distribution of $\sum_s \sum_g t_g S_{vgs}^P + \sum_t \sum_g t_g S_{vgt}^F$ is computed, assuming that $\{\sum_g t_g S_{vgs}^P\}$ and $\{\sum_g t_g S_{vgt}^F\}$ are independent over all s and t. Note that the contrast trend coefficients t_g must be integer valued. The p-values are exact under this independence assumption. However, the independence assumption is valid only asymptotically, which is why these p-values are called "approximate."

An exact permutation distribution is available only under the assumption of equal risk of censoring in all treatment groups; even then, computing this distribution can be cumbersome. Soper and Tonkonoh (1993) describe situations where the discrete approximation distribution closely fits the exact permutation distribution.

Fisher Exact Test

The CONTRAST statement in PROC MULTTEST enables you to compute Fisher exact tests for two-group comparisons. No stratification variable is allowed for this test. Note, however, that the FISHER exact test is a special case of the exact permutation tests performed by PROC MULTTEST and that these permutation tests allow a stratification variable. Recall that contrast coefficients can be –1, 0, or 1 for the Fisher test. The frequencies and sample sizes of the groups scored as –1 are combined, as are the frequencies and sample sizes of the groups scored as 1. Groups scored as 0 are excluded. The –1 group is then compared with the 1 group by using the Fisher exact test.

Letting x and m denote the frequency and sample size of the 1 group, and letting y and n denote those of the –1 group, the p-value is calculated as

$$\Pr(X \geq x \mid X + Y = x + y) = \sum_{i=x}^{m} \frac{\binom{m}{i}\binom{n}{x+y-i}}{\binom{m+n}{x+y}}$$

where X and Y are independent binomially distributed random variables with sample sizes m and n and common probability parameters. The hypergeometric distribution is used to determine the stated probability; Yates (1984) discusses this technique. PROC MULTTEST computes the two-tailed p-values by adding probabilities from both tails of the hypergeometric distribution. The first tail is from the observed x and y, and the other tail is chosen so that the resulting probability is as large as possible without exceeding the probability from the first tail. If the variable being tested has only one level, then the p-value is set to 1.

t Test for the Mean

For continuous variables, PROC MULTTEST automatically centers the contrast trend coefficients, as in the Freeman-Tukey test. These centered coefficients c_g are then used to form a t statistic contrasting the within-group means. Let n_{vgs} denote the sample size within group g and stratum s; it depends on variable v only when there are missing values. Determine the weights w_{vs} as in the Freeman-Tukey test with n_{vgs} replacing m_{vgs}. Define

$$\bar{X}_{vgs+} = \frac{1}{n_{vgs}} \sum_r X_{vgsr}$$

as the sample mean within a group-and-stratum combination, and let μ_{vgs} denote the treatment means. Write the null hypothesis as

$$\sum_s w_{vs} \sum_g c_g \mu_{vgs} = 0$$

Also define

$$s_v^2 = \frac{\sum_s \sum_g \sum_r \left(X_{vgsr} - \bar{X}_{vgs+}\right)^2}{\sum_s \sum_g \left(n_{vgs} - 1\right)}$$

as the pooled sample variance.

Homogeneous Variance

Assuming constant variance for all group-and-stratum combinations, the t statistic for the mean is

$$M_v = \frac{\sum_s w_{vs} \sum_g c_g \bar{X}_{vgs+}}{\sqrt{s_v^2 \left(\sum_s w_{vs}^2 \sum_g \frac{c_g^2}{n_{vgs}} \right)}}$$

Then under the null hypothesis and assuming normality, independence, and homoscedasticity, M_v follows a t distribution with $df_p = \sum_s \sum_g \left(n_{vgs} - 1\right)$ degrees of freedom.

Whenever a denominator of 0 is computed, the p-value is set to 1. When missing data force $n_{vgs} = 0$, the contribution to the denominator of the pooled variance is 0 and not –1. This is also true for the degrees of freedom.

Heterogeneous Variance

If you do not assume constant variance for all group-and-stratum combinations, then the approximate t test is

$$M_v = \frac{\sum_s w_{vs} \sum_g c_g \bar{X}_{vgs+}}{\sqrt{\sum_s w_{vs}^2 \sum_g c_g^2 \frac{s_{vgs}^2}{n_{vgs}}}}$$

Under the null hypothesis and assuming normality and independence, the Satterthwaite (1946) approximation for the degrees of freedom of the t test is given by

$$df_s = \frac{\left(\sum_s w_{vs}^2 \sum_g c_g^2 \frac{s_{vgs}^2}{n_{vgs}}\right)^2}{\sum_s \sum_g \frac{\left(w_{vs}^2 c_g^2 \frac{s_{vgs}^2}{n_{vgs}}\right)^2}{n_{vgs} - 1}}$$

under the restriction $1 \leq df_s \leq \sum_s \sum_g n_{vgs}$.

Whenever a denominator of 0 for M_v is computed, the p-value is set to 1. If the denominator for df_s is computed as 0, then set $df_s = df_p$. When missing data force $n_{vgs} \leq 1$, that group-and-stratum combination does not contribute to the df_s computation.

p-Value Adjustments

Suppose you test m null hypotheses, $H_{01}, \ldots, H_{0m}$, and obtain the p-values $p_1, \ldots, p_m$. Denote the ordered p-values as $p_{(1)} \leq \ldots \leq p_{(m)}$ and order the tests appropriately: $H_{0(1)}, \ldots, H_{0(m)}$. Suppose you know m_0 of the null hypotheses are true and $m_1 = m - m_0$ are false. Let R indicate the number of null hypotheses rejected by the tests, where V of these are incorrectly rejected (that is, V tests are Type I errors) and $R - V$ are correctly rejected (so $m_1 - R + V$ tests are Type II errors). This information is summarized in the following table:

	Null is Rejected	Null is Not Rejected	Total
Null is True	V	$m_0 - V$	m_0
Null is False	$R - V$	$m_1 - R + V$	m_1
Total	R	$m - R$	m

The *familywise error rate* (FWE) is the overall Type I error rate for all the comparisons (possibly under some restrictions); that is, it is the maximum probability of incorrectly rejecting one or more null hypotheses:

$$\text{FWE} = \Pr(V > 0)$$

The FWE is also known as the *maximum experimentwise error rate* (MEER), as discussed in the section "Pairwise Comparisons" on page 3695 in Chapter 47, "The GLM Procedure."

The *false discovery rate* (FDR) is the expected proportion of incorrectly rejected hypotheses among all rejected hypotheses:

$$\begin{aligned}\text{FDR} &= E\left(\frac{V}{R}\right) \quad \text{where } \frac{V}{R} = 0 \text{ when } V = R = 0 \\ &= E\left(\frac{V}{R} \mid R > 0\right) \Pr(R > 0)\end{aligned}$$

Under the overall null hypothesis (all the null hypotheses are true), the FDR=FWE since $V = R$ gives $E\left(\frac{V}{R}\right) = 1 \times \Pr\left(\frac{V}{R} = 1\right) = \Pr(V > 0)$. Otherwise, FDR is always less than FWE, and an FDR-controlling adjustment also controls the FWE. Another definition used is the *positive* false discovery rate:

$$\text{pFDR} = E\left(\frac{V}{R} \mid R > 0\right)$$

The p-value adjustment methods discussed in the following sections attempt to correct the raw p-values while controlling either the FWE or the FDR. Note that the methods might impose some restrictions in order to achieve this; restrictions are discussed along with the methods in the following sections. Discussions and comparisons of some of these methods are given in Dmitrienko et al. (2005), Dudoit, Shaffer, and Boldrick (2003), Westfall et al. (1999), and Brown and Russell (1997).

Familywise Error Rate Controlling Adjustments

PROC MULTTEST provides several p-value adjustments to control the familywise error rate. *Single-step* adjustment methods are computed without reference to the other hypothesis tests under consideration. The available single-step methods are the Bonferroni and Šidák adjustments, which are simple functions of the raw p-values that try to distribute the significance level α across all the tests, and the bootstrap and permutation resampling adjustments, which require the raw data. The Bonferroni and Šidák methods are calculated from the permutation distributions when exact permutation tests are used with the CA or Peto test.

Stepwise tests, or *sequentially rejective* tests, order the hypotheses in *step-up* (least significant to most significant) or *step-down* fashion, then sequentially determine acceptance or rejection of the nulls. These tests are more powerful than the single-step tests, and they do not always require you to perform every test. However, PROC MULTTEST still adjusts every p-value. PROC MULTTEST provides the following stepwise p-value adjustments: step-down Bonferroni (Holm), step-down Šidák, step-down bootstrap and permutation resampling, Hochberg's (1988) step-up, Hommel's (1988), Fisher's combination method, and the Stouffer-Liptak combination method. Adaptive versions of Holm's step-down Bonferroni and Hochberg's step-up Bonferroni methods, which require an estimate of the number of true null hypotheses, are also available.

Liu (1996) shows that all single-step and stepwise tests based on marginal p-values can be used to construct a *closed* test (Marcus, Peritz, and Gabriel 1976; Dmitrienko et al. 2005). Closed testing methods not only control the familywise error rate at size α, but are also more powerful than the tests on which they are based. Westfall and Wolfinger (2000) note that several of the methods available in PROC MULTTEST are closed—namely, the step-down methods, Hommel's method, and Fisher's combination; see that reference for conditions and exceptions.

All methods except the resampling methods are calculated by simple functions of the raw p-values or marginal permutation distributions; the permutation and bootstrap adjustments require the raw data. Because the resampling techniques incorporate distributional and correlational structures, they tend to be less conservative than the other methods.

When a resampling (bootstrap or permutation) method is used with only one test, the adjusted p-value is the bootstrap or permutation p-value for that test, with no adjustment for multiplicity, as described by Westfall and Soper (1994).

Bonferroni

The Bonferroni p-value for test $i, i = 1, \ldots, m$ is simply $\tilde{p}_i = mp_i$. If the adjusted p-value exceeds 1, it is set to 1. The Bonferroni test is conservative but always controls the familywise error rate.

If the unadjusted p-values are computed by using exact permutation distributions, then the Bonferroni adjustment for p_i is $\tilde{p}_i = p_1^* + \cdots + p_m^*$, where p_j^* is the largest p-value from the permutation distribution of test j satisfying $p_j^* \leq p_i$, or 0 if all permutational p-values of test j are greater than p_i. These adjustments are much less conservative than the ordinary Bonferroni adjustments because they incorporate the discrete distributional characteristics. However, they remain conservative in that they do not incorporate correlation structures between multiple contrasts and multiple variables (Westfall and Wolfinger 1997).

Šidák

A technique slightly less conservative than Bonferroni is the Šidák p-value (Šidák 1967), which is $\tilde{p}_i = 1 - (1 - p_i)^m$. It is exact when all of the p-values are uniformly distributed and independent, and it is conservative when the test statistics satisfy the positive orthant dependence condition (Holland and Copenhaver 1987).

If the unadjusted p-values are computed by using exact permutation distributions, then the Šidák adjustment for p_i is $\tilde{p}_i = 1 - (1 - p_1^*) \cdots (1 - p_m^*)$, where the p_j^* are as described previously. These adjustments are less conservative than the corresponding Bonferroni adjustments, but they do not incorporate correlation structures between multiple contrasts and multiple variables (Westfall and Wolfinger 1997).

Bootstrap

The bootstrap method creates pseudo-data sets by sampling observations with replacement from each within-stratum pool of observations. An entire data set is thus created, and p-values for all tests are computed on this pseudo-data set. A counter records whether the minimum p-value from the pseudo-data set is less than or equal to the actual p-value for each base test. (If there are m tests, then there are m such counters.) This process is repeated a large number of times, and the proportion of resampled data sets where the minimum pseudo-p-value is less than or equal to an actual p-value is the adjusted p-value reported by PROC MULTTEST. The algorithms are described in Westfall and Young (1993).

In the case of continuous data, the pooling of the groups is not likely to re-create the shape of the null hypothesis distribution, since the pooled data are likely to be multimodal. For this reason, PROC MULTTEST automatically mean-centers all continuous variables prior to resampling. Such mean-centering is akin to resampling residuals in a regression analysis, as discussed by Freedman (1981). You can specify the NOCENTER option if you do not want to center the data.

The bootstrap method implicitly incorporates all sources of correlation, from both the multiple contrasts and the multivariate structure. The adjusted p-values incorporate all correlations and distributional characteristics. This method always provides weak control of the familywise error rate, and it provides strong control when the *subset pivotality* condition holds; that is, for any subset of the null hypotheses, the joint distribution of the p-values for the subset is identical to that under the complete null (Westfall and Young 1993).

Permutation

The permutation-style-adjusted p-values are computed in identical fashion as the bootstrap-adjusted p-values, with the exception that the within-stratum resampling is performed without replacement instead of with replacement. This produces a rerandomization analysis such as in Brown and Fears (1981) and Heyse and Rom (1988). In the spirit of rerandomization analyses, the continuous variables are not centered prior to resampling. This default can be overridden by using the CENTER option.

The permutation method implicitly incorporates all sources of correlation, from both the multiple contrasts and the multivariate structure. The adjusted p-values incorporate all correlations and distributional characteristics. This method always provides weak control of the familywise error rate, and it provides strong control of the familywise error rate under the *subset pivotality* condition, as described in the preceding section.

Step-Down Methods

Step-down testing is available for the Bonferroni, Šidák, bootstrap, and permutation methods. The benefit of using step-down methods is that the tests are made more powerful (smaller adjusted p-values) while, in most cases, maintaining strong control of the familywise error rate. The step-down method was pioneered by Holm (1979) and further developed by Shaffer (1986), Holland and Copenhaver (1987), and Hochberg and Tamhane (1987).

The Bonferroni step-down (Holm) p-values $\tilde{p}_{(1)}, \ldots, \tilde{p}_{(m)}$ are obtained from

$$\tilde{p}_{(i)} = \begin{cases} m p_{(1)} & \text{for } i = 1 \\ \max\left(\tilde{p}_{(i-1)}, (m-i+1)p_{(i)}\right) & \text{for } i = 2, \ldots, m \end{cases}$$

As always, if any adjusted p-value exceeds 1, it is set to 1.

The Šidák step-down p-values are determined similarly:

$$\tilde{p}_{(i)} = \begin{cases} 1-(1-p_{(1)})^m & \text{for } i = 1 \\ \max\left(\tilde{p}_{(i-1)}, 1-(1-p_{(i)})^{m-i+1}\right) & \text{for } i = 2, \ldots, m \end{cases}$$

Step-down Bonferroni adjustments that use exact tests are defined as

$$\tilde{p}_{(i)} = \begin{cases} p^*_{(1)} + \cdots + p^*_{(m)} & \text{for } i = 1 \\ \max\left(\tilde{p}_{(i-1)}, p^*_{(i)} + \cdots + p^*_{(m)},\right) & \text{for } i = 2, \ldots, m \end{cases}$$

where the p^*_j are defined as before. Note that p^*_j is taken from the permutation distribution corresponding to the jth-smallest unadjusted p-value. Also, any $\tilde{p}_j$ greater than 1.0 is reduced to 1.0.

Step-down Šidák adjustments for exact tests are defined analogously by substituting $1-(1-p^*_{(j)})\cdots(1-p^*_{(m)})$ for $p^*_{(j)} + \cdots + p^*_{(m)}$.

The resampling-style step-down methods are analogous to the preceding step-down methods; the most extreme p-value is adjusted according to all m tests, the second-most extreme p-value is adjusted according to $(m-1)$ tests, and so on. The difference is that all correlational and distributional characteristics are incorporated when you use resampling methods. More specifically, assuming the same ordering of p-values as discussed previously, the resampling-style step-down-adjusted p-value for test i is the probability that the minimum pseudo-p-value of tests $i, \ldots, m$ is less than or equal to p_i.

This probability is evaluated by using Monte Carlo simulation, as are the previously described resampling-style-adjusted p-values. In fact, the computations for step-down-adjusted p-values are essentially no more

time-consuming than the computations for the non-step-down-adjusted *p*-values. After Monte Carlo, the step-down-adjusted *p*-values are corrected to ensure monotonicity; this correction leaves the first adjusted *p*-values alone, then corrects the remaining ones as needed. The step-down method approximately controls the familywise error rate, and it is described in more detail by Westfall and Young (1993), Westfall et al. (1999), and Westfall and Wolfinger (2000).

Hommel

Hommel's (1988) method is a closed testing procedure based on Simes' test (Simes 1986). The Simes *p*-value for a joint test of any set of S hypotheses with *p*-values $p_{(1)} \leq p_{(2)} \leq \ldots \leq p_{(S)}$ is $\min((S/1)p_{(1)}, (S/2)p_{(2)}, \ldots, (S/S)p_{(S)})$. The Hommel-adjusted *p*-value for test j is the maximum of all such Simes *p*-values, taken over all joint tests that include j as one of their components.

Hochberg-adjusted *p*-values are always as large or larger than Hommel-adjusted *p*-values. Sarkar and Chang (1997) shows that Simes' method is valid under independent or positively dependent *p*-values, so Hommel's and Hochberg's methods are also valid in such cases by the closure principle.

Hochberg

Assuming *p*-values are independent and uniformly distributed under their respective null hypotheses, Hochberg (1988) demonstrates that Holm's step-down adjustments control the familywise error rate even when calculated in *step-up* fashion. Since the adjusted *p*-values are uniformly smaller for Hochberg's method than for Holm's method, the Hochberg method is more powerful. However, this improved power comes at the cost of having to make the assumption of independence. Hochberg's method can be derived from Hommel's (Liu 1996), and is thus also derived from Simes' test (Simes 1986).

Hochberg-adjusted *p*-values are always as large or larger than Hommel-adjusted *p*-values. Sarkar and Chang (1997) showed that Simes' method is valid under independent or positively dependent *p*-values, so Hommel's and Hochberg's methods are also valid in such cases by the closure principle.

The Hochberg-adjusted *p*-values are defined in reverse order of the step-down Bonferroni:

$$\tilde{p}_{(i)} = \begin{cases} p_{(m)} & \text{for } i = m \\ \min\left(\tilde{p}_{(i+1)}, (m-i+1)p_{(i)}\right) & \text{for } i = m-1, \ldots, 1 \end{cases}$$

Fisher Combination

The FISHER_C option requests adjusted *p*-values by using closed tests, based on the idea of Fisher's combination test. The Fisher combination test for a joint test of any set of S hypotheses with *p*-values uses the chi-square statistic $\chi^2 = -2\sum \log(p_i)$, with $2S$ degrees of freedom. The FISHER_C adjusted *p*-value for test j is the maximum of all *p*-values for the combination tests, taken over all joint tests that include j as one of their components. Independence of *p*-values is required for the validity of this method.

Stouffer-Liptak Combination

The STOUFFER option requests adjusted *p*-values by using closed tests, based on the Stouffer-Liptak combination test. The Stouffer combination joint test of any set of S one-sided hypotheses with *p*-values, $p_1, \ldots, p_S$, yields the *p*-value, $1 - \Phi\left(\frac{1}{\sqrt{S}}\sum_i \Phi^{-1}(1-p_i)\right)$. The STOUFFER adjusted *p*-value for test j is the maximum of all *p*-values for the combination tests, taken over all joint tests that include j as one of their components.

Independence of the one-sided p-values is required for the validity of this method. Westfall (2005) shows that the Stouffer-Liptak adjustment might have more power than the Fisher combination and Simes' adjustments when the test results reinforce each other.

Adaptive Adjustments

Adaptive adjustments modify the FWE- and FDR-controlling procedures by taking an estimate of the number m_0 or proportion π_0 of true null hypotheses into account. The adjusted p-values for Holm's and Hochberg's methods involve the number of unadjusted p-values larger than (i), $m - i + 1$. So the minimal significance level at which the ith ordered p-value is rejected implies that the number of true null hypotheses is $m - i + 1$. However, if you know m_0, then you can replace $m - i + 1$ with $\min(m_0, m - i + 1)$, thereby obtaining more power while maintaining the original α-level significance.

Since m_0 is unknown, there are several methods used to estimate the value—see the NTRUENULL= option for more information. The estimation method described by Hochberg and Benjamini (1990) considers the graph of $1 - p_{(i)}$ versus i, where the $p_{(i)}$ are the ordered p-values of your tests. See Output 80.6.4 for an example. If all null hypotheses are actually true ($m_0 = m$), then the p-values behave like a sample from a uniform distribution and this graph should be a straight line through the origin. However, if points in the upper-right corner of this plot do not follow the initial trend, then some of these null hypotheses are probably false and $0 < m_0 < m$.

The ADAPTIVEHOLM option uses this estimate of m_0 to adjust the step-up Bonferroni method while the ADAPTIVEHOCHBERG option adjusts the step-down Bonferroni method. Both of these methods are due to Hochberg and Benjamini (1990). When m_0 is known, these procedures control the familywise error rate in the same manner as their nonadaptive versions but with more power; however, since m_0 must be estimated, the FWE control is only approximate. The ADAPTIVEFDR and PFDR options also use $\hat{m}_0$, and are described in the following section.

The adjusted p-values for the ADAPTIVEHOLM method are computed by

$$\tilde{p}_{(i)} = \begin{cases} \min(m, \hat{m}_0)\, p_{(1)} & \text{for } i = 1 \\ \max\left[\tilde{p}_{(i-1)}, \min(m - i + 1, \hat{m}_0)\, p_{(i)}\right] & \text{for } i = 2, \ldots, m \end{cases}$$

The adjusted p-values for the ADAPTIVEHOCHBERG method are computed by

$$\tilde{p}_{(i)} = \begin{cases} \min(1, \hat{m}_0)\, p_{(m)} & \text{for } i = m \\ \min\left[\tilde{p}_{(i+1)}, \min(m - i + 1, \hat{m}_0)\, p_{(i)}\right] & \text{for } i = m - 1, \ldots, 1 \end{cases}$$

False Discovery Rate Controlling Adjustments

Methods that control the *false discovery rate* (FDR) were described by Benjamini and Hochberg (1995). These adjustments do not necessarily control the familywise error rate (FWE). However, FDR-controlling methods are more powerful and more liberal, and hence reject more null hypotheses, than adjustments protecting the FWE. FDR-controlling methods are often used when you have a large number of null hypotheses. To control the FDR, Benjamini and Hochberg's (1995) linear step-up method is provided, as well as an adaptive version, a dependence version, and bootstrap and permutation resampling versions. Storey's (2002) pFDR methods are also provided.

The FDR option requests p-values that control the "false discovery rate" described by Benjamini and Hochberg (1995). These *linear step-up* adjustments are potentially much less conservative than the Hochberg adjustments.

The FDR-adjusted p-values are defined in step-up fashion, like the Hochberg adjustments, but with less conservative multipliers:

$$\tilde{p}_{(i)} = \begin{cases} p_{(m)} & \text{for } i = m \\ \min\left(\tilde{p}_{(i+1)}, \frac{m}{i} p_{(i)}\right) & \text{for } i = m-1, \ldots, 1 \end{cases}$$

The FDR method is guaranteed to control the false discovery rate at level $\leq \frac{m_0}{m}\alpha \leq \alpha$ when you have independent p-values that are uniformly distributed under their respective null hypotheses. Benjamini and Yekutieli (2001) show that the false discovery rate is also controlled at level $\leq \frac{m_0}{m}\alpha$ when the *positive regression dependent* condition holds on the set of the true null hypotheses, and they provide several examples where this condition is true.

NOTE: The positive regression dependent condition on the set of the true null hypotheses holds if the joint distribution of the test statistics $\mathbf{X} = (X_1, \ldots, X_m)$ for the null hypotheses $H_{01}, \ldots, H_{0m}$ satisfies: $\Pr(\mathbf{X} \in A | X_i = x)$ is nondecreasing in x for each X_i where H_{0i} is true, for any increasing set A. The set A is increasing if $\mathbf{x} \in A$ and $\mathbf{y} \geq \mathbf{x}$ implies $\mathbf{y} \in A$.

Dependent False Discovery Rate

The DEPENDENTFDR option requests a false discovery rate controlling method that is always valid for p-values under any kind of dependency (Benjamini and Yekutieli 2001), but is thus quite conservative. Let $\gamma = \sum_{i=1}^{m} \frac{1}{i}$. The DEPENDENTFDR procedure always controls the false discovery rate at level $\leq \frac{m_0}{m}\alpha\gamma$. The adjusted p-values are computed as

$$\tilde{p}_{(i)} = \begin{cases} \gamma p_{(m)} & \text{for } i = m \\ \min\left(\tilde{p}_{(i+1)}, \gamma \frac{m}{i} p_{(i)}\right) & \text{for } i = m-1, \ldots, 1 \end{cases}$$

False Discovery Rate Resampling Adjustments

Bootstrap and permutation resampling methods to control the false discovery rate are available with the FDRBOOT and FDRPERM options (Yekutieli and Benjamini 1999). These methods approximately control the false discovery rate when the *subset pivotality* condition holds, as discussed in the section "Bootstrap" on page 6443, and when the p-values corresponding to the true null hypotheses are independent of those for the false null hypotheses.

The resampling methodology for the BOOTSTRAP and PERMUTATION methods is used to create B resamples. For the bth resample, let $R^b(p)$ denote the number of p-values that are less than or equal to the observed p-value p. Let $r_\beta(p)$ be the $100(1-\beta)$ quantile of $\{R^1(p) \ldots R^b(p) \ldots R^B(p)\}$, and let $r(p)$ be the number of observed p-values less than or equal to p. Compute one of the following estimators:

local estimator

$$Q_1(p) = \begin{cases} \frac{1}{B}\sum_{b=1}^{B} \frac{R^b(p)}{R^b(p) + r(p) - pm} & \text{if } r(p) - r_\beta(p) \geq pm \\ \#\{R^b(p) \geq 1\}/B & \text{otherwise} \end{cases}$$

upper limit estimator

$$Q_\beta(p) = \begin{cases} \sup_{x \in [0,p]} \left(\frac{1}{B}\sum_{b=1}^{B} \frac{R^b(x)}{R^b(x) + r(x) - r_\beta(x)} \right) & \text{if } r(x) - r_\beta(x) \geq 0 \\ \#\{R^b(p) \geq 1\}/B & \text{otherwise} \end{cases}$$

where m is the number of tests and B is the number of resamples. Then for $Q = Q_1$ or Q_β, the adjusted p-values are computed as

$$\tilde{p}_{(i)} = \begin{cases} Q(p_{(m)}) & \text{for } i = m \\ \min\left(\tilde{p}_{(i+1)}, Q(p_{(i)})\right) & \text{for } i = m-1, \ldots, 1 \end{cases}$$

Adaptive False Discovery Rate

Since the FDR method controls the false discovery rate at $\leq \frac{m_0}{m}\alpha \leq \alpha$, knowledge of m_0 allows improvement of the power of the adjustment while still maintaining control of the false discovery rate. The ADAPTIVEFDR option requests adaptive adjusted p-values for approximate control of the false discovery rate, as discussed in Benjamini and Hochberg (2000). See the section "Adaptive Adjustments" on page 6446 for more details. These adaptive adjustments are also defined in step-up fashion but use an estimate $\hat{m}_0$ of the number of true null hypotheses:

$$\tilde{p}_{(i)} = \begin{cases} \frac{\hat{m}_0}{m} p_{(m)} & \text{for } i = m \\ \min\left(\tilde{p}_{(i+1)}, \frac{\hat{m}_0}{i} p_{(i)}\right) & \text{for } i = m-1, \ldots, 1 \end{cases}$$

Since $\hat{m}_0 \leq m$, the larger p-values are adjusted down. This means that, as defined, controlling the false discovery rate enables you to reject these tests at a level less than the observed p-value. However, by default, this reduction is prevented with an additional restriction: $\tilde{p}_{(i)} = \max\{\tilde{p}_{(i)}, p_{(i)}\}$.

To use this adjustment, Benjamini and Hochberg (2000) suggest first specifying the FDR option—if at least one test is rejected at your level, then apply the ADAPTIVEFDR adjustment. Alternatively, Benjamini, Krieger, and Yekutieli (2006) apply the FDR adjustment at level $\frac{\alpha}{\alpha+1}$, then specify the resulting number of true hypotheses with the NTRUENULL= option and apply the ADAPTIVEFDR adjustment; they show that this *two-stage linear step-up* procedure controls the false discovery rate at level α for independent test statistics.

Positive False Discovery Rate

The PFDR option computes the "q-values" $\hat{q}_\lambda(p_i)$ (Storey 2002; Storey, Taylor, and Siegmund 2004), which are adaptive adjusted p-values for strong control of the false discovery rate when the p-values corresponding to the true null hypotheses are independent and uniformly distributed. There are four versions of the PFDR available. Let $N(\lambda)$ be the number of observed p-values that are less than or equal to λ; let m be the number of tests; let $f = 1$ if the FINITE option is specified, and otherwise set $f = 0$; and denote the estimated proportion of true null hypotheses by

$$\hat{\pi}_0(\lambda) = \frac{m - N(\lambda) + f}{(1-\lambda)m}$$

The default estimate of FDR is

$$\widehat{\mathrm{FDR}}_\lambda(p) = \frac{\hat{\pi}_0(\lambda)\, p}{\max(N(p), 1)/m}$$

If you set $\lambda = 0$, then this is identical to the FDR adjustment.

The positive FDR is estimated by

$$\widehat{\mathrm{pFDR}}_\lambda(p) = \frac{\widehat{\mathrm{FDR}}_\lambda(p)}{1 - (1-p)^m}$$

The finite-sample versions of these two estimators for independent null p-values are given by

$$\widehat{\mathrm{FDR}}^*_\lambda(p) = \begin{cases} \dfrac{\hat{\pi}^*_0(\lambda)\,p}{\max(N(p),1)/m} & \text{if } p \le \lambda \\ 1 & \text{if } p > \lambda \end{cases}$$

$$\widehat{\mathrm{pFDR}}^*_\lambda(p) = \frac{\widehat{\mathrm{FDR}}^*_\lambda(p)}{1-(1-p)^m}$$

Finally, the adjusted p-values are computed as

$$\tilde{p}_i = \hat{q}_\lambda(p_i) = \inf_{p \ge p_i} \mathrm{FDR}_\lambda(p) \quad i = 1, \ldots, m$$

This method can produce adjusted p-values that are smaller than the raw p-values. This means that, as defined, controlling the false discovery rate enables you to reject these tests at a level less than the observed p-value. However, by default, this reduction is prevented with an additional restriction: $\tilde{p}_i = \max\{\tilde{p}_i, p_i\}$.

Missing Values

If a CLASS or STRATA variable has a missing value, then PROC MULTTEST removes that observation from the analysis.

When there are missing values for test variables, the within-group-and-stratum sample sizes can differ from variable to variable. In most cases this is not a problem; however, it is possible for all data to be missing for a particular group within a particular stratum. For continuous variables and Freeman-Tukey tests, PROC MULTTEST re-centers the contrast trend coefficients within strata where all data for a particular group are missing. Re-centering the MEAN tests could redefine your t tests in an undesirable fashion; for example, if you specify coefficients to contrast the first and third groups (`contrast -1 0 1`) but the third group is missing, then the re-centered coefficients become –0.5 and 0.5, thus contrasting the first and second groups. If this is the case, you can run your t tests in separate PROC MULTTEST invocations, then combine the data and adjust the p-values by using the INPVALUES= option. However, you might find this re-centering acceptable for the Freeman-Tukey trend tests, since the contrast still tests for an increasing trend. The Cochran-Armitage and Peto tests are unaffected by this situation.

PROC MULTTEST uses missing values for resampling if they exist in the original data set. If all variables have missing values for any observation, then PROC MULTTEST removes the observation prior to resampling. Otherwise, PROC MULTTEST treats all missing values as ordinary observations in the resampling. This means that different resampled data sets can have different group sizes. In some cases it means that a resampled data set can have all missing values for a particular variable in a particular group/stratum combination, even when values exist for that combination in the original data. For this reason, PROC MULTTEST recomputes all quantities within each pseudo-data set, including such items as centered scoring coefficients and degrees of freedom for p-values.

While PROC MULTTEST does provide analyses in missing value cases, you should not feel that it completely solves the missing-value problem. If you are concerned about the adverse effects of missing data on a particular analysis, you should consider using imputation and sensitivity analyses to assess the effects of the missing data.

Computational Resources

PROC MULTTEST keeps all of the data in memory to expedite resampling. A large portion of the memory requirement is thus 8*NOBS*NVAR bytes, where NOBS is the number of observations in the data set, and NVAR is the number of variables analyzed, including CLASS, FREQ, and STRATA variables.

If you specify PERMUTATION=*number* (for exact permutation distributions), then PROC MULTTEST requires additional memory. This requirement is approximately 4*NTEST*NSTRATA*CMAX**number**(*number* + 1) bytes, where NTEST is the number of contrasts, NSTRATA is the number of STRATA levels, and CMAX is the maximum contrast coefficient.

If you specify the FDRBOOT or FDRPERM option, then saving all the resamples in memory requires 8*NSAMPLE*NOBS bytes, where NSAMPLE is the number of resamples used.

The execution time is linear in the number of resamples; that is, 10,000 resamples will take 10 times longer than 1,000 resamples. The execution time is also linear in the sample size; that is, 100 resamples of size N will take 10 times longer than 100 resamples of size $10N$.

Output Data Sets

OUT= Data Set

The OUT= data set contains contrast names (_test_), variable names (_var_), the contrast label (_contrast_), raw p-values (raw_p or the value specified in the INPVALUES= option), and all requested adjusted p-values (bon_p, sid_p, boot_p, perm_p, stpbon_p, stpsid_p, stpbootp, stppermp, hom_p, hoc_p, fic_p, stouffer_p, aholm_p, ahoc_p, fdr_p, dfdr_p, fdrbootp, ufdbootp, fdrpermp, ufdpermp, afdr_p, or pfdr_p).

If a resampling-based adjusted p-value is requested, then the simulation standard error is included as either sim_se, stpsimse, fdrsimse, or ufdsimse, depending on whether single-step, step-down, or FDR adjustments are requested. The simulation standard errors are used to bound the true resampling-based adjusted p-value. For example, if the resampling-based estimate is 0.0312 and the simulation standard error is 0.00123, then a 95% confidence interval for the true adjusted p-value is $0.0312 \pm 1.96(0.00123)$, or 0.0288 to 0.0336.

Intermediate statistics used to calculate the p-values are also written to the OUT= data set. The statistics are separated by the _strat_ level. When _strat_ is reported as missing, the statistics refer to the pooled analysis over all _strat_ levels. The p-values are provided only for the pooled analyses and are therefore reported as missing for the strata-specific statistics.

For the Peto test, an additional variable, _tstrat_, is included to indicate whether the stratum is an incidental occurrence stratum (_tstrat_=0) or a fatal occurrence stratum (_tstrat_=1).

The statistic _value_ is the per-strata contribution to the numerator of the overall test statistic. In the case of the MEAN test, this is the contrast function of the sample means multiplied by the total number of observations within the stratum. For the FT test, _value_ is the contrast function of the double-arcsine transformed proportions, again multiplied by the total number of observations within the stratum. For the CA and Peto tests, _value_ is the observed value of the trend statistic within that stratum.

When either PETO or CA is requested, the variable _exp_ is included; this variable contains the expected value of the trend statistic for the given stratum.

The statistic _se_ is the square root of the variance of the per-strata _value_ statistic for any of the tests.

For MEAN tests, the variable _nval_ is included. When reported with an individual stratum level (that is, when the _strat_ value is nonmissing), the value _nval_ refers to the within-stratum sample size. For the combined analysis (that is, the value of the _strat_ is missing), the value _nval_ contains degrees of freedom of the *t* distribution used to compute the unadjusted *p*-value.

When the FISHER test is requested, the OUT= data set contains the variables _xval_, _mval_, _yval_, and _nval_, which define observations and sample sizes in the two groups defined by the CONTRAST statement.

For example, the OUT= data set from the drug example in the section "Getting Started: MULTTEST Procedure" on page 6413 is displayed in Figure 80.5.

Figure 80.5 Output Data for the MULTTEST Procedure

Obs	_test_	_var_	_contrast_	_value_	_exp_	_se_	raw_p	boot_p	sim_se
1	CA	SideEff1	Trend	8	5	1.54303	0.05187	0.33880	.003346749
2	CA	SideEff2	Trend	7	5	1.54303	0.19492	0.84030	.002590327
3	CA	SideEff3	Trend	10	7	1.63299	0.06619	0.51895	.003532994
4	CA	SideEff4	Trend	10	6	1.60357	0.01262	0.08840	.002007305
5	CA	SideEff5	Trend	7	4	1.44749	0.03821	0.24080	.003023370
6	CA	SideEff6	Trend	9	6	1.60357	0.06137	0.43825	.003508468
7	CA	SideEff7	Trend	9	5	1.54303	0.00953	0.05135	.001560660
8	CA	SideEff8	Trend	8	5	1.54303	0.05187	0.33880	.003346749
9	CA	SideEff9	Trend	7	5	1.54303	0.19492	0.84030	.002590327
10	CA	SideEff10	Trend	8	6	1.60357	0.21232	0.90300	.002092737

OUTPERM= Data Set

The OUTPERM= data set contains contrast names (_contrast_), variable names (_var_), and the associated permutation distributions (_value_ and upper_p). PROC MULTTEST computes the permutation distributions when you use the PERMUTATION= option with the CA or Peto test. The _value_ variable represents the support of the distributions, and upper_p represents their cumulative upper-tail probabilities. The size of this data set depends on the number of variables and the support of their permutation distributions.

For information about how this distribution is computed, see the section "Exact Permutation Test" on page 6435. For an illustration, see Example 80.1.

OUTSAMP= Data Set

The OUTSAMP= data set contains the data sets used in the resampling analysis, if such an analysis is requested. The variable _sample_ indicates the number of the resampled data set. This variable ranges from 1 to the value of the NSAMPLE= option. For each value of the _sample_ variable, an entire resampled data set is included, with _stratum_, _class_, and all other variables in the original data set. The values of the original variables are mean-centered for the mean test, if requested. The variable _obs_ indicates the observation's position in the original data set.

Each new data set is randomly drawn from the original data set, either with (bootstrap) or without (permutation) replacement. The size of this data set is, thus, the number of observations in the original data set times the number of samples.

Displayed Output

The output produced by PROC MULTTEST is divided into several tables. If the DATA= data set is specified, then the following tables are displayed:

- The "Model Information" table provides a list of the options and settings used for that particular invocation of the procedure. This table is not displayed if the INPVALUES= data set is specified. Included in this list are the following items:
  - statistical tests
  - support of the exact permutation distribution for the CA and Peto tests
  - continuity corrections used for the CA test
  - test tails
  - strata adjustment
  - *p*-value adjustments and specified suboptions
  - centering of continuous variables
  - number of samples and seed
- The "Contrast Coefficients" table lists the coefficients used in constructing the statistical tests. These coefficients are either specified in CONTRAST statements or generated by default. The coefficients apply to the levels of the CLASS statement variable. If a MEAN or FT test is specified in the TEST statement, the centered coefficients are displayed. Patterns of missing values in your data set might affect the coefficients used in your analysis; the displayed contrasts take missing value patterns into account. See the section "Missing Values" on page 6449 for more information.
- The "Variable Tabulations" tables provide summary statistics for each variable listed in the TEST statement. Included for discrete variables are the count, sample size, and percentage of occurrences. For continuous variables, the mean, sample standard deviation, and sample size are displayed. All of the previously mentioned statistics are computed for distinct combinations of the CLASS and STRATA statement variables.

If the INPVALUES= data set is specified, then the following tables are displayed:

- The "P-Value Adjustment Information" table provides a list of the specified *p*-value adjustments. If an adaptive adjustment is specified (see section "Adaptive Adjustments" on page 6446), then the following settings are also displayed when appropriate:
  - whether the finite-sample version of the PFDR is used (FINITE)
  - the number of tuning parameters to check (NLAMBDA=), the maximum tuning parameter (MAXLAMBDA=), or the specified tuning parameter (LAMBDA=)
  - the degrees of freedom of the spline (DF=) and the smoothing parameter
  - the number of bootstrap resamples (NBOOT=) and the seed (SEED=)

- If the bootstrap or spline method for estimating the number of true null hypotheses m_0 is used and the PLOTS= option is specified, the "Lambda Values" table displays the m_0 estimates as a function of the tuning parameter λ. If the bootstrap method is used, the table also displays the mean-squared errors, the minimum of which is used to select a specific λ. This table contains the values used in the "Lambda Functions" plot.
- The "Estimated Number of True Null Hypotheses" table displays the p-value adjustment, the method used to estimate the number of true nulls, and an estimate of the number and proportion of true null hypotheses in the data set.

The following table is displayed unless the NOPVALUE option is specified:

- The "p-Values" table is a collection of the raw and adjusted p-values from the run of PROC MULTTEST. The p-values are identified by variable and test.

ODS Table Names

PROC MULTTEST 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 the following table. For more information about ODS, see Chapter 20, "Using the Output Delivery System."

Table 80.3 ODS Tables Produced by PROC MULTTEST

ODS Table Name	Description	Statement or Option
Continuous	Continuous variable tabulations	TEST with MEAN
Contrasts	Contrast coefficients	default
Discrete	Discrete variable tabulations	TEST with CA, FT, PETO, or FISHER
LambdaValues	True null estimates	AHOLM, AHOC, AFDR, or PFDR
ModelInfo	Model information	default
NumTrueNull	Estimates of number of true nulls	AHOLM, AHOC, AFDR, or PFDR
pValues	p-values from the tests	default
pValueInfo	p-value adjustment information	INPVALUES=

ODS Graphics

Statistical procedures use ODS Graphics to create graphs as part of their output. ODS Graphics is described in detail in Chapter 21, "Statistical Graphics Using ODS."

Before you create graphs, ODS Graphics must be enabled (for example, by specifying the ODS GRAPHICS ON statement). For more information about enabling and disabling ODS Graphics, see the section "Enabling and Disabling ODS Graphics" on page 607 in Chapter 21, "Statistical Graphics Using ODS."

The overall appearance of graphs is controlled by ODS styles. Styles and other aspects of using ODS Graphics are discussed in the section "A Primer on ODS Statistical Graphics" on page 606 in Chapter 21, "Statistical Graphics Using ODS."

You must also specify the options in the PROC MULTTEST statement that are indicated in Table 80.4.

PROC MULTTEST assigns a name to each graph it creates using ODS. You can use these names to reference the graphs when using ODS. The names are listed in Table 80.4.

Table 80.4 Graphs Produced by PROC MULTTEST

ODS Graph Name	Plot Description	Option
AdjPlots	Panel of adjusted *p*-value plots	PLOTS=ADJUSTED
AdjByRawRank	Adjusted by rank of raw *p*-values	PLOTS=ADJUSTED(UNPACK)
AdjbyRawP	Adjusted by raw *p*-values	PLOTS=ADJUSTED(UNPACK)
AdjBySignificant	Proportion significant by adjusted	PLOTS=ADJUSTED(UNPACK)
FalsePosBySignificant	Expected number of false positives by proportion significant	PLOTS=ADJUSTED(UNPACK)
PByTest	*p*-values by test	PLOTS=PBYTEST
LambdaPlot	MSE or NTRUENULL by lambda	PLOTS=LAMBDA and (NTRUENULL=BOOTSTRAP or NTRUENULL=SPLINE or PFDR)
ManhattanPlots	$-\log_{10}$(adjusted *p*-values) by test	PLOTS=MANHATTAN
ManhattanPanel	Panel of Manhattan plots	PLOTS=MANHATTAN
RawUniformPlot	Raw *p*-values by rank and histogram	PLOTS=RAWPROB or AHOLM or AHOC or AFDR or PFDR
RawUniformPlot	Raw *p*-values by rank	PLOTS=RAWPROB(UNPACK) and AHOLM or AHOC or AFDR or PFDR
RawUniformHist	Histogram of raw *p*-values	PLOTS=RAWPROB(UNPACK) and AHOLM or AHOC or AFDR or PFDR

Examples: MULTTEST Procedure

Example 80.1: Cochran-Armitage Test with Permutation Resampling

This example, from Keith Soper at Merck, illustrates the exact permutation Cochran-Armitage test carried out on permutation resamples. In the following data set, each observation represents an animal. The binary variables S1 and S2 indicate two tumor types, with 0s indicating no tumor (failure) and 1 indicating a tumor (success); note that they have perfect negative association. The grouping variable is Dose.

```
data a;
   input S1 S2 Dose @@;
   datalines;
0 1 1   1 0 1   0 1 1
0 1 1   0 1 1   1 0 1
1 0 2   1 0 2   0 1 2
1 0 2   0 1 2   1 0 2
1 0 3   1 0 3   1 0 3
0 1 3   0 1 3   1 0 3
;

proc multtest data=a permutation nsample=10000 seed=36607 outperm=pmt;
   test ca(S1 S2 / permutation=10 uppertailed);
   class Dose;
   contrast 'CA Linear Trend' 0 1 2;
run;
proc print data=pmt;
run;
```

The PROC MULTTEST statement requests 10,000 permutation resamples. The OUTPERM= option creates an output SAS data set pmt used for the exact permutation distribution computed for the CA test.

The TEST statement specifies an upper-tailed Cochran-Armitage linear trend test for S1 and S2. The cutoff for exact permutation calculations is 10, as specified with the PERMUTATION= option in the TEST statement. Since S1 and S2 have 10 and 8 successes, respectively, PROC MULTTEST uses exact permutation distributions to compute the *p*-values for both variables.

The groups for the CA test are the levels of Dose from the CLASS statement. The trend coefficients applied to these groups are 0, 1, and 2, respectively, as specified in the CONTRAST statement.

Finally, the PROC PRINT statement displays the SAS data set pmt, which contains the permutation distributions.

The results from this analysis are displayed in Output 80.1.1 through Output 80.1.5. You should check the "Model Information" table to verify that the analysis specifications are correct.

Output 80.1.1 Cochran-Armitage Test with Permutation Resampling

The Multtest Procedure

Model Information	
Test for discrete variables	Cochran-Armitage
Exact permutation distribution used	Everywhere
Tails for discrete tests	Upper-tailed
Strata weights	None
P-value adjustment	Permutation
Number of resamples	10000
Seed	36607

The label and coefficients from the CONTRAST statement are shown in Output 80.1.2.

Output 80.1.2 Contrast Coefficients

Contrast Coefficients			
	Dose		
Contrast	**1**	**2**	**3**
CA Linear Trend	0	1	2

Output 80.1.3 displays summary statistics for the two test variables, S1 and S2. The Count column lists the number of successes for each level of the CLASS variable, Dose. The NumObs column lists the sample size, and the Percent column lists the percentage of successes in the sample.

Output 80.1.3 Summary Statistics

Discrete Variable Tabulations				
Variable	**Dose**	**Count**	**NumObs**	**Percent**
S1	1	2	6	33.33
S1	2	4	6	66.67
S1	3	4	6	66.67
S2	1	4	6	66.67
S2	2	2	6	33.33
S2	3	2	6	33.33

The Raw column in Output 80.1.4 contains the p-values from the CA test, and the Permutation column contains the permutation-adjusted p-values.

Output 80.1.4 Resulting p-Values

p-Values			
Variable	**Contrast**	**Raw**	**Permutation**
S1	CA Linear Trend	0.1993	0.4009
S2	CA Linear Trend	0.9220	1.0000

This table shows that, for S1, the adjusted p-value is approximately twice the raw p-value. In fact, resamples with small (large) p-values for S1 have large (small) p-values for S2 due to the perfect negative association

of the variables, and hence the permutation-adjusted p-value for S1 should be $2 \times 0.1993 = 0.3986$; the difference is due to resampling error. For the same reason, since the raw p-value for S2 is 0.9220, the adjusted p-value equals 1. The permutation p-values for S1 and S2 also happen to be the Bonferroni-adjusted p-values for this example.

The OUTPERM= data set is displayed in Output 80.1.5, which contains the exact permutation distributions for S1 and S2 in terms of cumulative probabilities.

Output 80.1.5 Exact Permutation Distribution

Obs	_contrast_	_var_	_value_	upper_p
1	CA Linear Trend	S1	0	1.00000
2	CA Linear Trend	S1	1	1.00000
3	CA Linear Trend	S1	2	1.00000
4	CA Linear Trend	S1	3	1.00000
5	CA Linear Trend	S1	4	1.00000
6	CA Linear Trend	S1	5	0.99966
7	CA Linear Trend	S1	6	0.99609
8	CA Linear Trend	S1	7	0.97827
9	CA Linear Trend	S1	8	0.92205
10	CA Linear Trend	S1	9	0.80070
11	CA Linear Trend	S1	10	0.61011
12	CA Linear Trend	S1	11	0.38989
13	CA Linear Trend	S1	12	0.19930
14	CA Linear Trend	S1	13	0.07795
15	CA Linear Trend	S1	14	0.02173
16	CA Linear Trend	S1	15	0.00391
17	CA Linear Trend	S1	16	0.00034
18	CA Linear Trend	S1	17	0.00000
19	CA Linear Trend	S1	18	0.00000
20	CA Linear Trend	S1	19	0.00000
21	CA Linear Trend	S1	20	0.00000
22	CA Linear Trend	S2	0	1.00000
23	CA Linear Trend	S2	1	1.00000
24	CA Linear Trend	S2	2	1.00000
25	CA Linear Trend	S2	3	0.99966
26	CA Linear Trend	S2	4	0.99609
27	CA Linear Trend	S2	5	0.97827
28	CA Linear Trend	S2	6	0.92205
29	CA Linear Trend	S2	7	0.80070
30	CA Linear Trend	S2	8	0.61011
31	CA Linear Trend	S2	9	0.38989
32	CA Linear Trend	S2	10	0.19930
33	CA Linear Trend	S2	11	0.07795
34	CA Linear Trend	S2	12	0.02173
35	CA Linear Trend	S2	13	0.00391
36	CA Linear Trend	S2	14	0.00034
37	CA Linear Trend	S2	15	0.00000
38	CA Linear Trend	S2	16	0.00000

Example 80.2: Freeman-Tukey and *t* Tests with Bootstrap Resampling

The data for this example are the same as for Example 80.1, except that a continuous variable T, which indicates the time of death of the animal, has been added.

```
data a;
   input S1 S2 T Dose @@;
   datalines;
0 1 104 1   1 0  80 1   0 1 104 1
0 1 104 1   0 1 100 1   1 0 104 1
1 0  85 2   1 0  60 2   0 1  89 2
1 0  96 2   0 1  96 2   1 0  99 2
1 0  60 3   1 0  50 3   1 0  80 3
0 1  98 3   0 1  99 3   1 0  50 3
;

proc multtest data=a bootstrap nsample=10000 seed=37081 outsamp=res;
   test ft(S1 S2 / lowertailed) mean(T / lowertailed);
   class Dose;
   contrast 'Linear Trend' 0 1 2;
run;

proc print data=res(obs=36);
run;
```

The BOOTSTRAP option in the PROC MULTTEST statement requests bootstrap resampling, and NSAMPLE=10000 requests 10,000 bootstrap samples. The SEED=37081 option provides a starting value for the random number generator. The OUTSAMP=res option creates an output SAS data set res containing the 10,000 bootstrap samples.

The TEST statement specifies the Freeman-Tukey test for S1 and S2 and specifies the *t* test for T. Both tests are lower-tailed. The grouping variable in the CLASS statement is Dose, and the coefficients across the levels of Dose are 0, 1, and 2, as specified in the CONTRAST statement. The PROC PRINT statement displays the first 36 observations of the res data set containing the bootstrap samples.

The results from this analysis are listed in Output 80.2.1 through Output 80.2.5.

The "Model Information" table in Output 80.2.1 corresponds to the specifications in the invocation of PROC MULTTEST.

Output 80.2.1 FT and *t* tests with Bootstrap Resampling

The Multtest Procedure

Model Information	
Test for discrete variables	Freeman-Tukey
Test for continuous variables	Mean t-test
Degrees of Freedom Method	Pooled
Tails for discrete tests	Lower-tailed
Tails for continuous tests	Lower-tailed
Strata weights	None
P-value adjustment	Bootstrap
Center continuous variables	Yes
Number of resamples	10000
Seed	37081

The "Contrast Coefficients" table in Output 80.2.2 shows the coefficients from the CONTRAST statement after centering, and they model a linear trend.

Output 80.2.2 Contrast Coefficients

Contrast Coefficients				
		Dose		
Contrast		**1**	**2**	**3**
Linear Trend	**Centered**	-1	0	1

The summary statistics are displayed in Output 80.2.3. The values for the discrete variables S1 and S2 are the same as those from Example 80.1. The mean, standard deviation, and sample size for the continuous variable T at each level of Dose are displayed in the "Continuous Variable Tabulations" table.

Output 80.2.3 Summary Statistics

Discrete Variable Tabulations				
Variable	**Dose**	**Count**	**NumObs**	**Percent**
S1	1	2	6	33.33
S1	2	4	6	66.67
S1	3	4	6	66.67
S2	1	4	6	66.67
S2	2	2	6	33.33
S2	3	2	6	33.33

Continuous Variable Tabulations				
Variable	**Dose**	**NumObs**	**Mean**	**Standard Deviation**
T	1	6	99.3333	9.6056
T	2	6	87.5000	14.4326
T	3	6	72.8333	22.7017

The *p*-values, displayed in Output 80.2.4, are from the Freeman-Tukey test for S1 and S2, and are from the *t* test for T.

Output 80.2.4 *p*-Values

p-Values			
Variable	**Contrast**	**Raw**	**Bootstrap**
S1	Linear Trend	0.8547	1.0000
S2	Linear Trend	0.1453	0.4605
T	Linear Trend	0.0070	0.0281

The Raw column in Output 80.2.4 contains the results from the tests on the original data, while the Bootstrap column contains the bootstrap resampled adjustment to raw_p. Note that the adjusted *p*-values are larger than the raw *p*-values for all three variables. The adjusted *p*-values more accurately reflect the correlation of the raw *p*-values, the small size of the data, and the multiple testing.

Output 80.2.5 displays the first 36 observations of the SAS data set resulting from the OUTSAMP=RES option in the PROC MULTTEST statement. The entire data set has 180,000 observations, which is 10,000 times the number of observations in the data set.

Output 80.2.5 Resampling Data Set

Obs	_sample_	_class_	_obs_	S1	S2	T
1	1	1	17	0	1	26.1667
2	1	1	8	1	0	-27.5000
3	1	1	5	0	1	0.6667
4	1	1	9	0	1	1.5000
5	1	1	7	1	0	-2.5000
6	1	1	3	0	1	4.6667
7	1	2	12	1	0	11.5000
8	1	2	12	1	0	11.5000
9	1	2	14	1	0	-22.8333
10	1	2	17	0	1	26.1667
11	1	2	1	0	1	4.6667
12	1	2	15	1	0	7.1667
13	1	3	4	0	1	4.6667
14	1	3	17	0	1	26.1667
15	1	3	14	1	0	-22.8333
16	1	3	15	1	0	7.1667
17	1	3	15	1	0	7.1667
18	1	3	6	1	0	4.6667
19	2	1	6	1	0	4.6667
20	2	1	17	0	1	26.1667
21	2	1	8	1	0	-27.5000
22	2	1	13	1	0	-12.8333
23	2	1	9	0	1	1.5000
24	2	1	8	1	0	-27.5000
25	2	2	9	0	1	1.5000
26	2	2	18	1	0	-22.8333
27	2	2	15	1	0	7.1667
28	2	2	14	1	0	-22.8333
29	2	2	9	0	1	1.5000
30	2	2	17	0	1	26.1667
31	2	3	16	0	1	25.1667
32	2	3	11	0	1	8.5000
33	2	3	14	1	0	-22.8333
34	2	3	18	1	0	-22.8333
35	2	3	18	1	0	-22.8333
36	2	3	10	1	0	8.5000

The _sample_ variable is the sample indicator and _class_ indicates the resampling group—that is, the level of the CLASS variable Dose assigned to the new observation. The number of the observation in the original data set is represented by _obs_. Also listed are the values of the original test variables, S1 and S2, and the mean-centered values of T.

Example 80.3: Peto Mortality-Prevalence Test

This example illustrates the use of the Peto mortality-prevalence test. The test is a combination of analyses about the prevalence of incidental tumors in the population and mortality due to fatal tumors.

In the following data set, each observation represents an animal. The variables S1–S3 are three tumor types, with a value of 0 indicating no tumor, 1 indicating an incidental (nonlethal) tumor, and 2 indicating a lethal tumor. The time variable T indicates the time of death of the animal, a strata variable B is constructed from T, and the grouping variable Dose is drug dosage.

```
data a;
   input S1-S3 T Dose @@;
   if T<=90 then B=1; else B=2;
   datalines;
0 0 0 104 0   2 0 1  80 0   0 0 1 104 0
0 0 0 104 0   0 2 0 100 0   1 0 0 104 0
2 0 0  85 1   2 1 0  60 1   0 1 0  89 1
2 0 1  96 1   0 0 0  96 1   2 0 1  99 1
2 1 1  60 2   2 0 0  50 2   2 0 1  80 2
0 0 2  98 2   0 0 1  99 2   2 1 1  50 2
;

proc multtest data=a notables out=p stepsid;
   test peto(S1-S3 / permutation=20 time=T uppertailed);
   class Dose;
   strata B;
   contrast 'mort-prev' 0  1  2;
run;
proc print data=p;
run;
```

The NOTABLES option in the PROC MULTTEST statement suppresses the display of the summary statistics for each variable. The OUT= option creates an output SAS data set p containing all *p*-values and intermediate statistics. The STEPSID option is used to adjust the *p*-values.

The TEST statement specifies an upper-tailed Peto test for S1–S3. The mortality strata are defined by TIME=T, the death times. The CLASS statement contains the grouping variable Dose. The prevalence strata are defined by the STRATA statement as the blocking variable B. The CONTRAST statement lists the default linear trend coefficients. The PROC PRINT statement displays the requested *p*-value data set.

The results from this analysis are listed in Output 80.3.1 through Output 80.3.4.

The "Model Information" table in Output 80.3.1 displays information corresponding to the PROC MULTTEST invocation. In this case the totals for all prevalence and fatality strata are less than 20, so exact permutation tests are used everywhere, and the STEPSID adjustments are computed from these permutation distributions.

Output 80.3.1 Peto Test

The Multtest Procedure

Model Information	
Test for discrete variables	Peto
Exact permutation distribution used	Everywhere
Tails for discrete tests	Upper-tailed
Strata weights	Sample size
P-value adjustment	Stepdown Sidak

The contrast trend coefficients are listed in Output 80.3.2. They happen to be the same as the levels of the Dose variable.

Output 80.3.2 Contrast Coefficients

Contrast Coefficients			
	Dose		
Contrast	**0**	**1**	**2**
mort-prev	0	1	2

In the "*p*-Values" table in Output 80.3.3, the *p*-values for the Peto tests are listed in the Raw column, and the step-down Šidák adjusted *p*-values are in the Stepdown Šidák column.

Output 80.3.3 *p*-Values

p-Values			
Variable	**Contrast**	**Raw**	**Stepdown Sidak**
S1	mort-prev	0.0681	0.0814
S2	mort-prev	0.5000	0.5000
S3	mort-prev	0.0363	0.0781

Significant *p*-values in the preceding table support the claim that higher dosage levels lead to higher mortality and prevalence. The raw Peto test is significant at the 5% level for S3, but the adjusted S3 test is no longer significant at 5%. The raw and adjusted *p*-values for S2 are the same because of the step-down technique.

The OUT= data set is displayed in Output 80.3.4.

Output 80.3.4 OUT= Data Set

Obs	_test_	_var_	_contrast_	_strat_	_tstrat_	_value_	_exp_	_se_	raw_p	stpsid_p
1	PETO	S1	mort-prev	1	0	0	0.00000	0.00000	.	.
2	PETO	S1	mort-prev	2	0	0	0.62500	0.85696	.	.
3	PETO	S1	mort-prev	50	1	4	2.00000	1.12022	.	.
4	PETO	S1	mort-prev	60	1	3	1.75000	1.06654	.	.
5	PETO	S1	mort-prev	80	1	2	1.57143	1.04978	.	.
6	PETO	S1	mort-prev	85	1	1	0.75000	0.72169	.	.
7	PETO	S1	mort-prev	96	1	1	0.70000	0.78102	.	.
8	PETO	S1	mort-prev	98	1	0	0.00000	0.00000	.	.
9	PETO	S1	mort-prev	99	1	1	0.42857	0.72843	.	.
10	PETO	S1	mort-prev	100	1	0	0.00000	0.00000	.	.
11	PETO	S2	mort-prev	1	0	6	5.50000	1.05221	.	.
12	PETO	S2	mort-prev	2	0	0	0.00000	0.00000	.	.
13	PETO	S2	mort-prev	50	1	0	0.00000	0.00000	.	.
14	PETO	S2	mort-prev	60	1	0	0.00000	0.00000	.	.
15	PETO	S2	mort-prev	80	1	0	0.00000	0.00000	.	.
16	PETO	S2	mort-prev	85	1	0	0.00000	0.00000	.	.
17	PETO	S2	mort-prev	96	1	0	0.00000	0.00000	.	.
18	PETO	S2	mort-prev	98	1	0	0.00000	0.00000	.	.
19	PETO	S2	mort-prev	99	1	0	0.00000	0.00000	.	.
20	PETO	S2	mort-prev	100	1	0	0.00000	0.00000	.	.
21	PETO	S3	mort-prev	1	0	6	5.50000	1.05221	.	.
22	PETO	S3	mort-prev	2	0	4	2.22222	1.08298	.	.
23	PETO	S3	mort-prev	50	1	0	0.00000	0.00000	.	.
24	PETO	S3	mort-prev	60	1	0	0.00000	0.00000	.	.
25	PETO	S3	mort-prev	80	1	0	0.00000	0.00000	.	.
26	PETO	S3	mort-prev	85	1	0	0.00000	0.00000	.	.
27	PETO	S3	mort-prev	96	1	0	0.00000	0.00000	.	.
28	PETO	S3	mort-prev	98	1	2	0.62500	0.85696	.	.
29	PETO	S3	mort-prev	99	1	0	0.00000	0.00000	.	.
30	PETO	S3	mort-prev	100	1	0	0.00000	0.00000	.	.
31	PETO	S1	mort-prev	.	.	12	7.82500	2.42699	0.06808	0.08140
32	PETO	S2	mort-prev	.	.	6	5.50000	1.05221	0.50000	0.50000
33	PETO	S3	mort-prev	.	.	12	8.34722	1.73619	0.03627	0.07811

The first 30 observations correspond to intermediate statistics used to compute the Peto *p*-values. The _test_ variable lists the name of the test, the _var_ variable lists the name of the TEST variables, and the _contrast_ variable lists the CONTRAST label. The _strat_ variable lists the level of the STRATA variable, and the _tstrat_ variable indicates whether or not the stratum corresponds to values of the TIME= variable. The _value_ variable is the observed contrast for a stratum, and the _exp_ variable is its expected value. The variable _se_ contains the square root of the variance terms summed to form the denominator of the Peto statistics.

The final three observations correspond to the three Peto tests, with their *p*-values listed under the raw_p variable. The stpsid_p variable contains the step-down Šidák-adjusted *p*-values.

Example 80.4: Fisher Test with Permutation Resampling

The following data, from Brown and Fears (1981), are the results of an 80-week carcinogenesis bioassay with female mice. Six tissue sites are examined at necropsy; 1 indicates the presence of a tumor and 0 the absence. A frequency variable Freq is included. A control and four different doses of a drug (in parts per milliliter) make up the levels of the grouping variable Dose.

```
data a;
   input Liver Lung Lymph Cardio Pitui Ovary Freq Dose$ @@;
   datalines;
1 0 0 0 0 0 8  CTRL   0 1 0 0 0 0 7  CTRL   0 0 1 0 0 0 6  CTRL
0 0 0 1 0 0 1  CTRL   0 0 0 0 0 1 2  CTRL   1 1 0 0 0 0 4  CTRL
1 0 1 0 0 0 1  CTRL   1 0 0 0 0 1 1  CTRL   0 1 1 0 0 0 1  CTRL
0 0 0 0 0 0 18 CTRL
1 0 0 0 0 0 9  4PPM   0 1 0 0 0 0 4  4PPM   0 0 1 0 0 0 7  4PPM
0 0 0 1 0 0 1  4PPM   0 0 0 0 1 0 2  4PPM   0 0 0 0 0 1 1  4PPM
1 1 0 0 0 0 4  4PPM   1 0 1 0 0 0 3  4PPM   1 0 0 0 1 0 1  4PPM
0 1 1 0 0 0 1  4PPM   0 1 0 1 0 0 1  4PPM   1 0 1 1 0 0 1  4PPM
0 0 0 0 0 0 15 4PPM
1 0 0 0 0 0 8  8PPM   0 1 0 0 0 0 3  8PPM   0 0 1 0 0 0 6  8PPM
0 0 0 1 0 0 3  8PPM   1 1 0 0 0 0 1  8PPM   1 0 1 0 0 0 2  8PPM
1 0 0 1 0 0 1  8PPM   1 0 0 0 1 0 1  8PPM   1 1 0 1 0 0 2  8PPM
1 1 0 0 0 1 2  8PPM   0 0 0 0 0 0 19 8PPM
1 0 0 0 0 0 4  16PPM  0 1 0 0 0 0 2  16PPM  0 0 1 0 0 0 9  16PPM
0 0 0 0 1 0 1  16PPM  0 0 0 0 0 1 1  16PPM  1 1 0 0 0 0 4  16PPM
1 0 1 0 0 0 1  16PPM  0 1 1 0 0 0 1  16PPM  0 1 0 1 0 0 1  16PPM
0 1 0 0 0 1 1  16PPM  0 0 1 1 0 0 1  16PPM  0 0 1 0 1 0 1  16PPM
1 1 1 0 0 0 2  16PPM  0 0 0 0 0 0 14 16PPM
1 0 0 0 0 0 8  50PPM  0 1 0 0 0 0 4  50PPM  0 0 1 0 0 0 8  50PPM
0 0 0 1 0 0 1  50PPM  0 0 0 0 0 1 4  50PPM  1 1 0 0 0 0 3  50PPM
1 0 1 0 0 0 1  50PPM  0 1 1 0 0 0 1  50PPM  0 1 0 0 1 1 1  50PPM
0 0 0 0 0 0 19 50PPM
;

proc multtest data=a order=data notables out=p
              permutation nsample=1000 seed=764511;
   test fisher(Liver Lung Lymph Cardio Pitui Ovary /
               lowertailed);
   class Dose;
   freq Freq;
run;
proc print data=p;
run;
```

In the PROC MULTTEST statement, the ORDER=DATA option is required to keep the levels of Dose in the order in which they appear in the data set. Without this option, the levels are sorted by their formatted value, resulting in an alphabetic ordering. The NOTABLES option suppresses the display of summary statistics, and the OUT= option produces an output data set p containing the *p*-values. The PERMUTATION option specifies permutation resampling, NSAMPLE=1000 requests 1000 samples, and SEED=764511 option provides a starting value for the random number generator. You should specify a seed if you need to duplicate resampling results.

To test for higher rates of tumor occurrence in the treatment groups compared to the control group, the LOWERTAILED option is specified in the FISHER option of the TEST statement to produce a lower-tailed Fisher exact test for the six tissue sites. The Fisher test is appropriate for comparing a treatment and a control, but multiple testing can be a problem. Brown and Fears (1981) use a multivariate permutation to evaluate the entire collection of tests. PROC MULTTEST adjusts the *p*-values by simulation.

The treatments make up the levels of the grouping variable Dose, listed in the CLASS statement. Since no CONTRAST statement is specified, PROC MULTTEST uses the default pairwise contrasts with the first level of Dose. The FREQ statement is used since these are summary data containing frequency counts of occurrences.

The results from this analysis are listed in Output 80.4.1 through Output 80.4.4. First, the PROC MULTTEST specifications are displayed in Output 80.4.1.

Output 80.4.1 Fisher Test with Permutation Resampling

The Multtest Procedure

Model Information	
Test for discrete variables	Fisher
Tails for discrete tests	Lower-tailed
Strata weights	None
P-value adjustment	Permutation
Number of resamples	1000
Seed	764511

The default contrasts for the Fisher test are displayed in Output 80.4.2. Note that each dose is compared with the control.

Output 80.4.2 Default Contrast Coefficients

Contrast Coefficients					
	Dose				
Contrast	**CTRL**	**4PPM**	**8PPM**	**16PPM**	**50PPM**
CTRL vs. 4PPM	1	-1	0	0	0
CTRL vs. 8PPM	1	0	-1	0	0
CTRL vs. 16PPM	1	0	0	-1	0
CTRL vs. 50PPM	1	0	0	0	-1

The "p-Values" table in Output 80.4.3 displays *p*-values for the Fisher exact tests and their permutation-based adjustments.

Output 80.4.3 p-Values

		p-Values	
Variable	**Contrast**	**Raw**	**Permutation**
Liver	CTRL vs. 4PPM	0.2828	0.9610
Liver	CTRL vs. 8PPM	0.3069	0.9670
Liver	CTRL vs. 16PPM	0.7102	1.0000
Liver	CTRL vs. 50PPM	0.7718	1.0000
Lung	CTRL vs. 4PPM	0.7818	1.0000
Lung	CTRL vs. 8PPM	0.8858	1.0000
Lung	CTRL vs. 16PPM	0.5469	0.9990
Lung	CTRL vs. 50PPM	0.8498	1.0000
Lymph	CTRL vs. 4PPM	0.2423	0.9280
Lymph	CTRL vs. 8PPM	0.5898	1.0000
Lymph	CTRL vs. 16PPM	0.0350	0.2680
Lymph	CTRL vs. 50PPM	0.4161	0.9930
Cardio	CTRL vs. 4PPM	0.3163	0.9710
Cardio	CTRL vs. 8PPM	0.0525	0.3710
Cardio	CTRL vs. 16PPM	0.4506	0.9960
Cardio	CTRL vs. 50PPM	0.7576	1.0000
Pitui	CTRL vs. 4PPM	0.1250	0.7540
Pitui	CTRL vs. 8PPM	0.4948	0.9970
Pitui	CTRL vs. 16PPM	0.2157	0.9080
Pitui	CTRL vs. 50PPM	0.5051	0.9970
Ovary	CTRL vs. 4PPM	0.9437	1.0000
Ovary	CTRL vs. 8PPM	0.8126	1.0000
Ovary	CTRL vs. 16PPM	0.7760	1.0000
Ovary	CTRL vs. 50PPM	0.3689	0.9930

As noted by Brown and Fears, only one of the 24 tests is significant at the 5% level (Lymph, CTRL vs. 16PPM). Brown and Fears report a 12% chance of observing at least one significant raw p-value for 16PPM and a 9% chance of observing at least one significant raw p-value for Lymph (both at the 5% level). Adjusted p-values exhibit much lower chances of false significances. For this example, none of the adjusted p-values are close to significant.

The OUT= data set is displayed in Output 80.4.4.

Output 80.4.4 OUT= Data Set

Obs	_test_	_var_	_contrast_	_xval_	_mval_	_yval_	_nval_	raw_p	perm_p	sim_se
1	FISHER	Liver	CTRL vs. 4PPM	14	49	18	50	0.28282	0.961	0.006122
2	FISHER	Liver	CTRL vs. 8PPM	14	49	17	48	0.30688	0.967	0.005649
3	FISHER	Liver	CTRL vs. 16PPM	14	49	11	43	0.71022	1.000	0.000000
4	FISHER	Liver	CTRL vs. 50PPM	14	49	12	50	0.77175	1.000	0.000000
5	FISHER	Lung	CTRL vs. 4PPM	12	49	10	50	0.78180	1.000	0.000000
6	FISHER	Lung	CTRL vs. 8PPM	12	49	8	48	0.88581	1.000	0.000000
7	FISHER	Lung	CTRL vs. 16PPM	12	49	11	43	0.54685	0.999	0.000999
8	FISHER	Lung	CTRL vs. 50PPM	12	49	9	50	0.84978	1.000	0.000000
9	FISHER	Lymph	CTRL vs. 4PPM	8	49	12	50	0.24228	0.928	0.008174
10	FISHER	Lymph	CTRL vs. 8PPM	8	49	8	48	0.58977	1.000	0.000000
11	FISHER	Lymph	CTRL vs. 16PPM	8	49	15	43	0.03498	0.268	0.014006
12	FISHER	Lymph	CTRL vs. 50PPM	8	49	10	50	0.41607	0.993	0.002636
13	FISHER	Cardio	CTRL vs. 4PPM	1	49	3	50	0.31631	0.971	0.005307
14	FISHER	Cardio	CTRL vs. 8PPM	1	49	6	48	0.05254	0.371	0.015276
15	FISHER	Cardio	CTRL vs. 16PPM	1	49	2	43	0.45061	0.996	0.001996
16	FISHER	Cardio	CTRL vs. 50PPM	1	49	1	50	0.75758	1.000	0.000000
17	FISHER	Pitui	CTRL vs. 4PPM	0	49	3	50	0.12496	0.754	0.013619
18	FISHER	Pitui	CTRL vs. 8PPM	0	49	1	48	0.49485	0.997	0.001729
19	FISHER	Pitui	CTRL vs. 16PPM	0	49	2	43	0.21572	0.908	0.009140
20	FISHER	Pitui	CTRL vs. 50PPM	0	49	1	50	0.50505	0.997	0.001729
21	FISHER	Ovary	CTRL vs. 4PPM	3	49	1	50	0.94372	1.000	0.000000
22	FISHER	Ovary	CTRL vs. 8PPM	3	49	2	48	0.81260	1.000	0.000000
23	FISHER	Ovary	CTRL vs. 16PPM	3	49	2	43	0.77596	1.000	0.000000
24	FISHER	Ovary	CTRL vs. 50PPM	3	49	5	50	0.36889	0.993	0.002636

The _test_, _var_, and _contrast_ variables provide the TEST name, TEST variable, and CONTRAST label, respectively. The _xval_, _mval_, _yval_, and _nval_ variables contain the components used to compute the Fisher exact tests from the hypergeometric distribution. The raw_p variable contains the *p*-values from the Fisher exact tests, and the perm_p variable contains their permutation-based adjustments. The variable sim_se is the simulation standard error from the permutation resampling.

Example 80.5: Inputting Raw p-Values

This example illustrates how to use PROC MULTTEST to multiplicity-adjust a collection of raw p-values obtained from some other source. This is a valuable option for those cases where PROC MULTTEST cannot compute the raw p-values directly. The data set a, which follows, contains the unadjusted p-values computed in Example 80.4. Note that the data set needs to have one variable containing the p-values, but the data set can contain other variables as well.

```
data a;
   input Test$ Raw_P @@;
   datalines;
test01  0.28282    test02  0.30688    test03  0.71022
test04  0.77175    test05  0.78180    test06  0.88581
test07  0.54685    test08  0.84978    test09  0.24228
test10  0.58977    test11  0.03498    test12  0.41607
test13  0.31631    test14  0.05254    test15  0.45061
test16  0.75758    test17  0.12496    test18  0.49485
test19  0.21572    test20  0.50505    test21  0.94372
test22  0.81260    test23  0.77596    test24  0.36889
;

proc multtest inpvalues=a holm hoc fdr;
run;
```

Note that the PROC MULTTEST statement is the only statement that can be specified with the p-value input mode. In this example, the raw p-values are adjusted by the Holm, Hochberg, and FDR methods.

The "P-Value Adjustment Information" table, displayed in Output 80.5.1, provides information about the requested adjustments and replaces the usual "Model Information" table. The adjusted p-values are displayed in Output 80.5.2

Output 80.5.1 Inputting Raw p-Values

The Multtest Procedure

P-Value Adjustment Information	
P-Value Adjustment	Stepdown Bonferroni
P-Value Adjustment	Hochberg
P-Value Adjustment	False Discovery Rate

Output 80.5.2 p-Values

p-Values				
Test	Raw	Stepdown Bonferroni	Hochberg	False Discovery Rate
1	0.2828	1.0000	0.9437	0.9243
2	0.3069	1.0000	0.9437	0.9243
3	0.7102	1.0000	0.9437	0.9243
4	0.7718	1.0000	0.9437	0.9243
5	0.7818	1.0000	0.9437	0.9243
6	0.8858	1.0000	0.9437	0.9243
7	0.5469	1.0000	0.9437	0.9243
8	0.8498	1.0000	0.9437	0.9243
9	0.2423	1.0000	0.9437	0.9243
10	0.5898	1.0000	0.9437	0.9243
11	0.0350	0.8395	0.8395	0.6305
12	0.4161	1.0000	0.9437	0.9243
13	0.3163	1.0000	0.9437	0.9243
14	0.0525	1.0000	0.9437	0.6305
15	0.4506	1.0000	0.9437	0.9243
16	0.7576	1.0000	0.9437	0.9243
17	0.1250	1.0000	0.9437	0.9243
18	0.4949	1.0000	0.9437	0.9243
19	0.2157	1.0000	0.9437	0.9243
20	0.5051	1.0000	0.9437	0.9243
21	0.9437	1.0000	0.9437	0.9437
22	0.8126	1.0000	0.9437	0.9243
23	0.7760	1.0000	0.9437	0.9243
24	0.3689	1.0000	0.9437	0.9243

Note that the adjusted p-values for the Hochberg method are less than or equal to those for the Holm (Step-down Bonferroni) method. In turn, the adjusted p-values for the FDR method (False Discovery Rate) are less than or equal to those for the Hochberg method. These comparisons hold generally for all p-value configurations. The FDR method controls the false discovery rate and not the familywise error rate. The Hochberg method controls the familywise error rate under independence. The Holm method controls the familywise error rate without assuming independence.

Example 80.6: Adaptive Adjustments and ODS Graphics

An experiment was performed using Affymetrix® gene chips on the CD4 lymphocyte white blood cells of patients with and without a hereditary allergy (atopy) and possibly with asthma. The Asthma-Atopy microarray data set and analysis are discussed in Gibson and Wolfinger (2004): a one-way ANOVA model of the log2mas5 variable ($\log_2$(MAS 5.0 summary statistics) is fit against a classification variable trt describing different asthma-atopy combinations in the patients, and the least squares means of the trt variable are computed.

For this example, a 1% random sample of least squares means having p-values exceeding 1E–6 is taken. The resulting data are recorded in the test data set, where the Probe_Set_ID variable identifies the probe and the Probt variable contains the p-values for the $m = 121$ tests, as follows:

```
data test;
   length Probe_Set_ID $9.;
   input Probe_Set_ID $ Probt @@;
   datalines;
200973_s_ .963316  201059_at .462754  201563_at .000409  201733_at .000819
201951_at .000252  202944_at .106550  203107_x_ .040396  203372_s_ .010911
203469_s_ .987234  203641_s_ .019296  203795_s_ .002276  204055_s_ .002328
205020_s_ .008628  205199_at .608129  205373_at .005209  205384_at .742381
205428_s_ .870533  205653_at .621671  205686_s_ .396440  205760_s_ .000002
206032_at .024661  206159_at .997627  206223_at .003702  206398_s_ .191682
206623_at .010030  206852_at .000004  207072_at .000214  207371_at .000013
207789_s_ .023623  207861_at .000002  207897_at .000007  208022_s_ .251999
208086_s_ .000361  208406_s_ .040182  208464_at .161468  209055_s_ .529824
209125_at .142276  209369_at .240079  209748_at .071750  209894_at .000042
209906_at .223282  210130_s_ .192187  210199_at .101623  210477_x_ .300038
210491_at .000078  210531_at .000784  210734_x_ .202931  210755_at .009644
210782_x_ .000011  211320_s_ .022896  211329_x_ .486869  211362_s_ .881798
211369_at .000030  211399_at .000008  211572_s_ .269788  211647_x_ .001301
213072_at .005019  213143_at .008711  213238_at .004824  213391_at .316133
213468_at .000172  213636_at .097133  213823_at .001678  213854_at .001921
213976_at .000299  214006_s_ .014616  214063_s_ .000361  214407_x_ .609880
214445_at .000009  214570_x_ .000002  214648_at .001255  214684_at .288156
214991_s_ .006695  215012_at .000499  215117_at .000136  215201_at .045235
215304_at .000816  215342_s_ .973786  215392_at .112937  215557_at .000007
215608_at .006204  215935_at .000027  215980_s_ .037382  216010_x_ .000354
216051_x_ .000003  216086_at .002310  216092_s_ .000056  216511_s_ .294776
216733_s_ .004823  216747_at .002902  216874_at .000117  216969_s_ .001614
217133_x_ .056851  217198_x_ .169196  217557_s_ .002966  217738_at .000005
218601_at .023817  218818_at .027554  219302_s_ .000039  219441_s_ .000172
219574_at .193737  219612_s_ .000075  219697_at .046476  219700_at .003049
219945_at .000066  219964_at .000684  220234_at .130064  220473_s_ .000017
220575_at .030223  220633_s_ .058460  220925_at .252465  221256_s_ .721731
221314_at .002307  221589_s_ .001810  221995_s_ .350859  222071_s_ .000062
222113_s_ .000023  222208_s_ .100961  222303_at .049265  37226_at  .000749
60474_at  .000423
run;
```

The following statements adjust the p-values in the test data set by using the adaptive adjustments (ADAPTIVEHOLM, ADAPTIVEHOCHBERG, ADAPTIVEFDR, and PFDR), which require an estimate of the

number of true null hypotheses ($\hat{m}_0$) or proportion of true null hypotheses ($\hat{\pi}_0$). This example illustrates some of the features and graphics for computing and evaluating these estimates. The NOPVALUE option is specified to suppress the display of the "p-Values" table.

```
ods graphics on;
proc multtest inpvalues(Probt)=test plots=all seed=518498000
              aholm ahoc afdr pfdr(positive) nopvalue;
   id Probe_Set_ID;
run;
ods graphics off;
```

Output 80.6.1 lists the requested p-value adjustments, along with the selected value of the "Lambda" tuning parameter and the seed (specified with the SEED= option) used in the bootstrap method of estimating the number of true null hypotheses. The "Lambda Values" table lists the estimated number of true nulls for each value of λ, where you can see that the minimum MSE (0.002315) occurs at $\lambda = 0.4$. Output 80.6.2 shows that the SPLINE method failed due to a large slope at $\lambda = 0.95$, so the bootstrap method is used and the MSE plot is displayed.

Output 80.6.1 p and Lambda Values

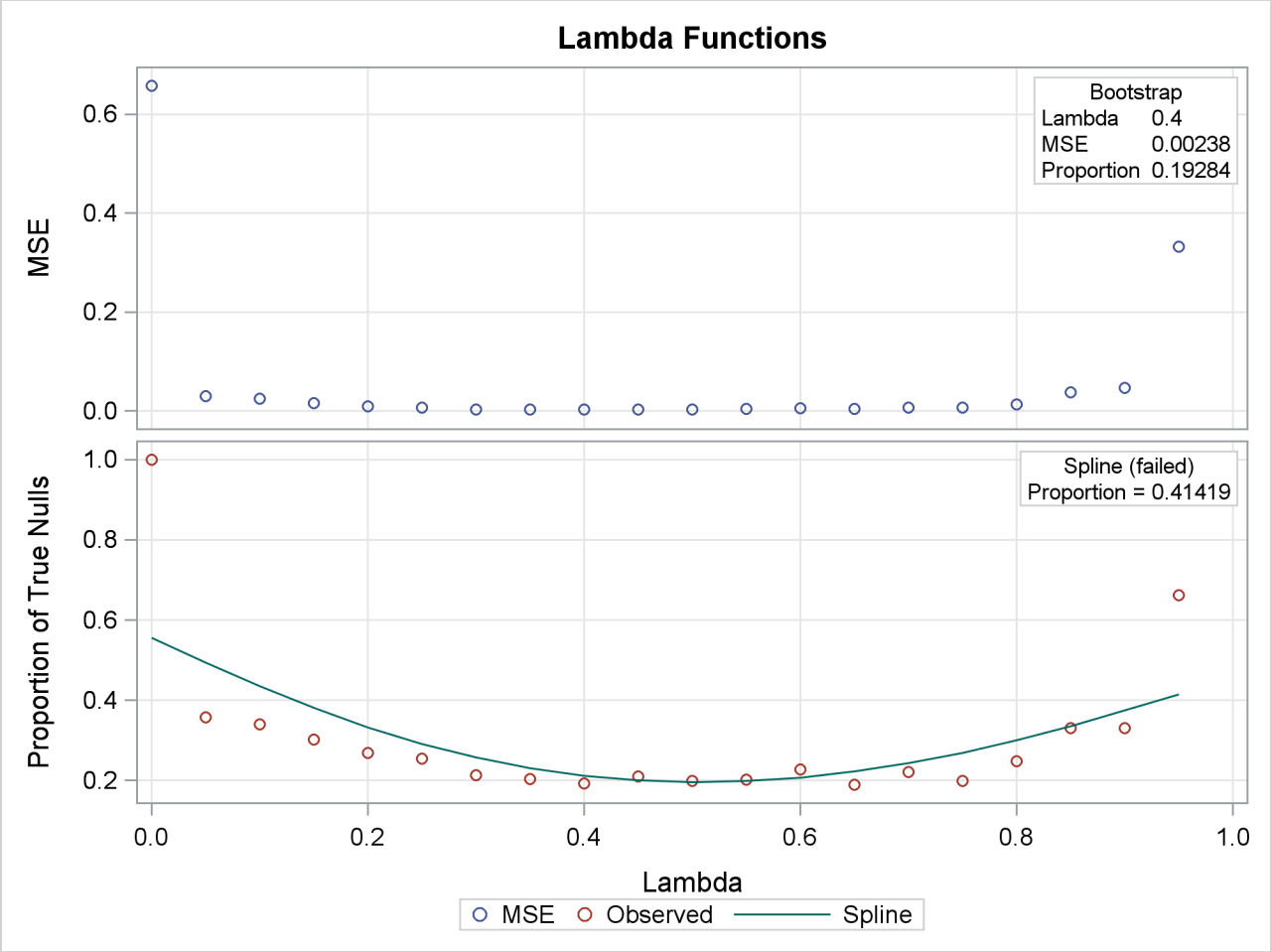
The Multtest Procedure

P-Value Adjustment Information	
P-Value Adjustment	Adaptive Holm
P-Value Adjustment	Adaptive Hochberg
P-Value Adjustment	Adaptive FDR
P-Value Adjustment	pFDR Q-Value
Lambda	0.4
Seed	518498000

Lambda Values

Lambda	MSE	NTrueNull Observed	NTrueNull Spline
0	0.657880	121.000000	67.318707
0.050000	0.030212	43.157895	59.812885
0.100000	0.024897	41.111111	52.636271
0.150000	0.014904	36.470588	46.033846
0.200000	0.008580	32.500000	40.172642
0.250000	0.006476	30.666667	35.157768
0.300000	0.002719	25.714286	31.046105
0.350000	0.002471	24.615385	27.861153
0.400000	0.002378	23.333333	25.595089
0.450000	0.003285	25.454545	24.217908
0.500000	0.003036	24.000000	23.687690
0.550000	0.003567	24.444444	23.965745
0.600000	0.005813	27.500000	25.016579
0.650000	0.004118	22.857143	26.809774
0.700000	0.006647	26.666667	29.321876
0.750000	0.006260	24.000000	32.512203
0.800000	0.013242	30.000000	36.315191
0.850000	0.037624	40.000000	40.618909
0.900000	0.046906	40.000000	45.274355
0.950000	0.332183	80.000000	50.117369

Output 80.6.2 Tuning Parameter Plots



Output 80.6.3 also shows that the bootstrap estimate is used for the PFDR adjustment. The other adjustments have different default methods for estimating the number of true nulls.

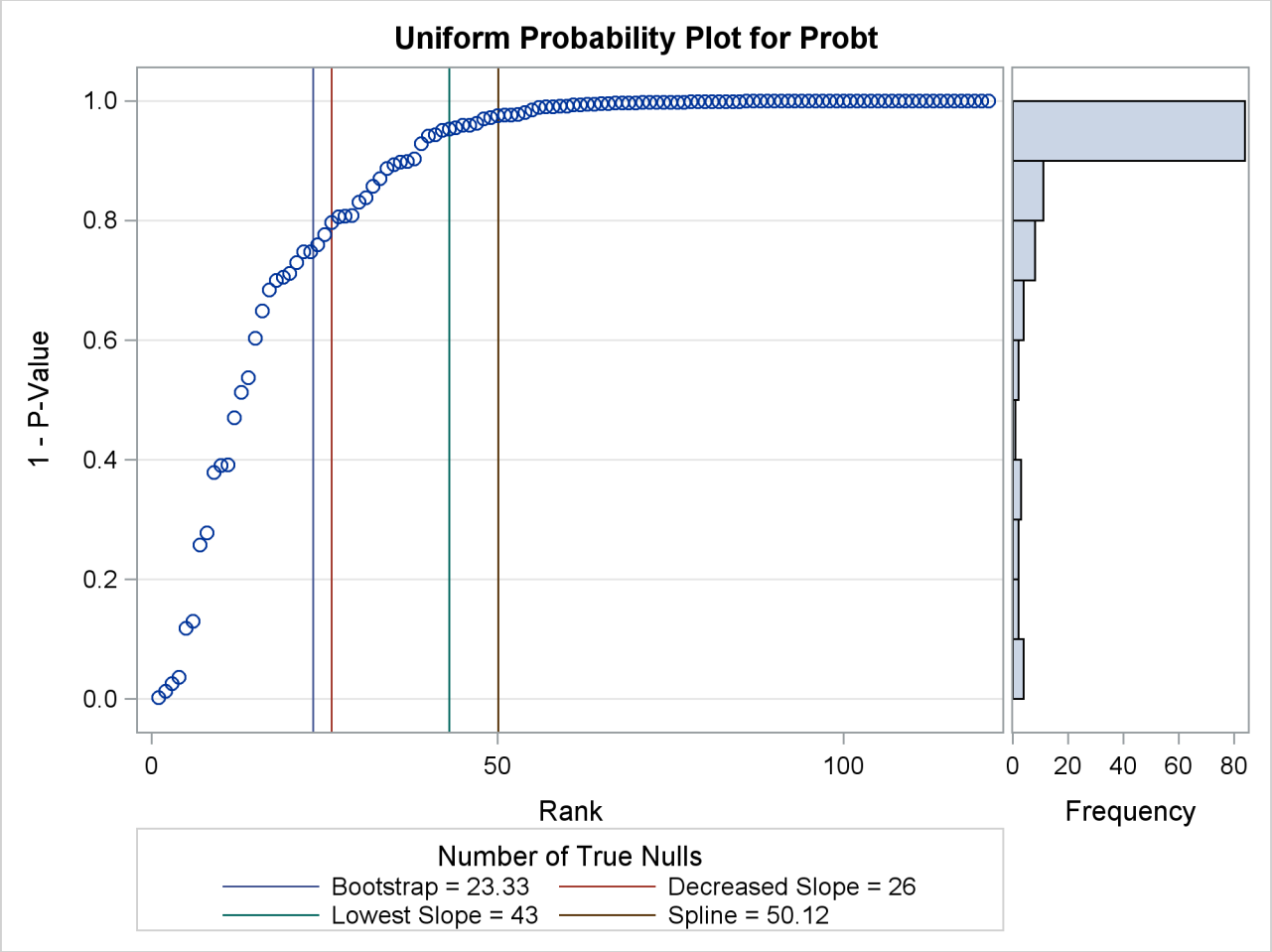
Output 80.6.3 Adjustments and Their Default Estimation Method

Estimated Number of True Null Hypotheses

P-Value Adjustment	Method	Estimate	Proportion
Adaptive Holm	**Decreased Slope**	26	0.21488
Adaptive Hochberg	**Decreased Slope**	26	0.21488
Adaptive FDR	**Lowest Slope**	43	0.35537
Positive FDR	**Bootstrap**	23.3333	0.19284

Output 80.6.4 displays the estimated number of true nulls $\hat{m}_0$ against a uniform probability plot of the unadjusted p-values (if the p-values are distributed uniformly, the points on the plot will all lie on a straight line). According to Schweder and Spjøtvoll (1982) and Hochberg and Benjamini (1990), the points on the left side of the plot should be approximately linear with slope $\frac{1}{m_0+1}$, so you can use this plot to evaluate whether your estimate of $\hat{m}_0$ seems reasonable.

Output 80.6.4 p-Value Distribution



The NTRUENULL= option provides several methods for estimating the number of true null hypotheses; the following table displays each method and its estimate for this example:

NTRUENULL=	Estimate
BOOTSTRAP	23.3
DECREASESLOPE	26
KSTEST	35
LEASTSQUARES	28
LOWESTSLOPE	43
MEANDIFF	42
SPLINE	50.1

Another method of estimating the number of true null hypotheses fits a finite mixture model (mixing a uniform with a beta) to the distribution of the unadjusted *p*-values (Allison et al. 2002). Osborne (2006) provides the following PROC NLMIXED statements to fit this model:

```
proc nlmixed data=test;
   parameters pi0=0.5 a=.1 b=.1;
   pi1= 1-pi0;
   bounds 0 <= pi0 <= 1;
   loglikelihood= log(pi0+pi1*pdf('beta',Probt,a,b));
   model Probt ~ general(loglikelihood);
run;
```

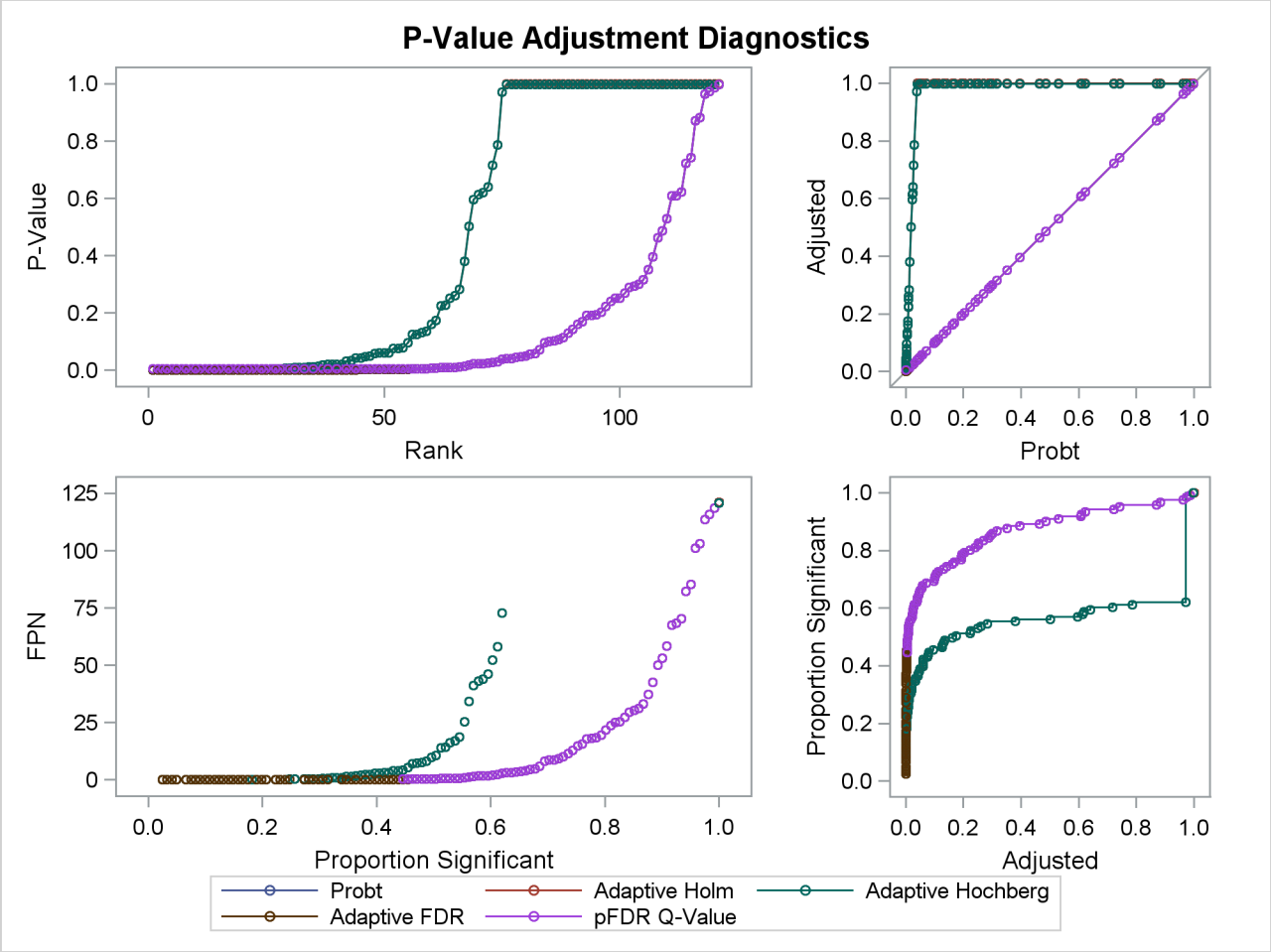
You might have to change the initial parameter values in the PARAMETERS statement to achieve convergence; see Chapter 83, "The NLMIXED Procedure," for more information. This mixture model estimates $\hat{\pi}_0 = 0$, meaning that the distribution of *p*-values is completely specified by a single beta distribution. If the estimate were, say, $\hat{\pi}_0 = 0.10$, you could then specify it as follows:

```
proc multtest inpvalues(Probt)=test ptruenull=0.10
              aholm ahoc afdr pfdr(positive) nopvalue;
   id Probe_Set_ID;
run;
```

A plot of the unadjusted and adjusted *p*-values for each test is also produced. Due to the large number of tests and adjustments, the plot is not very informative and is not displayed here.

The top two plots in Output 80.6.5 show how the adjusted values compare with each other and the unadjusted *p*-values. The PFDR and AFDR adjustments are eventually smaller than the unadjusted *p*-values since they control the false discovery rate. The adaptive Holm and Hochberg adjustments are almost identical, so the adaptive Holm values are mostly obscured in all four plots. The plot of the Proportion Significant versus the Adjusted *p*-values tells you how many of the tests are significant for each cutoff, while the plot of the number of false positives (FPN) versus the Proportion Significant tells you how many false positives you can expect for that cutoff.

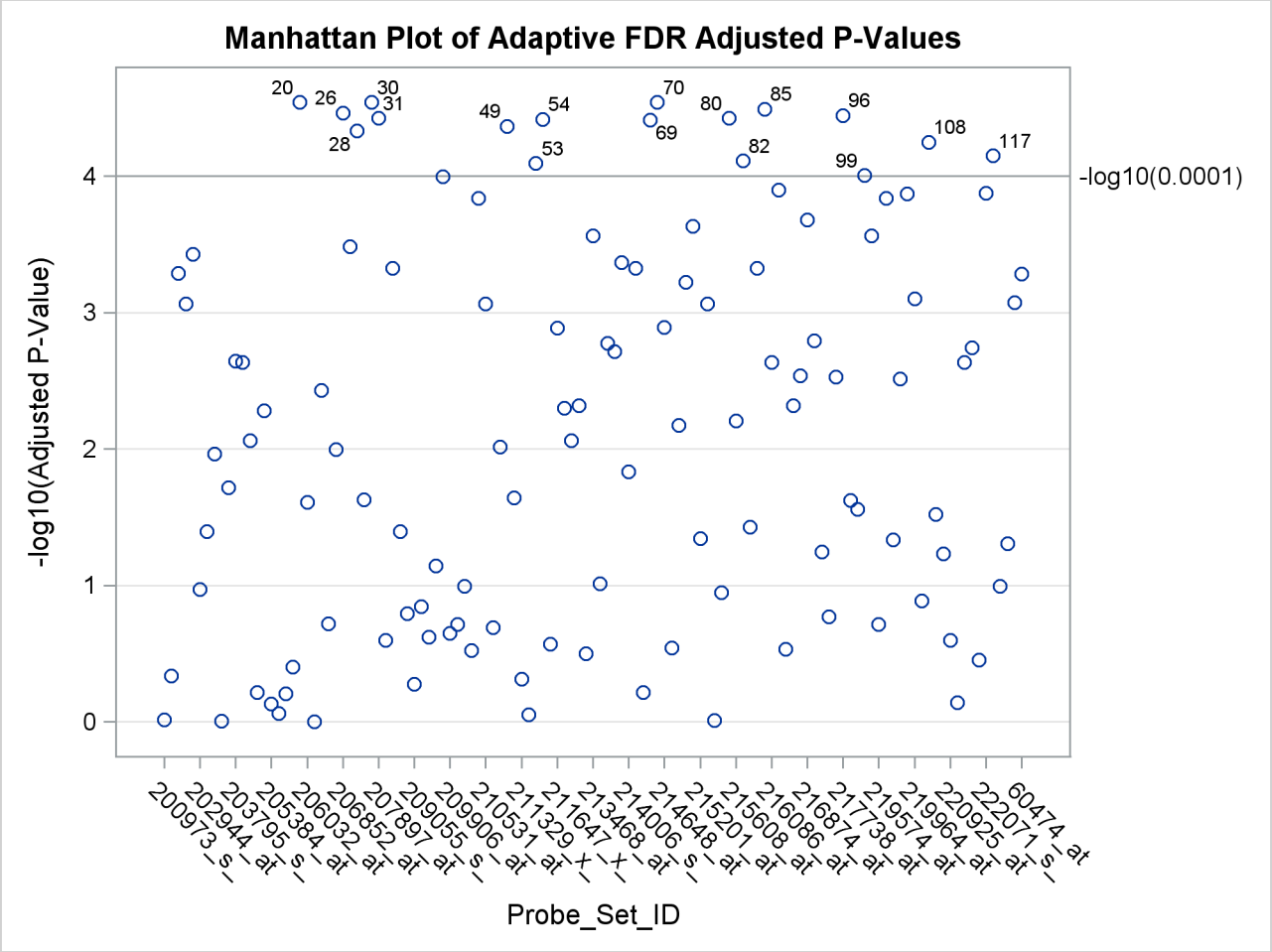
Output 80.6.5 Adjustment Diagnostics



A Manhattan plot displays $-\log_{10}$ of the adjusted *p*-values, so the most significant tests stand out at the top of the plot. The default plot is not displayed here. The following statements create a Manhattan plot of the adaptive FDR *p*-values, with the most significant tests labeled with their observation number. The ID values are displayed on the X axis, and the VREF= option specifies the significance level. This plot is typically created with many more *p*-values, and special ODS Graphics options such as the LABELMAX= option might be required to display the graph. If memory usage is an issue, you might want to store your *p*-values and use the SGPLOT procedure to create a similar graph.

```
ods graphics on / labelmax=1000;
proc multtest inpvalues(Probt)=test afdr nopvalue
   plots=Manhattan(label=obs vref=0.0001);
   id Probe_Set_ID;
run;
ods graphics off;
```

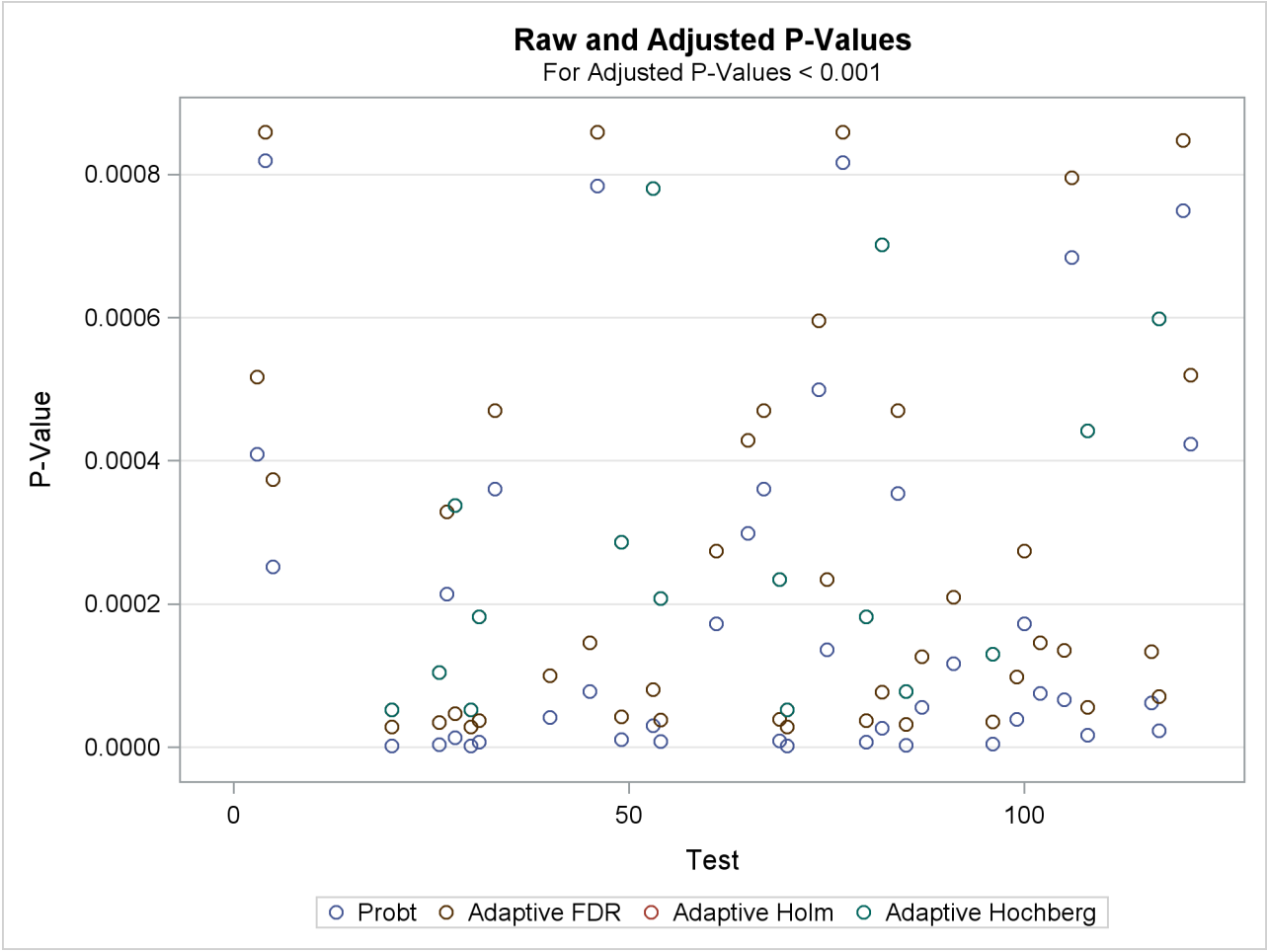
Output 80.6.6 Manhattan Plot



If you have a lot of tests, the "Raw and Adjusted *p*-Values" and "P-Value Adjustment Diagnostics" plots can be more informative if you suppress some of the tests. In the following statements, the SIGONLY=0.001 option selects tests with adjusted *p*-values < 0.001 for display. Output 80.6.7 displays tests with their "significant" adjusted *p*-values:

```
ods graphics on;
proc multtest inpvalues(Probt)=test plots(sigonly=0.001)=PByTest
              aholm ahoc afdr pfdr(positive) nopvalue;
run;
ods graphics off;
```

Output 80.6.7 Raw and Adjusted *p*-Values



References

Agresti, A. (2002). *Categorical Data Analysis*. 2nd ed. New York: John Wiley & Sons.

Allison, D. B., Gadbury, G. L., Moonseong, H., Fernández, J. R., Lee, C., Prolla, T. A., and Weindruch, R. (2002). "A Mixture Model Approach for the Analysis of Microarray Gene Expression Data." *Computational Statistics and Data Analysis* 39:1–20.

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

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.

Benjamini, Y., and Hochberg, Y. (2000). "On the Adaptive Control of the False Discovery Rate in Multiple Testing with Independent Statistics." *Journal of Educational and Behavioral Statistics* 25:60–83.

Benjamini, Y., Krieger, A. M., and Yekutieli, D. (2006). "Adaptive Linear Step-Up False Discovery Rate Controlling Procedures." *Biometrika* 93:491–507.

Benjamini, Y., and Yekutieli, D. (2001). "The Control of the False Discovery Rate in Multiple Testing under Dependency." *Annals of Statistics* 29:1165–1188.

Bickis, M., and Krewski, D. (1986). "Statistical Issues in the Analysis of the Long Term Carcinogenicity Bioassay in Small Rodents: An Empirical Evaluation of Statistical Decision Rules." Environmental Health Directorate, Health Protection Branch, Health and Welfare Canada, Ottawa, Ontario.

Brown, B. W., and Russell, K. (1997). "Methods Correcting for Multiple Testing: Operating Characteristics." *Statistics in Medicine* 16:2511–2528.

Brown, C. C., and Fears, T. R. (1981). "Exact Significance Levels for Multiple Binomial Testing with Application to Carcinogenicity Screens." *Biometrics* 37:763–774.

Cochran, W. G. (1954). "Some Methods for Strengthening the Common χ^2 Tests." *Biometrics* 10:417–451.

Dinse, G. E. (1985). "Testing for Trend in Tumor Prevalence Rates, Part 1: Nonlethal Tumors." *Biometrics* 41:751–770.

Dmitrienko, A., Molenberghs, G., Chuang-Stein, C., and Offen, W. (2005). *Analysis of Clinical Trials Using SAS: A Practical Guide*. Cary, NC: SAS Institute Inc.

Dudoit, S., Shaffer, J. P., and Boldrick, J. C. (2003). "Multiple Hypothesis Testing in Microarray Experiments." *Statistical Science* 18:71–103.

Freedman, D. A. (1981). "Bootstrapping Regression Models." *Annals of Statistics* 9:1218–1228.

Freeman, M. F., and Tukey, J. W. (1950). "Transformations Related to the Angular and the Square Root." *Annals of Mathematical Statistics* 21:607–611.

Gibson, G., and Wolfinger, R. D. (2004). "Gene Expression Profiling Using Mixed Models." In *Genetic Analysis of Complex Traits Using SAS*, edited by A. M. Saxton, 251–278. Cary, NC: SAS Institute Inc.

Good, I. J. (1987). "A Survey of the Use of the Fast Fourier Transform for Computing Distributions." *Journal of Statistical Computation and Simulation* 28:87–93.

Heyse, J., and Rom, D. (1988). "Adjusting for Multiplicity of Statistical Tests in the Analysis of Carcinogenicity Studies." *Biometrical Journal* 30:883–896.

Hochberg, Y. (1988). "A Sharper Bonferroni Procedure for Multiple Significance Testing." *Biometrika* 75:800–803.

Hochberg, Y., and Benjamini, Y. (1990). "More Powerful Procedures for Multiple Significance Testing." *Statistics in Medicine* 9:811–818.

Hochberg, Y., and Tamhane, A. C. (1987). *Multiple Comparison Procedures*. New York: John Wiley & Sons.

Hoel, D. G., and Walburg, H. E. (1972). "Statistical Analysis of Survival Experiments." *Journal of the National Cancer Institute* 49:361–372.

Holland, B. S., and Copenhaver, M. D. (1987). "An Improved Sequentially Rejective Bonferroni Test Procedure." *Biometrics* 43:417–424.

Holm, S. (1979). "A Simple Sequentially Rejective Multiple Test Procedure." *Scandinavian Journal of Statistics* 6:65–70.

Hommel, G. (1988). "A Comparison of Two Modified Bonferroni Procedures." *Biometrika* 75:383–386.

Hsueh, H., Chen, J. J., and Kodell, R. L. (2003). "Comparison of Methods for Estimating the Number of True Null Hypotheses in Multiplicity Testing." *Journal of Biopharmaceutical Statistics* 13:675–689.

Lagakos, S. W., and Louis, T. A. (1985). "The Statistical Analysis of Rodent Tumorigenicity Experiments." In *Toxicological Risk Assessment*, edited by D. B. Clayson, D. Krewski, and I. Munro, 144–163. Boca Raton, FL: CRC Press.

Liu, W. (1996). "Multiple Tests of a Non-hierarchical Finite Family of Hypotheses." *Journal of the Royal Statistical Society, Series B* 58:455–461.

Mantel, N. (1980). "Assessing Laboratory Evidence for Neoplastic Activity." *Biometrics* 36:381–399.

Mantel, N., and Haenszel, W. (1959). "Statistical Aspects of Analysis of Data from Retrospective Studies of Disease." *Journal of the National Cancer Institute* 22:719–748.

Marcus, R., Peritz, E., and Gabriel, K. R. (1976). "On Closed Testing Procedures with Special Reference to Ordered Analysis of Variance." *Biometrika* 63:655–660.

Miller, J. J. (1978). "The Inverse of the Freeman-Tukey Double Arcsine Transformation." *American Statistician* 32:138.

Osborne, J. A. (2006). "Estimating the False Discovery Rate Using SAS." In *Proceedings of the Thirty-First Annual SAS Users Group International Conference*. Cary, NC: SAS Institute Inc. http://www2.sas.com/proceedings/sugi31/190-31.pdf.

Pagano, M., and Tritchler, D. (1983). "On Obtaining Permutation Distributions in Polynomial Time." *Journal of the American Statistical Association* 78:435–440.

Peto, R., Pike, M. C., Day, N. E., Gray, R. G., Lee, P. N., Parish, S., Peto, J., Richards, S., and Wahrendorf, J. (1980). "Guidelines for Simple, Sensitive Significance Tests for Carcinogenic Effects in Long-Term Animal Experiments." In *Suppl. 2: Long-Term and Short-Term Screening Assays for Carcinogens—a Critical Appraisal*, 311–426. IARC Monographs on the Evaluation of Carcinogenic Risks to Humans. Lyon: International Agency for Research on Cancer.

Press, W. H., Teukolsky, S. A., Vetterling, W. T., and Flannery, B. P. (1992). *Numerical Recipes in C: The Art of Scientific Computing*. 2nd ed. Cambridge: Cambridge University Press.

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.

Satterthwaite, F. E. (1946). "An Approximate Distribution of Estimates of Variance Components." *Biometrics Bulletin* 2:110–114.

Schweder, T., and Spjøtvoll, E. (1982). "Plots of *P*-Values to Evaluate Many Tests Simultaneously." *Biometrika* 69:493–502.

Shaffer, J. P. (1986). "Modified Sequentially Rejective Multiple Test Procedures." *Journal of the American Statistical Association* 81:826–831.

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.

Soper, K. A., and Tonkonoh, N. (1993). "The Discrete Distribution Used for the Log-Rank Test Can Be Inaccurate." *Biometrical Journal* 35:291–298.

Storey, J. D. (2002). "A Direct Approach to False Discovery Rates." *Journal of the Royal Statistical Society, Series B* 64:479–498.

Storey, J. D., Taylor, J. E., and Siegmund, D. (2004). "Strong Control, Conservative Point Estimation, and Simultaneous Conservative Consistency of False Discovery Rates: A Unified Approach." *Journal of the Royal Statistical Society, Series B* 66:187–205.

Storey, J. D., and Tibshirani, R. (2003). "Statistical Significance for Genomewide Studies." *Proceedings of the National Academy of Sciences of the United States of America* 100:9440–9445.

Turkheimer, F. E., Smith, C. B., and Schmidt, K. (2001). "Estimation of the Number of 'True' Null Hypotheses in Multivariate Analysis of Neuroimaging Data." *NeuroImage* 13:920–930.

Westfall, P. H. (2005). "Combining P Values." In *Encyclopedia of Biostatistics*, 2nd ed., edited by P. Armitage and T. Colton. Chichester, UK: John Wiley & Sons.

Westfall, P. H., and Lin, Y. (1988). "Estimating Optimal Continuity Corrections in Run Time." In *Proceedings of the Statistical Computing Section*, 297–298. Alexandria, VA: American Statistical Association.

Westfall, P. H., and Soper, K. A. (1994). "Nonstandard Uses of PROC MULTTEST: Permutational Peto Tests; Permutational and Unconditional *t* and Binomial Tests." In *Proceedings of the Nineteenth Annual SAS Users Group International Conference*, 986–989. Cary, NC: SAS Institute Inc. http://www.sascommunity.org/sugi/SUGI94/Sugi-94-173%20Westfall%20Soper.pdf.

Westfall, P. H., Tobias, R. D., Rom, D., Wolfinger, R. D., and Hochberg, Y. (1999). *Multiple Comparisons and Multiple Tests Using the SAS System*. Cary, NC: SAS Institute Inc.

Westfall, P. H., and Wolfinger, R. D. (1997). "Multiple Tests with Discrete Distributions." *American Statistician* 51:3–8.

Westfall, P. H., and Wolfinger, R. D. (2000). "Closed Multiple Testing Procedures and PROC MULTTEST." *Observations* (June):21 pages. Paper available at http://support.sas.com/kb/22/addl/fusion22950_1_multtest.pdf.

Westfall, P. H., and Young, S. S. (1989). "*P*-Value Adjustments for Multiple Tests in Multivariate Binomial Models." *Journal of the American Statistical Association* 84:780–786.

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

Yates, F. (1984). "Tests of Significance for 2×2 Contingency Tables." *Journal of the Royal Statistical Society, Series A* 147:426–463.

Yekutieli, D., and Benjamini, Y. (1999). "Resampling-Based False Discovery Rate Controlling Multiple Test Procedures for Correlated Test Statistics." *Journal of Statistical Planning and Inference* 82:171–196.

Subject Index

adaptive FDR adjustment
    MULTTEST procedure, 6418
adaptive Hochberg adjustment
    MULTTEST procedure, 6418
adaptive Holm adjustment
    MULTTEST procedure, 6418
adaptive methods
    MULTTEST procedure, 6446
adjusted *p*-value
    MULTTEST procedure, 6412, 6441

Bonferroni adjustment
    MULTTEST procedure, 6419, 6443
bootstrap adjustment
    MULTTEST procedure, 6415, 6419, 6443, 6458
bootstrap FDR adjustment
    MULTTEST procedure, 6419

Cochran-Armitage test for trend
    continuity correction (MULTTEST), 6435
    MULTTEST procedure, 6432, 6434, 6455
    permutation distribution (MULTTEST), 6435
    two-tailed test (MULTTEST), 6437
computational resources
    MULTTEST procedure, 6450
convolution
    distribution (MULTTEST), 6436

dependent FDR adjustment
    MULTTEST procedure, 6419
double arcsine test
    MULTTEST procedure, 6437

exact tests
    permutation test (MULTTEST), 6435
expected trend
    MULTTEST procedure, 6437

false discovery rate, 6442
    adjustment (MULTTEST), 6446
familywise error rate, 6441
    adjustment (MULTTEST), 6442
fast Fourier transform
    MULTTEST procedure, 6436
FDR, *see* false discovery rate
Fisher combination
    adjustment (MULTTEST), 6445
Fisher exact test
    MULTTEST procedure, 6430, 6432, 6439, 6465

Freeman-Tukey test
    MULTTEST procedure, 6432, 6437, 6458
FWE, *see* familywise error rate

Hochberg
    adjustment (MULTTEST), 6445
Hommel
    adjustment (MULTTEST), 6445
hypergeometric
    distribution (MULTTEST), 6440
    variance (MULTTEST), 6434

Liptak combination
    adjustment (MULTTEST), 6445

missing values
    MULTTEST procedure, 6449
mortality test
    MULTTEST procedure, 6438, 6462
MULTTEST procedure
    adaptive FDR adjustment, 6418
    adaptive Hochberg adjustment, 6418
    adaptive Holm adjustment, 6418
    adaptive methods, 6446
    adjusted *p*-value, 6412, 6441
    Bonferroni adjustment, 6419, 6443
    bootstrap adjustment, 6415, 6419, 6443
    bootstrap FDR adjustment, 6419
    Cochran-Armitage test, 6432, 6434, 6437, 6455
    computational resources, 6450
    convolution distribution, 6436
    dependent FDR adjustment, 6419
    displayed output, 6452
    double arcsine test, 6437
    expected trend, 6437
    false discovery rate, 6442
    false discovery rate adjustment, 6446
    familywise error rate, 6441
    familywise error rate adjustment, 6442
    fast Fourier transform, 6436
    Fisher combination adjustment, 6445
    Fisher exact test, 6430, 6432, 6439
    Freeman-Tukey test, 6432, 6437, 6458
    Hochberg adjustment, 6445
    Hommel adjustment, 6445
    introductory example, 6413
    linear trend test, 6435
    Liptak combination adjustment, 6445
    missing values, 6449

ODS graph names, 6454
ODS table names, 6453
ordering of effects, 6423
output data sets, 6450
p-value adjustments, 6412, 6441
permutation adjustment, 6424, 6444, 6465
permutation FDR adjustment, 6420
Peto test, 6432, 6438, 6462
positive false discovery rate, 6442
positive FDR adjustment, 6424, 6448
resampled data sets, 6451
Sidak's adjustment, 6427, 6443
statistical tests, 6434
step-down methods, 6444
Stouffer combination adjustment, 6445
strata weights, 6437
t test, 6432, 6440, 6458

ODS graph names
MULTTEST procedure, 6454
output data sets
MULTTEST procedure, 6450, 6451

p-value adjustments
adaptive FDR (MULTTEST), 6418
adaptive Hochberg (MULTTEST), 6418
adaptive Holm (MULTTEST), 6418
Bonferroni (MULTTEST), 6419, 6443
bootstrap (MULTTEST), 6415, 6419, 6443, 6458
bootstrap FDR (MULTTEST), 6419
dependent FDR (MULTTEST), 6419
false discovery rate (MULTTEST), 6446
familywise error rate (MULTTEST), 6442
Fisher combination (MULTTEST), 6445
Hochberg (MULTTEST), 6445
Hommel (MULTTEST), 6445
Liptak combination (MULTTEST), 6445
MULTTEST procedure, 6412, 6441
permutation (MULTTEST), 6424, 6444, 6465
permutation FDR (MULTTEST), 6420
positive FDR (MULTTEST), 6424, 6448
Sidak (MULTTEST), 6427, 6443, 6462
Stouffer combination (MULTTEST), 6445
permutation
p-value adjustments (MULTTEST), 6424, 6444, 6465
permutation FDR adjustment
MULTTEST procedure, 6420
Peto test
MULTTEST procedure, 6432, 6438, 6462
pFDR, *see* positive false discovery rate
positive false discovery rate, 6442
positive FDR adjustment
MULTTEST procedure, 6424, 6448
prevalence test
MULTTEST procedure, 6438, 6462

resampled data sets
MULTTEST procedure, 6451

Sidak's adjustment
MULTTEST procedure, 6427, 6443, 6462
statistical
tests (MULTTEST), 6434
step-down methods
MULTTEST procedure, 6444, 6462
Stouffer combination
adjustment (MULTTEST), 6445
strata weights
MULTTEST procedure, 6437

t test
MULTTEST procedure, 6432, 6440, 6458

Syntax Index

ADAPTIVEFDR option
PROC MULTTEST statement, 6418, 6448
ADAPTIVEHOCHBERG option
PROC MULTTEST statement, 6418
ADAPTIVEHOLM option
PROC MULTTEST statement, 6418

BINOMIAL option
TEST statement (MULTTEST), 6433
BONFERRONI option
PROC MULTTEST statement, 6419, 6443
BOOTSTRAP option
PROC MULTTEST statement, 6414, 6419, 6443, 6458
BY statement
MULTTEST procedure, 6428

CA option
TEST statement (MULTTEST), 6432, 6434, 6455
CENTER option
PROC MULTTEST statement, 6419
CLASS statement
MULTTEST procedure, 6429
CONTINUITY= option
TEST statement (MULTTEST), 6433
CONTRAST statement
MULTTEST procedure, 6429

DATA= option
PROC MULTTEST statement, 6419
DDFM= option
TEST statement (MULTTEST), 6433
DEPENDENTFDR option
PROC MULTTEST statement, 6419, 6447

EPSILON= option
PROC MULTTEST statement, 6419

FDR option
PROC MULTTEST statement, 6419, 6446
FDRBOOT option
PROC MULTTEST statement, 6419, 6447
FDRPERM option
PROC MULTTEST statement, 6420, 6447
FISHER option
TEST statement (MULTTEST), 6430, 6432, 6439, 6465
FISHER_C option
PROC MULTTEST statement, 6420, 6445

FREQ statement
MULTTEST procedure, 6430
FT option
TEST statement (MULTTEST), 6432, 6437, 6458

HOC option
PROC MULTTEST statement, 6420, 6445
HOLM option
PROC MULTTEST statement, 6420, 6427
HOM option
PROC MULTTEST statement, 6420
HOMMEL option
PROC MULTTEST statement, 6445

ID statement
MULTTEST procedure, 6431
INPVALUES= option
PROC MULTTEST statement, 6420

LIPTAK option
PROC MULTTEST statement, 6420, 6445
LOWERTAILED option
TEST statement (MULTTEST), 6433

MEAN option
TEST statement (MULTTEST), 6432, 6440, 6458
MULTTEST procedure, 6416
syntax, 6416
MULTTEST procedure, BY statement, 6428
MULTTEST procedure, CLASS statement, 6429
TRUNCATE option, 6429
MULTTEST procedure, CONTRAST statement, 6429
MULTTEST procedure, FREQ statement, 6430
MULTTEST procedure, ID statement, 6431
MULTTEST procedure, PROC MULTTEST statement, 6417
ADAPTIVEFDR option, 6418, 6448
ADAPTIVEHOCHBERG option, 6418
ADAPTIVEHOLM option, 6418
BONFERRONI option, 6419, 6443
BOOTSTRAP option, 6414, 6419, 6443, 6458
CENTER option, 6419
DATA= option, 6419
DEPENDENTFDR option, 6419, 6447
EPSILON= option, 6419
FDR option, 6419, 6446
FDRBOOT option, 6419, 6447
FDRPERM option, 6420, 6447
FISHER_C option, 6420, 6445

HOC option, 6420, 6445
HOLM option, 6420, 6427
HOM option, 6420
HOMMEL option, 6445
INPVALUES= option, 6420
LIPTAK option, 6420, 6445
NOCENTER option, 6420
NOPRINT option, 6420
NOPVALUE option, 6421
NOTABLES option, 6421
NOZEROS option, 6421
NSAMPLE= option, 6421
NTRUENULL= option, 6421
ORDER= option, 6423, 6465
OUT= option, 6423, 6450
OUTPERM= option, 6424, 6451, 6455
OUTSAMP= option, 6424, 6451, 6458
PDATA= option, 6424
PERMUTATION option, 6424, 6444, 6455, 6465
PFDR option, 6424, 6448
PLOTS= option, 6425
PTRUENULL= option, 6427
RANUNI option, 6427
SEED= option, 6427
SIDAK option, 6427, 6443, 6462
STEPBON option, 6427
STEPBOOT option, 6428
STEPPERM option, 6428
STEPSID option, 6428, 6462
STOUFFER option, 6428, 6445
MULTTEST procedure, STRATA statement, 6431
WEIGHT= option, 6431, 6437
MULTTEST procedure, TEST statement, 6431
BINOMIAL option, 6433
CA option, 6432, 6434, 6455
CONTINUITY= option, 6433
DDFM= option, 6433
FISHER option, 6430, 6432, 6439, 6465
FT option, 6432, 6437, 6458
LOWERTAILED option, 6433
MEAN option, 6432, 6440, 6458
PERMUTATION= option, 6433, 6435, 6455
PETO option, 6432, 6438, 6462
TIME= option, 6433
UPPERTAILED option, 6433

NOCENTER option
PROC MULTTEST statement, 6420
NOPRINT option
PROC MULTTEST statement, 6420
NOPVALUE option
PROC MULTTEST statement, 6421
NOTABLES option
PROC MULTTEST statement, 6421
NOZEROS option
PROC MULTTEST statement, 6421
NSAMPLE= option
PROC MULTTEST statement, 6421
NTRUENULL= option
PROC MULTTEST statement, 6421

ORDER= option
PROC MULTTEST statement, 6423, 6465
OUT= option
PROC MULTTEST statement, 6423, 6450
OUTPERM= option
PROC MULTTEST statement, 6424, 6451, 6455
OUTSAMP= option
PROC MULTTEST statement, 6424, 6451, 6458

PDATA= option
PROC MULTTEST statement, 6424
PERMUTATION option
PROC MULTTEST statement, 6424, 6444, 6455, 6465
PERMUTATION= option
TEST statement (MULTTEST), 6433, 6435, 6455
PETO option
TEST statement (MULTTEST), 6432, 6438, 6462
PFDR option
PROC MULTTEST statement, 6424, 6448
PLOTS= option
PROC MULTTEST statement, 6425
PROC MULTTEST statement, *see* MULTTEST procedure
PTRUENULL= option
PROC MULTTEST statement, 6427

RANUNI option
PROC MULTTEST statement, 6427

SEED= option
PROC MULTTEST statement, 6427
SIDAK option
PROC MULTTEST statement, 6427, 6443, 6462
STEPBON option
PROC MULTTEST statement, 6427
STEPBOOT option
PROC MULTTEST statement, 6428
STEPPERM option
PROC MULTTEST statement, 6428
STEPSID option
PROC MULTTEST statement, 6428, 6462
STOUFFER option
PROC MULTTEST statement, 6428, 6445
STRATA statement
MULTTEST procedure, 6431

TEST statement

MULTTEST procedure, 6431
TIME= option
TEST statement (MULTTEST), 6433
TRUNCATE option
CLASS statement (MULTTEST), 6429

UPPERTAILED option
TEST statement (MULTTEST), 6433

WEIGHT= option
STRATA statement (MULTTEST), 6431, 6437