SAS/STAT® 15.1 User's Guide
The CATMOD Procedure

This document is an individual chapter from *SAS/STAT® 15.1 User's Guide*.

The correct bibliographic citation for this manual is as follows: SAS Institute Inc. 2018. *SAS/STAT® 15.1 User's Guide*. Cary, NC: SAS Institute Inc.

SAS/STAT® 15.1 User's Guide

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

November 2018

Chapter 33
The CATMOD Procedure

Contents

Overview: CATMOD Procedure

The CATMOD procedure performs categorical data modeling of data that can be represented by a contingency table. PROC CATMOD fits linear models to functions of response frequencies, and it can be used for linear modeling, log-linear modeling, logistic regression, and repeated measurement analysis. PROC CATMOD uses the following estimation methods:

- weighted least squares (WLS) estimation of parameters for a wide range of general linear models
- maximum likelihood (ML) estimation of parameters for log-linear models and the analysis of generalized logits

The CATMOD procedure provides a wide variety of categorical data analyses, many of which are generalizations of continuous data analysis methods. For example, analysis of variance, in the traditional sense, refers to the analysis of means and the partitioning of variation among the means into various sources. Here, the term *analysis of variance* is used in a generalized sense to denote the analysis of response functions and the partitioning of variation among those functions into various sources. The response functions might be mean scores if the dependent variables are ordinally scaled. But they can also be marginal probabilities, cumulative logits, or other functions that incorporate the essential information from the dependent variables.

NOTE: PROC CATMOD specializes in WLS modeling and analysis of a wide range of models on contingency tables. For ML modeling on standard models, especially with continuous predictors, it might be more appropriate to use a procedure such as PROC GENMOD or PROC LOGISTIC; see Chapter 48, "The GENMOD Procedure," and Chapter 76, "The LOGISTIC Procedure," for more information.

Types of Input Data

The data that PROC CATMOD analyzes are usually supplied in one of two ways. First, you can supply raw data, where each observation is a subject. Second, you can supply cell count data, where each observation is a cell in a contingency table. (A third way, which uses direct input of the covariance matrix, is also available; details are given in the section "Inputting Response Functions and Covariances Directly" on page 2165.)

Suppose detergent brand preference is related to three other categorical variables: water softness, water temperature, and previous use of a brand of detergent. In the raw data case, each observation in the input data set identifies a given respondent in the study and contains information about all four variables. The data set contains the same number of observations as the survey had respondents. In the cell count case, each observation identifies a given cell in the four-way table of water softness, water temperature, previous use of brand, and brand preference. A fifth variable contains the number of respondents in the cell. In the analysis, this fifth variable is identified in a WEIGHT statement. The data set contains the same number of observations as the number of cross-classifications formed by the four categorical variables. For more about this particular example, see Example 33.1. For additional details, see the section "Input Data Sets" on page 2163.

Most of the examples in this chapter use cell counts as input and use a WEIGHT statement.

Types of Statistical Analyses

This section illustrates, by example, the wide variety of categorical data analyses that PROC CATMOD provides. For each type of analysis, a brief description of the statistical problem and the SAS statements to provide the analysis are given. For each analysis, assume that the input data set consists of a set of cell counts from a contingency table. The variable specified in the WEIGHT statement contains these counts. In all these analyses, both the dependent and independent variables are categorical.

Linear Model Analysis

Suppose you want to analyze the relationship between the dependent variables (r1, r2) and the independent variables (a, b). Analyze the marginal probabilities of the dependent variables, and use a main-effects model:

```
proc catmod;
   weight wt;
   response marginals;
   model r1*r2=a b;
quit;
```

Log-Linear Model Analysis

Suppose you want to analyze the nominal dependent variables (r1, r2, r3) with a log-linear model. Use maximum likelihood analysis, include the main effects and the r1*r2 interaction in the model, and obtain the predicted cell frequencies:

```
proc catmod;
   weight wt;
   model r1*r2*r3=_response_ / pred=freq;
   loglin r1|r2 r3;
quit;
```

Logistic Regression

Suppose you want to analyze the relationship between the nominal dependent variable (r) and the independent variables (x1, x2) with a logistic regression analysis. Use maximum likelihood estimation:

```
proc catmod;
   weight wt;
   direct x1 x2;
   model r=x1 x2;
quit;
```

If x1 and x2 are continuous so that each observation has a unique value of these two variables, then it might be more appropriate to use the LOGISTIC or GENMOD procedure. (See the section "Logistic Regression" on page 2171.)

Repeated Measures Analysis

Suppose the dependent variables (r1, r2, r3) represent the same type of measurement taken at three different times. Analyze the relationship among the dependent variables, the repeated measurement factor (time), and the independent variable (a):

```
proc catmod;
   weight wt;
   response marginals;
   model r1*r2*r3=_response_|a;
   repeated time 3 / _response_=time;
quit;
```

Analysis of Variance

Suppose you want to investigate the relationship between the dependent variable (r) and the independent variables (a, b). Analyze the mean of the dependent variable, and include all main effects and interactions in the model:

```
proc catmod;
   weight wt;
   response mean;
   model r=a|b;
quit;
```

Linear Regression

PROC CATMOD can analyze the relationship between the dependent variables (r1, r2) and the independent variables (x1, x2). Use a linear regression analysis to analyze the marginal probabilities of the dependent variables:

```
proc catmod;
   weight wt;
   direct x1 x2;
   response marginals;
   model r1*r2=x1 x2;
quit;
```

Logistic Analysis of Ordinal Data

Suppose you want to analyze the relationship between the ordinally scaled dependent variable (r) and the independent variable (a). Use cumulative logits to take into account the ordinal nature of the dependent variable, and use weighted least squares estimation:

```
proc catmod;
   weight wt;
   response clogits;
   model r=_response_ a;
quit;
```

Sample Survey Analysis

Suppose the data set contains estimates of a vector of four functions and their covariance matrix, estimated in such a way as to correspond to the sampling process that is used. Analyze the functions with respect to the independent variables (a, b), and use a main-effects model:

```
proc catmod;
   response read b1-b10;
   model _f_=_response_;
   factors  a 2 , b 5 / _response_=a b;
quit;
```

Background: The Underlying Model

The CATMOD procedure analyzes data that can be represented by a two-dimensional contingency table. The rows of the table correspond to populations (or samples) formed on the basis of one or more independent variables. The columns of the table correspond to observed responses formed on the basis of one or more dependent variables. The frequency in the (i, j) cell is the number of subjects in the ith population that have the jth response. The frequencies in the table are assumed to follow a product multinomial distribution, corresponding to a sampling design in which a simple random sample is taken for each population. The contingency table can be represented as shown in Table 33.1.

Table 33.1 Contingency Table Representation

	Response				
Sample	**1**	**2**	$\cdots$	***r***	**Total**
1	n_{11}	n_{12}	$\cdots$	n_{1r}	n_1
2	n_{21}	n_{22}	$\cdots$	n_{2r}	n_2
$\vdots$	$\vdots$	$\vdots$	$\ddots$	$\vdots$	$\vdots$
s	n_{s1}	n_{s2}	$\cdots$	n_{sr}	n_s

For each sample i, the probability of the jth response (π_{ij}) is estimated by the sample proportion, $p_{ij} = n_{ij}/n_i$. The vector ($\mathbf{p}$) of all such proportions is then transformed into a vector of functions, denoted by

$\mathbf{F} = \mathbf{F}(\mathbf{p})$. If $\boldsymbol{\pi}$ denotes the vector of true probabilities for the entire table, then the functions of the true probabilities, denoted by $\mathbf{F}(\boldsymbol{\pi})$, are assumed to follow a linear model

$$\mathbf{E_A}(\mathbf{F}) = \mathbf{F}(\boldsymbol{\pi}) = \mathbf{X}\boldsymbol{\beta}$$

where $\mathbf{E_A}$ denotes asymptotic expectation, $\mathbf{X}$ is the design matrix containing fixed constants, and $\boldsymbol{\beta}$ is a vector of parameters to be estimated.

PROC CATMOD provides two estimation methods:

- The weighted least squares method minimizes the weighted residual sum of squares for the model. The weights are contained in the inverse covariance matrix of the functions $\mathbf{F}(\mathbf{p})$. According to central limit theory, if the sample sizes within populations are sufficiently large, the elements of $\mathbf{F}$ and $\mathbf{b}$ (the estimate of $\boldsymbol{\beta}$) are distributed approximately as multivariate normal. This allows the computation of statistics for testing the goodness of fit of the model and the significance of other sources of variation. For details of the theory, see Grizzle, Starmer, and Koch (1969) or Koch et al. (1977, Appendix 1). Weighted least squares estimation is available for all types of response functions.
- The maximum likelihood method estimates the parameters of the linear model so as to maximize the value of the joint multinomial likelihood function of the responses. Maximum likelihood estimation is available only for the standard response functions, logits and generalized logits, which are used for logistic regression analysis and log-linear model analysis. Two methods of maximization are available: Newton-Raphson and iterative proportional fitting. For details of the theory, see Bishop, Fienberg, and Holland (1975).

Following parameter estimation, hypotheses about linear combinations of the parameters can be tested. For that purpose, PROC CATMOD computes generalized Wald (1943) statistics, which are approximately chi-square distributed if the sample sizes are sufficiently large and the null hypotheses are true.

Linear Models Contrasted with Log-Linear Models

Linear model methods typified by the Grizzle, Starmer, and Koch (1969) approach make a very clear distinction between independent and dependent variables. The emphasis of these methods is estimation and hypothesis testing of the model parameters. Therefore, it is easy to test for differences among probabilities, perform repeated measures analysis, and test for marginal homogeneity, but it is difficult to test for independence and generalized independence. These methods are a natural extension of the usual ANOVA approach for continuous data.

In contrast, log-linear model methods typified by the Bishop, Fienberg, and Holland (1975) approach do not make an a priori distinction between independent and dependent variables, although model specifications that allow for the distinction can be made. The emphasis of these methods is on model building, goodness-of-fit tests, and estimation of cell frequencies or probabilities for the underlying contingency table. With these methods, it is easy to test independence and generalized independence, but it is difficult to test for differences among probabilities, do repeated measures analysis, and test for marginal homogeneity.

Using PROC CATMOD Interactively

You can use the CATMOD procedure interactively. After specifying a model with a MODEL statement and running PROC CATMOD with a RUN statement, you can execute any statement without reinvoking PROC CATMOD. You can execute the statements singly or in groups by following the single statement or group of statements with a RUN statement. Note that you can use more than one MODEL statement; this is an important difference from the GLM procedure.

If you use PROC CATMOD interactively, you can end the CATMOD procedure with a DATA step, another PROC step, an ENDSAS statement, or a QUIT statement. The syntax of the QUIT statement is as follows:

```
quit;
```

When you are using PROC CATMOD interactively, additional RUN statements do not end the procedure run but tell the procedure to execute additional statements.

When the CATMOD procedure detects a BY statement, it disables interactive processing; that is, once the BY statement and the next RUN statement are encountered, processing proceeds for each BY group in the data set, and no additional statements are accepted by the procedure. For example, the following statements perform three analyses: one for the entire data set, one for males, and one for females:

```
proc catmod;
   weight wt;
   response marginals;
   model r1*r2=a|b;
run;
   by sex;
run;
```

Note that the BY statement can appear after the first RUN statement; this is an important difference from PROC GLM, which requires that the BY statement appear before the first RUN statement.

Getting Started: CATMOD Procedure

The CATMOD procedure is a general modeling procedure for categorical data analysis, and it can be used for sophisticated analyses that require matrix specification of the response function and the design matrix. It can also be used to perform basic analysis-of-variance-type analyses that require only a few statements. The following is a basic example.

Weighted Least Squares Analysis of Mean Response

Consider the data in Table 33.2 (Stokes, Davis, and Koch 2000).

Table 33.2 Colds in Children

		Periods with Colds			
Sex	**Residence**	**0**	**1**	**2**	**Total**
Female	Rural	45	64	71	180
Female	Urban	80	104	116	300
Male	Rural	84	124	82	290
Male	Urban	106	117	87	310

For male and female children in rural and urban counties, the number of periods (of two) in which subjects report cold symptoms are recorded. So 45 subjects who are female and in rural counties report no cold symptoms, and 71 subjects who are female and from rural counties report colds in both periods.

The question of interest is whether the mean number of periods with colds reported is associated with gender or type of county. There is no reason to believe that the mean number of periods with colds is normally distributed, so a weighted least squares analysis of these data is performed with PROC CATMOD instead of an analysis of variance with PROC ANOVA or PROC GLM.

The input data for categorical data are often recorded in frequency form, with the counts for each particular profile being the input values. For the colds data, the input SAS data set colds is created with the following statements. The variable count contains the frequency of observations that have the particular profile described by the values of the other variables in that input line.

```
data colds;
   input sex $ residence $ periods count @@;
   datalines;
female rural 0  45  female rural 1  64  female rural 2  71
female urban 0  80  female urban 1 104  female urban 2 116
male   rural 0  84  male   rural 1 124  male   rural 2  82
male   urban 0 106  male   urban 1 117  male   urban 2  87
;
```

In order to fit a model to the mean number of periods with colds, you have to specify the response function in PROC CATMOD. The default response function is the logit if the response variable has two values, and it is generalized logits if the response variable has more than two values. If you want a different response function, then you specify that function in the RESPONSE statement. To request the mean number of periods with colds, you specify the MEANS option in the RESPONSE statement.

You can request a model consisting of the main effects and interaction of the variables sex and residence just as you would in the GLM procedure. Unlike the GLM procedure, PROC CATMOD does not require you to use a CLASS statement to treat a variable as a classification variable. In the CATMOD procedure, all variables in the MODEL statement are treated as classification variables unless you specify otherwise with a DIRECT statement. To verify that your model is specified correctly, you can specify the DESIGN option in the MODEL statement to display the design matrix.

The PROC CATMOD statements needed to model mean periods of colds with a main-effects and interaction model are as follows:

```
proc catmod data=colds;
   weight count;
   response means;
   model periods = sex residence sex*residence / design;
run;
```

The results of this analysis are shown in Figure 33.1 through Figure 33.3.

In Figure 33.1, the CATMOD procedure first displays a summary of the contingency table you are analyzing. The "Population Profiles" table lists the values of the explanatory variables that define each population, or row of the underlying contingency table, and labels each group with a sample number. The number of observations in each population is also displayed. The "Response Profiles" table lists the variable levels that define the response, or columns of the underlying contingency table.

Figure 33.1 Model Information and Profile Tables

The CATMOD Procedure

Data Summary			
Response	periods	**Response Levels**	3
Weight Variable	count	**Populations**	4
Data Set	COLDS	**Total Frequency**	1080
Frequency Missing	0	**Observations**	12

Population Profiles			
Sample	**sex**	**residence**	**Sample Size**
1	female	rural	180
2	female	urban	300
3	male	rural	290
4	male	urban	310

Response Profiles	
Response	**periods**
1	0
2	1
3	2

The "Response Functions and Design Matrix" table in Figure 33.2 contains the observed response functions—in this case, the mean number of periods with colds for each of the populations—and the design matrix. The first column of the design matrix contains the coefficients for the intercept parameter. The second column contains the coefficients for the sex parameter. (Note that the sum-to-zero constraint of the default full-rank parameterization PARAM=EFFECT implies that the coefficient for males is the negative of that for females; the parameter is called the *differential effect* for females.) The third column is similarly set up for residence, and the last column is for the interaction.

Figure 33.2 Observed Response Functions and Design Matrix

Response Functions and Design Matrix					
		Design Matrix			
Sample	Response Function	1	2	3	4
1	1.14444	1	1	1	1
2	1.12000	1	1	-1	-1
3	0.99310	1	-1	1	-1
4	0.93871	1	-1	-1	1

The model-fitting results are displayed in the "Analysis of Variance" table (Figure 33.3), which is similar to an ANOVA table. The effects from the right side of the MODEL statement are listed in the Source column.

Figure 33.3 ANOVA Table for the Saturated Model

Analysis of Variance			
Source	DF	Chi-Square	Pr > ChiSq
Intercept	1	1841.13	<.0001
sex	1	11.57	0.0007
residence	1	0.65	0.4202
sex*residence	1	0.09	0.7594
Residual	0	.	.

You can see in Figure 33.3 that the interaction effect is nonsignificant, so the data are reanalyzed using a main-effects model. Since PROC CATMOD is an interactive procedure, you can analyze the main-effects model by simply submitting the new MODEL statement as follows. The resulting tables are displayed in Figure 33.4 and Figure 33.5.

```
   model periods = sex residence / design;
run;
```

From the ANOVA table in Figure 33.4, you can see that the goodness-of-fit chi-square statistic is 0.09 with one degree of freedom and a p-value of 0.7594; hence, the model fits the data. Note that the chi-square tests in Figure 33.4 check whether all the parameters for a given effect are zero. In this model, each effect has only one parameter and therefore only one degree of freedom.

Figure 33.4 Main-Effects Model

The CATMOD Procedure

Data Summary			
Response	periods	Response Levels	3
Weight Variable	count	Populations	4
Data Set	COLDS	Total Frequency	1080
Frequency Missing	0	Observations	12

Figure 33.4 *continued*

Population Profiles

Sample	sex	residence	Sample Size
1	female	rural	180
2	female	urban	300
3	male	rural	290
4	male	urban	310

Response Profiles

Response	periods
1	0
2	1
3	2

Response Functions and Design Matrix

		Design Matrix		
Sample	Response Function	1	2	3
1	1.14444	1	1	1
2	1.12000	1	1	-1
3	0.99310	1	-1	1
4	0.93871	1	-1	-1

Analysis of Variance

Source	DF	Chi-Square	Pr > ChiSq
Intercept	1	1882.77	<.0001
sex	1	12.08	0.0005
residence	1	0.76	0.3839
Residual	1	0.09	0.7594

The "Analysis of Weighted Least Squares Estimates" table in Figure 33.5 lists the parameters and their estimates for the model, as well as the standard errors, Wald statistics, and p-values. These chi-square tests are one-degree-of-freedom tests that the individual parameter is equal to zero. They are equal to the tests shown in Figure 33.4 since each effect is composed of exactly one parameter.

Figure 33.5 Parameter Estimates for the Main-Effects Model

Analysis of Weighted Least Squares Estimates

Parameter		Estimate	Standard Error	Chi-Square	Pr > ChiSq
Intercept		1.0501	0.0242	1882.77	<.0001
sex	**female**	0.0842	0.0242	12.08	0.0005
residence	**rural**	0.0210	0.0241	0.76	0.3839

You can compute the mean number of periods with colds for the first population (Sample 1, females in rural residences) from Table 33.2 as follows:

$$\text{mean colds} = 0 \times \frac{45}{180} + 1 \times \frac{64}{180} + 2 \times \frac{71}{180} = 1.1444$$

This is the same value reported in the Response Function column for Sample 1 in the "Response Functions and Design Matrix" table displayed in Figure 33.4.

PROC CATMOD is fitting a model to the mean number of colds in each population as follows:

$$\begin{bmatrix} \text{Expected number of colds for rural females} \\ \text{urban females} \\ \text{rural males} \\ \text{urban males} \end{bmatrix} = \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & -1 \\ 1 & -1 & 1 \\ 1 & -1 & -1 \end{bmatrix} \begin{bmatrix} \beta_0 \\ \beta_1 \\ \beta_2 \end{bmatrix}$$

where the design matrix is the same one displayed in Figure 33.4, β_0 is the mean number of colds averaged over all the populations, β_1 is the differential effect for females, and β_2 is the differential effect for rural residences. The parameter estimates are shown in Figure 33.5; the expected number of periods with colds for rural females from this model is computed as

$$1 \times 1.0501 + 1 \times 0.0842 + 1 \times 0.0210 = 1.1553$$

and the expected number for rural males from this model is

$$1 \times 1.0501 - 1 \times 0.0842 + 1 \times 0.0210 = 0.9869$$

Notice also, in Figure 33.5, that the differential effect for residence is nonsignificant ($p = 0.3839$). If you continue the analysis by fitting a single-effect model (sex), you need to include a POPULATION statement to maintain the same underlying contingency table:

```
   population sex residence;
   model periods = sex;
run;
```

Generalized Logits Model

Over the course of one school year, third-graders from three different schools are exposed to three different styles of mathematics instruction: a self-paced computer-learning style, a team approach, and a traditional class approach. The students are asked which style they prefer, and their responses, classified by the type of program they are in (a regular school day versus a regular school day supplemented with an afternoon school program), are displayed in Table 33.3. The data set is from Stokes, Davis, and Koch (2000), and it is also analyzed in the section "Example 76.4: Nominal Response Data: Generalized Logits Model" on page 5926 in Chapter 76, "The LOGISTIC Procedure."

Table 33.3 School Program Data

		Learning Style Preference		
School	**Program**	**Self**	**Team**	**Class**
1	Regular	10	17	26
1	Afternoon	5	12	50
2	Regular	21	17	26
2	Afternoon	16	12	36
3	Regular	15	15	16
3	Afternoon	12	12	20

The levels of the response variable (self, team, and class) have no essential ordering, so a logistic regression is performed on the generalized logits. The model to be fit is

$$\log\left(\frac{\pi_{hij}}{\pi_{hir}}\right) = \alpha_j + \mathbf{x}'_{hi}\boldsymbol{\beta}_j$$

where π_{hij} is the probability that a student in school h and program i prefers teaching style j, $j \neq r$, and style r is the class style. There are separate sets of intercept parameters α_j and regression parameters β_j for each logit, and the matrix $\mathbf{x}_{hi}$ is the set of explanatory variables for the hi population. Thus, two logits are modeled for each school and program combination (population): the logit comparing self to class and the logit comparing team to class.

The following statements create the data set school and request the analysis. Generalized logits are the default response functions, and maximum likelihood estimation is the default method for analyzing generalized logits, so only the WEIGHT and MODEL statements are required. The option ORDER=DATA means that the response variable levels are ordered as they exist in the data set: self, team, and class; the logits are formed by comparing self to class and by comparing team to class. The results of this analysis are shown in Figure 33.6 and Figure 33.7.

```
data school;
   length Program $ 9;
   input School Program $ Style $ Count @@;
   datalines;
1 regular   self 10  1 regular   team 17  1 regular   class 26
1 afternoon self  5  1 afternoon team 12  1 afternoon class 50
2 regular   self 21  2 regular   team 17  2 regular   class 26
2 afternoon self 16  2 afternoon team 12  2 afternoon class 36
3 regular   self 15  3 regular   team 15  3 regular   class 16
3 afternoon self 12  3 afternoon team 12  3 afternoon class 20
;

proc catmod order=data;
   weight Count;
   model Style=School Program School*Program;
run;
```

A summary of the data set is displayed in Figure 33.6; the variable levels that form the three responses and six populations are listed in the "Response Profiles" and "Population Profiles" tables, respectively.

Figure 33.6 Model Information and Profile Tables

The CATMOD Procedure

Data Summary			
Response	Style	**Response Levels**	3
Weight Variable	Count	**Populations**	6
Data Set	SCHOOL	**Total Frequency**	338
Frequency Missing	0	**Observations**	18

Figure 33.6 *continued*

Population Profiles			
Sample	School	Program	Sample Size
1	1	regular	53
2	1	afternoon	67
3	2	regular	64
4	2	afternoon	64
5	3	regular	46
6	3	afternoon	44

Response Profiles	
Response	Style
1	self
2	team
3	class

The analysis of variance table is displayed in Figure 33.7. Since this is a saturated model, there are no degrees of freedom remaining for a likelihood ratio test, and missing values are displayed in the table. The interaction effect is clearly nonsignificant, so a main-effects model is fit.

Figure 33.7 Saturated Model: ANOVA Table

Maximum Likelihood Analysis of Variance			
Source	DF	Chi-Square	Pr > ChiSq
Intercept	2	40.05	<.0001
School	4	14.55	0.0057
Program	2	10.48	0.0053
School*Program	4	1.74	0.7827
Likelihood Ratio	0	.	.

Since PROC CATMOD is an interactive procedure, you can analyze the main-effects model by simply submitting the new MODEL statement as follows:

```
    model Style=School Program;
run;
```

You can check the population and response profiles (not shown) to confirm that they are the same as those in Figure 33.6. The analysis of variance table is shown in Figure 33.8. The likelihood ratio chi-square statistic is 1.78 with a p-value of 0.7766, indicating a good fit; the Wald chi-square tests for the school and program effects are also significant. Since School has three levels, two parameters are estimated for each of the two logits they modeled, for a total of four degrees of freedom. Since Program has two levels, one parameter is estimated for each of the two logits, for a total of two degrees of freedom.

Figure 33.8 Main-Effects Model: ANOVA Table

Maximum Likelihood Analysis of Variance			
Source	**DF**	**Chi-Square**	**Pr > ChiSq**
Intercept	2	39.88	<.0001
School	4	14.84	0.0050
Program	2	10.92	0.0043
Likelihood Ratio	4	1.78	0.7766

The parameter estimates and tests for individual parameters are displayed in Figure 33.9. The order of the parameters corresponds to the order of the population and response variables as shown in the profile tables (see Figure 33.6), with the levels of the response variables varying most rapidly. The first response function is the logit that compares self to class, and the corresponding parameters have Function Number=1. The second logit (Function Number=2) compares team to class. The School=1 parameters are the differential effects versus School=3 for their respective logits, and the School=2 parameters are likewise differential effects versus School=3. The Program parameters are the differential effects of 'regular' versus 'afternoon' for the two response functions.

Figure 33.9 Parameter Estimates

Analysis of Maximum Likelihood Estimates						
Parameter		**Function Number**	**Estimate**	**Standard Error**	**Chi-Square**	**Pr > ChiSq**
Intercept		**1**	-0.7979	0.1465	29.65	<.0001
		2	-0.6589	0.1367	23.23	<.0001
School	**1**	**1**	-0.7992	0.2198	13.22	0.0003
	1	**2**	-0.2786	0.1867	2.23	0.1356
	2	**1**	0.2836	0.1899	2.23	0.1352
	2	**2**	-0.0985	0.1892	0.27	0.6028
Program	**regular**	**1**	0.3737	0.1410	7.03	0.0080
	regular	**2**	0.3713	0.1353	7.53	0.0061

The Program variable has nearly the same effect on both logits, while School=1 has the largest effect of the schools.

Syntax: CATMOD Procedure

The following statements are available in the CATMOD procedure:

PROC CATMOD < *options* > **;**
DIRECT < *variables* > **;**
MODEL *response-effect* **=** *design-effects* < / *options* > **;**
CONTRAST '*label*' *row-description* < **,** . . . **,** *row-description* > < / *options* > **;**
BY *variables* **;**
FACTORS *factor-description* < **,** . . . **,** *factor-description* > < / *options* > **;**
LOGLIN *effects*< / *option* > **;**
POPULATION *variables* **;**
REPEATED *factor-description* < **,** . . . **,** *factor-description* > < / *options* > **;**
RESPONSE < *function* > < / *options* > **;**
RESTRICT *parameter***=***value* < . . . *parameter***=***value* > **;**
WEIGHT *variable* **;**

You can use all of the statements in PROC CATMOD interactively. The first RUN statement executes all of the previous statements. Any subsequent RUN statement executes only those statements that appear between the previous RUN statement and the current one. However, if you specify a BY statement, interactive processing is disabled. That is, all statements through the following RUN statement are processed for each BY group in the data set, but no additional statements are accepted by the procedure.

If more than one CONTRAST statement appears between two RUN statements, all the CONTRAST statements are processed. If more than one RESPONSE statement appears between two RUN statements, then analyses associated with each RESPONSE statement are produced. For all other statements, there can be only one occurrence of the statement between any two RUN statements. For example, if there are two LOGLIN statements between two RUN statements, the first LOGLIN statement is ignored.

The PROC CATMOD and MODEL statements are required. If specified, the DIRECT statement must precede the MODEL statement. As a result, if you use the DIRECT statement interactively, you need to specify a MODEL statement in the same RUN group. See the section "DIRECT Statement" on page 2140 for an example.

The CONTRAST statements, if any, must follow the MODEL statement.

You can specify only one of the LOGLIN, REPEATED, and FACTORS statements between any two RUN statements, because they all specify the same information: how to partition the variation among the response functions within a population.

A QUIT statement executes any statements that have not been processed and then ends the CATMOD procedure run.

The purpose of each statement, other than the PROC CATMOD statement, is summarized in the following list:

BY	determines groups in which data are to be processed separately.
CONTRAST	specifies a hypothesis to test.
DIRECT	specifies independent variables that are to be treated quantitatively (like continuous variables) rather than qualitatively (like classification or discrete variables). These variables

also help to determine the rows of the contingency table and distinguish response functions in one population from those in other populations.

FACTORS specifies (1) the factors that distinguish response functions from others in the same population and (2) model effects, based on these factors, which help to determine the design matrix.

LOGLIN specifies log-linear model effects.

MODEL specifies (1) dependent variables, which determine the columns of the contingency table, (2) independent variables, which distinguish response functions in one population from those in other populations, and (3) model effects, which determine the design matrix and the way in which total variation among the response functions is partitioned.

POPULATION specifies variables that determine the rows of the contingency table and distinguish response functions in one population from those in other populations.

REPEATED specifies (1) the repeated measurement factors that distinguish response functions from others in the same population and (2) model effects, based on these factors, which help to determine the design matrix.

RESPONSE determines the response functions that are to be modeled.

RESTRICT restricts values of parameters to the values you specify.

WEIGHT specifies a variable containing frequency counts.

PROC CATMOD Statement

PROC CATMOD < *options* > **;**

The PROC CATMOD statement invokes the CATMOD procedure. Table 33.4 summarizes the *options* available in the PROC CATMOD statement.

Table 33.4 PROC CATMOD Statement Options

Option	Description
DATA=	Names the input SAS data set
NAMELEN=	Specifies the length of effect names
NOPRINT	Suppresses the normal display of results
ORDER=	Specifies the sort order for the levels of classification variables

You can specify the following *options*.

DATA=*SAS-data-set*

names the SAS data set containing the data to be analyzed. By default, the CATMOD procedure uses the most recently created SAS data set. For details, see the section "Input Data Sets" on page 2163.

NAMELEN=*n*
specifies the length of effect names in tables and output data sets to be *n* characters long, where *n* is a value between 24 and 200. The default length is 24 characters.

NOPRINT
suppresses the normal display of results. The NOPRINT option is useful when you only want to create output data sets with the OUT= or OUTEST= option in the RESPONSE statement. A NOPRINT option is also available in the MODEL statement. Note that this option temporarily disables the Output Delivery System (ODS); see Chapter 20, "Using the Output Delivery System," for more information.

ORDER=DATA | FORMATTED | FREQ | INTERNAL
specifies the sort order for the levels of classification variables. This affects the ordering of the populations, responses, and parameters, as well as the definitions of the parameters. The default, ORDER=INTERNAL, orders the variable levels by their unformatted values (for example, numeric order or alphabetical order).

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

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

By default, ORDER=INTERNAL. For ORDER=FORMATTED and ORDER=INTERNAL, the sort order is machine dependent. See the section "Ordering of Populations and Responses" on page 2166 for more information and examples. 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*.

BY Statement

BY *variables* **;**

You can specify a BY statement in PROC CATMOD 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 in the CATMOD 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.

When you specify a BY statement with PROC CATMOD, no further interactive processing is possible. In other words, once the BY statement appears, all statements up to the associated RUN statement are executed for each BY group in the data set. After the RUN statement, no further statements are accepted by the procedure.

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*.

CONTRAST Statement

CONTRAST '*label*' *row-description* <, . . . , *row-description*> </ *options*> ;

where a *row-description* is defined as follows:

<@n> effect values <. . . <@n> effect values >

The CONTRAST statement constructs and tests linear functions of the parameters in the MODEL statement or effects listed in the LOGLIN statement. Each set of effects (separated by commas) specifies one row or set of rows of the matrix $\mathbf{C}$ that PROC CATMOD uses to test the hypothesis $\mathbf{C}\boldsymbol{\beta} = \mathbf{0}$.

CONTRAST statements must be preceded by the MODEL statement, and by the LOGLIN statement, if one is used. You can specify the following terms in the CONTRAST statement.

'*label*' specifies up to 256 characters of identifying information displayed with the test. The '*label*' is required.

effect is one of the effects specified in the MODEL or LOGLIN statement, INTERCEPT (for the intercept parameter), or ALL_PARMS (for the complete set of parameters).

The ALL_PARMS option is regarded as an effect with the same number of parameters as the number of columns in the design matrix. This is particularly useful when the design matrix is input directly, as in the following example:

```
model y=(1 0 0 0,
         1 0 1 0,
         1 1 0 0,
         1 1 1 1);
contrast 'Main Effect of B' all_parms 0 1 0 0;
contrast 'Main Effect of C' all_parms 0 0 1 0;
contrast 'B*C Interaction ' all_parms 0 0 0 1;
```

values are numbers that form the coefficients of the parameters associated with the given effect. If there are fewer values than parameters for an effect, the remaining coefficients become zero. For example, if you specify two values and the effect actually has five parameters, the final three are set to zero.

@n points to the parameters in the *n*th set when the model has a separate set of parameters for each of the response functions. The *@n* notation is seldom needed. It enables you to test the variation among response functions in the same population. However, it is usually easier to model and test such variation by using the _RESPONSE_ effect in the MODEL statement or by using the ALL_PARMS designation. Usually, contrasts are performed with respect to all of the response functions, and this is what the CONTRAST statement does by default (in this case, do not use the *@n* notation).

For example, if there are three response functions per population, then the following contrast results in a three-degree-of-freedom test comparing the first two levels of A simultaneously on the three response functions.

```
contrast 'Level 1 vs. Level 2' A 1 -1 0;
```

If, however, you want to specify a contrast with respect to the parameters in the *n*th set only, then use a single *@n* in a *row-description*. For example, the following statement tests that the first parameter of A and the first parameter of B are zero in the third response function:

```
contrast 'A=0, B=0, Function 3'  @3  A 1  B 1;
```

To specify a contrast with respect to parameters in two or more different sets of effects, use *@n* with each effect. For example:

```
contrast 'Average over Functions' @1 A 1 0 -1
                                  @2 A 1 1 -2;
```

When the model does not have a separate set of parameters for each of the response functions, the *@n* notation is invalid. This type of model is called AVERAGED. For details, see the description of the AVERAGED option and the section "Generation of the Design Matrix" on page 2177.

You can specify the following *options* in the CONTRAST statement after a slash.

ALPHA=*value*
specifies the significance level of the confidence interval for each contrast when the ESTIMATE= option is specified. The default is ALPHA=0.05, resulting in a 95% confidence interval for each contrast.

ESTIMATE=*keyword*
EST=*keyword*
requests that each individual contrast (that is, each row, $c_i\boldsymbol{\beta}$, of $\mathbf{C}\boldsymbol{\beta}$) or exponentiated contrast ($\exp(c_i\boldsymbol{\beta})$) be estimated and tested. PROC CATMOD displays the point estimate, its standard error, a Wald confidence interval, and a Wald chi-square test for each contrast. The significance level of the confidence interval is controlled by the ALPHA= option.

You can estimate the contrast or the exponentiated contrast, or both, by specifying one of the following *keywords*:

PARM specifies that the contrast itself be estimated.

EXP specifies that the exponentiated contrast be estimated.

BOTH specifies that both the contrast and the exponentiated contrast be estimated.

Specifying Contrasts

PROC CATMOD is parameterized differently than PROC GLM, so you must be careful not to use the same contrasts that you would with PROC GLM. Since PROC CATMOD uses full-rank parameterizations, all estimable parameters are directly estimable without involving other parameters.

For example, suppose a classification variable A has four levels and uses the default parameterization (PARAM=EFFECT). Then there are four parameters ($\alpha_1, \alpha_2, \alpha_3, \alpha_4$), of which PROC CATMOD uses only the first three. The fourth parameter is related to the others by the equation

$$\alpha_4 = -\alpha_1 - \alpha_2 - \alpha_3$$

To test the first versus the fourth level of A, you would test $\alpha_1 = \alpha_4$, which is

$$\alpha_1 = -\alpha_1 - \alpha_2 - \alpha_3$$

or, equivalently,

$$2\alpha_1 + \alpha_2 + \alpha_3 = 0$$

Therefore, you would use the following CONTRAST statement:

```
contrast '1 vs. 4' A 2 1 1;
```

To contrast the third level with the average of the first two levels, you would test

$$\frac{\alpha_1 + \alpha_2}{2} = \alpha_3$$

or, equivalently,

$$\alpha_1 + \alpha_2 - 2\alpha_3 = 0$$

Therefore, you would use the following CONTRAST statement:

```
contrast '1&2 vs. 3' A 1 1 -2;
```

Other CONTRAST statements are constructed similarly. For example:

```
contrast '1 vs. 2    '  A  1 -1  0;
contrast '1&2 vs. 4  '  A  3  3  2;
contrast '1&2 vs. 3&4'  A  2  2  0;
contrast 'Main Effect'  A  1  0  0,
                        A  0  1  0,
                        A  0  0  1;
```

The actual form of the **C** matrix depends on the effects in the model. The remaining examples in this section assume a single response function for each population.

Recall that the statements to test the first versus the fourth level of A are as follows:

```
proc catmod;
   model y=a;
   contrast '1 vs. 4' A 2 1 1;
run;
```

Since the first parameter corresponds to the intercept, the **C** matrix for the preceding statements is

$$\mathbf{C} = [\,0\;2\;1\;1\,]$$

But suppose you have a variable B with three levels and you use the following statements:

```
proc catmod;
   model y=b a;
   contrast '1 vs. 4' A 2 1 1;
run;
```

Then the CONTRAST statement produces the **C** matrix

$$\mathbf{C} = [\,0\,0\,0\,2\,1\,1\,]$$

since the first parameter corresponds to the intercept and the next two correspond to the B main effect.

You can also use the CONTRAST statement to test the joint effect of two or more effects in the MODEL statement. For example, the joint effect of A and B in the previous model has five degrees of freedom and is obtained by specifying the following:

```
contrast 'Joint Effect of A&B' A 1 0 0,
                               A 0 1 0,
                               A 0 0 1,
                               B 1 0,
                               B 0 1;
```

The ordering of variable levels is determined by the ORDER= option in the PROC CATMOD statement. Whenever you specify a contrast that depends on the order of the variable levels, you should verify the order from the "Population Profiles" table, the "Response Profiles" table, or the "One-Way Frequencies" table.

DIRECT Statement

DIRECT *variables* **;**

The DIRECT statement lists numeric independent variables to be treated in a quantitative, rather than qualitative, way. The DIRECT statement is useful for logistic regression, which is described in the section "Logistic Regression" on page 2171. For limitations of models involving continuous variables, see the section "Continuous Variables" on page 2172.

CAUTION: If a DIRECT variable is formatted, then the unformatted (internal) values are used in the analysis and the formatted values are displayed. If you use a format to group the internal values into one formatted value, then the first internal value is used in the analysis. If specified, the DIRECT statement must precede the MODEL statement. For example:

```
proc catmod;
   direct X;
   model Y=X;
run;
```

Suppose X has five levels. Then the main effect X adds only one column to the design matrix rather than four. The values inserted into the design matrix are the actual values of X.

You can interactively change the variables declared as DIRECT variables by using the statement without listing any variables. The following statements are valid:

```
proc catmod;
   direct X;
   model Y=X;
   weight wt;
run;
   direct;
   model Y=X;
run;
```

The first MODEL statement uses the actual values of X, and the second MODEL statement uses the four variables created when PROC CATMOD generates the design matrix. Note that the preceding statements can be run without a WEIGHT statement if the input data are raw data rather than cell counts.

For more details, see the discussions of main and direct effects in the section "Generation of the Design Matrix" on page 2177.

FACTORS Statement

FACTORS *factor-description* < **,** *. . .* **,** *factor-description* > < */ options* > **;**

where a *factor-description* is defined as follows:

factor-name < *$* > < *levels* >

and *factor-description*s are separated from each other by a comma. The $ is required for character-valued factors. The value of *levels* provides the number of levels of the factor identified by a given *factor-name*. For only one factor, *levels* is optional; for two or more factors, it is required.

The FACTORS statement identifies factors that distinguish response functions from others in the same population. It also specifies how those factors are incorporated into the model. You can use the FACTORS statement whenever there is more than one response function per population and the keyword _RESPONSE_ is specified in the MODEL statement. You can specify the name, type, and number of levels of each factor and the identification of each level.

The FACTORS statement is most useful when the response functions and their covariance matrix are read directly from the input data set. In this case, PROC CATMOD reads the response functions as though they are from one population (this poses no problem in the multiple-population case because the appropriately constructed covariance matrix is also read directly). Thus, you can use the FACTORS statement to partition the variation among the response functions into appropriate sources, even when the functions actually represent separate populations.

The format of the FACTORS statement is identical to that of the REPEATED statement. In fact, repeated measurement factors are simply special cases of factors in which some of the response functions correspond to multiple dependent variables that are measurements on the same experimental (or sampling) units.

You cannot specify the FACTORS statement for an analysis that also contains the REPEATED or LOGLIN statement since all of them specify the same information: how to partition the variation among the response functions within a population.

You can specify the following terms in the FACTORS statement:

factor-name names a factor that corresponds to two or more response functions. This name must be a valid SAS variable name, and it should not be the same as the name of a variable that already exists in the data set being analyzed.

$ indicates that the factor is character-valued. If the $ is omitted, then the CATMOD procedure assumes that the factor is numeric. The type of the factor is relevant only when you use the PROFILE= option or when the _RESPONSE_= option (described later in this section) specifies nested-by-value effects.

levels specifies the number of levels of the corresponding factor. If there is only one such factor, and the number is omitted, then PROC CATMOD assumes that the number of levels is equal to the number of response functions per population (q). Unless you specify the PROFILE= option, the number q must either be equal to or be a multiple of the product of the number of levels of all the factors.

You can specify the following *options* in the FACTORS statement after a slash.

PROFILE=(*matrix*)
specifies the values assumed by the factors for each response function. There should be one column for each factor, and the values in a given column (character or numeric) should match the type of the corresponding factor. Character values are restricted to 16 characters or less. If there are q response functions per population, then the matrix must have i rows, where q must either be equal to or be a multiple of i. Adjacent rows of the matrix should be separated by a comma.

The values in the PROFILE matrix are useful for specifying models in those situations where the study design is not a full factorial with respect to the factors. They can also be used to specify nested-by-value effects in the _RESPONSE_= option. If you specify character values in both places (the PROFILE= option and the _RESPONSE_= option), then the values must match with respect to whether or not they are enclosed in quotes (that is, enclosed in quotes in both places or in neither place).

For an example of using the PROFILE= option, see Example 33.10.

RESPONSE=*effects*
specifies design effects. The variables named in the effects must be *factor-names* that appear in the FACTORS statement. If the _RESPONSE_= option is omitted, then PROC CATMOD builds a full factorial _RESPONSE_ effect with respect to the factors.

TITLE=*'title'*
displays the *title* at the top of certain pages of output that correspond to the current FACTORS statement.

For an example of how the FACTORS statement is useful, consider the case where the response functions and their covariance matrix are read directly from the input data set. The TYPE=EST data set might be created in the following manner:

```
data direct(type=est);
   input b1-b4 _type_ $ _name_ $8.;
   datalines;
0.590463   0.384720   0.273269   0.136458   parms      .
0.001690   0.000911   0.000474   0.000432   cov        b1
0.000911   0.001823   0.000031   0.000102   cov        b2
0.000474   0.000031   0.001056   0.000477   cov        b3
0.000432   0.000102   0.000477   0.000396   cov        b4
;
```

Suppose the response functions correspond to four populations that represent the cross-classification of age (two groups) by sex. You can use the FACTORS statement to identify these two factors and to name the effects in the model. The statements required to fit a main-effects model to these data are as follows:

```
proc catmod data=direct;
   response read b1-b4;
   model _f_=_response_;
   factors age 2, sex 2 / _response_=age sex;
run;
```

If you want to specify some nested-by-value effects, you can change the FACTORS statement to the following:

```
factors age $ 2, sex $ 2 /
       _response_=age sex(age='under 30') sex(age='30 & over')
        profile=('under 30'   male,
                 'under 30'   female,
                 '30 & over'  male,
                 '30 & over'  female);
```

If, by design or by chance, the study contains no male subjects under 30 years of age, then there are only three response functions, and you can specify a main-effects model as follows:

```
proc catmod data=direct;
   response read b2-b4;
   model _f_=_response_;
   factors age $ 2, sex $ 2 / _response_=age sex
          profile=('under 30'   female,
                   '30 & over'  male,
                   '30 & over'  female);
run;
```

When you specify two or more factors and omit the PROFILE= option, PROC CATMOD presumes that the response functions are ordered so that the levels of the rightmost factor change most rapidly. For the preceding example, the order implied by the FACTORS statement is as follows:

Response Function	Dependent Variable	Age	Sex
1	b1	1	1
2	b2	1	2
3	b3	2	1
4	b4	2	2

For additional examples of how to use the FACTORS statement, see the section "Repeated Measures Analysis" on page 2174. All of the examples in that section are applicable, with the REPEATED statement replaced by the FACTORS statement.

LOGLIN Statement

LOGLIN *effects* < / *option* > **;**

The LOGLIN statement is used to define log-linear model effects. It can be used whenever the default response functions (generalized logits) are used.

In the LOGLIN statement, *effects* are design effects that contain dependent variables in the MODEL statement, including interaction, nested, and nested-by-value effects. You can use the bar (|) and at (@) operators as well. The following lists of effects are equivalent:

```
a b c a*b a*c b*c
```

and

```
a|b|c @2
```

When you use the LOGLIN statement, the keyword _RESPONSE_ should be specified in the MODEL statement. For further information about log-linear model analysis, see the section "Log-Linear Model Analysis" on page 2172.

You cannot specify the LOGLIN statement for an analysis that also contains the REPEATED or FACTORS statement since all of them specify the same information: how to partition the variation among the response functions within a population.

You can specify the following *option* in the LOGLIN statement after a slash.

TITLE=*'title'*
displays the *title* at the top of certain pages of output that correspond to this LOGLIN statement.

The following statements give an example of how to use the LOGLIN statement:

```
proc catmod;
   model a*b*c=_response_;
   loglin a|b|c @ 2;
run;
```

These statements yield a log-linear model analysis that contains all main effects and two-variable interactions. For more examples of log-linear model analysis, see the section "Log-Linear Model Analysis" on page 2172.

MODEL Statement

MODEL *response-effect* **=** < *design-effects* > < / *options* > **;**

PROC CATMOD requires a MODEL statement. You can specify the following in a MODEL statement:

response-effect can be either a single variable, a crossed effect with two or more variables joined by asterisks, or _F_. The _F_ specification indicates that the response functions and their estimated covariance matrix are to be read directly into the procedure (see the section "Inputting Response Functions and Covariances Directly" on page 2165 for details). The *response-effect* indicates the dependent variables that determine the response categories (the columns of the underlying contingency table).

design-effects specify potential sources of variation (such as main effects and interactions) in the model. These effects determine the number of model parameters, as well as the interpretation of such parameters. In addition, if there is no POPULATION statement, PROC CATMOD uses these variables to determine the populations (the rows of the underlying contingency table). When fitting the model, PROC CATMOD adjusts the independent effects in the model for all other independent effects in the model.

Design-effects can be any of those described in the section "Specification of Effects" on page 2167, or they can be defined by specifying the actual design matrix, enclosed in parentheses (see the section "Specifying the Design Matrix Directly" on page 2151). In addition, you can use the keyword _RESPONSE_ alone or as part of an effect. Effects cannot be nested within _RESPONSE_, so effects of the form A(_RESPONSE_) are invalid.

For more information, see the section "Log-Linear Model Analysis" on page 2172 and the section "Repeated Measures Analysis" on page 2174.

Some example MODEL statements are shown in the following table:

Example	Result
`model r=a b;`	Main effects only
`model r=a b a*b;`	Main effects with interaction
`model r=a b(a);`	Nested effect
`model r=a|b;`	Complete factorial
`model r=a b(a=1) b(a=2);`	Nested-by-value effects
`model r*s=_response_;`	Log-linear model
`model r*s=a _response_(a);`	Nested repeated measurement factor
`model _f_=_response_;`	Direct input of the response functions

The relationship between these specifications and the structure of the design matrix **X** is described in the section "Generation of the Design Matrix" on page 2177.

Table 33.5 summarizes the *options* available in the MODEL statement.

Table 33.5 MODEL Statement Options

Options	Task
Specify details of computation	
ML=	Generates the maximum likelihood estimates
GLS	Generates the weighted least squares estimates
WLS	
NOINT	Omits the intercept term from the model
PARAM=	Specifies the parameterization of classification variables
ADDCELL=	Adds a number to each cell frequency
AVERAGED	Averages the main effects across response functions
EPSILON=	Specifies the convergence criterion for maximum likelihood
MAXITER=	Specifies the number of iterations for maximum likelihood
MISSING=	Specifies how missing cells are treated
ZERO=	Specifies how zero cells are treated
Request additional computation and tables	
ALPHA=	Specifies the significance level of confidence intervals
CLPARM	Displays the Wald confidence intervals of estimates
CORRB	Displays the estimated correlation matrix of estimates
COV	Displays the covariance matrix of response functions
COVB	Displays the estimated covariance matrix of estimates
DESIGN	Displays the design and _RESPONSE_ matrix
FREQ	Displays the two-way frequency tables
ITPRINT	Displays the iterations for maximum likelihood
ONEWAY	Displays the one-way frequency tables
PRED=	Displays the predicted values
PREDICT	
PROB	Displays the probability estimates
PROFILE	Displays the population profiles
XPX	Displays the crossproducts matrix
TITLE=	Specifies the title
Suppress output	
NODESIGN	Suppresses the design matrix
NOPARM	Suppresses the parameter estimates
NOPREDVAR	Suppresses the variable levels
NOPROFILE	Suppresses the population and response profiles
NORESPONSE	Suppresses the _RESPONSE_ matrix

The following list describes these *options* in alphabetical order.

ADDCELL=*number*

adds *number* to the frequency count in each cell, where *number* is any positive number. This option has no effect on maximum likelihood analysis; it is used only for weighted least squares analysis.

ALPHA=*number*
sets the significance level for the Wald confidence intervals for parameter estimates. The value must be between 0 and 1. The default value of 0.05 results in the calculation of a 95% confidence interval. This option has no effect unless the CLPARM option is also specified.

AVERAGED
specifies that dependent variable effects can be modeled and that independent variable main effects are averaged across the response functions in a population. For further information about the effect of using (or not using) the AVERAGED option, see the section "Generation of the Design Matrix" on page 2177. Direct input of the design matrix or specification of the _RESPONSE_ keyword in the MODEL statement automatically uses an AVERAGED model type.

CLPARM
produces Wald confidence limits for the parameter estimates. The confidence coefficient can be specified with the ALPHA= option.

CORRB
displays the estimated correlation matrix of the parameter estimates.

COV
displays $\mathbf{S}_i$, which is the covariance matrix of the response functions for each population.

COVB
displays the estimated covariance matrix of the parameter estimates.

DESIGN
displays the design matrix $\mathbf{X}$ for WLS and ML analyses, and also displays the _RESPONSE_ matrix for log-linear models. For further information, see the section "Generation of the Design Matrix" on page 2177.

EPSILON=*number*
specifies the convergence criterion for the maximum likelihood estimation of the parameters. The iterative estimation process stops when the proportional change in the log likelihood is less than *number*, or after the number of iterations specified by the MAXITER= option, whichever comes first. By default, EPSILON=1E–8.

FREQ
produces the two-way frequency table for the cross-classification of populations by responses.

ITPRINT
displays parameter estimates and other information at each iteration of a maximum likelihood analysis.

MAXITER=*number*
specifies the maximum number of iterations used for the maximum likelihood estimation of the parameters. By default, MAXITER=20.

ML < = NR | IPF< (*ipf-options*) > >
computes maximum likelihood estimates (MLE) by using either a Newton-Raphson algorithm (NR) or an iterative proportional fitting algorithm (IPF).

The option ML=NR (or simply ML) is available when you use generalized logits, and also when you perform binary logistic regression with logits, cumulative logits, or adjacent category logits. For generalized logits (the default response functions), ML=NR is the default estimation method.

The option ML=IPF is available for fitting a hierarchical log-linear model with one population (no independent variables and no population variables). The use of bar notation to express the log-linear effects guarantees that the model is hierarchical (the presence of any interaction term in the model requires the presence of all its lower-order terms). If your table is *incomplete* (that is, your table has a zero or missing entry in at least one cell), then all missing cells and all cells with zero weight are treated as structural zeros by default; this behavior can be modified with the ZERO= and MISSING= options in the MODEL statement.

You can control the convergence of the two algorithms with the EPSILON= and MAXITER= options in the MODEL statement. You can select the convergence criterion for the IPF algorithm with the CONVCRIT= option.

NOTE: The RESTRICT statement is not available with the ML=IPF option.

You can specify the following *ipf-options* within parentheses after the ML=IPF option.

CONVCRIT=*keyword*
specifies the method that determines when convergence of the IPF algorithm occurs. You can specify one of the following *keywords*:

CELL termination requires the maximum absolute difference between consecutive cell estimates to be less than 0.001 (or the value of the EPSILON= option, if specified).

LOGL termination requires the relative difference between consecutive estimates of the log likelihood to be less than 1E–8 (or the value of the EPSILON= option, if specified). This is the default.

MARGIN termination requires the maximum absolute difference between consecutive margin estimates to be less than 0.001 (or the value of the EPSILON= option, if specified).

DF=*keyword*
specifies the method used to compute the degrees of freedom for the goodness-of-fit G^2 test (labeled "Likelihood Ratio" in the "Estimates" table).

For a *complete* table (a table having nonzero entries in every cell), the degrees of freedom are calculated as the number of cells in the table (n_c) minus the number of independent parameters specified in the model (n_p). For incomplete tables, these degrees of freedom can be adjusted by the number of fitted zeros (n_z, which includes the number of structural zeros) and the number of nonestimable parameters due to the zeros (n_n). If you are analyzing an incomplete table, you should verify that the degrees of freedom are correct.

You can specify one of the following *keywords*:

UNADJ computes the unadjusted degrees of freedom as $n_c - n_p$. These are the same degrees of freedom you would get if all cells in the table were positive.

ADJ computes the degrees of freedom as $(n_c - n_p) - (n_z - n_n)$ (Bishop, Fienberg, and Holland 1975), which adjusts for fitted zeros and nonestimable parameters. This is the default, and for complete tables it gives the same results as the UNADJ option.

ADJEST computes the degrees of freedom as $(n_c - n_p) - n_z$, which adjusts for fitted zeros only. This gives a lower bound on the true degrees of freedom.

PARM
computes parameter estimates, generates the "ANOVA," "Parameter Estimates," and "Predicted Values of Response Functions" tables, and includes the predicted standard errors in the "Predicted Values of Frequencies and Probabilities" tables.

When you specify the PARM option, the algorithm used to obtain the maximum likelihood parameter estimates is weighted least squares on the IPF-predicted frequencies. This algorithm can be much faster than the Newton-Raphson algorithm that is used if you specify the ML=NR option. In the resulting ANOVA table, the likelihood ratio is computed from the initial IPF fit while the degrees of freedom are generated from the WLS analysis; the DF= option can override this. Also, the initial response function, which the WLS method usually computes from the raw data, is computed from the IPF-predicted frequencies.

If there are any zero marginals in the configurations that define the model, then there are predicted cell frequencies of zero and WLS cannot be used to compute the estimates. In this case, PROC CATMOD automatically changes the algorithm from ML=IPF to ML=NR and prints a note in the log.

MISSING=*keyword*
MISS=*keyword*
specifies whether a missing cell is treated as a sampling or structural zero.

Structural zero cells are removed from the analysis since their expected values are zero, while sampling zero cells can have nonzero expected value and might be estimable. For a single population, the missing cells are treated as structural zeros by default. For multiple populations, as long as some population has a nonzero count for a given population and response profile, the missing values are treated as sampling zeros by default.

The following table displays the available *keywords* and summarizes how PROC CATMOD treats missing values for one or more populations:

MISSING=	One Population	Multiple Populations
STRUCTURAL (default)	Structural zeros	Sampling zeros
SAMP | **SAMPLING**	Sampling zeros	Sampling zeros
value	Sets missing weights and cells to *value*	Sets missing weights and cells to *value*

NODESIGN
suppresses the display of the design matrix **X** when the DESIGN option is also specified. This enables you to display only the _RESPONSE_ matrix for log-linear models.

NOINT
suppresses the intercept term in the model.

NOPARM
suppresses the display of the estimated parameters and the statistics for testing that each parameter is zero.

NOPREDVAR
suppresses the display of the variable levels in tables requested with the PRED= option and in the "Estimates" table. Population profiles are replaced with the sample number, classification variable levels are suppressed, and response profiles are replaced with a function number.

NOPRINT
suppresses the normal display of results. The NOPRINT option is useful when you only want to create output data sets with the OUT= or OUTEST= option in the RESPONSE statement. A NOPRINT option is also available in the PROC CATMOD statement. Note that this option temporarily disables the Output Delivery System (ODS); see Chapter 20, "Using the Output Delivery System," for more information.

NOPROFILE
suppresses the display of the population profiles and the response profiles.

NORESPONSE
suppresses the display of the _RESPONSE_ matrix for log-linear models when the DESIGN option is also specified. This enables you to display only the design matrix for log-linear models.

ONEWAY
produces a one-way table of frequencies for each variable used in the analysis. This table is useful in determining the order of the observed levels for each variable.

PARAM=EFFECT | REFERENCE
specifies the parameterization method for the classification variable or variables. The default is PARAM=EFFECT. Both the effect and reference parameterizations are full rank. See the section "Generation of the Design Matrix" on page 2177 for further details.

PREDICT

PRED=FREQ | PROB
displays the observed and predicted values of the response functions for each population, together with their standard errors and the residuals (observed minus predicted). In addition, if the response functions are the standard ones (generalized logits), then the PRED=FREQ option specifies the computation and display of predicted cell frequencies, while PRED=PROB (or just PREDICT) specifies the computation and display of predicted cell probabilities.

The OUT= data set always contains the predicted probabilities. If the response functions are the generalized logits, the predicted cell probabilities are output unless the option PRED=FREQ is specified, in which case the predicted cell frequencies are output.

PROB
produces the two-way table of probability estimates for the cross-classification of populations by responses. These estimates sum to one across the response categories for each population.

PROFILE
displays all of the population profiles. If you have more than 60 populations, then by default only the first 40 profiles are displayed; the PROFILE option overrides this default behavior.

TITLE='*title***'**
displays the *title* at the top of certain pages of output that correspond to this MODEL statement.

WLS
GLS
computes weighted least squares estimates. This type of estimation is also called generalized least squares estimation. For response functions other than the default (of generalized logits), WLS is the default estimation method.

XPX
displays $\mathbf{X}'\mathbf{S}^{-1}\mathbf{X}$, the crossproducts matrix for the normal equations.

ZERO=*keyword*
specifies whether a nonmissing cell with zero weight in the data set is treated as a sampling or structural zero.

Structural zero cells are removed from the analysis since their expected values are zero, while sampling zero cells have nonzero expected value and might be estimable. For a single population, the zero cells are treated as structural zeros by default; with multiple populations, as long as some population has a nonzero count for a given population and response profile, the zeros are treated as sampling zeros by default.

The following table displays the available *keywords* and summarizes how PROC CATMOD treats zeros for one or more populations:

ZERO=	One Population	Multiple Populations
STRUCTURAL (default)	Structural zeros	Sampling zeros
SAMP | **SAMPLING**	Sampling zeros	Sampling zeros
value	Sets zero weights to *value*	Sets zero weights to *value*

Specifying the Design Matrix Directly

If you specify the design matrix directly, adjacent rows of the matrix must be separated by a comma, and the matrix must have $q \times s$ rows, where s is the number of populations and q is the number of response functions per population. The first q rows correspond to the response functions for the first population, the second set of q rows corresponds to the functions for the second population, and so forth. The following is an example of using direct specification of the design matrix.

```
proc catmod;
   model R=(1 0,
            1 1,
            1 2,
            1 3);
run;
```

These statements are appropriate for the case of one population and for R with five levels (generating four response functions), so that $4 \times 1 = 4$. These statements are also appropriate for a situation with two populations and two response functions per population, giving $2 \times 2 = 4$ rows of the design matrix. (To accommodate more than one population, the POPULATION statement is needed.)

When you input the design matrix directly, you also have the option of specifying that any subsets of the parameters be tested for equality to zero. Indicate each subset by specifying the appropriate column numbers of the design matrix, followed by an equal sign and a label (24 characters or less, in quotes) that describes the subset. Adjacent subsets are separated by a comma, and the entire specification is enclosed in parentheses and placed after the design matrix. For example:

```
proc catmod;
   population Group Time;
   model R=(1  1  0  0,
            1  1  0  1,
            1  1  0  2,
            1  0  1  0,
            1  0  1  1,
            1  0  1  2,
            1 -1 -1  0,
            1 -1 -1  1,
            1 -1 -1  2) (1  ='Intercept',
                         2 3='Group main effect',
                         4  ='Linear effect of Time');
run;
```

The preceding statements are appropriate when Group and Time each have three levels and R is dichotomous. The POPULATION statement produces nine populations, and $q = 1$ (since R is dichotomous), so $q \times s = 1 \times 9 = 9$.

If you input the design matrix directly but do not specify any subsets of the parameters to be tested, then PROC CATMOD tests the effect of MODEL | MEAN, which represents the significance of the model beyond what is explained by an overall mean. For the previous example, the MODEL | MEAN effect is the same as that obtained by specifying the following at the end of the MODEL statement:

```
(2 3 4='model|mean');
```

POPULATION Statement

POPULATION *variables* **;**

The POPULATION statement specifies that populations are to be based only on cross-classifications of the specified *variables*. If you do not specify the POPULATION statement, then populations are based only on cross-classifications of the independent variables in the MODEL statement.

The POPULATION statement has two major uses:

- When you enter the design matrix directly, there are no independent variables in the MODEL statement; therefore, the POPULATION statement is the only way to produce more than one population.
- When you fit a reduced model, the POPULATION statement might be necessary if you want to form the same number of populations as there are for the saturated model.

To illustrate the first use, suppose you specify the following statements:

```
data one;
   input A $ B $ wt @@;
   datalines;
yes yes 23   yes no 31   no yes 47   no no 50
;

proc catmod;
   weight wt;
   population B;
   model A=(1 0,
            1 1);
run;
```

Since the dependent variable A has two levels, there is one response function per population. Since the variable B has two levels, there are two populations. The MODEL statement is valid since the number of rows in the design matrix (2) is the same as the total number of response functions. If the POPULATION statement is omitted, there would be only one population and one response function, and the MODEL statement would be invalid.

To illustrate the second use, suppose you specify the following statements:

```
data two;
   input A $ B $ Y wt @@;
   datalines;
yes  yes  1  23       yes  yes  2  63
yes  no   1  31       yes  no   2  70
no   yes  1  47       no   yes  2  80
no   no   1  50       no   no   2  84
;

proc catmod;
   weight wt;
   model Y=A B A*B / wls;
run;
```

These statements form four populations and produce the following design matrix and analysis of variance table:

$$\mathbf{X} = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & 1 & -1 & -1 \\ 1 & -1 & 1 & -1 \\ 1 & -1 & -1 & 1 \end{bmatrix}$$

Source	DF	Chi-Square	Pr > ChiSq
Intercept	1	48.10	<.0001
A	1	3.47	0.0625
B	1	0.25	0.6186
A*B	1	0.19	0.6638
Residual	0		

Since the B and A*B effects are nonsignificant ($p > 0.10$), fit the reduced model that contains only the A effect:

```
proc catmod;
   weight wt;
   model Y=A / wls;
run;
```

Now only two populations are formed, and the design matrix and the analysis of variance table are as follows:

$$\mathbf{X} = \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}$$

Source	DF	Chi-Square	Pr > ChiSq
Intercept	1	47.94	<.0001
A	1	3.33	0.0678
Residual	0		

However, you can form four populations by adding the POPULATION statement to the analysis:

```
proc catmod;
   weight wt;
   population A B;
   model Y=A / wls;
run;
```

The design matrix and the analysis of variance table resulting from these statements are as follows:

$$\mathbf{X} = \begin{bmatrix} 1 & 1 \\ 1 & 1 \\ 1 & -1 \\ 1 & -1 \end{bmatrix}$$

Source	DF	Chi-Square	Pr > ChiSq
Intercept	1	47.76	<.0001
A	1	3.30	0.0694
Residual	2	0.35	0.8374

The advantage of the latter analysis is that it retains four populations for the reduced model, thereby creating a built-in goodness-of-fit test: the residual chi-square. Such a test is important because the cumulative (or joint) effect of deleting two or more effects from the model can be significant, even if the individual effects are not.

The resulting differences between the two analyses are due to the fact that the latter analysis uses pure weighted least squares estimates with respect to the four populations that are actually sampled. The former analysis pools populations and therefore uses parameter estimates that can be regarded as weighted least squares estimates of maximum likelihood predicted cell frequencies. In any case, the estimation methods are asymptotically equivalent; therefore, the results are very similar. If you specify the ML option (instead of the WLS option) in the preceding MODEL statements, then the parameter estimates are identical for the two analyses.

CAUTION: If your model has different covariate profiles within any population, then the first profile is used in the analysis.

REPEATED Statement

REPEATED *factor-description* <, . . . , *factor-description*> </ *options*> ;

where a *factor-description* is defined as follows:

factor-name <$> < *levels*>

and *factor-description*s are separated from each other by a comma. The $ is required for character-valued factors. The value of *levels* provides the number of levels of the repeated measurement factor identified by a given *factor-name*. For only one repeated measurement factor, *levels* is optional; for two or more repeated measurement factors, it is required. The REPEATED statement incorporates repeated measurement factors into the model. You can use this statement whenever there is more than one dependent variable and the

keyword _RESPONSE_ is specified in the MODEL statement. If the dependent variables correspond to one or more repeated measurement factors, you can use the REPEATED statement to define _RESPONSE_ in terms of those factors. You can specify the name, type, and number of levels of each factor, as well as the identification of each level.

You cannot specify the REPEATED statement for an analysis that also contains the FACTORS or LOGLIN statement since all of them specify the same information: how to partition the variation among the response functions within a population.

You can specify the following terms in the REPEATED statement:

factor-name names a repeated measurement factor that corresponds to two or more response functions. This name must be a valid SAS variable name, and it should not be the same as the name of a variable that already exists in the data set being analyzed.

$ indicates that the factor is character-valued. If the $ is omitted, then the CATMOD procedure assumes that the factor is numeric. The type of the factor is relevant only when you use the PROFILE= option or when the _RESPONSE_= option specifies nested-by-value effects.

levels specifies the number of levels of the corresponding repeated measurement factor. If there is only one such factor and the number is omitted, then PROC CATMOD assumes that the number of levels is equal to the number of response functions per population (q). Unless you specify the PROFILE= option, the number q must either be equal to or be a multiple of the product of the number of levels of all the factors.

You can specify the following *options* in the REPEATED statement after a slash.

PROFILE=(*matrix***)**
specifies the values assumed by the factors for each response function. There should be one column for each factor, and the values in a given column should match the type (character or numeric) of the corresponding factor. Character values are restricted to 16 characters or less. If there are q response functions per population, then the matrix must have i rows, where q must either be equal to or be a multiple of i. Adjacent rows of the matrix should be separated by a comma.

The values in the PROFILE matrix are useful for specifying models in those situations where the study design is not a full factorial with respect to the factors. They can also be used to specify nested-with-value effects in the _RESPONSE_= option. If you specify character values in both the PROFILE= option and the _RESPONSE_= option, then the values must match with respect to whether or not they are enclosed in quotes (that is, they must be enclosed in quotes in both places or in neither place).

RESPONSE=*effects*
specifies design effects. The variables named in the effects must be *factor-names* that appear in the REPEATED statement. If the _RESPONSE_= option is omitted, then PROC CATMOD builds a full factorial _RESPONSE_ effect with respect to the repeated measurement factors. For example, the following two statements are equivalent in that they produce the same parameter estimates:

```
repeated Time 2, Treatment 2;
repeated Time 2, Treatment 2 / _response_=Time|Treatment;
```

However, the second statement produces tests of the Time, Treatment, and Time*Treatment effects in the "Analysis of Variance" table, whereas the first statement produces a single test for the combined effects in _RESPONSE_.

TITLE=*'title'*
displays the *title* at the top of certain pages of output that correspond to this REPEATED statement.

For further information and numerous examples of the REPEATED statement, see the section "Repeated Measures Analysis" on page 2174.

RESPONSE Statement

RESPONSE < *function* > < / *options* > **;**

The RESPONSE statement specifies functions of the response probabilities. The procedure models these response functions as linear combinations of the parameters.

By default, PROC CATMOD uses the standard response functions (generalized logits, which are explained in detail in the section "Understanding the Standard Response Functions" on page 2162). With these standard response functions, the default estimation method is maximum likelihood, but you can use the WLS option in the MODEL statement to request weighted least squares estimation. With other response functions (specified in the RESPONSE statement), the default (and only) estimation method is weighted least squares.

You can specify more than one RESPONSE statement, in which case each RESPONSE statement produces a separate analysis. If the computed response functions for any population are linearly dependent (yielding a singular covariance matrix), then PROC CATMOD displays an error message and stops processing. See the section "Cautions" on page 2186 for methods of dealing with this.

The *function* specification can be any of the items in the following list. For an example of response functions generated and formulas for q (the number of response functions), see the section "More on Response Functions" on page 2158.

Table 33.6 summarizes the *options* available in the RESPONSE statement.

Table 33.6 RESPONSE Statement Options

Option	Description
ALOGIT	Specifies response functions as adjacent-category logits
CLOGIT	Specifies that the response functions are cumulative logits
JOINT	Specifies that the response functions are the joint response probabilities
LOGIT	Specifies that the response functions are generalized logits
MARGINAL	Specifies that the response functions are marginal probabilities
MEAN	Specifies that the response functions are the means dependent variables
READ	Directly reads the response functions and their covariance matrix from the input data set
OUT=	Produces a SAS data set that contains predicted values, standard errors and residuals
OUTEST=	Produces a SAS data set that contains the estimated parameter vector and its estimated covariance matrix
TITLE=	Displays the *title*

ALOGIT

ALOGITS

specifies response functions as adjacent-category logits of the marginal probabilities for each of the dependent variables. For each dependent variable, the response functions are a set of linearly independent adjacent-category logits, obtained by taking the logarithms of the ratios of two probabilities. The denominator of the kth ratio is the marginal probability corresponding to the kth level of the variable, and the numerator is the marginal probability corresponding to the $(k + 1)$ level. If a dependent variable has two levels, then the adjacent-category logit is the negative of the generalized logit.

CLOGIT

CLOGITS

specifies that the response functions are cumulative logits of the marginal probabilities for each of the dependent variables. For each dependent variable, the response functions are a set of linearly independent cumulative logits, obtained by taking the logarithms of the ratios of two probabilities. The denominator of the kth ratio is the cumulative probability, c_k, corresponding to the kth level of the variable, and the numerator is $1 - c_k$ (Agresti 1984, 113–114). If a dependent variable has two levels, then PROC CATMOD computes its cumulative logit as the negative of its generalized logit. You should use cumulative logits only when the dependent variables are ordinally scaled.

JOINT

specifies that the response functions are the joint response probabilities. A linearly independent set is created by deleting the last response probability. For the case of one dependent variable, the JOINT and MARGINALS specifications are equivalent.

LOGIT

LOGITS

specifies that the response functions are generalized logits of the marginal probabilities for each of the dependent variables. For each dependent variable, the response functions are a set of linearly independent generalized logits, obtained by taking the logarithms of the ratios of two probabilities. The denominator of each ratio is the marginal probability corresponding to the last observed level of the variable, and the numerators are the marginal probabilities corresponding to each of the other levels. If there is one dependent variable, then specifying LOGIT is equivalent to using the standard response functions.

MARGINAL

MARGINALS

specifies that the response functions are marginal probabilities for each of the dependent variables in the MODEL statement. For each dependent variable, the response functions are a set of linearly independent marginals, obtained by deleting the marginal probability corresponding to the last level.

MEAN

MEANS

specifies that the response functions are the means of the dependent variables in the MODEL statement. This specification requires that all of the dependent variables be numeric.

READ *variables*
specifies that the response functions and their covariance matrix are to be read directly from the input data set with one response function for each variable named. See the section "Inputting Response Functions and Covariances Directly" on page 2165 for more information.

transformation
specifies response functions that can be expressed by using successive applications of the four operations: **LOG**, **EXP**, $*$ matrix literal, or $+$ matrix literal. The operations are described in detail in the section "Using a Transformation to Specify Response Functions" on page 2160.

You can specify the following *options* in the RESPONSE statement after a slash.

OUT=*SAS-data-set*
produces a SAS data set that contains, for each population, the observed and predicted values of the response functions, their standard errors, and the residuals. Moreover, if you use the standard response functions, the data set also includes observed and predicted values of the cell frequencies or the cell probabilities. For further information, see the section "Output Data Sets" on page 2169.

OUTEST=*SAS-data-set*
produces a SAS data set that contains the estimated parameter vector and its estimated covariance matrix. For further information, see the section "Output Data Sets" on page 2169.

TITLE=*'title'*
displays the *title* at the top of certain pages of output that correspond to this RESPONSE statement.

More on Response Functions

Suppose the dependent variable A has three levels and is the only *response-effect* in the MODEL statement. The following table shows the proportions upon which the response functions are defined:

Value of A:	1	2	3
Proportions:	p_1	p_2	p_3

Note that $\sum_j p_j = 1$. The following table shows the response functions generated for each population:

Function Specification	**Value of q**	**Response Function**
none*	2	$\ln\left(\frac{p_1}{p_3}\right),\ \ln\left(\frac{p_2}{p_3}\right)$
ALOGITS	2	$\ln\left(\frac{p_2}{p_1}\right),\ \ln\left(\frac{p_3}{p_2}\right)$
CLOGITS	2	$\ln\left(\frac{1-p_1}{p_1}\right),\ \ln\left(\frac{1-(p_1+p_2)}{p_1+p_2}\right)$
JOINT	2	$p_1,\ p_2$
LOGITS	2	$\ln\left(\frac{p_1}{p_3}\right),\ \ln\left(\frac{p_2}{p_3}\right)$
MARGINAL	2	$p_1,\ p_2$
MEAN	1	$1p_1 + 2p_2 + 3p_3$

*Without a function specification, the default response functions are generalized logits.

Now, suppose the dependent variables A and B each have three levels (valued 1, 2, and 3 each) and the *response-effect* in the MODEL statement is A*B. The following table shows the proportions upon which the response functions are defined:

Value of A:	1	1	1	2	2	2	3	3	3
Value of B:	1	2	3	1	2	3	1	2	3
Proportions:	p_1	p_2	p_3	p_4	p_5	p_6	p_7	p_8	p_9

The marginal totals for the preceding table are defined as follows:

$$\begin{aligned} p_{1\cdot} &= p_1 + p_2 + p_3 & \qquad p_{\cdot 1} &= p_1 + p_4 + p_7 \\ p_{2\cdot} &= p_4 + p_5 + p_6 & \qquad p_{\cdot 2} &= p_2 + p_5 + p_8 \\ p_{3\cdot} &= p_7 + p_8 + p_9 & \qquad p_{\cdot 3} &= p_3 + p_6 + p_9 \end{aligned}$$

where $\sum_j p_j = 1$. The following table shows the response functions generated for each population:

Function Specification	**Value of q**	**Response Function**
none*	8	$\ln\left(\frac{p_1}{p_9}\right),\ \ln\left(\frac{p_2}{p_9}\right),\ \ln\left(\frac{p_3}{p_9}\right), \ldots,\ \ln\left(\frac{p_8}{p_9}\right)$
ALOGITS	4	$\ln\left(\frac{p_{2\cdot}}{p_{1\cdot}}\right),\ \ln\left(\frac{p_{3\cdot}}{p_{2\cdot}}\right),\ \ln\left(\frac{p_{\cdot 2}}{p_{\cdot 1}}\right),\ \ln\left(\frac{p_{\cdot 3}}{p_{\cdot 2}}\right)$
CLOGITS	4	$\ln\left(\frac{1-p_{1\cdot}}{p_{1\cdot}}\right),\ \ln\left(\frac{1-(p_{1\cdot}+p_{2\cdot})}{p_{1\cdot}+p_{2\cdot}}\right),\ \ln\left(\frac{1-p_{\cdot 1}}{p_{\cdot 1}}\right),\ \ln\left(\frac{1-(p_{\cdot 1}+p_{\cdot 2})}{p_{\cdot 1}+p_{\cdot 2}}\right)$
JOINT	8	$p_1,\ p_2,\ p_3,\ p_4,\ p_5,\ p_6,\ p_7,\ p_8$
LOGITS	4	$\ln\left(\frac{p_{1\cdot}}{p_{3\cdot}}\right),\ \ln\left(\frac{p_{2\cdot}}{p_{3\cdot}}\right),\ \ln\left(\frac{p_{\cdot 1}}{p_{\cdot 3}}\right),\ \ln\left(\frac{p_{\cdot 2}}{p_{\cdot 3}}\right)$
MARGINAL	4	$p_{1\cdot},\ p_{2\cdot},\ p_{\cdot 1},\ p_{\cdot 2}$
MEAN	2	$1p_{1\cdot} + 2p_{2\cdot} + 3p_{3\cdot},\ 1p_{\cdot 1} + 2p_{\cdot 2} + 3p_{\cdot 3}$

* Without a function specification, the default response functions are generalized logits.

The READ and *transformation* function specifications are not shown in the preceding table. For these two situations, there is not a general response function; the response functions that are generated depend on what you specify.

Another important aspect of the function specification is the number of response functions generated per population, q. Let m_i represent the number of levels for the ith dependent variable in the MODEL statement, and let d represent the number of dependent variables in the MODEL statement. Then, if the function specification is ALOGITS, CLOGITS, LOGITS, or MARGINALS, the number of response functions is

$$q = \sum_{i=1}^{d} (m_i - 1)$$

If the function specification is JOINT or the default (generalized logits), the number of response functions per population is

$$q = r - 1$$

where r is the number of response profiles. If every possible cross-classification of the dependent variables is observed in the samples, then

$$r = \prod_{i=1}^{d} m_i$$

Otherwise, r is the number of cross-classifications actually observed.

If the function specification is MEANS, the number of response functions per population is $q = d$.

Response Statement Examples

Some example response statements are shown in the following table:

Example	Result
`response marginals;`	Marginals for each dependent variable
`response means;`	The mean of each dependent variable
`response logits;`	Generalized logits of the marginal probabilities
`response clogits;`	Cumulative logits of the marginal probabilities
`response alogits;`	Adjacent-category logits of the marginal probabilities
`response joint;`	The joint probabilities
`response 1 -1 log;`	The logit
`response;`	Generalized logits
`response 1 2 3;`	The mean score, with scores of 1, 2, and 3 corresponding to the three response levels
`response read b1-b4;`	Four response functions and their covariance matrix, read directly from the input data set

Using a Transformation to Specify Response Functions

If you specify a *transformation*, it is applied to the vector that contains the sample proportions in each population. The *transformation* can be any combination of the following four operations:

Operation	Specification
linear combination	$*$ matrix literal
linear combination	matrix literal
logarithm	**LOG**
exponential	**EXP**
adding constant	$+$ matrix literal

If more than one operation is specified, then PROC CATMOD applies the operations consecutively from right to left.

A matrix literal is a matrix of numbers with each row of the matrix separated from the next by a comma. If you specify a linear combination, in most cases the $*$ is not needed. The following statement defines the response function $p_1 + 1$. The $*$ is needed to separate the two matrix literals '1' and '1 0'.

```
response + 1 * 1 0;
```

The **LOG** of a vector transforms each element of the vector into its natural logarithm; the **EXP** of a vector transforms each element into its exponential function (antilogarithm).

In order to specify a linear response function for data that have $r = 3$ response categories, you can specify either of the following RESPONSE statements:

```
response  * 1 0 0 , 0 1 0;
response    1 0 0 , 0 1 0;
```

The matrix literal in the preceding statements specifies a 2×3 matrix, which is applied to each population as follows:

$$\begin{bmatrix} F_1 \\ F_2 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix} * \begin{bmatrix} p_1 \\ p_2 \\ p_3 \end{bmatrix}$$

where p_1, p_2, and p_3 are sample proportions for the three response categories in a population, and F_1 and F_2 are the two response functions computed for that population. Therefore, this response function sets $F1 = p_1$ and $F2 = p_2$ in each population.

As another example of the linear response function, suppose you have two dependent variables corresponding to two observers who evaluate the same subjects. If the observers grade on the same three-point scale and if all nine possible responses are observed, then the following RESPONSE statement would compute the probability that the observers agree on their assessments:

```
response 1 0 0 0 1 0 0 0 1;
```

This response function is then computed as

$$F = p_{11} + p_{22} + p_{33} = \begin{bmatrix} 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 \end{bmatrix} * \begin{bmatrix} p_{11} \\ p_{12} \\ p_{13} \\ p_{21} \\ p_{22} \\ p_{23} \\ p_{31} \\ p_{32} \\ p_{33} \end{bmatrix}$$

where p_{ij} denotes the probability that a subject gets a grade of i from the first observer and j from the second observer.

If the function is a compound function, requiring more than one operation to specify it, then the operations should be listed so that the first operation to be applied is on the right and the last operation to be applied is on the left. For example, if there are two response levels, you can have the following response function:

```
response 1 -1 log;
```

This is equivalent to the matrix expression

$$F = \begin{bmatrix} 1 & -1 \end{bmatrix} * \begin{bmatrix} \log(p_1) \\ \log(p_2) \end{bmatrix} = \log(p_1) - \log(p_2) = \log\left(\frac{p_1}{p_2}\right)$$

which is the logit response function since $p_2 = 1 - p_1$ when there are only two response levels.

The following statement specifies another example of a compound response function:

```
response exp 1 -1 * 1 0 0 1, 0 1 1 0 log;
```

This is equivalent to the matrix expression

$$F = \mathbf{EXP}(\mathbf{A} * \mathbf{B} * \mathbf{LOG}(\mathbf{P}))$$

where $\mathbf{P}$ is the vector of sample proportions for some population,

$$\mathbf{A} = \begin{bmatrix} 1 & -1 \end{bmatrix} \text{ and } \mathbf{B} = \begin{bmatrix} 1 & 0 & 0 & 1 \\ 0 & 1 & 1 & 0 \end{bmatrix}$$

If the four responses are based on two dependent variables, each with two levels, then the function can also be written as

$$F = \frac{p_{11}p_{22}}{p_{12}p_{21}}$$

which is the odds (crossproduct) ratio for a 2×2 table.

Understanding the Standard Response Functions

If no RESPONSE statement is specified, PROC CATMOD computes the standard response functions, which contrast the log of each response probability with the log of the probability for the last response category. If there are r response categories, then there are $r - 1$ standard response functions. For example, if there are four response categories, using no RESPONSE statement is equivalent to specifying the following:

```
response  1 0 0 -1,
          0 1 0 -1,
          0 0 1 -1  log;
```

This results in three response functions:

$$F = \begin{bmatrix} F_1 \\ F_2 \\ F_3 \end{bmatrix} = \begin{bmatrix} \log(p_1/p_4) \\ \log(p_2/p_4) \\ \log(p_3/p_4) \end{bmatrix}$$

If there are only two response levels, the resulting response function would be a logit, which is why the standard response functions are called generalized logits. They are useful in dealing with the log-linear model:

$$\boldsymbol{\pi} = \mathbf{EXP}(\mathbf{X}\boldsymbol{\beta})$$

If $\mathbf{C}$ denotes the matrix in the preceding RESPONSE statement, then because of the restriction that the probabilities sum to 1, it follows that an equivalent model is

$$\mathbf{C} * \mathbf{LOG}(\boldsymbol{\pi}) = (\mathbf{CX})\boldsymbol{\beta}$$

But $\mathbf{C} * \mathbf{LOG}(\mathbf{P})$ is simply the vector of standard response functions. Thus, fitting a log-linear model on the cell probabilities is equivalent to fitting a linear model on the generalized logits.

RESTRICT Statement

RESTRICT *parameter*=*value* < . . . *parameter*=*value* > **;**

where *parameter* is the letter B followed by a number; for example, B3 specifies the third parameter in the model. The *value* is the value to which the parameter is restricted. The RESTRICT statement restricts values of parameters to the values you specify, so that the estimation of the remaining parameters is subject to these restrictions. Consider the following statement:

```
restrict b1=1 b4=0 b6=0;
```

This restricts the values of three parameters. The first parameter is set to 1, and the fourth and sixth parameters are set to zero.

The RESTRICT statement is interactive. A new RESTRICT statement replaces any previous ones. In addition, if you submit two or more MODEL, LOGLIN, FACTORS, or REPEATED statements, then the subsequent occurrences of these statements also delete the previous RESTRICT statement.

WEIGHT Statement

WEIGHT *variable* **;**

You can use a WEIGHT statement to refer to a variable containing the cell frequencies, which do not need to be integers. The WEIGHT statement lets you use summary data sets containing a count variable. See the section "Input Data Sets" on page 2163 for further information about the WEIGHT statement.

Details: CATMOD Procedure

Missing Values

Observations with missing values for any variable listed in the MODEL or POPULATION statement are omitted from the analysis.

If the WEIGHT variable for an observation has a missing value, the observation is by default omitted from the analysis. You can modify this behavior by specifying the MISSING= option in the MODEL statement. The option MISSING=*value* sets all missing weights to *value* and all missing cells to *value*. The option MISSING=SAMPLING causes all missing cells in a contingency table to be treated as sampling zeros.

Any observation with nonpositive weight is also, by default, omitted from the analysis. If it has zero weight, then you can specify the ZERO= option in the MODEL statement.

Input Data Sets

Data to be analyzed by PROC CATMOD must be in a SAS data set containing one of the following:

- raw data values (variable values for every subject)
- frequency counts and the corresponding variable values
- response function values and their covariance matrix

If you specify a WEIGHT statement, then PROC CATMOD uses the values of the WEIGHT variable as the frequency counts. If the READ function is specified in the RESPONSE statement, then the procedure expects the input data set to contain the values of response functions and their covariance matrix. Otherwise, PROC CATMOD assumes that the SAS data set contains raw data values.

Raw Data Values

If you use raw data, PROC CATMOD first counts the number of observations having each combination of values for all variables specified in the MODEL or POPULATION statement. For example, suppose the variables A and B each take on the values 1 and 2, and their frequencies can be represented as follows:

		A	
		1	**2**
B	**1**	2	1
	2	3	1

The SAS data set Raw containing the raw data might be as follows:

Observation	A	B
1	1	1
2	1	1
3	1	2
4	1	2
5	1	2
6	2	1
7	2	2

And the statements for PROC CATMOD are as follows:

```
proc catmod data=Raw;
   model A=B;
run;
```

For discussions of how to handle structural and random zeros with raw data as input data, see the section "Zero Frequencies" on page 2187 and Example 33.5.

Frequency Counts

If your data set contains frequency counts, then use the WEIGHT statement to specify the variable containing the frequencies. For example, you could create and analyze the Summary data set as follows:

```
data Summary;
   input A B Count;
   datalines;
1 1 2
1 2 3
2 1 1
2 2 1
;
```

```
proc catmod data=Summary;
   weight Count;
   model A=B;
run;
```

The data set Summary can also be created from the data set Raw by using the FREQ procedure:

```
proc freq data=Raw;
   tables A*B / out=Summary;
run;
```

Inputting Response Functions and Covariances Directly

If you want to read in the response functions and their covariance matrix, rather than have PROC CATMOD compute them, create a TYPE=EST data set. In addition to having one variable name for each function, the data set should have two additional variables: _TYPE_ and _NAME_, both character variables of length 8. The variable _TYPE_ should have the value 'PARMS' when the observation contains the response functions; it should have the value 'COV' when the observation contains elements of the covariance matrix of the response functions. The variable _NAME_ is used only when _TYPE_=COV, in which case it should contain the name of the variable that has its covariance elements stored in that observation. In the following data set, for example, the covariance between the second and fourth response functions is 0.000102:

```
data direct(type=est);
   input b1-b4 _type_ $ _name_ $8.;
   datalines;
0.590463   0.384720   0.273269   0.136458   PARMS      .
0.001690   0.000911   0.000474   0.000432   COV        B1
0.000911   0.001823   0.000031   0.000102   COV        B2
0.000474   0.000031   0.001056   0.000477   COV        B3
0.000432   0.000102   0.000477   0.000396   COV        B4
;
```

In order to tell PROC CATMOD that the input data set contains the values of response functions and their covariance matrix, do the following:

- specify the READ function in the RESPONSE statement
- specify _F_ as the dependent variable in the MODEL statement

For example, suppose the response functions correspond to four populations that represent the cross-classification of two age groups by two race groups. You can use the FACTORS statement to identify these two factors and to name the effects in the model. The following statements are required to fit a main-effects model to these data:

```
proc catmod data=direct;
   response read b1-b4;
   model _f_=_response_;
   factors age 2, race 2 / _response_=age race;
run;
```

Ordering of Populations and Responses

By default, populations and responses are sorted in standard SAS order as follows:

- alphabetical order for character variables
- increasing numeric order for numeric variables

Suppose you specify the following statements:

```
data one;
   length A B $ 6;
   input A $ B $ wt @@;
   datalines;
low       low  23  low    medium  31 low    high  38
medium    low  40  medium medium  42 medium high  50
high      low  52  high   medium  54 high   high  61
;

proc catmod;
   weight wt;
   model A=B / oneway;
run;
```

The ordering of populations and responses corresponds to the alphabetical order of the levels of the character variables. You can specify the ONEWAY option to display the ordering of the variables, while the "Population Profiles" and "Response Profiles" tables display the ordering of the populations and the responses, respectively.

Population Profiles		Response Profiles	
Sample	B	Response	A
1	high	1	high
2	low	2	low
3	medium	3	medium

In this example, if you want to have the levels ordered in the natural order of 'low,' 'medium,' 'high,' you can specify the ORDER=DATA option:

```
proc catmod order=data;
   weight wt;
   model a=b / oneway;
run;
```

The resulting ordering of populations and responses is as follows:

Population Profiles		Response Profiles	
Sample	B	Response	A
1	low	1	low
2	medium	2	medium
3	high	3	high

You can use the ORDER=DATA option to ensure that populations and responses are ordered in a specific way. But since this also affects the definitions and the ordering of the parameters, you must exercise caution when using the _RESPONSE_ effect, the CONTRAST statement, or direct input of the design matrix.

An alternative method of ensuring that populations and responses are ordered in a specific way is to assign a format to your variables and specify the ORDER=FORMATTED option. The levels are then ordered according to their formatted values.

Another method is to replace any character variables with numeric variables and to assign formatted values such as 'yes' and 'no' to the numeric levels. Since ORDER=INTERNAL is the default ordering, PROC CATMOD orders the populations and responses according to the numeric values but displays the formatted values.

Specification of Effects

By default, the CATMOD procedure treats all variables as classification variables. As a result, there is no CLASS statement in PROC CATMOD. The values of a classification variable can be numeric or character. PROC CATMOD builds a set of effects-coded variables to represent the levels of the classification variable and then uses these to fit the model (for details, see the section "Generation of the Design Matrix" on page 2177). You can modify the default by using the DIRECT statement to treat numeric independent continuous variables as continuous variables. The classification variables, combinations of classification variables, and continuous variables are then used in fitting linear models to data.

The parameters of a linear model are generally divided into subsets that correspond to meaningful sources of variation in the response functions. These sources, called *effects*, can be specified in the MODEL, LOGLIN, FACTORS, REPEATED, and CONTRAST statements. Effects can be specified in any of the following ways:

- A main effect is a single classification variable (that is, it produces class levels): A B C.
- A crossed effect (or interaction) is two or more classification variables joined by asterisks—for example: A*B A*B*C.
- A nested effect is a main effect or an interaction, followed by a parenthetical field containing a main effect or an interaction. Multiple variables within the parentheses are assumed to form a crossed effect even when the asterisk is absent. In the following list, the last two effects are identical: B(A) C(A*B) A*B(C*D) A*B(C D).
- A nested-by-value effect is the same as a nested effect except that any variable in the parentheses can be followed by an equal sign and a value: B(A=1) C(A B=1) C*D(A=1 B=1) A(C='low').
- A direct effect is a variable specified in a DIRECT statement: X Y.
- Direct effects can be crossed with other effects: X*Y X*X*X X*A*B(C D=1).

The variables for crossed and nested effects remain in the order in which they are first encountered. For example, in the following model, the effect A*B is reported as B*A since B appears before A in the statement:

```
model R=B A A*B C(A B);
```

Also, C(A B) is interpreted as C(A*B) and is therefore reported as C(B*A).

Bar Notation

You can shorten the specification of multiple effects by using bar notation. For example, the following statements illustrate two methods of writing a full three-way factorial model:

```
proc catmod;
   model y=a b c a*b a*c b*c a*b*c;
run;

proc catmod;
   model y=a|b|c;
run;
```

When you use the bar (|) notation, the right and left sides become effects, and the interaction between them becomes an effect. Multiple bars are permitted. The expressions are expanded from left to right, using rules 1 through 4 given in Searle (1971, p. 390):

- Multiple bars are evaluated left to right. For example, A|B|C is evaluated as follows:
  A | B | C → {A | B} | C
  → {A B A*B} | C
  → A B A*B C A*C B*C A*B*C
- Crossed and nested groups of variables are combined. For example, A(B) | C(D) generates A*C(B D), among other terms.
- Duplicate variables are removed. For example, A(C) | B(C) generates A*B(C C), among other terms, and the extra C is removed.
- Effects are discarded if a variable occurs on both the crossed and nested sides of an effect. For instance, A(B) | B(D E) generates A*B(B D E), but this effect is deleted.

You can also specify the maximum number of variables involved in any effect that results from bar evaluation by specifying that maximum number, preceded by an @ sign, at the end of the bar effect. For example, the specification A | B | C @ 2 would result in only those effects that contain two or fewer variables; in this case, the effects A, B, A*B, C, A*C, and B*C are generated.

Other examples of the bar notation follow:

A | C(B)	is equivalent to	A C(B) A*C(B)
A(B) | C(B)	is equivalent to	A(B) C(B) A*C(B)
A(B) | B(D E)	is equivalent to	A(B) B(D E)
A | B(A) | C	is equivalent to	A B(A) C A*C B*C(A)
A | B(A) | C@2	is equivalent to	A B(A) C A*C
A | B | C | D@2	is equivalent to	A B A*B C A*C B*C D A*D B*D C*D

For details about how the effects specified lead to a design matrix, see the section "Generation of the Design Matrix" on page 2177.

Output Data Sets

OUT= Data Set

For each population, the OUT= data set contains the observed and predicted values of the response functions, their standard errors, the residuals, and variables that describe the population and response profiles. In addition, if you use the standard response functions, the data set includes observed and predicted values for the cell frequencies or the cell probabilities, together with their standard errors and residuals.

Number of Observations

For the standard response functions, there are $s \times (2q - 1)$ observations in the data set for each BY group, where s is the number of populations and q is the number of response functions per population. Otherwise, there are $s \times q$ observations in the data set for each BY group.

Variables in the OUT= Data Set

The data set contains the following variables:

BY variables	If you use a BY statement, the BY variables are included in the OUT= data set.
dependent variables	If the response functions are the default ones (generalized logits), then the dependent variables, which describe the response profiles, are included in the OUT= data set. When _TYPE_=FUNCTION, the values of these variables are missing.
independent variables	The independent variables, which describe the population profiles, are included in the OUT= data set.
NUMBER	the sequence number of the response function or the cell probability or the cell frequency
OBS	the observed value
PRED	the predicted value
RESID	the residual (observed minus predicted)
SAMPLE	the population number. This matches the sample number in the "Population Profile" section of the output.
SEOBS	the standard error of the observed value
SEPRED	the standard error of the predicted value
TYPE	specifies a character variable with three possible values. When _TYPE_=FUNCTION, the observed and predicted values are values of the response functions. When _TYPE_=PROB, they are values of the cell probabilities. When _TYPE_=FREQ, they are values of the cell frequencies. Cell probabilities or frequencies are provided only when the default response functions are modeled. In this case, cell probabilities are provided by default, and cell frequencies are provided if you specify the option PRED=FREQ.

OUTEST= Data Set

This TYPE=EST output data set contains the estimated parameter vector and its estimated covariance matrix. If you specify both the ML and WLS options in the MODEL statement, the OUTEST= data set contains both sets of estimates. For each BY group, there are $p + 1$ observations in the data set for each estimation method, where p is the number of estimated parameters. The data set contains the following variables:

B1, B2, and so on	variables for the estimated parameters. The OUTEST= data set contains one variable for each estimated parameter.
BY variables	If you use a BY statement, the BY variables are included in the OUT= data set.
METHOD	the method used to obtain parameter estimates. For weighted least squares estimation, _METHOD_=WLS, and for maximum likelihood estimation, _METHOD_=ML.
NAME	identifies parameter names. When _TYPE_=PARMS, _NAME_ is blank, but when _TYPE_=COV, _NAME_ has one of the values B1, B2, and so on, corresponding to the parameter names.
STATUS	indicates whether the estimates have converged
TYPE	identifies the statistics contained in the variables for parameter estimates (B1, B2, and so on). When _TYPE_=PARMS, the variables contain parameter estimates; when _TYPE_=COV, they contain covariance estimates.

The variables _METHOD_, _NAME_, and _TYPE_ are character variables; the BY variables can be either character or numeric; and the variables for estimated parameters are numeric.

See Appendix A, "Special SAS Data Sets," for more information about special SAS data sets.

Logistic Analysis

In a logistic analysis, the response functions are the logits of the dependent variable.

PROC CATMOD can compute the three following types of logits with the use of *keywords* in the RESPONSE statement. Note that other types of response functions can be generated by specifying appropriate transformations in the RESPONSE statement.

- Generalized logits are used primarily for nominally scaled dependent variables, but they can also be used for ordinal data modeling. Maximum likelihood estimation is available for the analysis of these logits.
- Cumulative logits are used for ordinally scaled dependent variables. Except for dependent variables with two response levels, only weighted least squares estimation is available for the analysis of these logits.
- Adjacent-category logits are equivalent to generalized logits, but they have some advantages for ordinal data analysis because they automatically incorporate integer scores for the levels of the dependent variable. Except for dependent variables with two response levels, only weighted least squares estimation is available for the analysis of these logits.

If the dependent variable has only two responses, then the cumulative logit and the adjacent-category logit are the negative of the generalized logit, as computed by PROC CATMOD. Consequently, parameter estimates obtained by using these logits are the negative of those obtained from using generalized logits. A simple logistic analysis of variance uses statements like the following:

```
proc catmod;
   model r=a|b;
run;
```

Logistic Regression

If the independent variables are treated quantitatively (like continuous variables), then a logistic analysis is known as a *logistic regression*. If you want PROC CATMOD to treat the independent variables as quantitative variables, specify them in both the DIRECT and MODEL statements, as follows:

```
proc catmod;
   direct x1 x2 x3;
   model r=x1 x2 x3;
run;
```

Since the preceding statements do not include a RESPONSE statement, generalized logits are computed. See Example 33.3 for another example.

The parameter estimates from the CATMOD procedure are the same as those from a logistic regression program such as PROC LOGISTIC (see Chapter 76, "The LOGISTIC Procedure"). The chi-square statistics and the predicted values are also identical. In the binary response case, PROC CATMOD can be made to model the probability of the maximum value by either (1) organizing the input data so that the maximum value occurs first and specifying ORDER=DATA in the PROC CATMOD statement or (2) specifying cumulative logits (CLOGITS) in the RESPONSE statement.

CAUTION: Computational difficulties might occur if you use a continuous variable with a large number of unique values in a DIRECT statement. See the section "Continuous Variables" on page 2172 for more details.

Cumulative Logits

If your dependent variable is ordinally scaled, you can specify the analysis of cumulative logits that take into account the ordinal nature of the dependent variable:

```
proc catmod;
   response clogits;
   direct x;
   model r=a x;
run;
```

The preceding statements correspond to a simple analysis that addresses the question of existence of an association between the independent variables and the ordinal dependent variable. However, there are some commonly used models for the analysis of ordinal data (Agresti 1984) that address the structure of association (in terms of odds ratios), as well as its existence.

If the independent variables are classification variables, a typical analysis for such a model uses the following statements:

```
proc catmod;
   weight wt;
   response clogits;
   model r=_response_ a b;
run;
```

On the other hand, if the independent variables are ordinally scaled, you might specify numeric scores in variables x1 and x2, and use the following statements:

```
proc catmod;
   weight wt;
   direct x1 x2;
   response clogits;
   model r=_response_ x1 x2;
run;
```

See Agresti (1984) for additional details of estimation, testing, and interpretation.

Continuous Variables

Computational difficulties might occur if you have a continuous variable with a large number of unique values and you use this variable in a DIRECT statement, since an observation often represents a separate population of size one. At this extreme of sparseness, the weighted least squares method is inappropriate since there are too many zero frequencies. Therefore, you should use the maximum likelihood method. PROC CATMOD is not designed optimally for continuous variables; therefore, it might be less efficient and unable to allocate sufficient memory to handle this problem, as compared with a procedure designed specifically to handle continuous data. In these situations, consider using the LOGISTIC or GENMOD procedure to analyze your data.

Log-Linear Model Analysis

When the response functions are the default generalized logits, then inclusion of the keyword _RESPONSE_ in every effect in the right side of the MODEL statement fits a log-linear model. The keyword _RESPONSE_ tells PROC CATMOD that you want to model the variation among the dependent variables. You then specify the actual model in the LOGLIN statement.

When you perform log-linear model analysis, you can request weighted least squares estimates, maximum likelihood estimates, or both. By default, PROC CATMOD calculates maximum likelihood estimates when the default response functions are used. The following table provides appropriate MODEL statements for the combinations of types of estimates:

Estimation Desired	MODEL Statement
Maximum likelihood (Newton-Raphson)	`model a*b=_response_;`
Maximum likelihood (Iterative Proportional Fitting)	`model a*b=_response_ / ml=ipf;`
Weighted least squares	`model a*b=_response_ / wls;`
Maximum likelihood and weighted least squares	`model a*b=_response_ / wls ml;`

CAUTION: Sampling zeros in the input data set should be specified with the ZERO= option to ensure that these sampling zeros are not treated as structural zeros. Alternatively, you can replace cell counts for sampling zeros with some positive number close to zero (such as 1E–20) in a DATA step. Data containing sampling zeros should be analyzed with maximum likelihood estimation. See the section "Cautions" on page 2186 and Example 33.5 for further information and an illustration that uses both cell count data and raw data.

One Population

The usual log-linear model analysis has one population, which means that all of the variables are dependent variables. For example, the following statements yield a maximum likelihood analysis of a saturated log-linear model for the dependent variables r1 and r2:

```
proc catmod;
   weight wt;
   model r1*r2=_response_;
   loglin r1|r2;
run;
```

If you want to fit a reduced model with respect to the dependent variables (for example, a model of independence or conditional independence), specify the reduced model in the LOGLIN statement. For example, the following statements yield a main-effects log-linear model analysis of the factors r1 and r2:

```
proc catmod;
   weight wt;
   model r1*r2=_response_ / pred;
   loglin r1 r2;
run;
```

The output includes Wald statistics for the individual effects r1 and r2, as well as predicted cell probabilities. Moreover, the goodness-of-fit statistic is the likelihood ratio test for the hypothesis of independence between r1 and r2 or, equivalently, a test of r1*r2.

Multiple Populations

You can do log-linear model analysis with multiple populations by using a POPULATION statement or by including effects on the right side of the MODEL statement that contain independent variables. Each effect must include the _RESPONSE_ keyword.

For example, suppose the dependent variables r1 and r2 are dichotomous, and the independent variable group has three levels. Then the following statements specify a saturated model (three degrees of freedom for _RESPONSE_ and six degrees of freedom for the interaction between _RESPONSE_ and group):

```
proc catmod;
   weight wt;
   model r1*r2=_response_ group*_response_;
   loglin r1|r2;
run;
```

From another point of view, _RESPONSE_*group can be regarded as a main effect for group with respect to the three response functions, while _RESPONSE_ can be regarded as an intercept effect with respect to the functions. In other words, the following statements give essentially the same results as the logistic analysis:

```
proc catmod;
   weight wt;
   model r1*r2=group;
run;
```

The ability to model the interaction between the independent and the dependent variables becomes particularly useful when a reduced model is specified for the dependent variables. For example, the following statements specify a model with two degrees of freedom for _RESPONSE_ (one for r1 and one for r2) and four degrees of freedom for the interaction of _RESPONSE_*group:

```
proc catmod;
   weight wt;
   model r1*r2=_response_ group*_response_;
   loglin r1 r2;
run;
```

The likelihood ratio goodness-of-fit statistic (three degrees of freedom) tests the hypothesis that r1 and r2 are independent in each of the three groups.

Iterative Proportional Fitting

You can use the iterative proportional fitting (IPF) algorithm to fit a hierarchical log-linear model with no independent variables and no population variables.

The advantage of IPF over the Newton-Raphson (NR) algorithm and over the weighted least squares (WLS) method is that, when the contingency table has several dimensions and the parameter vector is large, you can obtain the log likelihood, the goodness-of-fit G^2, and the predicted frequencies or probabilities without performing potentially expensive parameter estimation and covariance matrix calculations. This enables you to do the following:

- compare two models by computing the likelihood ratio statistics to test the significance of the contribution of the variables in one model that are not in the other model
- compute predicted values of the cell probabilities or frequencies for the final model

Each iteration of the IPF algorithm is generally faster than an iteration of the NR algorithm; however, the IPF algorithm converges to the MLEs more slowly than the NR algorithm. Both NR and WLS are more general methods that are able to perform more complex analyses than IPF can.

Repeated Measures Analysis

If there are multiple dependent variables and the variables represent repeated measurements of the same observational unit, then the variation among the dependent variables can be attributed to one or more repeated measurement factors. The factors can be included in the model by specifying _RESPONSE_ on the right side of the MODEL statement and by using a REPEATED statement to identify the factors.

To perform a repeated measures analysis, you also need to specify a RESPONSE statement, since the standard response functions (generalized logits) cannot be used. Typically, the MEANS or MARGINALS response functions are specified in a repeated measures analysis, but other response functions can also be reasonable.

One Population

Consider an experiment in which each subject is measured at three times, and the response functions are marginal probabilities for each of the dependent variables. If the dependent variables each have k levels, then PROC CATMOD computes $k-1$ response functions for each time. Differences among the response functions with respect to these times could be attributed to the repeated measurement factor Time. To incorporate the Time variation into the model, specify the following statements:

```
proc catmod;
   response marginals;
   model t1*t2*t3=_response_;
   repeated Time 3 / _response_=Time;
run;
```

These statements produce a Time effect that has $2(k-1)$ degrees of freedom since there are $k-1$ response functions at each time point. For a dichotomous variable, the Time effect has two degrees of freedom.

Now suppose that at each time point, each subject has X-rays taken, and the X-rays are read by two different radiologists. This creates six dependent variables that represent the 3×2 cross-classification of the repeated measurement factors Time and Reader. A saturated model with respect to these factors can be obtained by specifying the following statements:

```
proc catmod;
   response marginals;
   model r11*r12*r21*r22*r31*r32=_response_;
   repeated Time 3, Reader 2
      / _response_=Time Reader Time*Reader;
run;
```

If you want to fit a main-effects model with respect to Time and Reader, then change the REPEATED statement to the following:

```
   repeated Time 3, Reader 2 / _response_=Time Reader;
```

If you want to fit a main-effects model for Time but for only one of the readers, the REPEATED statement might look like the following:

```
   repeated Time $ 3, Reader $ 2
            /_response_=Time(Reader=Smith)
              profile  =('1'  Smith,
                         '1'  Jones,
                         '2'  Smith,
                         '2'  Jones,
                         '3'  Smith,
                         '3'  Jones);
```

If Jones had been unavailable for a reading at time 3, then there would be only $5(k-1)$ response functions, even though PROC CATMOD would be expecting some multiple of 6 ($= 3 \times 2$). In that case, the PROFILE= option would be necessary to indicate which repeated measurement profiles were actually represented:

```
   repeated Time $ 3, Reader $ 2
            /_response_=Time(Reader=Smith)
              profile  =('1'  Smith,
                         '1'  Jones,
                         '2'  Smith,
                         '2'  Jones,
                         '3'  Smith);
```

When two or more repeated measurement factors are specified, PROC CATMOD presumes that the response functions are ordered so that the levels of the rightmost factor change most rapidly. This means that the dependent variables should be specified in the same order. For this example, the order implied by the REPEATED statement is as follows, where the variable r_{ij} corresponds to Time i and Reader j:

Response Function	Dependent Variable	Time	Reader
1	r_{11}	1	1
2	r_{12}	1	2
3	r_{21}	2	1
4	r_{22}	2	2
5	r_{31}	3	1
6	r_{32}	3	2

The order of dependent variables in the MODEL statement must agree with the order implied by the REPEATED statement.

Multiple Populations

When there are variables specified in the POPULATION statement or on the right side of the MODEL statement, these variables produce multiple populations. PROC CATMOD can then model these independent variables, the repeated measurement factors, and the interactions between the two.

For example, suppose that there are five groups of subjects, that each subject in the study is measured at three different times, and that the dichotomous dependent variables are labeled t1, t2, and t3. The following statements compute three response functions for each population:

```
proc catmod;
   weight wt;
   population Group;
   response marginals;
   model t1*t2*t3=_response_;
   repeated Time / _response_=Time;
run;
```

PROC CATMOD then regards _RESPONSE_ as a variable with three levels corresponding to the three response functions in each population and forms an effect with two degrees of freedom. The MODEL and REPEATED statements tell PROC CATMOD to fit the main effect of Time.

In general, the MODEL statement tells PROC CATMOD how to integrate the independent variables and the repeated measurement factors into the model. For example, again suppose that there are five groups of subjects, that each subject is measured at three times, and that the dichotomous independent variables are labeled t1, t2, and t3. If you use the same WEIGHT, POPULATION, RESPONSE, and REPEATED statements as in the preceding program, the following MODEL statements result in the indicated analyses:

`model t1*t2*t3=Group / averaged;`	Specifies the Group main effect (with 4 degrees of freedom)
`model t1*t2*t3=_response_;`	Specifies the Time main effect (with 2 degrees of freedom)
`model t1*t2*t3=_response_*Group;`	Specifies the interaction between Time and Group (with 8 degrees of freedom)
`model t1*t2*t3=_response_|Group;`	Specifies both main effects, and the interaction between Time and Group (with a total of 14 degrees of freedom)
`model t1*t2*t3=_response_(Group);`	Specifies a Time main effect within each Group (with 10 degrees of freedom)

However, the following MODEL statement is invalid since effects cannot be nested within _RESPONSE_:

```
model t1*t2*t3=Group(_response_);
```

Generation of the Design Matrix

Each row of the design matrix (corresponding to a population) is generated by a unique combination of independent variable values. Each column of the design matrix corresponds to a model parameter. The columns are produced from the effect specifications in the MODEL, LOGLIN, FACTORS, and REPEATED statements. For details about effect specifications, see the section "Specification of Effects" on page 2167.

This section is divided into three parts:

- one response function per population
- two or more response functions per population (excluding log-linear models), beginning on page 2180
- log-linear models, beginning on page 2183

This section assumes that the default effect parameterization is used. Specifying the reference parameterization replaces the "–1"s with zeros in the design matrix for the main effects of classification variables, and makes appropriate changes to interaction terms.

You can display the design matrix by specifying the DESIGN option in the MODEL statement.

One Response Function per Population

Intercept

When there is one response function per population, all design matrices start with a column of 1s for the intercept unless the NOINT option is specified or the design matrix is input directly.

Main Effects

If a classification variable A has k levels, then its main effect has $k-1$ degrees of freedom, and the design matrix has $k-1$ columns that correspond to the first $k-1$ levels of A. The ith column contains a 1 in the ith row, a –1 in the last row, and 0s everywhere else. If α_i denotes the parameter that corresponds to the ith level of variable A, then the $k-1$ columns yield estimates of the independent parameters, $\alpha_1, \alpha_i, \ldots, \alpha_{k-1}$. The

last parameter is not needed because PROC CATMOD constrains the k parameters to sum to zero. In other words, PROC CATMOD uses a full-rank center-point parameterization to build design matrices. Here are two examples:

Variable	Data Levels	Effect Parameterization Design Matrix	
A	1	1	0
	2	0	1
	3	–1	–1
B	1	–1	
	2	–1	

For an effect with three levels, such as A, PROC CATMOD produces two parameter estimates for each response function. By default, the first (corresponding to the first row in the design columns) estimates the effect of level 1 of A compared to the average effect of the three levels of A. The second (corresponding to the second row in the design columns) estimates the effect of level 2 of A compared to the average effect of the three levels of A. The sum-to-zero constraint requires the effect of level 3 of A to be the negative of the sum of the level 1 and 2 effects (as shown by the third row in the design columns).

Crossed Effects (Interactions)

Crossed effects (such as A*B) are formed by the horizontal direct products of main effects, as illustrated in the following table:

Data Levels		Design Matrix				
A	**B**	**A**		**B**	**A*B**	
1	1	1	0	1	1	0
1	2	1	0	–1	–1	0
2	1	0	1	1	0	1
2	2	0	1	–1	0	–1
3	1	–1	–1	1	–1	–1
3	2	–1	–1	–1	1	1

The number of degrees of freedom for a crossed effect (that is, the number of design matrix columns) is equal to the product of the numbers of degrees of freedom for the separate effects.

Nested Effects

The effect A(B) is read "A within B" and is the same as specifying an A main effect for every value of B. If n_a and n_b are the number of levels in A and B, respectively, then the number of columns for A(B) is $(n_a - 1)n_b$ when every combination of levels exists in the data. The following table gives an example:

Data Levels		Design Matrix			
B	**A**	**A(B)**			
1	1	1	0	0	0
1	2	0	1	0	0
1	3	–1	–1	0	0
2	1	0	0	1	0
2	2	0	0	0	1
2	3	0	0	–1	–1

CAUTION: PROC CATMOD actually allocates a column for all possible combinations of values even though some combinations are not present in the data. This can be of particular concern if the data are not balanced with respect to the nested levels.

Nested-by-Value Effects

Instead of nesting an effect within all values of the main effect, you can nest an effect within specified values of the nested variable (A(B=1), for example). The four degrees of freedom for the A(B) effect shown in the preceding section can also be obtained by specifying the two separate nested effects with values, as the following table shows:

Data Levels		Design Matrix			
B	A	A(B=1)		A(B=2)	
1	1	1	0	0	0
1	2	0	1	0	0
1	3	–1	–1	0	0
2	1	0	0	1	0
2	2	0	0	0	1
2	3	0	0	–1	–1

Each effect has $n_a - 1$ degrees of freedom, assuming a complete combination, so each effect in this example has two degrees of freedom.

The procedure compares nested values to data values on the basis of formatted values. If a format is not specified for the variable, the procedure formats internal data values to BEST16, left-justified. The nested values specified in nested-by-value effects are also converted to a BEST16 formatted value, left-justified.

For example, if the numeric variable B has internal data values 1 and 2, then A(B=1), A(B=1.0), and A(B=1E0) are all valid nested-by-value effects. However, if the data value 1 is formatted as 'one', then A(B='one') is a valid effect, but A(B=1) is not since the formatted nested value (1) does not match the formatted data value (one).

To ensure correct nested-by-value effects, look at the tables of population and response profiles. These are displayed by default, and they contain the formatted data values. In addition, the population and response profiles are displayed when you specify the ONEWAY option in the MODEL statement.

Direct Effects

To request that the actual values of a variable be inserted into the design matrix, declare the variable in a DIRECT statement, and specify the effect by the variable name. For example, specifying the effects X1 and X2 in both the MODEL and DIRECT statements results in the following:

Data Levels		Design Columns	
X1	X2	X1	X2
1	1	1	1
2	4	2	4
3	9	3	9

Unless there is a POPULATION statement that excludes the direct variables, the direct variables help to define the sample populations. In general, the variables should not be continuous in the sense that every subject has a different value because this would create a separate population for each subject (note, however, that such a strategy is used purposely for logistic regression).

If there is a POPULATION statement that omits mention of the direct variables, then the values of the direct variables must be identical for all subjects in a given population since there can be only one independent variable profile for each population.

Two or More Response Functions per Population

When there is more than one response function per population, the structure of the design matrix depends on whether or not the model type is AVERAGED (see the AVERAGED option in the MODEL statement). The model type is AVERAGED if independent variable effects are averaged over the multiple responses within a population rather than being nested in them.

The following subsections illustrate the effect of specifying (or not specifying) an AVERAGED model type. This section does not apply to log-linear models; for these models, see the section "Log-Linear Model Design Matrices" on page 2183.

Model Type Not AVERAGED

Suppose the variable A has two levels, and you specify the following statements:

```
proc catmod;
   model Y=A / design;
run;
```

If the variable Y has two levels, then there is only one response function per population, and the design matrix is as follows:

	Design Matrix	
Sample	**Intercept**	**A**
1	1	1
2	1	–1

But if the variable Y has three levels, then there are two response functions per population, and the preceding design matrix is assumed to hold for each of the two response functions. The response functions are always ordered so that the multiple response functions within a population are grouped together. For this example, the design matrix would be as follows:

	Response Function	Design Matrix			
Sample	**Number**	**Intercept**		**A**	
1	1	1	0	1	0
1	2	0	1	0	1
2	1	1	0	–1	0
2	2	0	1	0	–1

Since the same submatrix applies to each of the multiple response functions, PROC CATMOD displays only the submatrix (that is, the one it would create if there were only one response function per population) rather than the entire design matrix. PROC CATMOD displays

$$\begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}$$

Ordering of Parameters

This grouping of multiple response functions within populations also has an effect in the table of parameter estimates displayed by PROC CATMOD. The following table shows some parameter estimates, where the four rows of the table correspond to the four columns in the preceding design matrix:

Effect	Parameter	Estimate
Intercept	1	1.4979
	2	0.8404
A	3	0.1116
	4	–0.3296

Notice that the intercept and the A effect each have two parameter estimates associated with them. The first estimate in each pair is associated with the first response function, and the second in each pair is associated with the second response function. Consequently, 0.1116 is the effect of the first level of A on the first response function. In any table of parameter estimates displayed by PROC CATMOD, as you read down the column of estimates, the response function level changes before levels of the variables making up the effect.

Model Type AVERAGED

When the model type is AVERAGED (for example, when the AVERAGED option is specified in the MODEL statement, when _RESPONSE_ is used in the MODEL statement, or when the design matrix is input directly in the MODEL statement), PROC CATMOD does not assume that the same submatrix applies to each of the q response functions per population. Rather, it averages any independent variable effects across the functions, and it enables you to study variation among the q functions. The first column of the design matrix is always a column of 1s corresponding to the intercept, unless the NOINT option is specified in the MODEL statement or the design matrix is input directly. Also, since the design matrix does not have any special submatrix structure, PROC CATMOD displays the entire matrix.

For example, suppose the dependent variable Y has three levels, the independent variable A has two levels, and you specify the following statements:

```
proc catmod;
   response marginals;
   model y=a / averaged design;
run;
```

Then there are two response functions per population, and the response functions are always ordered so that the multiple response functions within a population are grouped together. For this example, the design matrix would be as follows:

Sample	Response Function Number	Design Matrix	
		Intercept	A
1	1	1	1
1	2	1	1
2	1	1	–1
2	2	1	–1

Note that the model now has only two degrees of freedom. The remaining two degrees of freedom in the residual correspond to variation among the three levels of the dependent variable. Generally, that variation tends to be statistically significant and therefore should not be left out of the model. You can include it in the

model by including the two effects, _RESPONSE_ and _RESPONSE_*A, but if the study is not a repeated measures study, those sources of variation tend to be uninteresting. The usual solution for this type of study (one dependent variable) is to exclude the AVERAGED option from the MODEL statement.

An AVERAGED model type is automatically used whenever you use the _RESPONSE_ keyword in the MODEL statement. The _RESPONSE_ effect models variation among the q response functions per population. If there is no REPEATED, FACTORS, or LOGLIN statement, then PROC CATMOD builds a main effect with $q - 1$ degrees of freedom. For example, three response functions would produce the following design columns:

Response Function Number	Design Columns _RESPONSE_	
1	1	0
2	0	1
3	–1	–1

If there is more than one population, then the _RESPONSE_ effect is averaged over the populations. Also, the _RESPONSE_ effect can be crossed with any other effect, or it can be nested within an effect.

If there is a REPEATED statement that contains only one repeated measurement factor, then PROC CATMOD builds the design columns for _RESPONSE_ in the same way, except that the output labels the main effect with the factor name rather than with the word _RESPONSE_. For example, suppose an independent variable A has two levels, and the input statements are as follows:

```
proc catmod;
   response marginals;
   model Time1*Time2=A _response_ A*_response_ / design;
   repeated Time 2 / _response_=Time;
run;
```

If Time1 and Time2 each have two levels (so that they each have one independent marginal probability), then the RESPONSE statement makes PROC CATMOD compute two response functions per population. The design matrix is as follows:

Sample	Response Function Number	Design Matrix Intercept	A	Time	A*Time
1	1	1	1	1	1
1	2	1	1	–1	–1
2	1	1	–1	1	–1
2	2	1	–1	–1	1

However, if Time1 and Time2 each have three levels (so that they each have two independent marginal probabilities), then the RESPONSE statement causes PROC CATMOD to compute four response functions per population. In that case, since Time has two levels, PROC CATMOD groups the functions into sets of 2 $(= 4/2)$ and constructs the preceding submatrix for each function in the set. This results in the following design matrix, which is obtained from the previous one by multiplying each element by an identity matrix of order two:

Sample	Response Function	Design Matrix							
		Intercept		A		Time		A*Time	
1	P(Time1=1)	1	0	1	0	1	0	1	0
1	P(Time1=2)	0	1	0	1	0	1	0	1
1	P(Time2=1)	1	0	1	0	–1	0	–1	0
1	P(Time2=2)	0	1	0	1	0	–1	0	–1
2	P(Time1=1)	1	0	–1	0	1	0	–1	0
2	P(Time1=2)	0	1	0	–1	0	1	0	–1
2	P(Time2=1)	1	0	–1	0	–1	0	1	0
2	P(Time2=2)	0	1	0	–1	0	–1	0	1

If there is a REPEATED statement that contains two or more repeated measurement factors, then PROC CATMOD builds the design columns for _RESPONSE_ according to the definition of _RESPONSE_ in the REPEATED statement. For example, suppose you specify the following statements:

```
proc catmod;
   response marginals;
   model R11*R12*R21*R22=_response_ / design;
   repeated Time 2, Place 2 / _response_=Time Place;
run;
```

If each of the dependent variables has two levels, then PROC CATMOD builds four response functions. The _RESPONSE_ effect generates a main-effects model with respect to Time and Place, and the design matrix is as follows:

Response Function Number	Variable	Time	Place	Design Matrix	
				Intercept	_RESPONSE_
1	R11	1	1	1	1
2	R12	1	2	1	–1
3	R21	2	1	1	–1
4	R22	2	2	1	–1

Log-Linear Model Design Matrices

When the response functions are the standard ones (generalized logits), then inclusion of the keyword _RESPONSE_ in every design effect fits a log-linear model. The design matrix for a log-linear model looks different from a standard design matrix because the standard one is transformed by the same linear transformation that converts the r response probabilities to $r-1$ generalized logits. For example, suppose the dependent variables X and Y each have two levels, and you specify a saturated log-linear model analysis:

```
proc catmod;
   model X*Y=_response_ / design;
   loglin X Y X*Y;
run;
```

Then the cross-classification of X and Y yields four response probabilities, p_{11}, p_{12}, p_{21}, and p_{22}, which are then reduced to three generalized logit response functions, $F_1 = \log(p_{11}/p_{22})$, $F_2 = \log(p_{12}/p_{22})$, and $F_3 = \log(p_{21}/p_{22})$.

Since the saturated log-linear model implies that

$$\begin{bmatrix} \log(p_{11}) \\ \log(p_{12}) \\ \log(p_{21}) \\ \log(p_{22}) \end{bmatrix} = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & 1 & -1 & -1 \\ 1 & -1 & 1 & -1 \\ 1 & -1 & -1 & 1 \end{bmatrix} \boldsymbol{\gamma} - \lambda \begin{bmatrix} 1 \\ 1 \\ 1 \\ 1 \end{bmatrix}$$

$$= \begin{bmatrix} 1 & 1 & 1 \\ 1 & -1 & -1 \\ -1 & 1 & -1 \\ -1 & -1 & 1 \end{bmatrix} \boldsymbol{\beta} - \delta \begin{bmatrix} 1 \\ 1 \\ 1 \\ 1 \end{bmatrix}$$

where $\boldsymbol{\gamma}$ and $\boldsymbol{\beta}$ are parameter vectors, and λ and δ are normalizing constants required by the restriction that the probabilities sum to 1, it follows that the MODEL statement yields

$$\begin{bmatrix} F_1 \\ F_2 \\ F_3 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & -1 \\ 0 & 1 & 0 & -1 \\ 0 & 0 & 1 & -1 \end{bmatrix} \times \begin{bmatrix} \log(p_{11}) \\ \log(p_{12}) \\ \log(p_{21}) \\ \log(p_{22}) \end{bmatrix}$$

$$= \begin{bmatrix} 1 & 0 & 0 & -1 \\ 0 & 1 & 0 & -1 \\ 0 & 0 & 1 & -1 \end{bmatrix} \times \begin{bmatrix} 1 & 1 & 1 \\ 1 & -1 & -1 \\ -1 & 1 & -1 \\ -1 & -1 & 1 \end{bmatrix} \boldsymbol{\beta}$$

$$= \begin{bmatrix} 2 & 2 & 0 \\ 2 & 0 & -2 \\ 0 & 2 & -2 \end{bmatrix} \boldsymbol{\beta}$$

The design matrix is as follows:

Sample	Response Function Number	Design Matrix X	Y	X*Y
1	1	2	2	0
1	2	2	0	–2
1	3	0	2	–2

Design matrices for reduced models are constructed similarly. For example, suppose you request a main-effects log-linear model analysis of the factors X and Y:

```
proc catmod;
   model X*Y=_response_ / design;
   loglin X Y;
run;
```

Since the main-effects log-linear model implies that

$$\begin{bmatrix} \log(p_{11}) \\ \log(p_{12}) \\ \log(p_{21}) \\ \log(p_{22}) \end{bmatrix} = \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & -1 \\ 1 & -1 & 1 \\ 1 & -1 & -1 \end{bmatrix} \boldsymbol{\gamma} - \lambda \begin{bmatrix} 1 \\ 1 \\ 1 \\ 1 \end{bmatrix}$$

$$= \begin{bmatrix} 1 & 1 \\ 1 & -1 \\ -1 & 1 \\ -1 & -1 \end{bmatrix} \boldsymbol{\beta} - \delta \begin{bmatrix} 1 \\ 1 \\ 1 \\ 1 \end{bmatrix}$$

it follows that the MODEL statement yields

$$\begin{bmatrix} F_1 \\ F_2 \\ F_3 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & -1 \\ 0 & 1 & 0 & -1 \\ 0 & 0 & 1 & -1 \end{bmatrix} \times \begin{bmatrix} \log(p_{11}) \\ \log(p_{12}) \\ \log(p_{21}) \\ \log(p_{22}) \end{bmatrix}$$

$$= \begin{bmatrix} 1 & 0 & 0 & -1 \\ 0 & 1 & 0 & -1 \\ 0 & 0 & 1 & -1 \end{bmatrix} \times \begin{bmatrix} 1 & 1 \\ 1 & -1 \\ -1 & 1 \\ -1 & -1 \end{bmatrix} \boldsymbol{\beta}$$

$$= \begin{bmatrix} 2 & 2 \\ 2 & 0 \\ 0 & 2 \end{bmatrix} \boldsymbol{\beta}$$

Therefore, the corresponding design matrix is as follows:

Sample	Response Function Number	Design Matrix X	Design Matrix Y
1	1	2	2
1	2	2	0
1	3	0	2

Since it is difficult to tell from the final design matrix whether PROC CATMOD used the parameterization that you intended, the procedure displays the untransformed _RESPONSE_ matrix for log-linear models. For example, specifying the main-effects model in the preceding example displays the following matrix:

Response Function Number	_RESPONSE_ Matrix 1	_RESPONSE_ Matrix 2
1	1	1
2	1	–1
3	–1	1
4	–1	–1

You can suppress the display of this matrix by specifying the NORESPONSE option in the MODEL statement.

Cautions

Effective Sample Size

Since the method depends on asymptotic approximations, you need to be careful that the sample sizes are sufficiently large to support the asymptotic normal distributions of the response functions. A general guideline is that you would like to have an effective sample size of at least 25 to 30 for each response function that is being analyzed. For example, if you have one dependent variable and $r = 4$ response levels, and you use the standard response functions to compute three generalized logits for each population, then you would like the sample size of each population to be at least 75. Moreover, the subjects should be dispersed throughout the table so that less than 20 percent of the response functions have an effective sample size less than 5. For example, if each population had less than 5 subjects in the first response category, then it would be wiser to pool this category with another category rather than to assume the asymptotic normality of the first response function. Or, if the dependent variable is ordinally scaled, an alternative is to request the mean score response function rather than three generalized logits.

If there is more than one dependent variable, and you specify RESPONSE MEANS, then the effective sample size for each response function is the same as the actual sample size. Thus, a sample size of 30 could be sufficient to support four response functions, provided that the functions are the means of four dependent variables.

A Singular Covariance Matrix

If there is a singular (noninvertible) covariance matrix for the response functions in any population, then PROC CATMOD writes an error message and stops processing. You have several options available to correct this problem:

- You can reduce the number of response functions according to how many can be supported by the populations with the smallest sample sizes.
- If there are three or more levels for any independent variable, you can pool the levels into a fewer number of categories, thereby reducing the number of populations. However, your interpretation of results must be done more cautiously since such pooling implies a different sampling scheme and masks any differences that existed among the pooled categories.
- If there are two or more independent variables, you can delete at least one of them from the model. However, this is just another form of pooling, and the same cautions that apply to the previous option also apply here.
- If there is one independent variable, then, in some situations, you might simply eliminate the populations that are causing the covariance matrices to be singular.
- You can use the ADDCELL= option in the MODEL statement to add a small amount (for example, 0.5) to every cell frequency, but this can seriously bias the results if the cell frequencies are small.

Zero Frequencies

There are two types of zero cells in a contingency table: structural and sampling. A structural zero cell has an expected value of zero, while a sampling zero cell can have nonzero expected value and can be estimable.

If you use the standard response functions and there are zero frequencies, you should use maximum likelihood estimation (the default is ML=NR) rather than weighted least squares to analyze the data. For weighted least squares analysis, the CATMOD procedure always computes the observed response functions and might need to take the logarithm of a zero proportion. In this case, PROC CATMOD issues a warning and then takes the log of a small value ($0.5/n_i$ for the probability) in order to continue, but this can produce invalid results if the cells contain too few observations. Maximum likelihood analysis, on the other hand, does not require computation of the observed response functions and therefore yields valid results for the parameter estimates and all of the predicted values.

For a log-linear model analysis with WLS or ML=NR, PROC CATMOD creates response profiles only for the observed profiles. For any log-linear model analysis with one population (the usual case), the contingency table does not contain zeros, which means that all zero frequencies are treated as structural zeros. If there is more than one population, then a zero in the body of the contingency table is treated as a sampling zero (as long as some population has a nonzero count for that profile). If you fit the log-linear model by using ML=IPF, the contingency table is incomplete and the zeros are treated like structural zeros. If you want zero frequencies that PROC CATMOD would normally treat as structural zeros to be interpreted as sampling zeros, you can specify the ZERO=SAMPLING and MISSING=SAMPLING options in the MODEL statement. Alternatively, you can specify ZERO=1E–20 and MISSING=1E–20.

See Bishop, Fienberg, and Holland (1975) for a discussion of the issues and Example 33.5 for an illustration of a log-linear model analysis of data that contain both structural and sampling zeros.

If you perform a weighted least squares analysis on a contingency table that contains zero cell frequencies, then avoid using the LOG transformation as the first transformation on the observed proportions. In general, it is better to change the response functions or to pool some of the response categories than to settle for the 0.5 correction or to use the ADDCELL= option.

Testing the Wrong Hypothesis

If you use the keyword _RESPONSE_ in the MODEL statement, and you specify MARGINALS, LOGITS, ALOGITS, or CLOGITS in your RESPONSE statement, you might receive the following warning message:

```
Warning: The _RESPONSE_ effect may be testing the wrong
         hypothesis since the marginal levels of the
         dependent variables do not coincide. Consult the
         response profiles and the CATMOD documentation.
```

The following examples illustrate situations in which the _RESPONSE_ effect tests the wrong hypothesis.

Zeros in the Marginal Frequencies

Suppose you specify the following statements:

```
data A1;
   input Time1 Time2 @@;
   datalines;
1 2    2 3    1 3
;
```

```
proc catmod;
   response marginals;
   model Time1*Time2=_response_;
   repeated Time 2 / _response_=Time;
run;
```

One marginal probability is computed for each dependent variable, resulting in two response functions. The model is a saturated one: one degree of freedom for the intercept and one for the main effect of Time. Except for the warning message, PROC CATMOD produces an analysis with no apparent errors, but the "Response Profiles" table displayed by PROC CATMOD is as follows:

Response Profiles

Response	Time1	Time2
1	1	2
2	1	3
3	2	3

Since RESPONSE MARGINALS yields marginal probabilities for every level but the last, the two response functions being analyzed are Prob(Time1=1) and Prob(Time2=2). The Time effect is testing the hypothesis that Prob(Time1=1)=Prob(Time2=2). What it *should* be testing is the hypothesis that

```
Prob(Time1=1) = Prob(Time2=1)
Prob(Time1=2) = Prob(Time2=2)
Prob(Time1=3) = Prob(Time2=3)
```

but there are not enough data to support the test (assuming that none of the probabilities are structural zeros by the design of the study).

The ORDER=DATA Option

Suppose you specify the following statements:

```
data a1;
   input Time1 Time2 @@;
   datalines;
2 1    2 2    1 1    1 2    2 1
;

proc catmod order=data;
   response marginals;
   model Time1*Time2=_response_;
   repeated Time 2 / _response_=Time;
run;
```

As in the preceding example, one marginal probability is computed for each dependent variable, resulting in two response functions. The model is also the same: one degree of freedom for the intercept and one for the main effect of Time. PROC CATMOD issues the warning message and displays the following "Response Profiles" table:

Response Profiles		
Response	**Time1**	**Time2**
1	2	1
2	2	2
3	1	1
4	1	2

Although the marginal levels are the same for the two dependent variables, they are not in the same order because the ORDER=DATA option specified that they be ordered according to their appearance in the input stream. Since RESPONSE MARGINALS yields marginal probabilities for every level except the last, the two response functions being analyzed are Prob(Time1=2) and Prob(Time2=1). The Time effect is testing the hypothesis that Prob(Time1=2)=Prob(Time2=1). What it *should* be testing is the hypothesis that

```
Prob(Time1=1) = Prob(Time2=1)
Prob(Time1=2) = Prob(Time2=2)
```

Whenever the warning message appears, look at the "Response Profiles" table or the "One-Way Frequencies" table to determine what hypothesis is actually being tested. For the latter example, a correct analysis can be obtained by deleting the ORDER=DATA option or by reordering the data so that the (1,1) observation is first.

Computational Method

The notation used in PROC CATMOD differs slightly from that used in the literature. The following table provides a summary of the basic dimensions and the notation for a contingency table. See the section "Computational Formulas" on page 2190 for a complete description.

Summary of Basic Dimensions

s = number of populations or samples (= number of rows in the underlying contingency table)

r = number of response categories (= number of columns in the underlying contingency table)

q = number of response functions computed for each population

d = number of parameters

Notation

$\mathbf{j}$	Denotes a column vector of 1s.
$\mathbf{J}$	Denotes a square matrix of 1s.
$\sum_k$	Denotes the sum over all the possible values of k.
n_i	Denotes the row sum $\sum_j n_{ij}$.
$\mathbf{DIAG}_n(\mathbf{p})$	Denotes the diagonal matrix formed from the first n elements of the vector $\mathbf{p}$.
$\mathbf{DIAG}_n^{-1}(\mathbf{p})$	Denotes the inverse of $\mathbf{DIAG}_n(\mathbf{p})$.
$\mathbf{DIAG}(\mathbf{A}_1, \mathbf{A}_2, \ldots, \mathbf{A}_k)$	Denotes a block diagonal matrix with the $\mathbf{A}$ matrices on the main diagonal.

Input data can be represented by a contingency table, as shown in Table 33.7.

Table 33.7 Input Data Represented by a Contingency Table

	Response				
Population	**1**	**2**	$\cdots$	***r***	**Total**
1	n_{11}	n_{12}	$\cdots$	n_{1r}	n_1
2	n_{21}	n_{22}	$\cdots$	n_{2r}	n_2
$\vdots$	$\vdots$	$\vdots$	$\ddots$	$\vdots$	$\vdots$
s	n_{s1}	n_{s2}	$\cdots$	n_{sr}	n_s

Computational Formulas

The following formulas are shown for each population and for all populations combined.

Source	Formula	Dimension
Probability Estimates		
*j*th response	$p_{ij} = \dfrac{n_{ij}}{n_i}$	1×1
*i*th population	$\mathbf{p}_i = \begin{bmatrix} p_{i1} \\ p_{i2} \\ \vdots \\ p_{ir} \end{bmatrix}$	$r \times 1$
all populations	$\mathbf{p} = \begin{bmatrix} \mathbf{p}_1 \\ \mathbf{p}_2 \\ \vdots \\ \mathbf{p}_s \end{bmatrix}$	$sr \times 1$
Variance of Probability Estimates		
*i*th population	$\mathbf{V}_i = \dfrac{1}{n_i}(\mathbf{DIAG}(\mathbf{p}_i) - \mathbf{p}_i\mathbf{p}_i')$	$r \times r$
all populations	$\mathbf{V} = \mathbf{DIAG}(\mathbf{V}_1, \mathbf{V}_2, \ldots, \mathbf{V}_s)$	$sr \times sr$
Response Functions		
*i*th population	$\mathbf{F}_i = \mathbf{F}(\mathbf{p}_i)$	$q \times 1$
all populations	$\mathbf{F} = \begin{bmatrix} \mathbf{F}_1 \\ \mathbf{F}_2 \\ \vdots \\ \mathbf{F}_s \end{bmatrix}$	$sq \times 1$

Source	Formula	Dimension
Derivative of Function with Respect to Probability Estimates		
ith population	$\mathbf{H}_i = \dfrac{\partial \mathbf{F}(\mathbf{p}_i)}{\partial \mathbf{p}_i}$	$q \times r$
all populations	$\mathbf{H} = \mathbf{DIAG}(\mathbf{H}_1, \mathbf{H}_2, \ldots, \mathbf{H}_s)$	$sq \times sr$
Variance of Functions		
ith population	$\mathbf{S}_i = \mathbf{H}_i \mathbf{V}_i \mathbf{H}_i'$	$q \times q$
all populations	$\mathbf{S} = \mathbf{DIAG}(\mathbf{S}_1, \mathbf{S}_2, \ldots, \mathbf{S}_s)$	$sq \times sq$
Inverse Variance of Functions		
ith population	$\mathbf{S}^i = (\mathbf{S}_i)^{-1}$	$q \times q$
all populations	$\mathbf{S}^{-1} = \mathbf{DIAG}(\mathbf{S}^1, \mathbf{S}^2, \ldots, \mathbf{S}^s)$	$sq \times sq$

Derivative Table for Compound Functions: Y=F(G(p))

In the following table, let $\mathbf{G}(\mathbf{p})$ be a vector of functions of $\mathbf{p}$, and let $\mathbf{D}$ denote $\partial \mathbf{G}/\partial \mathbf{p}$, which is the first derivative matrix of $\mathbf{G}$ with respect to $\mathbf{p}$:

Function	$\mathbf{Y} = \mathbf{F}(\mathbf{G})$	Derivative $(\partial \mathbf{Y}/\partial \mathbf{p})$
Multiply matrix	$\mathbf{Y} = \mathbf{A} * \mathbf{G}$	$\mathbf{A} * \mathbf{D}$
Logarithm	$\mathbf{Y} = \mathbf{LOG}(\mathbf{G})$	$\mathbf{DIAG}^{-1}(\mathbf{G}) * \mathbf{D}$
Exponential	$\mathbf{Y} = \mathbf{EXP}(\mathbf{G})$	$\mathbf{DIAG}(\mathbf{Y}) * \mathbf{D}$
Add constant	$\mathbf{Y} = \mathbf{G} + \mathbf{A}$	$\mathbf{D}$

Default Response Functions: Generalized Logits

In the following table, subscripts i for the population are suppressed. Also denote $f_j = \log\left(\dfrac{p_j}{p_r}\right)$ for $j = 1, \ldots, r-1$ for each population $i = 1, \ldots, s$.

Formula	
Inverse of Response Functions for a Population	
$p_j = \dfrac{\exp(f_j)}{1 + \sum_k \exp(f_k)}$	for $j = 1, \ldots, r-1$
$p_r = \dfrac{1}{1 + \sum_k \exp(f_k)}$	

Formula
Form of F and Derivative for a Population
$\mathbf{F} = \mathbf{KLOG(p)} = (\mathbf{I}_{r-1}, -\mathbf{j})\ \mathbf{LOG(p)}$
$\mathbf{H} = \frac{\partial \mathbf{F}}{\partial \mathbf{p}} = \left(\mathbf{DIAG}_{r-1}^{-1}(\mathbf{p}), \frac{-1}{p_r}\mathbf{j}\right)$
Covariance Results for a Population
$\mathbf{S} = \mathbf{HVH}'$
$= \frac{1}{n}\left(\mathbf{DIAG}_{r-1}^{-1}(\mathbf{p}) + \frac{1}{p_r}\mathbf{J}_{r-1}\right)$ where $\mathbf{V}, \mathbf{H},$ and $\mathbf{J}$ are as previously defined.
$\mathbf{S}^{-1} = n(\mathbf{DIAG}_{r-1}(\mathbf{p}) - \mathbf{qq}')$, where $\mathbf{q} = \mathbf{DIAG}_{r-1}(\mathbf{p})\ \mathbf{j}$
$\mathbf{S}^{-1}\mathbf{F} = n\mathbf{DIAG}_{r-1}(\mathbf{p})\mathbf{F} - (n\sum_j p_j f_j)\ \mathbf{q}$
$\mathbf{F}'\mathbf{S}^{-1}\mathbf{F} = n\sum_j p_j f_j^2 - n(\sum_j p_j f_j)^2$

The following calculations are shown for each population and then for all populations combined:

Source	Formula	Dimension
Design Matrix		
ith population	$\mathbf{X}_i$	$q \times d$
all populations	$\mathbf{X} = \begin{bmatrix} \mathbf{X}_1 \\ \mathbf{X}_2 \\ \vdots \\ \mathbf{X}_s \end{bmatrix}$	$sq \times d$
Crossproduct of Design Matrix		
ith population	$\mathbf{C}_i = \mathbf{X}_i{}'\mathbf{S}^i\mathbf{X}_i$	$d \times d$
all populations	$\mathbf{C} = \mathbf{X}'\mathbf{S}^{-1}\mathbf{X} = \sum_i \mathbf{C}_i$	$d \times d$

In the following table, z_p is the 100pth percentile of the standard normal distribution:

Formula	Dimension
Crossproduct of Design Matrix with Function	
$\mathbf{R} = \mathbf{X}'\mathbf{S}^{-1}\mathbf{F} = \sum_i \mathbf{X}_i{}'\mathbf{S}^i\mathbf{F}_i$	$d \times 1$

Formula	Dimension
Weighted Least Squares Estimates	
$\mathbf{b} = \mathbf{C}^{-1}\mathbf{R} = (\mathbf{X}'\mathbf{S}^{-1}\mathbf{X})^{-1}(\mathbf{X}'\mathbf{S}^{-1}\mathbf{F})$	$d \times 1$
Covariance of Weighted Least Squares Estimates	
$\mathbf{COV}(\mathbf{b}) = \mathbf{C}^{-1}$	$d \times d$
Wald Confidence Limits for Parameter Estimates	
$b_k \pm z_{1-\alpha/2}\mathbf{C}^{-1}_{kk}$	$k = 1, \ldots, d$
Predicted Response Functions	
$\hat{\mathbf{F}} = \mathbf{X}\mathbf{b}$	$sq \times 1$
Covariance of Predicted Response Functions	
$\mathbf{V}_{\hat{\mathbf{F}}} = \mathbf{X}\mathbf{C}^{-1}\mathbf{X}'$	$sq \times sq$
Residual Chi-Square	
$\mathrm{RSS} = \mathbf{F}'\mathbf{S}^{-1}\mathbf{F} - \hat{\mathbf{F}}'\mathbf{S}^{-1}\hat{\mathbf{F}}$	1×1
Chi-Square for H_0: $\mathbf{L}\boldsymbol{\beta} = \mathbf{0}$	
$\mathrm{Q} = (\mathbf{L}\mathbf{b})'(\mathbf{L}\mathbf{C}^{-1}\mathbf{L}')^{-1}(\mathbf{L}\mathbf{b})$	1×1

Maximum Likelihood Method

Let $\mathbf{C}$ be the Hessian matrix and $\mathbf{G}$ be the gradient of the log-likelihood function (both functions of $\boldsymbol{\pi}$ and the parameters $\boldsymbol{\beta}$). Let $\mathbf{p}_i^*$ denote the vector containing the first $r - 1$ sample proportions from population i, and let $\boldsymbol{\pi}_i^*$ denote the corresponding vector of probability estimates from the current iteration. Starting with the least squares estimates $\mathbf{b_0}$ of $\boldsymbol{\beta}$ (if you use the ML and WLS options; with the ML option alone, the procedure starts with $\mathbf{0}$), the probabilities $\boldsymbol{\pi}(\mathbf{b})$ are computed, and $\mathbf{b}$ is calculated iteratively by the Newton-Raphson method until it converges (see the EPSILON= option). The factor λ is a step-halving factor that equals one at the start of each iteration. For any iteration in which the likelihood decreases, PROC CATMOD uses a series of subiterations in which λ is iteratively divided by two. The subiterations continue until the likelihood is greater than that of the previous iteration. If the likelihood has not reached that point after 10 subiterations, then convergence is assumed, and a warning message is displayed.

Sometimes, infinite parameters are present in the model, either because of the presence of one or more zero frequencies or because of a poorly specified model with collinearity among the estimates. If an estimate is tending toward infinity, then PROC CATMOD flags the parameter as infinite and holds the estimate fixed in subsequent iterations. PROC CATMOD regards a parameter to be infinite when two conditions apply:

- The absolute value of its estimate exceeds five divided by the range of the corresponding variable.
- The standard error of its estimate is at least three times greater than the estimate itself.

The estimator of the asymptotic covariance matrix of the maximum likelihood predicted probabilities is given by Imrey, Koch, and Stokes (1981, eq. 2.18).

The following equations summarize the method:

$$\mathbf{b}_{k+1} = \mathbf{b}_k - \lambda \mathbf{C}^{-1}\mathbf{G}$$

where

$$\begin{aligned} \mathbf{C} &= \mathbf{X}'\mathbf{S}^{-1}(\boldsymbol{\pi})\mathbf{X} \\ \mathbf{N} &= \begin{bmatrix} n_1(\mathbf{p}_1^* - \boldsymbol{\pi}_1^*) \\ \vdots \\ n_s(\mathbf{p}_s^* - \boldsymbol{\pi}_s^*) \end{bmatrix} \\ \mathbf{G} &= \mathbf{X}'\mathbf{N} \end{aligned}$$

Iterative Proportional Fitting

The algorithm used by PROC CATMOD for iterative proportional fitting is described in Bishop, Fienberg, and Holland (1975); Haberman (1972); Agresti (2002). To illustrate the method, consider the observed three-dimensional table $\{n_{ijk}\}$ for the variables X, Y, and Z, and the following hierarchical model:

$$\log(m_{ijk}) = \mu + \lambda_i^X + \lambda_j^Y + \lambda_k^Z + \lambda_{ij}^{XY} + \lambda_{ik}^{XZ} + \lambda_{jk}^{YZ}$$

The following statements request that PROC CATMOD use IPF to fit the preceding model:

```
model X*Y*Z = _response_ / ml=ipf;
loglin X|Y|Z@2;
```

Begin with a table of initial cell estimates $\{\hat{m}_{ijk}^{(0)}\}$. PROC CATMOD produces the initial estimates by setting the n_{sz} structural zero cells to 0 and all other cells to $n/(n_c - n_{sz})$, where n is the total weight of the table and n_c is the total number of cells in the table. Iteratively adjust the estimates at step $s-1$ to the observed marginal tables specified in the model by cycling through the following three-stage process to produce the estimates at step s:

$$\begin{aligned} \hat{m}_{ijk}^{(s_1)} &= \hat{m}_{ijk}^{(s-1)} \frac{n_{ij\cdot}}{\hat{m}_{ij\cdot}^{(s-1)}} \\ \hat{m}_{ijk}^{(s_2)} &= \hat{m}_{ijk}^{(s_1)} \frac{n_{i\cdot k}}{\hat{m}_{i\cdot k}^{(s_1)}} \\ \hat{m}_{ijk}^{(s)} &= \hat{m}_{ijk}^{(s_2)} \frac{n_{\cdot jk}}{\hat{m}_{\cdot jk}^{(s_2)}} \end{aligned}$$

The subscript "·" indicates summation over the missing subscript. The log-likelihood l_s is estimated at each step s by

$$l_s = \sum_{i,j,k} n_{ijk} \log\left(\frac{\hat{m}_{ijk}^{(s)}}{n}\right)$$

When the function $|(l_{s-1} - l_s)/l_{s-1}|$ is less than 10^{-8}, the iterations terminate. You can change the comparison value with the EPSILON= option, and you can change the convergence criterion with the CONVCRIT= option. The option CONVCRIT=CELL uses the maximum cell difference

$$\max_{i,j,k} |\hat{m}_{ijk}^{(s-1)} - \hat{m}_{ijk}^{(s)}|$$

as the criterion while the option CONVCRIT=MARGIN computes the maximum difference of the margins

$$\text{Maximum of } \left\{ \max_{i,j} |\hat{m}_{ij\cdot}^{(s-1)} - \hat{m}_{ij\cdot}^{(s)}|, \max_{i,k} |\hat{m}_{i\cdot k}^{(s-1)} - \hat{m}_{i\cdot k}^{(s)}|, \max_{j,k} |\hat{m}_{\cdot jk}^{(s-1)} - \hat{m}_{\cdot jk}^{(s)}| \right\}$$

Memory and Time Requirements

The memory and time required by PROC CATMOD are proportional to the number of parameters in the model.

Displayed Output

PROC CATMOD displays the following information in the "Data Summary" table:

- the response effect
- the weight variable, if one is specified
- the data set name
- the number of response levels
- the number of samples or populations
- the total frequency, which is the total sample size
- the number of observations from the data set (the number of data records)
- the frequency of missing observations, labeled as "Frequency Missing"

Except for the analysis of variance table, all of the following items can be displayed or suppressed, depending on your specification of statements and *options*.

- The ONEWAY option produces the "One-Way Frequencies" table, which displays the frequencies of each variable value used in the analysis.

- The populations (or samples) are defined in a table labeled "Population Profiles." The sample size and the values of the defining variables are displayed for each sample. This table is suppressed if the NOPROFILE option is specified.
- The observed responses are defined in a table labeled "Response Profiles." The values of the defining variables are displayed for each response. This table is suppressed if the NOPROFILE option is specified.
- If the FREQ option is specified, then the "Response Frequencies" table is displayed, which shows the frequency of each response for each population.
- If the PROB option is specified, then the "Response Probabilities" table is produced. This table displays the probability of each response for each population.
- If the COV option is specified, the "Response Functions, Covariance Matrix" table, which shows the covariance matrix of the response functions for each sample, is displayed.
- If the DESIGN option is specified, the response functions are displayed in the "Response Functions, Design Matrix" table. If the COV option is also specified, the response functions are displayed in the "Response Functions, Covariance Matrix" table.
- If the DESIGN option is specified, the design matrix is displayed in the "Response Functions, Design Matrix" table, and if a log-linear model is being fit, the _RESPONSE_ matrix is displayed in the "_Response_ Matrix" table. If the model type is AVERAGED, then the design matrix is displayed with $q * s$ rows, assuming q response functions for each of s populations. Otherwise, the design matrix is displayed with only s rows since the model is the same for each of the q response functions.
- The "X′*Inv(S)*X" matrix is displayed for weighted least squares analyses if the XPX option is specified.
- The "Analysis of Variance" table for the weighted least squares analysis reports the results of significance tests for each of the *design-effects* on the right side of the MODEL statement. If _RESPONSE_ is a *design-effect* and is defined explicitly in the LOGLIN, FACTORS, or REPEATED statement, then the table contains test statistics for the individual effects constituting the _RESPONSE_ effect. If the design matrix is input directly, then the content of the displayed output depends on whether you specify any subsets of the parameters to be tested. If you specify one or more subsets, then the table contains one test for each subset. Otherwise, the table contains one test for the effect MODEL | MEAN. In every case, the table also contains the residual goodness-of-fit test. Produced for each test of significance are the source of variation, the number of degrees of freedom (DF), the Wald chi-square value, and the significance probability (Pr > ChiSq).
- The "Analysis of Weighted Least Squares Estimates" table lists, for each parameter in the model, the least squares estimate, the estimated standard error of the parameter estimate, the Wald chi-square value (calculated as $((\text{parameter estimate})/(\text{standard error}))^2$) for testing that the parameter is zero, and the significance probability (Pr > ChiSq) of the test. If the CLPARM option is specified, then 95% Wald confidence intervals are displayed.

  Each row in the table is labeled with the parameter (the model effect and the class levels) and the response function number; however, if the NOPREDVAR option or a REPEATED or FACTORS statement is specified or if the design matrix is directly input, the rows are labeled by the effect in the model for which parameters are formed and the parameter number.

- The "Covariance Matrix of the Parameter Estimates" table for the weighted least squares analysis displays the estimated covariance matrix of the least squares estimates of the parameters, provided that the COVB option is specified.

- The "Correlation Matrix of the Parameter Estimates" table for the weighted least squares analysis displays the estimated correlation matrix of the least squares estimates of the parameters, provided that the CORRB option is specified.

- The "Maximum Likelihood Analysis" table is produced when the ML and ITPRINT options are specified for the standard response functions. It displays the iteration number, the number of step-halving sub-iterations, –2 log likelihood for that iteration, the convergence criterion, and the parameter estimates for each iteration.

- The "Maximum Likelihood Analysis of Variance" table, displayed when the ML option is specified for the standard response functions, is similar to the table produced for the least squares analysis. The Wald chi-square test for each effect is based on the information matrix from the likelihood calculations. The likelihood ratio statistic compares the specified model with the unrestricted (saturated) model and is an appropriate goodness-of-fit test for the model.

- The "Analysis of Maximum Likelihood Estimates" table, displayed when the ML option is specified for the standard response functions, is similar to the one produced for the least squares analysis. The table includes the maximum likelihood estimates, the estimated standard errors based on the information matrix, and the Wald chi-square statistics based on estimated standard errors.

- The "Covariance Matrix of the Maximum Likelihood Estimates" table displays the estimated covariance matrix of the maximum likelihood estimates of the parameters, provided that the COVB and ML options are specified for the standard response functions.

- The "Correlation Matrix of the Maximum Likelihood Estimates" table displays the estimated correlation matrix of the maximum likelihood estimates of the parameters, provided that the CORRB and ML options are specified for the standard response functions.

- For each source of variation specified in a CONTRAST statement, the "Contrasts" table lists the label for the source (Contrast), the number of degrees of freedom (DF), the Wald chi-square value, and the significance probability (Pr > ChiSq). If the ESTIMATE= option is specified, the "Analysis of Contrasts" table displays, for each row of the contrast, the label (Contrast), the type (PARM or EXP), the row of the contrast, the estimate and its standard error, a Wald confidence interval, the Wald chi-square, and the *p*-value (Pr > ChiSq) for 1 degree of freedom.

- Specification of the PREDICT option in the MODEL statement has the following effect. Produced for each response function within each population are the observed and predicted function values, their standard errors, and the residual (observed minus predicted). If the response functions are the default ones (generalized logits), additional information displayed for each response within each population includes the observed and predicted cell probabilities, their standard errors, and the residual. However, specifying PRED=FREQ in the MODEL statement results in the display of the predicted cell frequencies rather than the predicted cell probabilities. The displayed output includes the population profiles and, for the response function table, the function number, while the probability and frequency tables display the response profiles. If the NOPREDVAR option is specified in the MODEL statement, the population profiles are replaced with the sample numbers, and the response profiles are replaced with the labels "P*n*" for the *n*th cell probability, and "F*n*" for the *n*th cell frequency.

- When there are multiple RESPONSE statements, the output for each statement starts on a new page. For each RESPONSE statement, the corresponding title, if specified, is displayed at the top of each page.
- If the ADDCELL= option is specified in the MODEL statement, and if there is a weighted least squares analysis specified, the adjusted sample size for each population (with number added to each cell) is labeled "Adjusted Sample Size" in the "Population Profiles" table. Similarly, the adjusted response frequencies and probabilities are displayed in the "Adjusted Response Frequencies" and "Adjusted Response Probabilities" tables, respectively.
- If _RESPONSE_ is defined explicitly in the LOGLIN, FACTORS, or REPEATED statement, then the definition is displayed as a note whenever _RESPONSE_ appears in the output.

ODS Table Names

PROC CATMOD assigns a name to each table it creates. You can use these names to reference the table when using the Output Delivery System (ODS) to select tables and create output data sets. These names are listed in the following table. For more information about ODS, see Chapter 20, "Using the Output Delivery System."

Table 33.12 ODS Tables Produced by PROC CATMOD

ODS Table Name	Description	Statement	Option
ANOVA	Analysis of variance	MODEL	default
Contrasts	Contrasts	CONTRAST	default
ContrastEstimates	Analysis of contrasts	CONTRAST	ESTIMATE=
ConvergenceStatus	Convergence status	MODEL	ML
CorrB	Correlation matrix of the estimates	MODEL	CORRB
CovB	Covariance matrix of the estimates	MODEL	COVB
DataSummary	Data summary	PROC	default
Estimates	Analysis of estimates	MODEL	default, unless NOPARM
MaxLikelihood	Maximum likelihood analysis	MODEL	ML and ITPRINT
OneWayFreqs	One-way frequencies	MODEL	ONEWAY
PopProfiles	Population profiles	MODEL	default, unless NOPROFILE
PredictedFreqs	Predicted frequencies	MODEL	PRED=FREQ
PredictedProbs	Predicted probabilities	MODEL	PREDICT or PRED=PROB
PredictedValues	Predicted values	MODEL	PREDICT or PRED=
ResponseCov	Response functions, covariance matrix	MODEL	COV
ResponseDesign	Response functions, design matrix	MODEL	DESIGN, unless NODESIGN
ResponseFreqs	Response frequencies	MODEL	FREQ

Table 33.12 (continued)

ODS Table Name	Description	Statement	Option
ResponseMatrix	_RESPONSE_ matrix	MODEL and LOGLIN	DESIGN, unless NORESPONSE
ResponseProbs	Response probabilities	MODEL	PROB
ResponseProfiles	Response profiles	MODEL	default, unless NOPROFILE
XPX	**X**'*Inv(**S**)***X** matrix	MODEL	XPX, for WLS*

* WLS estimation is the default for response functions other than the default (generalized logits).

Examples: CATMOD Procedure

Example 33.1: Linear Response Function, r=2 Responses

In an example from Ries and Smith (1963), the choice of detergent brand (Brand = M or X) is related to three other categorical variables: the softness of the laundry water (Softness = soft, medium, or hard), the temperature of the water (Temperature = high or low), and whether the subject was a previous user of Brand M (Previous = yes or no). The linear response function, which could also be specified as RESPONSE MARGINALS, yields one probability, Pr(brand preference=M), as the response function to be analyzed. Two models are fit in this example: the first model is a saturated one, containing all of the main effects and interactions, while the second is a reduced model containing only the main effects. The following statements produce Output 33.1.1 through Output 33.1.4:

```
data detergent;
   input Softness $ Brand $ Previous $ Temperature $ Count @@;
   datalines;
soft X yes high 19   soft X yes low 57
soft X no  high 29   soft X no  low 63
soft M yes high 29   soft M yes low 49
soft M no  high 27   soft M no  low 53
med  X yes high 23   med  X yes low 47
med  X no  high 33   med  X no  low 66
med  M yes high 47   med  M yes low 55
med  M no  high 23   med  M no  low 50
hard X yes high 24   hard X yes low 37
hard X no  high 42   hard X no  low 68
hard M yes high 43   hard M yes low 52
hard M no  high 30   hard M no  low 42
;

title 'Detergent Preference Study';
proc catmod data=detergent;
   response 1 0;
   weight Count;
   model Brand=Softness|Previous|Temperature / freq prob;
   title2 'Saturated Model';
run;
```

The "Data Summary" table (Output 33.1.1) indicates that you have two response levels and twelve populations.

Output 33.1.1 Detergent Preference Study: Linear Model Analysis

Detergent Preference Study
Saturated Model

The CATMOD Procedure

Data Summary			
Response	Brand	**Response Levels**	2
Weight Variable	Count	**Populations**	12
Data Set	DETERGENT	**Total Frequency**	1008
Frequency Missing	0	**Observations**	24

The "Population Profiles" table in Output 33.1.2 displays the ordering of independent variable levels as used in the table of parameter estimates.

Output 33.1.2 Population Profiles

Population Profiles				
Sample	**Softness**	**Previous**	**Temperature**	**Sample Size**
1	hard	no	high	72
2	hard	no	low	110
3	hard	yes	high	67
4	hard	yes	low	89
5	med	no	high	56
6	med	no	low	116
7	med	yes	high	70
8	med	yes	low	102
9	soft	no	high	56
10	soft	no	low	116
11	soft	yes	high	48
12	soft	yes	low	106

Since Brand M is the first level in the "Response Profiles" table (Output 33.1.3), the RESPONSE statement causes Pr(Brand=M) to be the single response function modeled.

Output 33.1.3 Response Profiles, Frequencies, and Probabilities

Response Profiles	
Response	**Brand**
1	M
2	X

Output 33.1.3 *continued*

Response Frequencies

	Response Number	
Sample	**1**	**2**
1	30	42
2	42	68
3	43	24
4	52	37
5	23	33
6	50	66
7	47	23
8	55	47
9	27	29
10	53	63
11	29	19
12	49	57

Response Probabilities

	Response Number	
Sample	**1**	**2**
1	0.41667	0.58333
2	0.38182	0.61818
3	0.64179	0.35821
4	0.58427	0.41573
5	0.41071	0.58929
6	0.43103	0.56897
7	0.67143	0.32857
8	0.53922	0.46078
9	0.48214	0.51786
10	0.45690	0.54310
11	0.60417	0.39583
12	0.46226	0.53774

The "Analysis of Variance" table in Output 33.1.4 shows that all of the interactions are nonsignificant.

Output 33.1.4 Analysis of Variance

Analysis of Variance			
Source	**DF**	**Chi-Square**	**Pr > ChiSq**
Intercept	1	983.13	<.0001
Softness	2	0.09	0.9575
Previous	1	22.68	<.0001
Softness*Previous	2	3.85	0.1457
Temperature	1	3.67	0.0555
Softness*Temperature	2	0.23	0.8914
Previous*Temperature	1	2.26	0.1324
Softnes*Previou*Temperat	2	0.76	0.6850
Residual	0	.	.

Therefore, a main-effects model is fit with the following statements:

```
    model Brand=Softness Previous Temperature
        / clparm noprofile design;
    title2 'Main-Effects Model';
run;
quit;
```

The PROC CATMOD statement is not required due to the interactive capability of the CATMOD procedure. The NOPROFILE option suppresses the redisplay of the "Response Profiles" table. The CLPARM option produces 95% confidence limits for the parameter estimates. Output 33.1.5 through Output 33.1.7 are produced.

The design matrix in Output 33.1.5 displays the results of the differential-effects modeling used in PROC CATMOD.

Output 33.1.5 Main-Effects Design Matrix

Detergent Preference Study
Main-Effects Model

The CATMOD Procedure

Data Summary			
Response	Brand	**Response Levels**	2
Weight Variable	Count	**Populations**	12
Data Set	DETERGENT	**Total Frequency**	1008
Frequency Missing	0	**Observations**	24

Output 33.1.5 *continued*

Response Functions and Design Matrix

		Design Matrix				
Sample	Response Function	1	2	3	4	5
1	0.41667	1	1	0	1	1
2	0.38182	1	1	0	1	-1
3	0.64179	1	1	0	-1	1
4	0.58427	1	1	0	-1	-1
5	0.41071	1	0	1	1	1
6	0.43103	1	0	1	1	-1
7	0.67143	1	0	1	-1	1
8	0.53922	1	0	1	-1	-1
9	0.48214	1	-1	-1	1	1
10	0.45690	1	-1	-1	1	-1
11	0.60417	1	-1	-1	-1	1
12	0.46226	1	-1	-1	-1	-1

The analysis of variance table in Output 33.1.6 shows that previous use of Brand M, together with the temperature of the laundry water, is a significant factor in whether a subject prefers Brand M laundry detergent. The table also shows that the additive model fits since the goodness-of-fit statistic (the residual chi-square) is nonsignificant.

Output 33.1.6 ANOVA Table for the Main-Effects Model

Analysis of Variance

Source	DF	Chi-Square	Pr > ChiSq
Intercept	1	1004.93	<.0001
Softness	2	0.24	0.8859
Previous	1	20.96	<.0001
Temperature	1	3.95	0.0468
Residual	7	8.26	0.3100

The chi-square test in Output 33.1.7 shows that the Softness parameters are not significantly different from zero; as expected, the Wald confidence limits for these two estimates contain zero. So softness of the water is not a factor in choosing Brand M.

Output 33.1.7 WLS Estimates for the Main-Effects Model

Analysis of Weighted Least Squares Estimates

Parameter		Estimate	Standard Error	Chi-Square	Pr > ChiSq	95% Confidence Limits	
Intercept		0.5080	0.0160	1004.93	<.0001	0.4766	0.5394
Softness	hard	-0.00256	0.0218	0.01	0.9066	-0.0454	0.0402
	med	0.0104	0.0218	0.23	0.6342	-0.0323	0.0530
Previous	no	-0.0711	0.0155	20.96	<.0001	-0.1015	-0.0407
Temperature	high	0.0319	0.0161	3.95	0.0468	0.000446	0.0634

The negative coefficient for Previous (–0.0711) indicates that the first level of Previous (which is shown to be 'no') is associated with a smaller probability of preferring Brand M than the second level of Previous (with coefficient constrained to be 0.0711 since the parameter estimates for a given effect must sum to zero). In other words, previous users of Brand M are much more likely to prefer it than those who have never used it before.

Similarly, the positive coefficient for Temperature indicates that the first level of Temperature (which, from the "Population Profiles" table, is 'high') has a larger probability of preferring Brand M than the second level of Temperature. In other words, those who do their laundry in hot water are more likely to prefer Brand M than those who do their laundry in cold water.

Example 33.2: Mean Score Response Function, r=3 Responses

Four surgical operations for duodenal ulcers are compared in a clinical trial at four hospitals. The operations performed are as follows: Treatment = a, drainage and vagotomy; Treatment = b, 25% resection and vagotomy; Treatment = c, 50% resection and vagotomy; and Treatment = d, 75% resection. The response is severity of an undesirable complication called "dumping syndrome." The data in the following statements are from Grizzle, Starmer, and Koch (1969, pp. 489–504).

```
data operate;
   input Hospital Treatment $ Severity $ wt @@;
   datalines;
1 a none 23    1 a slight  7    1 a moderate 2
1 b none 23    1 b slight 10    1 b moderate 5
1 c none 20    1 c slight 13    1 c moderate 5
1 d none 24    1 d slight 10    1 d moderate 6
2 a none 18    2 a slight  6    2 a moderate 1
2 b none 18    2 b slight  6    2 b moderate 2
2 c none 13    2 c slight 13    2 c moderate 2
2 d none  9    2 d slight 15    2 d moderate 2
3 a none  8    3 a slight  6    3 a moderate 3
3 b none 12    3 b slight  4    3 b moderate 4
3 c none 11    3 c slight  6    3 c moderate 2
3 d none  7    3 d slight  7    3 d moderate 4
4 a none 12    4 a slight  9    4 a moderate 1
4 b none 15    4 b slight  3    4 b moderate 2
4 c none 14    4 c slight  8    4 c moderate 3
4 d none 13    4 d slight  6    4 d moderate 4
;
```

The response variable (Severity) is ordinally scaled with three levels, so assignment of scores is appropriate (0 = none, 0.5 = slight, 1 = moderate). For these scores, the response function yields the mean score. The following statements produce Output 33.2.1 through Output 33.2.3:

```
title 'Dumping Syndrome Data';
proc catmod data=operate order=data ;
   weight wt;
   response 0  0.5  1;
   model Severity=Treatment Hospital / freq oneway design;
   title2 'Main-Effects Model';
quit;
```

The ORDER= option is specified so that the levels of the response variable remain in the correct order. A main-effects model is fit. The ONEWAY option produces a table of the number of subjects within each variable level, and the FREQ option displays the frequency of each response within each sample (Output 33.2.1).

Output 33.2.1 Surgical Data: Analysis of Mean Scores

Dumping Syndrome Data
Main-Effects Model

The CATMOD Procedure

Data Summary			
Response	Severity	**Response Levels**	3
Weight Variable	wt	**Populations**	16
Data Set	OPERATE	**Total Frequency**	417
Frequency Missing	0	**Observations**	48

One-Way Frequencies		
Variable	**Value**	**Frequency**
Severity	none	240
	slight	129
	moderate	48
Treatment	a	96
	b	104
	c	110
	d	107
Hospital	1	148
	2	105
	3	74
	4	90

Population Profiles			
Sample	**Treatment**	**Hospital**	**Sample Size**
1	a	1	32
2	a	2	25
3	a	3	17
4	a	4	22
5	b	1	38
6	b	2	26
7	b	3	20
8	b	4	20
9	c	1	38
10	c	2	28
11	c	3	19
12	c	4	25
13	d	1	40
14	d	2	26
15	d	3	18
16	d	4	23

Output 33.2.1 *continued*

Response Profiles	
Response	**Severity**
1	none
2	slight
3	moderate

Response Frequencies			
	Response Number		
Sample	**1**	**2**	**3**
1	23	7	2
2	18	6	1
3	8	6	3
4	12	9	1
5	23	10	5
6	18	6	2
7	12	4	4
8	15	3	2
9	20	13	5
10	13	13	2
11	11	6	2
12	14	8	3
13	24	10	6
14	9	15	2
15	7	7	4
16	13	6	4

You can use the one-way frequencies and the response profiles from Output 33.2.1 to verify that the response levels are in the desired order (none, slight, moderate) so that the response scores (0, 0.5, 1.0) are applied appropriately. If the ORDER=DATA option had not been used, the levels would have been in a different order.

The analysis of variance table (Output 33.2.2) shows that the additive model fits (since the residual chi-square is not significant), that the Treatment effect is significant, and that the Hospital effect is not significant.

Output 33.2.2 Surgical Data: Analysis of Mean Scores

Response Functions and Design Matrix

Sample	Response Function	Design Matrix 1	2	3	4	5	6	7
1	0.17188	1	1	0	0	1	0	0
2	0.16000	1	1	0	0	0	1	0
3	0.35294	1	1	0	0	0	0	1
4	0.25000	1	1	0	0	-1	-1	-1
5	0.26316	1	0	1	0	1	0	0
6	0.19231	1	0	1	0	0	1	0
7	0.30000	1	0	1	0	0	0	1
8	0.17500	1	0	1	0	-1	-1	-1
9	0.30263	1	0	0	1	1	0	0
10	0.30357	1	0	0	1	0	1	0
11	0.26316	1	0	0	1	0	0	1
12	0.28000	1	0	0	1	-1	-1	-1
13	0.27500	1	-1	-1	-1	1	0	0
14	0.36538	1	-1	-1	-1	0	1	0
15	0.41667	1	-1	-1	-1	0	0	1
16	0.30435	1	-1	-1	-1	-1	-1	-1

Analysis of Variance

Source	DF	Chi-Square	Pr > ChiSq
Intercept	1	248.77	<.0001
Treatment	3	8.90	0.0307
Hospital	3	2.33	0.5065
Residual	9	6.33	0.7069

The coefficients of Treatment in Output 33.2.3 show that the first two treatments (with negative coefficients) have lower mean scores than the last two treatments (the fourth coefficient, not shown, must be positive since the four coefficients must sum to zero). In other words, the less severe treatments (the first two) cause significantly less severe dumping syndrome complications.

Output 33.2.3 Surgical Data: Analysis of Mean Scores

Analysis of Weighted Least Squares Estimates

Parameter		Estimate	Standard Error	Chi-Square	Pr > ChiSq
Intercept		0.2724	0.0173	248.77	<.0001
Treatment	**a**	-0.0552	0.0270	4.17	0.0411
	b	-0.0365	0.0289	1.59	0.2073
	c	0.0248	0.0280	0.78	0.3757
Hospital	**1**	-0.0204	0.0264	0.60	0.4388
	2	-0.0178	0.0268	0.44	0.5055
	3	0.0531	0.0352	2.28	0.1312

Example 33.3: Logistic Regression, Standard Response Function

In this data set, from Cox and Snell (1989), ingots are prepared with different heating and soaking times and tested for their readiness to be rolled. The following DATA step creates a response variable Y with value 1 for ingots that are not ready and value 0 otherwise. The explanatory variables are Heat and Soak.

```
data ingots;
   input Heat Soak nready ntotal @@;
   Count=nready;
   Y=1;
   output;
   Count=ntotal-nready;
   Y=0;
   output;
   drop nready ntotal;
   datalines;
7 1.0 0 10   14 1.0 0 31   27 1.0 1 56   51 1.0 3 13
7 1.7 0 17   14 1.7 0 43   27 1.7 4 44   51 1.7 0  1
7 2.2 0  7   14 2.2 2 33   27 2.2 0 21   51 2.2 0  1
7 2.8 0 12   14 2.8 0 31   27 2.8 1 22   51 4.0 0  1
7 4.0 0  9   14 4.0 0 19   27 4.0 1 16
;
```

Logistic regression analysis is often used to investigate the relationship between discrete response variables and continuous explanatory variables. For logistic regression, the continuous *design-effects* are declared in a DIRECT statement. The following statements produce Output 33.3.1 through Output 33.3.6:

```
title 'Maximum Likelihood Logistic Regression';
proc catmod data=ingots;
   weight Count;
   direct Heat Soak;
   model Y=Heat Soak / freq covb corrb itprint design;
quit;
```

You can verify that the populations are defined as you intended by looking at the "Population Profiles" table in Output 33.3.1.

Output 33.3.1 Maximum Likelihood Logistic Regression

Maximum Likelihood Logistic Regression

The CATMOD Procedure

Data Summary			
Response	Y	**Response Levels**	2
Weight Variable	Count	**Populations**	19
Data Set	INGOTS	**Total Frequency**	387
Frequency Missing	0	**Observations**	25

Output 33.3.1 *continued*

Population Profiles			
Sample	**Heat**	**Soak**	**Sample Size**
1	7	1	10
2	7	1.7	17
3	7	2.2	7
4	7	2.8	12
5	7	4	9
6	14	1	31
7	14	1.7	43
8	14	2.2	33
9	14	2.8	31
10	14	4	19
11	27	1	56
12	27	1.7	44
13	27	2.2	21
14	27	2.8	22
15	27	4	16
16	51	1	13
17	51	1.7	1
18	51	2.2	1
19	51	4	1

Since the "Response Profiles" table in Output 33.3.2 shows the response level ordering as 0, 1, the default response function, the logit, is defined as $\log\left(\frac{p_{Y=0}}{p_{Y=1}}\right)$.

Output 33.3.2 Response Summaries

Response Profiles	
Response	**Y**
1	0
2	1

Output 33.3.2 *continued*

Response Frequencies		
	Response Number	
Sample	1	2
1	10	0
2	17	0
3	7	0
4	12	0
5	9	0
6	31	0
7	43	0
8	31	2
9	31	0
10	19	0
11	55	1
12	40	4
13	21	0
14	21	1
15	15	1
16	10	3
17	1	0
18	1	0
19	1	0

The values of the continuous variable are inserted into the design matrix (Output 33.3.3).

Output 33.3.3 Design Matrix

Response Functions and Design Matrix				
		Design Matrix		
Sample	Response Function	1	2	3
1	2.99573	1	7	1
2	3.52636	1	7	1.7
3	2.63906	1	7	2.2
4	3.17805	1	7	2.8
5	2.89037	1	7	4
6	4.12713	1	14	1
7	4.45435	1	14	1.7
8	2.74084	1	14	2.2
9	4.12713	1	14	2.8
10	3.63759	1	14	4
11	4.00733	1	27	1
12	2.30259	1	27	1.7
13	3.73767	1	27	2.2
14	3.04452	1	27	2.8
15	2.70805	1	27	4
16	1.20397	1	51	1
17	0.69315	1	51	1.7
18	0.69315	1	51	2.2
19	0.69315	1	51	4

Seven Newton-Raphson iterations are required to find the maximum likelihood estimates (Output 33.3.4).

Output 33.3.4 Iteration History

Maximum Likelihood Analysis						
				Parameter Estimates		
Iteration	Sub Iteration	-2 Log Likelihood	Convergence Criterion	1	2	3
0	0	536.49592	1.0000	0	0	0
1	0	152.58961	0.7156	2.1594	-0.0139	-0.003733
2	0	106.76066	0.3003	3.5334	-0.0363	-0.0120
3	0	96.692171	0.0943	4.7489	-0.0640	-0.0299
4	0	95.383825	0.0135	5.4138	-0.0790	-0.0498
5	0	95.345659	0.000400	5.5539	-0.0819	-0.0564
6	0	95.345613	4.8289E-7	5.5592	-0.0820	-0.0568
7	0	95.345613	7.732E-13	5.5592	-0.0820	-0.0568

Maximum likelihood computations converged.

The analysis of variance table (Output 33.3.5) shows that the model fits since the likelihood ratio goodness-of-fit test is nonsignificant. It also shows that the length of heating time is a significant factor with respect to readiness but that length of soaking time is not.

Output 33.3.5 Analysis of Variance Table

Maximum Likelihood Analysis of Variance			
Source	**DF**	**Chi-Square**	**Pr > ChiSq**
Intercept	1	24.65	<.0001
Heat	1	11.95	0.0005
Soak	1	0.03	0.8639
Likelihood Ratio	16	13.75	0.6171

From the table of maximum likelihood estimates in Output 33.3.6, the fitted model is

$$\mathrm{E}(\mathrm{logit}(p)) = 5.559 - 0.082(\mathrm{Heat}) - 0.057(\mathrm{Soak})$$

For example, for Sample 1 with Heat = 7 and Soak = 1, the estimate is

$$\mathrm{E}(\mathrm{logit}(p)) = 5.559 - 0.082(7) - 0.057(1) = 4.9284$$

Output 33.3.6 Maximum Likelihood Estimates, Covariances, and Correlations

Analysis of Maximum Likelihood Estimates				
Parameter	**Estimate**	**Standard Error**	**Chi-Square**	**Pr > ChiSq**
Intercept	5.5592	1.1197	24.65	<.0001
Heat	-0.0820	0.0237	11.95	0.0005
Soak	-0.0568	0.3312	0.03	0.8639

Covariance Matrix of the Maximum Likelihood Estimates				
Row	**Parameter**	**Col1**	**Col2**	**Col3**
1	Intercept	1.2537133	-0.0215664	-0.2817648
2	Heat	-0.0215664	0.0005633	0.0026243
3	Soak	-0.2817648	0.0026243	0.1097020

Correlation Matrix of the Maximum Likelihood Estimates				
Row	**Parameter**	**Col1**	**Col2**	**Col3**
1	Intercept	1.00000	-0.81152	-0.75977
2	Heat	-0.81152	1.00000	0.33383
3	Soak	-0.75977	0.33383	1.00000

Predicted values of the logits, as well as the probabilities of readiness, could be obtained by specifying PRED=PROB in the MODEL statement. For the example of Sample 1 with Heat = 7 and Soak = 1, PRED=PROB would give an estimate of the probability of readiness equal to 0.9928 since

$$4.9284 = \log\left(\frac{\hat{p}}{1-\hat{p}}\right)$$

implies that

$$\hat{p} = \frac{e^{4.9284}}{1+e^{4.9284}} = 0.9928$$

As another consideration, since soaking time is nonsignificant, you could fit another model that deleted the variable Soak.

Example 33.4: Log-Linear Model, Three Dependent Variables

This analysis reproduces the predicted cell frequencies for Bartlett's data by using a log-linear model of no three-variable interaction (Bishop, Fienberg, and Holland 1975, p. 89). Cuttings of two different lengths (Length=short or long) are planted at one of two time points (Time=now or spring), and their survival status (Status=dead or alive) is recorded.

As in the text, the variable levels are simply labeled 1 and 2. The following statements produce Output 33.4.1 through Output 33.4.3:

```
data bartlett;
   input Length Time Status wt @@;
   datalines;
1 1 1 156     1 1 2  84     1 2 1 84     1 2 2 156
2 1 1 107     2 1 2 133     2 2 1 31     2 2 2 209
;

title 'Bartlett''s Data';
proc catmod data=bartlett;
   weight wt;
   model Length*Time*Status=_response_
         / noparm pred=freq;
   loglin Length|Time|Status @ 2;
   title2 'Model with No 3-Variable Interaction';
quit;
```

Output 33.4.1 Analysis of Bartlett's Data: Log-Linear Model

Bartlett's Data
Model with No 3-Variable Interaction

The CATMOD Procedure

Data Summary			
Response	Length*Time*Status	**Response Levels**	8
Weight Variable	wt	**Populations**	1
Data Set	BARTLETT	**Total Frequency**	960
Frequency Missing	0	**Observations**	8

Population Profiles	
Sample	**Sample Size**
1	960

Output 33.4.1 *continued*

Response Profiles			
Response	**Length**	**Time**	**Status**
1	1	1	1
2	1	1	2
3	1	2	1
4	1	2	2
5	2	1	1
6	2	1	2
7	2	2	1
8	2	2	2

Maximum Likelihood Analysis
Maximum likelihood computations converged.

Maximum Likelihood Analysis of Variance			
Source	**DF**	**Chi-Square**	**Pr > ChiSq**
Length	1	2.64	0.1041
Time	1	5.25	0.0220
Length*Time	1	5.25	0.0220
Status	1	48.94	<.0001
Length*Status	1	48.94	<.0001
Time*Status	1	95.01	<.0001
Likelihood Ratio	1	2.29	0.1299

The analysis of variance table shows that the model fits since the likelihood ratio test for the three-variable interaction is nonsignificant. All of the two-variable interactions, however, are significant; this shows that there is mutual dependence among all three variables.

The predicted values table (Output 33.4.2) displays observed and predicted values for the generalized logits.

Output 33.4.2 Response Function Predicted Values

Maximum Likelihood Predicted Values for Response Functions					
	Observed		**Predicted**		
Function Number	**Function**	**Standard Error**	**Function**	**Standard Error**	**Residual**
1	-0.29248	0.105806	-0.23565	0.098486	-0.05683
2	-0.91152	0.129188	-0.94942	0.129948	0.037901
3	-0.91152	0.129188	-0.94942	0.129948	0.037901
4	-0.29248	0.105806	-0.23565	0.098486	-0.05683
5	-0.66951	0.118872	-0.69362	0.120172	0.024113
6	-0.45199	0.110921	-0.3897	0.102267	-0.06229
7	-1.90835	0.192465	-1.73146	0.142969	-0.17688

The predicted frequencies table (Output 33.4.3) displays observed and predicted cell frequencies, their standard errors, and residuals.

Output 33.4.3 Predicted Frequencies

Maximum Likelihood Predicted Values for Frequencies

			Observed		Predicted		
Length	Time	Status	Frequency	Standard Error	Frequency	Standard Error	Residual
1	1	1	156	11.43022	161.0961	11.07379	-5.09614
1	1	2	84	8.754999	78.90386	7.808613	5.096139
1	2	1	84	8.754999	78.90386	7.808613	5.096139
1	2	2	156	11.43022	161.0961	11.07379	-5.09614
2	1	1	107	9.750588	101.9039	8.924304	5.096139
2	1	2	133	10.70392	138.0961	10.33434	-5.09614
2	2	1	31	5.47713	36.09614	4.826315	-5.09614
2	2	2	209	12.78667	203.9039	12.21285	5.09614

Example 33.5: Log-Linear Model, Structural and Sampling Zeros

This example illustrates a log-linear model of independence, by using data that contain structural zero frequencies as well as sampling (random) zero frequencies.

In a population of six squirrel monkeys, the joint distribution of genital display with respect to active or passive role was observed. The data are from Fienberg (1980, Table 8-2). Since a monkey cannot have both the active and passive roles in the same interaction, the diagonal cells of the table are structural zeros. See Agresti (2002) for more information about the quasi-independence model.

The DATA step replaces the structural zeros with missing values, and the MISSING=STRUCTURAL option is specified in the MODEL statement to remove these zeros from the analysis. The ZERO=SAMPLING option treats the off-diagonal zeros as sampling zeros. Also, the row for Monkey 't' is deleted since it contains all zeros; therefore, the cell frequencies predicted by a model of independence are also zero. In addition, the CONTRAST statement compares the behavior of the two monkeys labeled 'u' and 'v'. See the section "Structural and Sampling Zeros with Raw Data" on page 2219 for information about how to perform this analysis when you have raw data. The following statements produce Output 33.5.1 through Output 33.5.8:

```
data Display;
   input Active $ Passive $ wt @@;
   if Active ne 't';
   if Active eq Passive then wt=.;
   datalines;
r r  0   r s  1   r t  5   r u  8   r v  9   r w  0
s r 29   s s  0   s t 14   s u 46   s v  4   s w  0
t r  0   t s  0   t t  0   t u  0   t v  0   t w  0
u r  2   u s  3   u t  1   u u  0   u v 38   u w  2
v r  0   v s  0   v t  0   v u  0   v v  0   v w  1
w r  9   w s 25   w t  4   w u  6   w v 13   w w  0
;

title 'Behavior of Squirrel Monkeys';
proc catmod data=Display;
   weight wt;
   model Active*Passive=_response_ /
         missing=structural zero=sampling
```

```
      freq pred=freq noparm oneway;
   loglin Active Passive;
   contrast 'Passive, U vs. V' Passive 0 0 0 1 -1;
   contrast 'Active,  U vs. V' Active  0 0 1 -1;
   title2 'Test Quasi-Independence for the Incomplete Table';
quit;
```

Output 33.5.1 Log-Linear Model Analysis with Zero Frequencies

Behavior of Squirrel Monkeys
Test Quasi-Independence for the Incomplete Table

The CATMOD Procedure

Data Summary			
Response	Active*Passive	**Response Levels**	25
Weight Variable	wt	**Populations**	1
Data Set	DISPLAY	**Total Frequency**	220
Frequency Missing	0	**Observations**	25

The results of the ONEWAY option are shown in Output 33.5.2. Monkey 't' does not show up as a value for the Active variable since that row was removed.

Output 33.5.2 Output from the ONEWAY option

One-Way Frequencies		
Variable	**Value**	**Frequency**
Active	r	23
	s	93
	u	46
	v	1
	w	57
Passive	r	40
	s	29
	t	24
	u	60
	v	64
	w	3

Sampling zeros are displayed as 0 in Output 33.5.4. The Response Number column corresponds to the value displayed in the "Response Profiles" table in Output 33.5.3.

Output 33.5.3 Profiles

Population Profiles	
Sample	**Sample Size**
1	220

Output 33.5.3 *continued*

Response Profiles		
Response	**Active**	**Passive**
1	r	s
2	r	t
3	r	u
4	r	v
5	r	w
6	s	r
7	s	t
8	s	u
9	s	v
10	s	w
11	u	r
12	u	s
13	u	t
14	u	v
15	u	w
16	v	r
17	v	s
18	v	t
19	v	u
20	v	w
21	w	r
22	w	s
23	w	t
24	w	u
25	w	v

Output 33.5.4 Frequency of Response by Response Number

Response Frequencies																									
										Response Number															
Sample	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**13**	**14**	**15**	**16**	**17**	**18**	**19**	**20**	**21**	**22**	**23**	**24**	**25**
1	1	5	8	9	0	29	14	46	4	0	2	3	1	38	2	0	0	0	0	1	9	25	4	6	13

The analysis of variance table (Output 33.5.5) shows that the model of independence does not fit since the likelihood ratio test for the interaction is significant. In other words, active and passive behaviors of the squirrel monkeys are dependent behavior roles.

Output 33.5.5 Analysis of Variance Table

Maximum Likelihood Analysis of Variance			
Source	**DF**	**Chi-Square**	**Pr > ChiSq**
Active	4	56.58	<.0001
Passive	5	47.94	<.0001
Likelihood Ratio	15	135.17	<.0001

If the model fit these data, then the contrasts in Output 33.5.6 show that monkeys ‘u’ and ‘v’ appear to have similar passive behavior patterns but very different active behavior patterns.

Output 33.5.6 Contrasts between Monkeys ‘u’ and ‘v’

Contrasts of Maximum Likelihood Estimates

Contrast	DF	Chi-Square	Pr > ChiSq
Passive, U vs. V	1	1.31	0.2524
Active, U vs. V	1	14.87	0.0001

Output 33.5.7 displays the predicted response functions and Output 33.5.8 displays predicted cell frequencies (from the PRED=FREQ option), but since the model does not fit, these should be ignored. Note that, since the response function is the generalized logit with the 25th response as the baseline, the observed response functions for the sampling zeros are missing.

Output 33.5.7 Response Function Predicted Values

Maximum Likelihood Predicted Values for Response Functions

	Observed		Predicted		
Function Number	Function	Standard Error	Function	Standard Error	Residual
1	-2.56495	1.037749	-0.97355	0.339019	-1.5914
2	-0.95551	0.526235	-1.72504	0.345438	0.769529
3	-0.48551	0.449359	-0.52751	0.309254	0.042007
4	-0.36772	0.433629	-0.73927	0.249006	0.371543
5	.	.	-3.56052	0.634104	.
6	0.802346	0.333775	0.320589	0.26629	0.481758
7	0.074108	0.385164	-0.29934	0.295634	0.37345
8	1.263692	0.314105	0.898184	0.250857	0.365508
9	-1.17865	0.571772	0.686431	0.173396	-1.86509
10	.	.	-2.13482	0.608071	.
11	-1.8718	0.759555	-0.2415	0.287218	-1.63031
12	-1.46634	0.640513	-0.10994	0.303568	-1.3564
13	-2.56495	1.037749	-0.86143	0.314794	-1.70352
14	1.072637	0.321308	0.124346	0.204345	0.94829
15	-1.8718	0.759555	-2.6969	0.617433	0.8251
16	.	.	-4.14787	1.024508	.
17	.	.	-4.01632	1.030062	.
18	.	.	-4.76781	1.032457	.
19	.	.	-3.57028	1.020794	.
20	-2.56495	1.037749	-6.60328	1.161289	4.038332
21	-0.36772	0.433629	-0.36584	0.202959	-0.00188
22	0.653926	0.34194	-0.23429	0.232794	0.888212
23	-1.17865	0.571772	-0.98577	0.239408	-0.19288
24	-0.77319	0.493548	0.211754	0.185007	-0.98494

Output 33.5.8 Predicted Frequencies

Maximum Likelihood Predicted Values for Frequencies

		Observed		Predicted		
Active	Passive	Frequency	Standard Error	Frequency	Standard Error	Residual
r	s	1	0.997725	5.259508	1.36156	-4.25951
r	t	5	2.210512	2.480726	0.691066	2.519274
r	u	8	2.776525	8.215948	1.855146	-0.21595
r	v	9	2.937996	6.648049	1.50932	2.351951
r	w	0	0	0.395769	0.240268	-0.39577
s	r	29	5.017696	19.18599	3.147915	9.814007
s	t	14	3.620648	10.32172	2.169599	3.678284
s	u	46	6.031734	34.18463	4.428706	11.81537
s	v	4	1.981735	27.66096	3.722788	-23.661
s	w	0	0	1.6467	0.952712	-1.6467
u	r	2	1.407771	10.9364	2.12322	-8.9364
u	s	3	1.720201	12.47407	2.554336	-9.47407
u	t	1	0.997725	5.883583	1.380655	-4.88358
u	v	38	5.606814	15.7673	2.684692	22.2327
u	w	2	1.407771	0.938652	0.551645	1.061348
v	r	0	0	0.219966	0.221779	-0.21997
v	s	0	0	0.250893	0.253706	-0.25089
v	t	0	0	0.118338	0.120314	-0.11834
v	u	0	0	0.391924	0.393255	-0.39192
v	w	1	0.997725	0.018879	0.021728	0.981121
w	r	9	2.937996	9.657645	1.808656	-0.65765
w	s	25	4.707344	11.01553	2.275019	13.98447
w	t	4	1.981735	5.195638	1.184452	-1.19564
w	u	6	2.415857	17.2075	2.772098	-11.2075
w	v	13	3.497402	13.92369	2.24158	-0.92369

Structural and Sampling Zeros with Raw Data

The preceding PROC CATMOD step uses cell count data as input. Prior to invoking the CATMOD procedure, structural and sampling zeros are easily identified and manipulated in a single DATA step. For the situation where structural or sampling zeros (or both) exist and the input data set is raw data, use the following steps:

1. Run PROC FREQ on the raw data (see Chapter 44, "The FREQ Procedure"). In the TABLES statement, list all dependent and independent variables, separated by asterisks, and use the SPARSE option and the OUT= option. This creates an output data set that contains all possible zero frequencies. Since the tabled output can be huge, you should also specify the NOPRINT option in the TABLES statement.

2. Use a DATA step to change the zero frequencies associated with either sampling zeros or structural zeros to missing.

3. Use the resulting data set as input to PROC CATMOD, specify the statement WEIGHT COUNT to use adjusted frequencies, and specify the ZERO= and MISSING= options to define your sampling and structural zeros.

For example, suppose the data set RawDisplay contains the raw data for the squirrel monkey data. The following statements show how to obtain the same analysis as shown previously:

```
proc freq data=RawDisplay;
   tables Active*Passive / sparse out=Combos noprint;
run;

data Combos2;
   set Combos;
   if Active ne 't';
   if Active eq Passive then count=.;
run;

proc catmod data=Combos2;
   weight count;
   model Active*Passive=_response_ /
         zero=sampling missing=structural
         freq pred=freq noparm noresponse;
   loglin Active Passive;
quit;
```

The first IF statement in the DATA step is needed only for this particular example; since observations for Monkey 't' were deleted from the Display data set, they also need to be deleted from Combos2.

Example 33.6: Repeated Measures, 2 Response Levels, 3 Populations

In this multiple-population repeated measures example, from Guthrie (1981), subjects from three groups have their responses (0 or 1) recorded in each of four trials. The analysis of the marginal probabilities is directed at assessing the main effects of the repeated measurement factor (Trial) and the independent variable (Group), as well as their interaction. Although the contingency table is incomplete (only 13 of the 16 possible responses are observed), this poses no problem in the computation of the marginal probabilities. The following statements produce Output 33.6.1:

```
data group;
   input a b c d Group wt @@;
   datalines;
1 1 1 1 2 2     0 0 0 0 2 2     0 0 1 0 1 2     0 0 1 0 2 2
0 0 0 1 1 4     0 0 0 1 2 1     0 0 0 1 3 3     1 0 0 1 2 1
0 0 1 1 1 1     0 0 1 1 2 2     0 0 1 1 3 5     0 1 0 0 1 4
0 1 0 0 2 1     0 1 0 1 2 1     0 1 0 1 3 2     0 1 1 0 3 1
1 0 0 0 1 3     1 0 0 0 2 1     0 1 1 1 2 1     0 1 1 1 3 2
1 0 1 0 1 1     1 0 1 1 2 1     1 0 1 1 3 2
;

title 'Multiple-Population Repeated Measures';
proc catmod data=group;
   weight wt;
   response marginals;
   model a*b*c*d=Group _response_ Group*_response_
         / freq;
   repeated Trial 4;
   title2 'Saturated Model';
run;
```

Output 33.6.1 Analysis of Multiple-Population Repeated Measures

Multiple-Population Repeated Measures
Saturated Model

The CATMOD Procedure

Data Summary			
Response	a*b*c*d	**Response Levels**	13
Weight Variable	wt	**Populations**	3
Data Set	GROUP	**Total Frequency**	45
Frequency Missing	0	**Observations**	23

Population Profiles		
Sample	**Group**	**Sample Size**
1	1	15
2	2	15
3	3	15

Response Profiles				
Response	**a**	**b**	**c**	**d**
1	0	0	0	0
2	0	0	0	1
3	0	0	1	0
4	0	0	1	1
5	0	1	0	0
6	0	1	0	1
7	0	1	1	0
8	0	1	1	1
9	1	0	0	0
10	1	0	0	1
11	1	0	1	0
12	1	0	1	1
13	1	1	1	1

Response Frequencies													
	Response Number												
Sample	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**13**
1	0	4	2	1	4	0	0	0	3	0	1	0	0
2	2	1	2	2	1	1	0	1	1	1	0	1	2
3	0	3	0	5	0	2	1	2	0	0	0	2	0

Analysis of Variance			
Source	**DF**	**Chi-Square**	**Pr > ChiSq**
Intercept	1	354.88	<.0001
Group	2	24.79	<.0001
Trial	3	21.45	<.0001
Group*Trial	6	18.71	0.0047
Residual	0	.	.

Output 33.6.1 *continued*

Analysis of Weighted Least Squares Estimates					
Effect	Parameter	Estimate	Standard Error	Chi-Square	Pr > ChiSq
Intercept	1	0.5833	0.0310	354.88	<.0001
Group	2	0.1333	0.0335	15.88	<.0001
	3	-0.0333	0.0551	0.37	0.5450
Trial	4	0.1722	0.0557	9.57	0.0020
	5	0.1056	0.0647	2.66	0.1028
	6	-0.0722	0.0577	1.57	0.2107
Group*Trial	7	-0.1556	0.0852	3.33	0.0679
	8	-0.0556	0.0800	0.48	0.4877
	9	-0.0889	0.0953	0.87	0.3511
	10	0.0111	0.0866	0.02	0.8979
	11	0.0889	0.0822	1.17	0.2793
	12	-0.0111	0.0824	0.02	0.8927

The analysis of variance table in Output 33.6.1 shows that there is a significant interaction between the independent variable Group and the repeated measurement factor Trial. An intermediate model (not shown) is fit in which the effects Trial and Group* Trial are replaced by Trial(Group=1), Trial(Group=2), and Trial(Group=3). Of these three effects, only the last is significant, so it is retained in the final model. The following statements produce Output 33.6.2 and Output 33.6.3:

```
    model a*b*c*d=Group _response_(Group=3)
          / noprofile noparm design;
    title2 'Trial Nested within Group 3';
quit;
```

Output 33.6.2 displays the design matrix resulting from retaining the nested effect.

Output 33.6.2 Final Model: Design Matrix

Multi-Population Repeated Measures
Trial Nested within Group 3

The CATMOD Procedure

Data Summary			
Response	a*b*c*d	Response Levels	13
Weight Variable	wt	Populations	3
Data Set	GROUP	Total Frequency	45
Frequency Missing	0	Observations	23

Output 33.6.2 *continued*

Response Functions and Design Matrix

			Design Matrix					
Sample	Function Number	Response Function	1	2	3	4	5	6
1	1	0.73333	1	1	0	0	0	0
	2	0.73333	1	1	0	0	0	0
	3	0.73333	1	1	0	0	0	0
	4	0.66667	1	1	0	0	0	0
2	1	0.66667	1	0	1	0	0	0
	2	0.66667	1	0	1	0	0	0
	3	0.46667	1	0	1	0	0	0
	4	0.40000	1	0	1	0	0	0
3	1	0.86667	1	-1	-1	1	0	0
	2	0.66667	1	-1	-1	0	1	0
	3	0.33333	1	-1	-1	0	0	1
	4	0.06667	1	-1	-1	-1	-1	-1

The residual goodness-of-fit statistic tests the joint effect of Trial(Group=1) and Trial(Group=2). The analysis of variance table in Output 33.6.3 shows that the final model fits, that there is a significant Group effect, and that there is a significant Trial effect in Group 3.

Output 33.6.3 ANOVA Table

Analysis of Variance

Source	DF	Chi-Square	Pr > ChiSq
Intercept	1	386.94	<.0001
Group	2	25.42	<.0001
Trial(Group=3)	3	75.07	<.0001
Residual	6	5.09	0.5319

Example 33.7: Repeated Measures, 4 Response Levels, 1 Population

This example illustrates a repeated measures analysis in which there are more than two levels of response. In this study, from Grizzle, Starmer, and Koch (1969, p. 493), 7,477 women aged 30–39 are tested for vision in both right and left eyes. Since there are four response levels for each dependent variable, the RESPONSE statement computes three marginal probabilities for each dependent variable, resulting in six response functions for analysis. Since the model contains a repeated measurement factor (Side) with two levels (Right, Left), PROC CATMOD groups the functions into sets of three (=6/2). Therefore, the Side effect has three degrees of freedom (one for each marginal probability), and it is the appropriate test of marginal homogeneity. The following statements produce Output 33.7.1:

```
title 'Vision Symmetry';
data vision;
   input Right Left count @@;
   datalines;
1 1 1520    1 2  266    1 3  124    1 4  66
2 1  234    2 2 1512    2 3  432    2 4  78
```

```
3 1  117    3 2  362    3 3 1772    3 4 205
4 1   36    4 2   82    4 3  179    4 4 492
;

proc catmod data=vision;
   weight count;
   response marginals;
   model Right*Left=_response_ / freq design;
   repeated Side 2;
   title2 'Test of Marginal Homogeneity';
quit;
```

Output 33.7.1 Vision Study: Analysis of Marginal Homogeneity

Vision Symmetry
Test of Marginal Homogeneity

The CATMOD Procedure

Data Summary			
Response	Right*Left	**Response Levels**	16
Weight Variable	count	**Populations**	1
Data Set	VISION	**Total Frequency**	7477
Frequency Missing	0	**Observations**	16

Population Profiles	
Sample	**Sample Size**
1	7477

Response Profiles		
Response	**Right**	**Left**
1	1	1
2	1	2
3	1	3
4	1	4
5	2	1
6	2	2
7	2	3
8	2	4
9	3	1
10	3	2
11	3	3
12	3	4
13	4	1
14	4	2
15	4	3
16	4	4

Response Frequencies																
	Response Number															
Sample	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**13**	**14**	**15**	**16**
1	1520	266	124	66	234	1512	432	78	117	362	1772	205	36	82	179	492

Output 33.7.1 *continued*

Response Functions and Design Matrix

Sample	Function Number	Response Function	Design Matrix 1	2	3	4	5	6
1	1	0.26428	1	0	0	1	0	0
	2	0.30173	0	1	0	0	1	0
	3	0.32847	0	0	1	0	0	1
	4	0.25505	1	0	0	-1	0	0
	5	0.29718	0	1	0	0	-1	0
	6	0.33529	0	0	1	0	0	-1

Analysis of Variance

Source	DF	Chi-Square	Pr > ChiSq
Intercept	3	78744.17	<.0001
Side	3	11.98	0.0075
Residual	0	.	.

Analysis of Weighted Least Squares Estimates

Effect	Parameter	Estimate	Standard Error	Chi-Square	Pr > ChiSq
Intercept	1	0.2597	0.00468	3073.03	<.0001
	2	0.2995	0.00464	4160.17	<.0001
	3	0.3319	0.00483	4725.25	<.0001
Side	4	0.00461	0.00194	5.65	0.0174
	5	0.00227	0.00255	0.80	0.3726
	6	-0.00341	0.00252	1.83	0.1757

The analysis of variance table in Output 33.7.1 shows that the Side effect is significant, so there is not marginal homogeneity between left-eye vision and right-eye vision. In other words, the distribution of the quality of right-eye vision differs significantly from the distribution of the quality of left-eye vision in the same subjects. The test of the Side effect is equivalent to Bhapkar's test (Agresti 1990) .

Example 33.8: Repeated Measures, Logistic Analysis of Growth Curve

The following data, from a longitudinal study reported in Koch et al. (1977), are from patients in four populations (2 diagnostic groups × 2 treatments) who are measured at three times to assess their response (n=normal or a=abnormal) to treatment:

```
title 'Growth Curve Analysis';
data growth2;
   input Diagnosis $ Treatment $ week1 $ week2 $ week4 $ count @@;
   datalines;
mild std n n n 16    severe std n n n  2
mild std n n a 13    severe std n n a  2
mild std n a n  9    severe std n a n  8
mild std n a a  3    severe std n a a  9
mild std a n n 14    severe std a n n  9
```

```
mild std a n a  4    severe std a n a 15
mild std a a n 15    severe std a a n 27
mild std a a a  6    severe std a a a 28
mild new n n n 31    severe new n n n  7
mild new n n a  0    severe new n n a  2
mild new n a n  6    severe new n a n  5
mild new n a a  0    severe new n a a  2
mild new a n n 22    severe new a n n 31
mild new a n a  2    severe new a n a  5
mild new a a n  9    severe new a a n 32
mild new a a a  0    severe new a a a  6
;
```

The analysis is directed at assessing the effect of the repeated measurement factor, Time, as well as the independent variables, Diagnosis (mild or severe) and Treatment (std or new). The RESPONSE statement is used to compute the logits of the marginal probabilities. The times used in the design matrix (0, 1, 2) correspond to the logarithms (base 2) of the actual times (1, 2, 4). The following statements produce Output 33.8.1 through Output 33.8.4:

```
proc catmod data=growth2 order=data;
   title2 'Reduced Logistic Model';
   weight count;
   population Diagnosis Treatment;
   response logit;
   model week1*week2*week4=(1 0 0 0,  /* mild, std */
                            1 0 1 0,
                            1 0 2 0,

                            1 0 0 0,  /* mild, new */
                            1 0 0 1,
                            1 0 0 2,

                            0 1 0 0,  /* severe, std */
                            0 1 1 0,
                            0 1 2 0,

                            0 1 0 0,  /* severe, new */
                            0 1 0 1,
                            0 1 0 2)
          (1='Mild diagnosis, week 1',
           2='Severe diagnosis, week 1',
           3='Time effect for std trt',
           4='Time effect for new trt')
           / freq design;
   contrast 'Diagnosis effect, week 1' all_parms 1 -1 0 0;
   contrast 'Equal time effects' all_parms 0 0 1 -1;
quit;
```

The samples and the response numbers are defined in Output 33.8.1, and the frequency distribution of the response numbers within the samples is displayed.

Output 33.8.1 Logistic Analysis of Growth Curve

Growth Curve Analysis
Reduced Logistic Model

The CATMOD Procedure

Data Summary			
Response	week1*week2*week4	**Response Levels**	8
Weight Variable	count	**Populations**	4
Data Set	GROWTH2	**Total Frequency**	340
Frequency Missing	0	**Observations**	29

Population Profiles			
Sample	**Diagnosis**	**Treatment**	**Sample Size**
1	mild	std	80
2	mild	new	70
3	severe	std	100
4	severe	new	90

Response Profiles			
Response	**week1**	**week2**	**week4**
1	n	n	n
2	n	n	a
3	n	a	n
4	n	a	a
5	a	n	n
6	a	n	a
7	a	a	n
8	a	a	a

Output 33.8.2 displays the design matrix specified in the MODEL statement, and the observed logits of the marginal probabilities are displayed in the Response Function column.

Output 33.8.2 Response Frequencies

Response Frequencies								
	Response Number							
Sample	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**
1	16	13	9	3	14	4	15	6
2	31	0	6	0	22	2	9	0
3	2	2	8	9	9	15	27	28
4	7	2	5	2	31	5	32	6

Output 33.8.2 *continued*

Response Functions and Design Matrix						
			Design Matrix			
Sample	Function Number	Response Function	1	2	3	4
1	1	0.05001	1	0	0	0
	2	0.35364	1	0	1	0
	3	0.73089	1	0	2	0
2	1	0.11441	1	0	0	0
	2	1.29928	1	0	0	1
	3	3.52636	1	0	0	2
3	1	-1.32493	0	1	0	0
	2	-0.94446	0	1	1	0
	3	-0.16034	0	1	2	0
4	1	-1.53148	0	1	0	0
	2	0.00000	0	1	0	1
	3	1.60944	0	1	0	2

The analysis of variance table in Output 33.8.3 shows that the data can be adequately modeled by two parameters that represent diagnosis effects at week 1 and two log-linear time effects (one for each treatment). Both of the time effects are significant.

Since the estimate of the logit for the severe diagnosis effect (parameter 2) is more negative than it is for the mild diagnosis effect (parameter 1), there is a smaller predicted probability of the first response (normal) for the severe diagnosis group.

Output 33.8.3 ANOVA and Parameter Estimates

Analysis of Variance			
Source	DF	Chi-Square	Pr > ChiSq
Mild diagnosis, week 1	1	0.28	0.5955
Severe diagnosis, week 1	1	100.48	<.0001
Time effect for std trt	1	26.35	<.0001
Time effect for new trt	1	125.09	<.0001
Residual	8	4.20	0.8387

Analysis of Weighted Least Squares Estimates					
Effect	Parameter	Estimate	Standard Error	Chi-Square	Pr > ChiSq
Model	1	-0.0716	0.1348	0.28	0.5955
	2	-1.3529	0.1350	100.48	<.0001
	3	0.4944	0.0963	26.35	<.0001
	4	1.4552	0.1301	125.09	<.0001

The analysis of contrasts (Output 33.8.4) shows that the diagnosis effect at week 1 is highly significant. In other words, those subjects with a severe diagnosis have a significantly higher probability of abnormal response at week 1 than those subjects with a mild diagnosis.

Output 33.8.4 Contrasts

Analysis of Contrasts			
Contrast	**DF**	**Chi-Square**	**Pr > ChiSq**
Diagnosis effect, week 1	1	77.02	<.0001
Equal time effects	1	59.12	<.0001

The analysis of contrasts (Output 33.8.4) also shows that the time effect for the standard treatment is significantly different from the one for the new treatment. The table of parameter estimates (Output 33.8.3) shows that the time effect for the new treatment (parameter 4) is stronger than it is for the standard treatment (parameter 3).

Example 33.9: Repeated Measures, Two Repeated Measurement Factors

This example, from MacMillan et al. (1981), illustrates a repeated measures analysis in which there are two repeated measurement factors. Two diagnostic procedures (standard and test) are performed on each subject, and the results of both are evaluated at each of two times as being positive or negative. In the following DATA step, std1 and std2 are the two measurements of the standard procedure, and test1 and test2 are the two measurements of the test procedure:

```
data a;
   input std1 $ test1 $ std2 $ test2 $ wt @@;
   datalines;
neg neg neg neg 509  neg neg neg pos  4  neg neg pos neg  17
neg neg pos pos   3  neg pos neg neg 13  neg pos neg pos   8
neg pos pos pos   8  pos neg neg neg 14  pos neg neg pos   1
pos neg pos neg  17  pos neg pos pos  9  pos pos neg neg   7
pos pos neg pos   4  pos pos pos neg  9  pos pos pos pos 170
;
```

For the initial model, the response functions are marginal probabilities, and the repeated measurement factors are Time and Treatment. The model is a saturated one, containing effects for Time, Treatment, and Time*Treatment. The following statements produce Output 33.9.1:

```
proc catmod data=a;
   title2 'Marginal Symmetry, Saturated Model';
   weight wt;
   response marginals;
   model std1*test1*std2*test2=_response_ / freq design noparm;
   repeated Time 2, Treatment 2 / _response_=Time Treatment
            Time*Treatment;
run;
```

The analysis of variance table in Output 33.9.1 shows that there is no significant effect of Time, either by itself or in its interaction with Treatment. The second model includes only the Treatment effect. Again, the response functions are marginal probabilities, and the repeated measurement factors are Time and Treatment.

Output 33.9.1 Diagnosis Data: Two Repeated Measurement Factors

Diagnostic Procedure Comparison
Marginal Symmetry, Saturated Model

The CATMOD Procedure

Data Summary

Response	std1*test1*std2*test2	**Response Levels**	15
Weight Variable	wt	**Populations**	1
Data Set	A	**Total Frequency**	793
Frequency Missing	0	**Observations**	15

Population Profiles

Sample	Sample Size
1	793

Response Profiles

Response	std1	test1	std2	test2
1	neg	neg	neg	neg
2	neg	neg	neg	pos
3	neg	neg	pos	neg
4	neg	neg	pos	pos
5	neg	pos	neg	neg
6	neg	pos	neg	pos
7	neg	pos	pos	pos
8	pos	neg	neg	neg
9	pos	neg	neg	pos
10	pos	neg	pos	neg
11	pos	neg	pos	pos
12	pos	pos	neg	neg
13	pos	pos	neg	pos
14	pos	pos	pos	neg
15	pos	pos	pos	pos

Response Frequencies

	Response Number													
Sample	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**13**	**14**	**15**
1	509	4	17	3	13	8	8	14	1	17	9	7	4	9	170

Response Functions and Design Matrix

			Design Matrix			
Sample	**Function Number**	**Response Function**	**1**	**2**	**3**	**4**
1	**1**	0.70870	1	1	1	1
	2	0.72383	1	1	-1	-1
	3	0.70618	1	-1	1	-1
	4	0.73897	1	-1	-1	1

Output 33.9.1 *continued*

Analysis of Variance			
Source	**DF**	**Chi-Square**	**Pr > ChiSq**
Intercept	1	2385.34	<.0001
Time	1	0.85	0.3570
Treatment	1	8.20	0.0042
Time*Treatment	1	2.40	0.1215
Residual	0	.	.

A main effect model with respect to Treatment is fit. The following statements produces Output 33.9.2:

```
    title2 'Marginal Symmetry, Reduced Model';
    model std1*test1*std2*test2=_response_ / corrb design noprofile;
    repeated Time 2, Treatment 2 / _response_=Treatment;
run;
```

The analysis of variance table for the reduced model (Output 33.9.2) shows that the model fits (since the residual chi-square is nonsignificant) and that the treatment effect is significant. The negative parameter estimate for Treatment shows that the first level of treatment (std) has a smaller probability of the first response level (neg) than the second level of treatment (test). In other words, the standard diagnostic procedure gives a significantly higher probability of a positive response than the test diagnostic procedure.

Output 33.9.2 Diagnosis Data: Reduced Model

Diagnostic Procedure Comparison
Marginal Symmetry, Reduced Model

The CATMOD Procedure

Data Summary			
Response	std1*test1*std2*test2	**Response Levels**	15
Weight Variable	wt	**Populations**	1
Data Set	A	**Total Frequency**	793
Frequency Missing	0	**Observations**	15

Response Functions and Design Matrix				
			Design Matrix	
Sample	**Function Number**	**Response Function**	**1**	**2**
1	**1**	0.70870	1	1
	2	0.72383	1	-1
	3	0.70618	1	1
	4	0.73897	1	-1

Analysis of Variance			
Source	**DF**	**Chi-Square**	**Pr > ChiSq**
Intercept	1	2386.97	<.0001
Treatment	1	9.55	0.0020
Residual	2	3.51	0.1731

Output 33.9.2 *continued*

Analysis of Weighted Least Squares Estimates

Effect	Parameter	Estimate	Standard Error	Chi-Square	Pr > ChiSq
Intercept	1	0.7196	0.0147	2386.97	<.0001
Treatment	2	-0.0128	0.00416	9.55	0.0020

Correlation Matrix of the Parameter Estimates

Row	Col1	Col2
1	1.00000	0.04194
2	0.04194	1.00000

The next example illustrates a RESPONSE statement that, at each time, computes the sensitivity and specificity of the test diagnostic procedure with respect to the standard procedure. Since these are measures of the relative accuracy of the two diagnostic procedures, the repeated measurement factors in this case are labeled Time and Accuracy. Only 15 of the 16 possible responses are observed, so additional care must be taken in formulating the RESPONSE statement for computation of sensitivity and specificity.

The following statements produce Output 33.9.3 and Output 33.9.4:

```
      title2 'Sensitivity and Specificity Analysis, '
             'Main-Effects Model';
      model std1*test1*std2*test2=_response_ / covb design noprofile;
      repeated Time 2, Accuracy 2 / _response_=Time Accuracy;
      response exp 1 -1  0  0  0  0  0  0,
                   0  0  1 -1  0  0  0  0,
                   0  0  0  0  1 -1  0  0,
                   0  0  0  0  0  0  1 -1

                log 0 0 0 0   0 0  0   0 0 0 0   1 1 1 1,
                    0 0 0 0   0 0  0   1 1 1 1   1 1 1 1,
                    1 1 1 1   0 0  0   0 0 0 0   0 0 0 0,
                    1 1 1 1   1 1  1   0 0 0 0   0 0 0 0,
                    0 0 0 1   0 0  1   0 0 0 1   0 0 0 1,
                    0 0 1 1   0 0  1   0 0 1 1   0 0 1 1,
                    1 0 0 0   1 0  0   1 0 0 0   1 0 0 0,
                    1 1 0 0   1 1  0   1 1 0 0   1 1 0 0;
quit;
```

For the sensitivity and specificity analysis, the four response functions displayed next to the design matrix (Output 33.9.3) represent the following:

1. sensitivity, time 1
2. specificity, time 1
3. sensitivity, time 2
4. specificity, time 2

The sensitivities and specificities are for the test diagnostic procedure relative to the standard procedure.

Output 33.9.3 Diagnosis Data: Sensitivity and Specificity Analysis

Diagnostic Procedure Comparison
Sensitivity and Specificity Analysis, Main-Effects Model

The CATMOD Procedure

Data Summary			
Response	std1*test1*std2*test2	**Response Levels**	15
Weight Variable	wt	**Populations**	1
Data Set	A	**Total Frequency**	793
Frequency Missing	0	**Observations**	15

Response Functions and Design Matrix

			Design Matrix		
Sample	**Function Number**	**Response Function**	**1**	**2**	**3**
1	**1**	0.82251	1	1	1
	2	0.94840	1	1	-1
	3	0.81545	1	-1	1
	4	0.96964	1	-1	-1

Analysis of Variance

Source	**DF**	**Chi-Square**	**Pr > ChiSq**
Intercept	1	6448.79	<.0001
Time	1	4.10	0.0428
Accuracy	1	38.81	<.0001
Residual	1	1.00	0.3178

The ANOVA table in Output 33.9.3 shows that an additive model fits, that there is a significant effect of time, and that the sensitivity is significantly different from the specificity.

Output 33.9.4 shows that the predicted sensitivities and specificities are lower for time 1 (since parameter 2 is negative). It also shows that the sensitivity is significantly less than the specificity.

Output 33.9.4 Parameter Estimates

Analysis of Weighted Least Squares Estimates

Effect	**Parameter**	**Estimate**	**Standard Error**	**Chi-Square**	**Pr > ChiSq**
Intercept	**1**	0.8892	0.0111	6448.79	<.0001
Time	**2**	-0.00932	0.00460	4.10	0.0428
Accuracy	**3**	-0.0702	0.0113	38.81	<.0001

Output 33.9.4 *continued*

Covariance Matrix of the Parameter Estimates			
Row	Col1	Col2	Col3
1	0.00012260	0.00000229	0.00010137
2	0.00000229	0.00002116	-.00000587
3	0.00010137	-.00000587	0.00012697

Example 33.10: Direct Input of Response Functions and Covariance Matrix

This example illustrates the ability of PROC CATMOD to operate on an existing vector of functions and the corresponding covariance matrix. The estimates under investigation are composite indices summarizing the responses to 18 psychological questions pertaining to general well-being. These estimates are computed for domains corresponding to an age-by-sex cross-classification, and the covariance matrix is calculated using the method of balanced repeated replications. The analysis is directed at obtaining a description of the variation among these domain estimates. The data are from Koch and Stokes (1979).

In the following statements, the first row of the fbeing data set contains the response functions for the variables b1–b10, while the remaining rows contain the covariance matrix. From the PROC CATMOD statements, the READ option in the RESPONSE statement says that you are inputting the response functions and their covariance matrix, while the PROFILE= option in the FACTORS statement tells you that the variables b1–b5 correspond to the effects for sex='male' at the five different age groupings, and b6–b10 likewise correspond to the effects for sex='female'. See the section "Inputting Response Functions and Covariances Directly" on page 2165 for more information about using the READ option.

```
data fbeing(type=est);
   input   b1-b5   _type_ $  _name_ $  b6-b10 #2;
   datalines;
 7.93726   7.92509   7.82815   7.73696   8.16791  parms     .
 7.24978   7.18991   7.35960   7.31937   7.55184
 0.00739   0.00019   0.00146  -0.00082   0.00076  cov       b1
 0.00189   0.00118   0.00140  -0.00140   0.00039
 0.00019   0.01172   0.00183   0.00029   0.00083  cov       b2
-0.00123  -0.00629  -0.00088  -0.00232   0.00034
 0.00146   0.00183   0.01050  -0.00173   0.00011  cov       b3
 0.00434  -0.00059  -0.00055   0.00023  -0.00013
-0.00082   0.00029  -0.00173   0.01335   0.00140  cov       b4
 0.00158   0.00212   0.00211   0.00066   0.00240
 0.00076   0.00083   0.00011   0.00140   0.01430  cov       b5
-0.00050  -0.00098   0.00239  -0.00010   0.00213
 0.00189  -0.00123   0.00434   0.00158  -0.00050  cov       b6
 0.01110   0.00101   0.00177  -0.00018  -0.00082
 0.00118  -0.00629  -0.00059   0.00212  -0.00098  cov       b7
 0.00101   0.02342   0.00144   0.00369   0.00253
 0.00140  -0.00088  -0.00055   0.00211   0.00239  cov       b8
 0.00177   0.00144   0.01060   0.00157   0.00226
-0.00140  -0.00232   0.00023   0.00066  -0.00010  cov       b9
-0.00018   0.00369   0.00157   0.02298   0.00918
 0.00039   0.00034  -0.00013   0.00240   0.00213  cov      b10
-0.00082   0.00253   0.00226   0.00918   0.01921
;
```

The following statements produce Output 33.10.1:

```
proc catmod data=fbeing;
   title 'Complex Sample Survey Analysis';
   response read b1-b10;
   factors sex $ 2, age $ 5 / _response_=sex age
                              profile=(male     '25-34',
                                       male     '35-44',
                                       male     '45-54',
                                       male     '55-64',
                                       male     '65-74',
                                       female   '25-34',
                                       female   '35-44',
                                       female   '45-54',
                                       female   '55-64',
                                       female   '65-74');
   model _f_=_response_
         / design title='Main Effects for Sex and Age';
run;
```

Output 33.10.1 Health Survey Data: Using Direct Input

Complex Sample Survey Analysis

Main Effects for Sex and Age

The CATMOD Procedure

Response Functions and Design Matrix

Sample	Function Number	Response Function	Design Matrix 1	2	3	4	5	6
1	1	7.93726	1	1	1	0	0	0
	2	7.92509	1	1	0	1	0	0
	3	7.82815	1	1	0	0	1	0
	4	7.73696	1	1	0	0	0	1
	5	8.16791	1	1	-1	-1	-1	-1
	6	7.24978	1	-1	1	0	0	0
	7	7.18991	1	-1	0	1	0	0
	8	7.35960	1	-1	0	0	1	0
	9	7.31937	1	-1	0	0	0	1
	10	7.55184	1	-1	-1	-1	-1	-1

Analysis of Variance

Source	DF	Chi-Square	Pr > ChiSq
Intercept	1	28089.07	<.0001
sex	1	65.84	<.0001
age	4	9.21	0.0561
Residual	4	2.92	0.5713

Output 33.10.1 *continued*

Analysis of Weighted Least Squares Estimates					
Effect	Parameter	Estimate	Standard Error	Chi-Square	Pr > ChiSq
Intercept	1	7.6319	0.0455	28089.07	<.0001
sex	2	0.2900	0.0357	65.84	<.0001
age	3	-0.00780	0.0645	0.01	0.9037
	4	-0.0465	0.0636	0.54	0.4642
	5	-0.0343	0.0557	0.38	0.5387
	6	-0.1098	0.0764	2.07	0.1506

The analysis of variance table in Output 33.10.1 shows that the additive model fits and that there is a significant effect of both sex and age. The following statements produce Output 33.10.2:

```
    contrast 'No Age Effect for Age<65' all_parms 0 0 1 0 0 -1,
                                        all_parms 0 0 0 1 0 -1,
                                        all_parms 0 0 0 0 1 -1;
run;
```

The analysis of the contrast shows that there is no significant difference among the four age groups that are under age 65.

Output 33.10.2 Health Survey Data: Age<65 Contrast

Complex Sample Survey Analysis

Main Effects for Sex and Age

The CATMOD Procedure

Analysis of Contrasts			
Contrast	DF	Chi-Square	Pr > ChiSq
No Age Effect for Age<65	3	0.72	0.8678

The next model contains a binary age effect (under 65 versus 65 and over). The following statements produce Output 33.10.3:

```
    model _f_=(1  1  1,
               1  1  1,
               1  1  1,
               1  1  1,
               1  1 -1,
               1 -1  1,
               1 -1  1,
               1 -1  1,
               1 -1  1,
               1 -1 -1)
                       (1='Intercept' ,
                        2='Sex'       ,
                        3='Age (25-64 vs. 65-74)')
          / design title='Binary Age Effect (25-64 vs. 65-74)' ;
run;
quit;
```

Output 33.10.3 Health Survey Data: Age<65 Model

Complex Sample Survey Analysis

Binary Age Effect (25-64 vs. 65-74)

The CATMOD Procedure

Response Functions and Design Matrix

Sample	Function Number	Response Function	Design Matrix 1	2	3
1	1	7.93726	1	1	1
	2	7.92509	1	1	1
	3	7.82815	1	1	1
	4	7.73696	1	1	1
	5	8.16791	1	1	-1
	6	7.24978	1	-1	1
	7	7.18991	1	-1	1
	8	7.35960	1	-1	1
	9	7.31937	1	-1	1
	10	7.55184	1	-1	-1

Analysis of Variance

Source	DF	Chi-Square	Pr > ChiSq
Intercept	1	19087.16	<.0001
Sex	1	72.64	<.0001
Age (25-64 vs. 65-74)	1	8.49	0.0036
Residual	7	3.64	0.8198

Analysis of Weighted Least Squares Estimates

Effect	Parameter	Estimate	Standard Error	Chi-Square	Pr > ChiSq
Model	1	7.7183	0.0559	19087.16	<.0001
	2	0.2800	0.0329	72.64	<.0001
	3	-0.1304	0.0448	8.49	0.0036

The analysis of variance table in Output 33.10.3 shows that the model fits (note that the goodness-of-fit statistic is the sum of the previous one (Output 33.10.1) plus the chi-square for the contrast matrix in Output 33.10.2). The age and sex effects are significant. Since the second parameter in the table of estimates is positive, males (the first level for the sex variable) have a higher predicted index of well-being than females. Since the third parameter estimate is negative, those younger than age 65 (the first level of age) have a lower predicted index of well-being than those 65 and older.

Example 33.11: Predicted Probabilities

Suppose you have collected marketing research data to examine the relationship between a prospect's likelihood of buying your product and the person's education and income. Specifically, the variables are as follows:

Variable	Levels	Interpretation
Education	high, low	Prospect's education level
Income	high, low	Prospect's income level
Purchase	yes, no	Did prospect purchase product?

The following statements first create a data set, loan, that contains the marketing research data. Then the CATMOD procedure fits a model, obtains the parameter estimates, and obtains the predicted probabilities of interest. These statements produce Output 33.11.1 and Output 33.11.2.

```
data loan;
   input Education $ Income $ Purchase $ wt;
   datalines;
high  high  yes    54
high  high  no     23
high  low   yes    41
high  low   no     12
low   high  yes    35
low   high  no     42
low   low   yes    19
low   low   no      8
;

ods output PredictedValues=Predicted (keep=Education Income PredFunction);
proc catmod data=loan order=data;
   weight wt;
   response marginals;
   model Purchase=Education Income / pred design;
run;

proc sort data=Predicted;
   by descending PredFunction;
run;
proc print data=Predicted;
run;
```

Notice that the preceding statements use the Output Delivery System (ODS) to output the parameter estimates instead of the OUT= option, though either can be used.

Output 33.11.1 Marketing Research Data: Obtaining Predicted Probabilities

The CATMOD Procedure

Data Summary			
Response	Purchase	**Response Levels**	2
Weight Variable	wt	**Populations**	4
Data Set	LOAN	**Total Frequency**	234
Frequency Missing	0	**Observations**	8

Output 33.11.1 *continued*

Population Profiles

Sample	Education	Income	Sample Size
1	high	high	77
2	high	low	53
3	low	high	77
4	low	low	27

Response Profiles

Response	Purchase
1	yes
2	no

Response Functions and Design Matrix

		Design Matrix		
Sample	Response Function	1	2	3
1	0.70130	1	1	1
2	0.77358	1	1	-1
3	0.45455	1	-1	1
4	0.70370	1	-1	-1

Analysis of Variance

Source	DF	Chi-Square	Pr > ChiSq
Intercept	1	418.36	<.0001
Education	1	8.85	0.0029
Income	1	4.70	0.0302
Residual	1	1.84	0.1745

Analysis of Weighted Least Squares Estimates

Parameter		Estimate	Standard Error	Chi-Square	Pr > ChiSq
Intercept		0.6481	0.0317	418.36	<.0001
Education	**high**	0.0924	0.0311	8.85	0.0029
Income	**high**	-0.0675	0.0312	4.70	0.0302

Predicted Values for Response Functions

			Observed		Predicted		
Education	Income	Function Number	Function	Standard Error	Function	Standard Error	Residual
high	**high**	**1**	0.701299	0.052158	0.67294	0.047794	0.028359
high	**low**	**1**	0.773585	0.057487	0.808034	0.051586	-0.03445
low	**high**	**1**	0.454545	0.056744	0.48811	0.051077	-0.03356
low	**low**	**1**	0.703704	0.087877	0.623204	0.064867	0.080499

Output 33.11.2 Predicted Probabilities Data Set

Obs	Education	Income	PredFunction
1	high	low	0.808034
2	high	high	0.67294
3	low	low	0.623204
4	low	high	0.48811

You can use the predicted values (values of PredFunction in Output 33.11.2) as scores representing the likelihood that a randomly chosen subject from one of these populations will purchase the product. Notice that the "Response Profiles" table in Output 33.11.1 shows you that the first sorted level of Purchase is 'yes', indicating that the predicted probabilities are for Pr(Purchase='yes'). For example, someone with high education and low income has an estimated probability of purchase of 0.808. Like any response function estimate given by PROC CATMOD, this estimate can be obtained by cross-multiplying the row from the design matrix corresponding to the sample (sample number 2 in this case) with the vector of parameter estimates: $(1 * 0.6481) + (1 * 0.0924) + (-1 * (-0.0675))$.

This ranking of scores can help in decision making (for example, with respect to allocation of advertising dollars, choice of advertising media, choice of print media, and so on).

References

Agresti, A. (1984). *Analysis of Ordinal Categorical Data*. New York: John Wiley & Sons.

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

Agresti, A. (1996). *An Introduction to Categorical Data Analysis*. New York: John Wiley & Sons.

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

Bishop, Y. M. M., Fienberg, S. E., and Holland, P. W. (1975). *Discrete Multivariate Analysis: Theory and Practice*. Cambridge, MA: MIT Press.

Christensen, R. (1997). *Log-Linear Models and Logistic Regression*. 2nd ed. New York: Springer-Verlag.

Cox, D. R., and Snell, E. J. (1989). *The Analysis of Binary Data*. 2nd ed. London: Chapman & Hall.

Fienberg, S. E. (1980). *The Analysis of Cross-Classified Categorical Data*. 2nd ed. Cambridge, MA: MIT Press.

Forthofer, R. N., and Koch, G. G. (1973). "An Analysis of Compounded Functions of Categorical Data." *Biometrics* 29:143–157.

Forthofer, R. N., and Lehnen, R. G. (1981). *Public Program Analysis: A New Categorical Data Approach*. Belmont, CA: Wadsworth.

Freeman, D. H., Jr. (1987). *Applied Categorical Data Analysis*. New York: Marcel Dekker.

Grizzle, J. E., Starmer, C. F., and Koch, G. G. (1969). "Analysis of Categorical Data by Linear Models." *Biometrics* 25:489–504.

Guthrie, D. (1981). "Analysis of Dichotomous Variables in Repeated Measures Experiments." *Psychological Bulletin* 90:189–195.

Haberman, S. J. (1972). "Algorithm AS 51: Log-Linear Fit for Contingency Tables." *Journal of the Royal Statistical Society, Series C* 21:218–225.

Haslett, S. (1990). "Degrees of Freedom and Parameter Estimability in Hierarchical Models for Sparse Complete Contingency Tables." *Computational Statistics and Data Analysis* 9:179–195.

Imrey, P. B., Koch, G. G., and Stokes, M. E. (1981). "Categorical Data Analysis: Some Reflections on the Log Linear Model and Logistic Regression, Part I: Historical and Methodological Overview." *International Statistical Review* 49:265–283.

Koch, G. G., Landis, J. R., Freeman, J. L., Freeman, D. H., and Lehnen, R. G. (1977). "A General Methodology for the Analysis of Experiments with Repeated Measurement of Categorical Data." *Biometrics* 33:133–158.

Koch, G. G., and Stokes, M. E. (1979). *Annotated Computer Applications of Weighted Least Squares Methods for Illustrative Analyses of Examples Involving Health Survey Data*. Technical report, prepared for the U.S. National Center for Health Studies.

Landis, J. R., Stanish, W. M., Freeman, J. L., and Koch, G. G. (1976). "A Computer Program for the Generalized Chi-Square Analysis of Categorical Data Using Weighted Least Squares (GENCAT)." *Computer Programs in Biomedicine* 6:196–231.

MacMillan, J., Becker, C., Koch, G. G., Stokes, M. E., and Vandiviere, H. M. (1981). "An Application of Weighted Least Squares Methods to the Analysis of Measurement Process Components of Variability in an Observational Study." In *Proceedings of Survey Research Methods*, 680–685. Alexandria, VA: American Statistical Association.

Ries, P. N., and Smith, H. (1963). "The Use of Chi-Square for Preference Testing in Multidimensional Problems." *Chemical Engineering Progress* 59:39–43.

Searle, S. R. (1971). "Topics in Variance Component Estimation." *Biometrics* 26:1–76.

Stanish, W. M., and Koch, G. G. (1984). "The Use of CATMOD for Repeated Measurement Analysis of Categorical Data." In *Proceedings of the Ninth Annual SAS Users Group International Conference*, 761–770. Cary, NC: SAS Institute Inc. http://www.sascommunity.org/sugi/SUGI84/Sugi-84-142%20Stanish%20Koch.pdf.

Stokes, M. E., Davis, C. S., and Koch, G. G. (2000). *Categorical Data Analysis Using the SAS System*. 2nd ed. Cary, NC: SAS Institute Inc.

Wald, A. (1943). "Tests of Statistical Hypotheses Concerning Several Parameters When the Number of Observations Is Large." *Transactions of the American Mathematical Society* 54:426–482.

Subject Index

Syntax Index