Usage Note *24447:* Examples of writing CONTRAST and ESTIMATE statements

Details | About | Rate It

Examples of Writing CONTRAST and ESTIMATE Statements

Printing this document: *Because some of the tables in this document are wide, you might need to print it in landscape mode to avoid truncation of the right edge.*

Introduction

To properly test a hypothesis such as "The effect of treatment A in group 1 is equal to the treatment A effect in group 2," it is necessary to translate it correctly into a mathematical hypothesis using the fitted model. Specifically, you need to construct the linear combination of model parameters that corresponds to the hypothesis. Such linear combinations can be estimated and tested using the CONTRAST and/or ESTIMATE statements available in many modeling procedures. While only certain procedures are illustrated below, this discussion applies to any modeling procedure that allows these statements.

Though assisting with the translation of a stated hypothesis into the needed linear combination is beyond the scope of the services that are provided by Technical Support at SAS, we hope that the following discussion and examples will help you. Technical Support can assist you with syntax and other questions that relate to CONTRAST and ESTIMATE statements.

As you'll see in the examples that follow, there are some important steps in properly writing a CONTRAST or ESTIMATE statement:

- Write down the model that you are using the procedure to fit. There are two crucial parts to this:

  Parameterization
  : How design variables that are generated by the CLASS statement are coded. The two most commonly used parameterizations that are available in SAS are *indicator* (or *dummy*) coding and *effects* (or *deviation from mean*) coding.

  Parameter Ordering
  : The order of the parameters within effects that have multiple parameters (such as CLASS variables and interactions of CLASS variables). The ordering typically depends on the order in which the variables are specified in the CLASS statement and the setting of the ORDER= option in the PROC or CLASS statement. Use the Parameter Estimates or Class Level Information table in the modeling procedure's displayed results to confirm the parameter order.

- Write down the hypothesis to be tested or quantity to be estimated in terms of the model's parameters and simplify. When testing, write the null hypothesis in the form *contrast = 0* before simplifying the left-hand side. For example, to compare two means, specify the null hypothesis as $\mu_1 - \mu_2 = 0$ and then write $\mu_1 - \mu_2$ in terms of the model parameters.

- Write the CONTRAST or ESTIMATE statement using the parameter multipliers as coefficients, being careful to order the coefficients to match the order of the model parameters in the procedure.

Writing CONTRAST and ESTIMATE statements can become difficult when interaction or nested effects are part of the model. The following examples concentrate on using the steps above in this situation. In the simpler case of a main-effects-only model, writing CONTRAST and ESTIMATE statements to make simple pairwise comparisons is more intuitive. Examples of this simpler situation can be found in the example titled "Randomized Complete Blocks with Means Comparisons and Contrasts" in the PROC GLM documentation and in this note which uses PROC GENMOD.

The CONTRAST and ESTIMATE statements allow for estimation and testing of any linear combination of model parameters. However, a common subclass of interest involves comparison of means and most of the examples below are from this class. While examples in this class provide good examples of the above process for determining coefficients for CONTRAST and ESTIMATE statements, there are other statements available that perform means comparisons more easily. These statements include the LSMEANS, LSMESTIMATE, and SLICE statements that are available beginning with SAS/STAT 9.22 in SAS 9.2 TS2M3 in many procedures. The ODDSRATIO statement in PROC LOGISTIC and the similar HAZARDRATIO statement in PROC PHREG are also available. While the main purpose of this note is to illustrate how to write proper CONTRAST and ESTIMATE statements, these additional statements are also presented when they can provide equivalent analyses.

Example 1: A Two-Factor Model with Interaction

Consider a model for two factors: A with five levels and B with two levels:

$$Y_{ijk} = \mu + \alpha_i + \beta_j + \alpha\beta_{ij} + \varepsilon_{ijk} \quad (1)$$

where i=1,2,...,5, j=1,2, k=1, 2,...,n_{ij} . The response, Y, is normally distributed with constant variance. The statements below generate observations from such a model:

```
data test;
   seed=6342454;
   do a=1 to 5;
      do b=1 to 2;
         do rep=1 to ceil(ranuni(seed)*5)+5;
            y=5 + a + b + a*b + rannor(seed);
            output;
         end;
      end;
   end;
run;
```

The following statements fit the main effects and interaction model. The LSMEANS statement computes the cell means for the 10 A*B cells in this example. The E option shows how each cell mean is formed by displaying the coefficient vectors that are used in calculating the LS-means.

```
proc mixed data=test;
   class a b;
   model y=a b a*b / solution;
   lsmeans a*b / e;
run;
```

Least Squares Means

Effect	a	b	Estimate	Standard Error	DF	t Value	Pr > |t|
a*b	1	1	7.5193	0.2905	74	25.88	<.0001
a*b	1	2	10.0341	0.2598	74	38.62	<.0001
a*b	2	1	10.4189	0.2739	74	38.04	<.0001
a*b	2	2	12.5812	0.3355	74	37.50	<.0001
a*b	3	1	11.5853	0.3355	74	34.54	<.0001
a*b	3	2	15.7347	0.2598	74	60.55	<.0001
a*b	4	1	14.5552	0.3355	74	43.39	<.0001
a*b	4	2	19.3499	0.2598	74	74.47	<.0001
a*b	5	1	16.3459	0.2739	74	59.68	<.0001
a*b	5	2	21.6220	0.2598	74	83.21	<.0001

Computing the Cell Means Using the ESTIMATE Statement

The cell means can also be obtained by using the ESTIMATE statement to compute the appropriate linear combinations of model parameters. Means for the AB_{11} and AB_{12} cells (highlighted in the above table) are computed below using the ESTIMATE statement. The coefficients that are needed in the ESTIMATE statement are determined by writing what you want to estimate in terms of the fitted model. Using model (1) above, the AB_{12} cell mean, μ_{12}, is:

$$\mu_{12} = (\Sigma_k Y_{12k})/n_{12}$$

$$= (\Sigma_k \mu)/n_{12} + (\Sigma_k \alpha_1)/n_{12} + (\Sigma_k \beta_2)/n_{12} + (\Sigma_k \alpha\beta_{12})/n_{12} + (\Sigma_k \varepsilon_{12k})/n_{12}$$

Because averages of the errors (ε_{ijk}) are assumed to be zero:

$$= \mu + \alpha_1 + \beta_2 + \alpha\beta_{12}$$

Similarly, the AB_{11} cell mean is written this way:

$$\mu_{11} = \mu + \alpha_1 + \beta_1 + \alpha\beta_{11}$$

So, to get an estimate of the AB_{12} mean, you need to add together the estimates of μ, α_1, β_2, and $\alpha\beta_{12}$. This can be done by multiplying the vector of parameter estimates (the *solution* vector) by a vector of coefficients such that their product is this sum. The solution vector in PROC MIXED is requested with the SOLUTION option in the MODEL statement and appears as the Estimate column in the Solution for Fixed Effects table:

Solution for Fixed Effects

Effect	a	b	Estimate	Standard Error	DF	t Value	Pr > |t|
Intercept			21.6220	0.2598	74	83.21	<.0001

Solution for Fixed Effects

Effect	a	b	Estimate	Standard Error	DF	t Value	Pr > |t|
a	1		-11.5879	0.3675	74	-31.53	<.0001
a	2		-9.0408	0.4243	74	-21.31	<.0001
a	3		-5.8874	0.3675	74	-16.02	<.0001
a	4		-2.2722	0.3675	74	-6.18	<.0001
a	5		0	.	.	.	.
b		1	-5.2762	0.3775	74	-13.97	<.0001
b		2	0	.	.	.	.
a*b	1	1	2.7613	0.5426	74	5.09	<.0001
a*b	1	2	0	.	.	.	.
a*b	2	1	3.1139	0.5745	74	5.42	<.0001
a*b	2	2	0	.	.	.	.
a*b	3	1	1.1268	0.5680	74	1.98	0.0510
a*b	3	2	0	.	.	.	.
a*b	4	1	0.4815	0.5680	74	0.85	0.3994
a*b	4	2	0	.	.	.	.
a*b	5	1	0	.	.	.	.
a*b	5	2	0	.	.	.	.

For this model, the solution vector of parameter estimates contains 18 elements. The first element is the estimate of the intercept, μ. The next five elements are the parameter estimates for the levels of A, α_1 through α_5. The next two elements are the parameter estimates for the levels of B, β_1 and β_2. The last 10 elements are the parameter estimates for the 10 levels of the A*B interaction, $\alpha\beta_{11}$ through $\alpha\beta_{52}$.

Now choose a coefficient vector, also with 18 elements, that will multiply the solution vector: Choose a coefficient of 1 for the intercept (μ), coefficients of (1 0 0 0 0) for the A term to pick up the α_1 estimate, coefficients of (0 1) for the B term to pick up the β_2 estimate, and coefficients of (0 1 0 0 0 0 0 0 0 0) for the A*B interaction term to pick up the $\alpha\beta_{12}$ estimate.

The ESTIMATE statement syntax enables you to specify the coefficient vector in sections as just described, with one section for each model effect:

```
estimate 'AB12' intercept 1
                      a 1 0 0 0 0
                      b 0 1
                    a*b 0 1 0 0 0 0 0 0 0 0;
```

Note that this same coefficient vector is given in the table of LS-means coefficients, which was requested by the E option in the LSMEANS statement. Zeros in this table are shown as blanks for clarity. Notice that Row2 is the coefficient vector for computing the mean of the AB_{12} cell.

Coefficients for a*b Least Squares Means

Effect	a	b	Row1	Row2	Row3	Row4	Row5	Row6	Row7	Row8	Row9	Row10
Intercept			1	1	1	1	1	1	1	1	1	1
a	1		1	1								
a	2				1	1						
a	3						1	1				
a	4								1	1		
a	5										1	1
b		1	1		1		1		1		1	
b		2		1		1		1		1		1
a*b	1	1	1									
a*b	1	2		1								
a*b	2	1			1							
a*b	2	2				1						

Coefficients for a*b Least Squares Means

Effect	a	b	Row1	Row2	Row3	Row4	Row5	Row6	Row7	Row8	Row9	Row10
a*b	3	1					1					
a*b	3	2						1				
a*b	4	1							1			
a*b	4	2								1		
a*b	5	1									1	
a*b	5	2										1

It is important to know how variable levels change within the set of parameter estimates for an effect. For example, in the set of parameter estimates for the A*B interaction effect, notice that the second estimate is the estimate of $\alpha\beta_{12}$, because the levels of B change before the levels of A. Had B preceded A in the CLASS statement, the levels of A would have changed before the levels of B, resulting in the second estimate being for $\alpha\beta_{21}$. In this case, the $\alpha\beta_{12}$ estimate is the sixth estimate in the A*B effect requiring a change in the coefficient vector that you specify in the ESTIMATE statement.

An ESTIMATE statement for the AB_{11} cell mean can be written as above by rewriting the cell mean in terms of the model yielding the appropriate linear combination of parameter estimates. The result is Row1 in the table of LS-means coefficients. The following statements fit the model and compute the AB_{11} and AB_{12} cell means by using the LSMEANS statement and equivalent ESTIMATE statements:

```
proc mixed data=test;
   class a b;
   model y=a b a*b;
   lsmeans a*b;
   estimate 'AB11' intercept 1
                         a 1 0 0 0 0
                         b 1 0
                       a*b 1 0 0 0 0 0 0 0 0 0;
   estimate 'AB12' intercept 1
                         a 1 0 0 0 0
                         b 0 1
                       a*b 0 1 0 0 0 0 0 0 0 0;
run;
```

Estimates

Label	Estimate	Standard Error	DF	t Value	Pr > |t|
AB11	7.5193	0.2905	74	25.88	<.0001
AB12	10.0341	0.2598	74	38.62	<.0001

Estimating and Testing a Difference of Means

Suppose you want to test that the AB_{11} and AB_{12} cell means are equal. This is the null hypothesis to test:

$$H_0: \mu_{11} = \mu_{12}$$

or equivalently:

$$H_0: \mu_{11} - \mu_{12} = 0$$

Writing this contrast in terms of model parameters:

$$\mu_{11} - \mu_{12} = (\mu + \alpha_1 + \beta_1 + \alpha\beta_{11}) - (\mu + \alpha_1 + \beta_2 + \alpha\beta_{12}) = \beta_1 - \beta_2 + \alpha\beta_{11} - \alpha\beta_{12}$$

Note that the coefficients for the INTERCEPT and A effects cancel out, removing those effects from the final coefficient vector. However, coefficients for the B effect remain in addition to coefficients for the A*B interaction effect. Use the resulting coefficients in a CONTRAST statement to test that the difference in means is zero. Be careful to order the coefficients to match the order of the model parameters in the procedure.

You can also duplicate the results of the CONTRAST statement with an ESTIMATE statement. Note that the ESTIMATE statement displays the estimated difference in cell means (–2.5148) and a *t*-test that this difference is equal to zero, while the CONTRAST statement provides only an *F*-test of the difference. The tests are equivalent.

For simple pairwise contrasts like this involving a single effect, there are several other ways to obtain the test. You can use the DIFF option in the LSMEANS statement. You can specify a contrast of the LS-means themselves, rather than the model parameters, by using the LSMESTIMATE statement. Finally, you can use the SLICE statement. Note that the CONTRAST and ESTIMATE statements are the most flexible allowing for any linear combination of model parameters. Only these two statements may be flexible enough to estimate or test sufficiently complex linear combinations of model parameters.

The following statements show all five ways of computing and testing this contrast. The DIFF option in the LSMEANS statement provides all pairwise comparisons of the ten LS-means. The contrast of the ten LS-means specified in the LSMESTIMATE statement estimates and tests the difference between the AB11 and AB12 LS-means. The DIFF and SLICEBY(A='1') options in the SLICE statement estimate the differences in LS-means at A=1. Finally, the CONTRAST and ESTIMATE statements use the contrast determined above to compute the AB11 - AB12 difference. All produce equivalent results.

```
proc mixed data=test;
   class a b;
   model y= a b a*b;
   lsmeans a*b / diff;
   lsmestimate a*b 'AB11 - AB12' 1 -1 0 0 0 0 0 0 0 0;
   slice a*b / sliceby(a='1') diff;
   contrast 'AB11 - AB12' b 1 -1
                       a*b 1 -1 0 0 0 0 0 0 0 0;
   estimate 'AB11 - AB12' b 1 -1
                       a*b 1 -1 0 0 0 0 0 0 0 0;
run;
```

These results are from the SLICE statement:

Simple Differences of a*b Least Squares Means						
Slice	b	_b	Estimate	Standard Error	DF	t Value	Pr > |t|
a 1	1	2	-2.5148	0.3898	74	-6.45	<.0001

The LSMESTIMATE statement produces these results:

Least Squares Means Estimate						
Effect	Label	Estimate	Standard Error	DF	t Value	Pr > |t|
a*b	AB11 - AB12	-2.5148	0.3898	74	-6.45	<.0001

Following are the relevant sections of the CONTRAST, ESTIMATE, and LSMEANS statement results:

Estimates					
Label	Estimate	Standard Error	DF	t Value	Pr > |t|
AB11 - AB12	-2.5148	0.3898	74	-6.45	<.0001

Contrasts				
Label	Num DF	Den DF	F Value	Pr > F
AB11 - AB12	1	74	41.63	<.0001

Differences of Least Squares Means								
Effect	a	b	_a	_b	Estimate	Standard Error	DF	t Value	Pr > |t|
a*b	1	1	1	2	-2.5148	0.3898	74	-6.45	<.0001

A More Complex Contrast

Suppose you want to test the average of AB_{11} and AB_{12} versus the average of AB_{21} and AB_{22}.

$$H_0: \tfrac{1}{2}(\mu_{11} + \mu_{12}) = \tfrac{1}{2}(\mu_{21} + \mu_{22})$$

The coefficients for the mean estimates of AB_{11} and AB_{12} are again determined by writing them in terms of the model. The individual AB_{11} and AB_{12} cell means are:

$$\mu_{11} = \mu + \alpha_1 + \beta_1 + \alpha\beta_{11}$$

$$\mu_{12} = \mu + \alpha_1 + \beta_2 + \alpha\beta_{12}$$

and the mean of the two cell means is:

$$\tfrac{1}{2}(\mu_{11}+\mu_{12}) = \tfrac{1}{2}[\,(\mu + \alpha_1 + \beta_1 + \alpha\beta_{11}) + (\mu + \alpha_1 + \beta_2 + \alpha\beta_{12})\,]$$
$$= \mu + \alpha_1 + \tfrac{1}{2}\beta_1 + \tfrac{1}{2}\beta_2 + \tfrac{1}{2}\alpha\beta_{11} + \tfrac{1}{2}\alpha\beta_{12}$$

The coefficients for the average of the AB_{21} and AB_{22} cells are determined in the same fashion. Then, as before, subtracting the two coefficient vectors yields the coefficient vector for testing the difference of these two averages. The final coefficients appear in ESTIMATE and CONTRAST statements below. Note that within a set of coefficients for an effect you can leave off any trailing zeros.

The LSMESTIMATE statement can also be used. Since the contrast involves only the ten LS-means, it is much more straight-forward to specify. Again, trailing zero coefficients can be omitted. The SLICE and LSMEANS statements cannot be used for this more complex contrast.

```
proc mixed data=test;
   class a b;
   model y= a b a*b;
   estimate 'avg AB11,AB12' intercept 1
                                  a 1
                                  b .5 .5
                                a*b .5 .5;
   estimate 'avg AB21,AB22' intercept 1
                                  a 0 1
                                  b .5 .5
                                a*b 0 0 .5 .5;
   contrast 'avg AB11,AB12 - avg AB21+AB22' a 1 -1
                                      a*b .5 .5 -.5 -.5;
   lsmestimate a*b 'avg AB11,AB12 - avg AB21+AB22' 1 1 -1 -1 / divisor=2;
run;
```

Estimates					
Label	Estimate	Standard Error	DF	t Value	Pr > |t|
avg AB11,AB12	8.7767	0.1949	74	45.04	<.0001
avg AB21,AB22	11.5001	0.2165	74	53.11	<.0001

Contrasts				
Label	Num DF	Den DF	F Value	Pr > F
avg AB11,AB12 - avg AB21+AB22	1	74	87.39	<.0001

These results come from the LSMESTIMATE statement. The *t* statistic value is the square root of the F statistic from the CONTRAST statement producing an equivalent test.

Least Squares Means Estimate						
Effect	Label	Estimate	Standard Error	DF	t Value	Pr > |t|
a*b	avg AB11,AB12 - avg AB21+AB22	-2.7233	0.2913	74	-9.35	<.0001

Comparing One Interaction Mean to the Average of All Interaction Means

Suppose A has two levels and B has three levels and you want to test if the AB_{12} cell mean is different from the average of all six cell means.

$$H_0: \mu_{12} - 1/6\ \Sigma_{ij}\mu_{ij} = 0$$

The model is the same as model (1) above with just a change in the subscript ranges. You can use the same method of writing the AB_{12} cell mean in terms of the model:

$$\mu_{12} = \mu + \alpha_1 + \beta_2 + \alpha\beta_{12}$$

You can write the average of cell means in terms of the model:

$$\Sigma_{ij}\mu_{ij} = 1/6\ \Sigma_i\Sigma_j(\mu + \alpha_i + \beta_j + \alpha\beta_{ij}) = \mu + 3/6\ \Sigma_i\alpha_i + 2/6\ \Sigma_j\beta_j + 1/6\ \Sigma_i\Sigma_j\alpha\beta_{ij}$$

So, the coefficient for the A parameters is 1/2; for B it is 1/3; and for AB it is 1/6. However, if you write the ESTIMATE statement like this

```
    estimate 'avg ABij' intercept 1
                              a .5 .5
                              b .333 .333 .333
                            a*b .167 .167 .167 .167 .167 .167;
```

then the procedure provides no results, either displaying `Non-est` in the table of results or issuing this message in the log:

```
NOTE: avg ABij is not estimable.
```

The estimate is declared nonestimable simply because the coefficients 1/3 and 1/6 are not represented precisely enough. To avoid this problem, use the DIVISOR= option. The value that you specify in the option divides all the coefficients that are provided in the ESTIMATE statement.

Finally, writing the hypothesis $\mu_{12} - 1/6\ \Sigma_{ij}\mu_{ij}$ in terms of the model results in these contrast coefficients: 0 for μ, 1/2 and –1/2 for A, –1/3, 2/3, and –1/3 for B, and –1/6, 5/6, –1/6, –1/6, and –1/6 for AB. The statements below fit the model, estimate each part of the hypothesis, and estimate and test the hypothesis. The DIVISOR= option is used to ensure precision and avoid nonestimability.

The LSMESTIMATE statement again makes this easier. The necessary contrast coefficients are stated in the null hypothesis above: (0 1 0 0 0 0) - (1/6 1/6 1/6 1/6 1/6 1/6) , which simplifies to the contrast shown in the LSMESTIMATE statement below.

```
proc mixed data=test;
    class a b;
    model y=a b a*b;
    estimate 'AB12' intercept 1
                            a 1 0
                            b 0 1 0
                          a*b 0 1 0 0 0 0;
    estimate 'avg ABij' intercept 6
                              a 3 3
                              b 2 2 2
                            a*b 1 1 1 1 1 1 / divisor=6;
    estimate 'AB12 vs avg ABij' a 3 -3
                              b -2 4 -2
                            a*b -1 5 -1 -1 -1 -1 / divisor=6;
    lsmestimate a*b 'AB12 vs avg ABij' -1 5 -1 -1 -1 -1 / divisor=6;
run;
```

Parameter	Estimate	Standard Error	t Value	Pr > |t|
AB12	10.0341436	0.25521923	39.32	<.0001
avg ABij	11.3122544	0.11799152	95.87	<.0001
AB12 vs avg ABij	-1.2781108	0.23947144	-5.34	<.0001

Example 2: A Three-Factor Model with Interactions

Now consider a model in three factors, with five, two, and three levels, respectively. Here is the model that includes main effects and all interactions:

$$Y_{ijkl} = \mu + \alpha_i + \beta_j + \gamma_k + \alpha\beta_{ij} + \alpha\gamma_{ik} + \beta\gamma_{jk} + \alpha\beta\gamma_{ijk} + \varepsilon_{ijkl} \quad (2)$$

where i=1,2,...,5, j=1,2, k=1,2,3, and l=1,2,...,N_{ijk} .

These statements generate data from the above model:

```
data test;
   seed=8422636;
```

```
   do a=1 to 5;
      do b=1 to 2;
         do c=1 to 3;
            do rep=1 to ceil(ranuni(seed)*3)+3;
               y=5 + a + b + c + a*b + a*c + b*c + a*b*c + rannor(seed);
               output;
            end;
         end;
      end;
   end;
run;
```

The following statements fit model (2) and display the solution vector and cell means. Note that there are 5 × 2 × 3 = 30 cell means.

```
proc mixed data=test;
   class a b c;
   model y=a|b|c / solution;
   lsmeans a*b*c;
run;
```

Suppose it is of interest to test the null hypothesis that cell means ABC_{121} and ABC_{212} are equal — that is, H_0: $\mu_{121} - \mu_{212} = 0$. Note that these are the fourth and eighth cell means in the Least Squares Means table. Writing the means and their difference in terms of model (2):

$$\mu_{121} = \mu + \alpha_1 + \beta_2 + \gamma_1 + \alpha\beta_{12} + \alpha\gamma_{11} + \beta\gamma_{21} + \alpha\beta\gamma_{121}$$

$$\mu_{212} = \mu + \alpha_2 + \beta_1 + \gamma_2 + \alpha\beta_{21} + \alpha\gamma_{22} + \beta\gamma_{12} + \alpha\beta\gamma_{212}$$

$$\mu_{121} - \mu_{212} = \alpha_1 - \alpha_2 - \beta_1 + \beta_2 + \gamma_1 - \gamma_2 + \alpha\beta_{12} - \alpha\beta_{21} + \alpha\gamma_{11} - \alpha\gamma_{22} + \beta\gamma_{21} - \beta\gamma_{12} + \alpha\beta\gamma_{121} - \alpha\beta\gamma_{212}$$

The following ESTIMATE and CONTRAST statements estimate these means, their difference, and also test that the difference is equal to zero. The test of the difference is more easily obtained using the LSMESTIMATE statement. The simple contrast shown in the LSMESTIMATE statement below compares the fourth and eighth means as desired.

```
proc mixed data=test;
   class a b c;
   model y=a|b|c;
   estimate 'ABC121' intercept 1 a 1 b 0 1 c 1
                     a*b 0 1 a*c 1 b*c 0 0 0 1
                     a*b*c 0 0 0 1;
   estimate 'ABC212' intercept 1 a 0 1 b 1 c 0 1
                     a*b 0 0 1 a*c 0 0 0 0 1 b*c 0 1
                     a*b*c 0 0 0 0 0 0 0 1;
   contrast 'ABC121 - ABC212' a 1 -1 b -1 1 c 1 -1
                              a*b 0 1 -1 a*c 1 0 0 0 -1
                              b*c 0 -1 0 1
                              a*b*c 0 0 0 1 0 0 0 -1;
   estimate 'ABC121 - ABC212' a 1 -1 b -1 1 c 1 -1
                              a*b 0 1 -1 a*c 1 0 0 0 -1
                              b*c 0 -1 0 1
                              a*b*c 0 0 0 1 0 0 0 -1;
   lsmestimate a*b*c 'ABC121 - ABC212' 0 0 0 1 0 0 0 -1;
run;
```

Estimates

Label	Estimate	Standard Error	DF	t Value	Pr > |t|
ABC121	17.0454	0.4118	125	41.40	<.0001
ABC212	21.8270	0.4118	125	53.01	<.0001
ABC121 - ABC212	-4.7816	0.5823	125	-8.21	<.0001

Contrasts

Label	Num DF	Den DF	F Value	Pr > F
ABC121 - ABC212	1	125	67.42	<.0001

Least Squares Means

Effect	a	b	c	Estimate	Standard Error	DF	t Value	Pr > |t|
a*b*c	1	2	1	17.0454	0.4118	125	41.40	<.0001
a*b*c	2	1	2	21.8270	0.4118	125	53.01	<.0001

Differences of Least Squares Means

Effect	a	b	c	_a	_b	_c	Estimate	Standard Error	DF	t Value	Pr > |t|
a*b*c	1	2	1	2	1	2	-4.7816	0.5823	125	-8.21	<.0001

Example 3: A Two-Factor Logistic Model with Interaction Using Dummy and Effects Coding

Logistic models are in the class of generalized linear models. You can fit many kinds of logistic models in many procedures including LOGISTIC, GENMOD, GLIMMIX, PROBIT, CATMOD, and others. For these models, the response is no longer modeled directly. Instead, you model a function of the response distribution's mean. In logistic models, the response distribution is binomial and the log odds (or *logit* of the binomial mean, p) is the response function that you model:

$$\text{logit}(p_i) \equiv \log(\text{Odds}_i) \equiv \log[p_i / (1-p_i)]$$

For more information about logistic models, see these references.

Consider the following medical example in which patients with one of two diagnoses (*complicated* or *uncomplicated*) are treated with one of three treatments (*A*, *B*, or *C*) and the result (*cured* or *not* cured) is observed.

```
data uti;
   input diagnosis : $13. treatment $ response $ count @@;
   datalines;
complicated    A  cured 78  complicated   A not 28
complicated    B  cured 101 complicated   B not 11
complicated    C  cured 68  complicated   C not 46
uncomplicated  A  cured 40  uncomplicated A not 5
uncomplicated  B  cured 54  uncomplicated B not 5
uncomplicated  C  cured 34  uncomplicated C not 6
;
```

Dummy Coding

Indicator or *dummy* coding of a predictor replaces the actual variable in the design matrix (or model matrix) with a set of variables that use values of 0 or 1 to indicate the level of the original variable. One variable is created for each level of the original variable. A main effect parameter is interpreted as the difference in the level's effect compared to the reference level. This is the default coding scheme for CLASS variables in most procedures including GLM, MIXED, GLIMMIX, and GENMOD. Some procedures allow multiple types of coding. In PROC LOGISTIC, use the PARAM=GLM option in the CLASS statement to request dummy coding of CLASS variables. A full-rank version of indicator coding (called *reference* coding) that omits the indicator variable for the reference level (by default, the last level) is also available in PROC LOGISTIC, PROC GENMOD, PROC CATMOD, and some other procedures via the PARAM=REF option.

Using dummy coding, the right-hand side of the logistic model looks like it does when modeling a normally distributed response as in Example 1:

$$\log(\text{Odds}_{ij}) = \mu + \alpha_i + \beta_j + \alpha\beta_{ij} \qquad (3a)$$

where i=1,2,...,5, j=1,2, k=1, 2,...,N_{ij} . But an equivalent representation of the model is:

$$\log(\text{Odds}_{ij}) = \mu + \Sigma_i\alpha_i A_i + \Sigma_j\beta_j B_j + \Sigma_{ij}\gamma_{ij}A_iB_j \qquad (3b)$$

where A_i and B_j are sets of design variables that are defined as follows using dummy coding:

- A_i = 1 if A = i ; otherwise A_i = 0 ,
  where i = 1, 2, 3, 4, or 5 (although the levels might be coded differently).
- B_j = 1 if B = j ; otherwise B_j = 0 ,
  where j = 1 or 2 (although the levels might be coded differently).

For the medical example above, model 3b for the odds of being cured are:

$$\log(\text{Odds}_{dt}) = \mu + d_1\text{O} + d_2\text{U} + t_1\text{A} + t_2\text{B} + t_3\text{C} + g_1\text{OA} + g_2\text{OB} + g_3\text{OC} + g_4\text{UA} + g_5\text{UB} + g_6\text{UC} \qquad (3c)$$

where

- O is the dummy variable for the complicated diagnosis
- U is the dummy variable for the uncomplicated diagnosis
- A, B, and C are the dummy variables for the three treatments
- OA through UC are the products of the diagnosis and treatment dummy variables, jointly representing the diagnosis by treatment interaction

Estimating and Testing Odds Ratios with Dummy Coding

Because log odds are being modeled instead of means, we talk about estimating or testing contrasts of log odds rather than means as in PROC MIXED or PROC GLM. However, the process of constructing CONTRAST statements is the same: write the hypothesis of interest in terms of the fitted model to determine the coefficients for the statement. Note that the CONTRAST statement in PROC LOGISTIC provides an estimate of the contrast as well as a test that it equals zero, so an ESTIMATE statement is not provided. The GENMOD and GLIMMIX procedures provide separate CONTRAST and ESTIMATE statements.

Note that the difference in log odds is equivalent to the log of the odds ratio:

$$\log(\text{Odds}_i) - \log(\text{Odds}_j) = \log(\text{Odds}_i / \text{Odds}_j) = \log(\text{OR}_{ij})$$

So, by exponentiating the estimated difference in log odds, an estimate of the odds ratio is provided. In PROC LOGISTIC, the ESTIMATE=BOTH option in the CONTRAST statement requests estimates of both the contrast (difference in log odds or log odds ratio) and the exponentiated contrast (odds ratio). In PROC GENMOD or PROC GLIMMIX, use the EXP option in the ESTIMATE statement.

For the medical example, suppose we are interested in the odds ratio for treatment A versus treatment C in the complicated diagnosis. We write the null hypothesis this way:

$$H_0: \log(\text{Odds}_{OA}) = \log(\text{Odds}_{OC})$$

or

$$H_0: \log(\text{Odds}_{OA}) - \log(\text{Odds}_{OC}) = 0$$

The following table summarizes the data within the complicated diagnosis:

Table of treatment by response

Table of treatment by response			
treatment	response		Total
	cured	not	
A	78	28	106
C	68	46	114
Total	146	74	220

The odds ratio can be computed from the data as:

```
   78/28   78·46
   ----- = ----- = 1.8845
   68/46   68·28
```

This means that, when the diagnosis is complicated, the odds of being cured by treatment A are 1.8845 times the odds of being cured by treatment C. The following statements display the table above and compute the odds ratio:

```
proc freq data=uti;
   where diagnosis = "complicated" and treatment in ("A","C");
   table treatment * response / relrisk norow nocol nopercent;
   weight count;
   run;
```

Estimates of the Relative Risk (Row1/Row2)			
Type of Study	Value	95% Confidence Limits	
Case-Control (Odds Ratio)	1.8845	1.0643	3.3367
Cohort (Col1 Risk)	1.2336	1.0210	1.4906
Cohort (Col2 Risk)	0.6546	0.4440	0.9652

To estimate and test this same contrast of log odds using model 3c, follow the same process as in Example 1 to obtain the contrast coefficients that are needed in the CONTRAST or ESTIMATE statement. See the Analysis of Maximum Likelihood Estimates table to verify the order of the design variables. This is critical for properly ordering the coefficients in the CONTRAST or ESTIMATE statement. For this example, the table confirms that the parameters are ordered as shown in model 3c.

The log odds for treatment A in the complicated diagnosis are:

$$\log(\text{Odds}_{OA}) = \mu + d_1 + t_1 + g_1$$

The log odds for treatment C in the complicated diagnosis are:

$$\log(\text{Odds}_{OC}) = \mu + d_1 + t_3 + g_3$$

Subtracting these gives the difference in log odds, or equivalently, the log odds ratio:

$$\log(\text{Odds}_{OA}) - \log(\text{Odds}_{OC}) = t_1 - t_3 + g_1 - g_3$$

The following statements use PROC LOGISTIC to fit model 3c and estimate the contrast. The contrast estimate is exponentiated to yield the odds ratio estimate.

```
proc logistic data=uti;
   freq count;
   class diagnosis treatment / param=glm;
   model response(event='cured') = diagnosis treatment diagnosis*treatment;
   contrast 'trt A vs C in comp' treatment 1 0 -1
            diagnosis*treatment 1 0 -1 0 0 0 / estimate=both;
   output out=out xbeta=xbeta;
   run;
```

Contrast Rows Estimation and Testing Results								
Contrast	Type	Row	Estimate	Standard Error	Alpha	Lower Limit	Upper Limit	Wald Chi-Square	Pr > ChiSq
trt A vs C in comp	PARM	1	0.6336	0.2915	0.05	0.0623	1.2050	4.7246	0.0297
trt A vs C in comp	EXP	1	1.8845	0.5493	0.05	1.0643	3.3367	4.7246	0.0297

The XBETA= option in the OUTPUT statement requests the linear predictor, **x'β**, for each observation. This is the log odds. The following statements print the log odds for treatments A and C in the complicated diagnosis.

```
proc print data=out noobs;
   where diagnosis="complicated" and response="cured" and
         treatment in ("A","C");
   var diagnosis treatment xbeta;
   run;
```

diagnosis	treatment	xbeta
complicated	A	1.02450

diagnosis	treatment	xbeta
complicated	C	0.39087

Notice that the difference in log odds for these two cells (1.02450 – 0.39087 = 0.63363) is the same as the log odds ratio estimate that is provided by the CONTRAST statement. Exponentiating this value (exp[.63363] = 1.8845) yields the exponentiated contrast value (the odds ratio estimate) from the CONTRAST statement.

PROC GENMOD can also be used to estimate this odds ratio. Specify the DIST=BINOMIAL option to specify a logistic model

```
proc genmod data=uti;
   freq count;
   class diagnosis treatment;
   model response = diagnosis treatment diagnosis*treatment / dist=binomial;
   estimate 'trt A vs C in comp' treatment 1 0 -1
            diagnosis*treatment 1 0 -1 0 0 0 / exp;
   run;
```

As shown in Example 1, tests of simple effects within an interaction can be done using any of several statements other than the CONTRAST and ESTIMATE statements. In PROC LOGISTIC, odds ratio estimates for variables involved in interactions can be most easily obtained using the ODDSRATIO statement. The following ODDSRATIO statement provides the same estimate of the treatment A vs. treatment C odds ratio in the complicated diagnosis as above (along with odds ratio estimates for the other treatment pairs in that diagnosis).

```
   oddsratio treatment / at(diagnosis='complicated');
```

As in Example 1, you can also use the LSMEANS, LSMESTIMATE, and SLICE statements in PROC LOGISTIC, PROC GENMOD, and PROC GLIMMIX when dummy coding (PARAM=GLM) is used. The ILINK option in the LSMEANS statement provides estimates of the probabilities of cure for each combination of treatment and diagnosis. The DIFF option estimates and tests each pairwise difference of log odds. The EXP option exponentiates each difference providing odds ratio estimates for each pair. The LSMESTIMATE statement allows you to request specific comparisons. Since treatment A and treatment C are the first and third in the LSMEANS list, the contrast in the LSMESTIMATE statement estimates and tests their difference. The EXP option provides the odds ratio estimate by exponentiating the difference. Similarly, the SLICEBY, DIFF, and EXP options in the SLICE statement estimate and test differences and odds ratios in the complicated diagnosis.

```
   lsmeans diagnosis*treatment / ilink exp diff;
   lsmestimate diagnosis*treatment 'A vs C complicated' 1 0 -1 / exp;
   slice diagnosis*treatment / sliceby(diagnosis='complicated') diff exp;
```

An example of using the LSMEANS and LSMESTIMATE statements to estimate odds ratios in a repeated measures (GEE) model in PROC GENMOD is available.

A Nested Model

Rather than the usual main effects and interaction model (3c), the same tasks can be accomplished using an equivalent nested model:

$$\log(\text{Odds}_{dt}) = \mu + d_d + t(d)_{dt}$$

or

$$\log(\text{Odds}_{dt}) = \mu + d_1\text{O} + d_2\text{U} + g_1\text{OA} + g_2\text{OB} + g_3\text{OC} + g_4\text{UA} + g_5\text{UB} + g_6\text{UC} \qquad (3d)$$

The nested term uses the same degrees of freedom as the treatment and interaction terms in the previous model. The design variables that are generated for the nested term are the same as those generated by the interaction term previously. But the nested term makes it more obvious that you are contrasting levels of treatment within each level of diagnosis. See the "Parameterization of PROC GLM Models" section in the PROC GLM documentation for some important details on how the design variables are created. As before, it is vital to know the order of the design variables that are created for an effect so that you properly order the contrast coefficients in the CONTRAST statement.

You write the contrast of log odds in terms of the nested model (3d):

$$\log(\text{Odds}_{\text{OA}}) = \mu + d_1 + g_1$$

The log odds for treatment C in the complicated diagnosis are:

$$\log(\text{Odds}_{\text{OC}}) = \mu + d_1 + g_3$$

Subtracting these gives the difference in log odds, or equivalently, the log odds ratio:

$$\log(\text{Odds}_{\text{OA}}) - \log(\text{Odds}_{\text{OC}}) = g_1 - g_3$$

Notice that this simple contrast is exactly the same contrast that is estimated for a main effect parameter — a comparison of the level's effect versus the effect of the last (reference) level. Therefore, this contrast is also estimated by the parameter for treatment A within the complicated diagnosis in the nested effect. The following statements fit the nested model and compute the contrast. The Analysis of Maximum Likelihood Estimates table confirms the ordering of design variables in model 3d. The first three parameters of the nested effect are the effects of treatments within the complicated diagnosis. The second three parameters are the effects of the treatments within the uncomplicated diagnosis. The EXPB option adds a column in the parameter estimates table that contains exponentiated values of the corresponding parameter estimates.

```
proc logistic data=uti;
   freq count;
   class diagnosis treatment / param=glm;
   model response(event='cured') = diagnosis treatment(diagnosis) / expb;
   contrast 'trt A vs C in comp'
            treatment(diagnosis) 1 0 -1 0 0 0 / estimate=both;
   run;
```

The contrast table that shows the log odds ratio and odds ratio estimates is exactly as before. Notice that the parameter estimate for treatment A within complicated diagnosis is the same as the estimated contrast and the exponentiated parameter estimate is the same as the exponentiated contrast.

Analysis of Maximum Likelihood Estimates

Parameter	DF	Estimate	Standard Error	Wald Chi-Square	Pr > ChiSq	Exp(Est)
Intercept	1	1.7346	0.4428	15.3451	<.0001	5.667

Analysis of Maximum Likelihood Estimates

Parameter			DF	Estimate	Standard Error	Wald Chi-Square	Pr > ChiSq	Exp(Est)
diagnosis	complicated		1	-1.3437	0.4822	7.7653	0.0053	0.261
diagnosis	uncomplicated		0	0	.	.	.	.
treatment(diagnosis)	A	complicated	1	0.6336	0.2915	4.7246	0.0297	1.884
treatment(diagnosis)	B	complicated	1	1.8262	0.3705	24.3005	<.0001	6.210
treatment(diagnosis)	C	complicated	0	0	.	.	.	.
treatment(diagnosis)	A	uncomplicated	1	0.3448	0.6489	0.2824	0.5952	1.412
treatment(diagnosis)	B	uncomplicated	1	0.6445	0.6438	1.0020	0.3168	1.905
treatment(diagnosis)	C	uncomplicated	0	0	.	.	.	.

The same results can be obtained using the ESTIMATE statement in PROC GENMOD.

```
proc genmod data=uti;
   freq count;
   class diagnosis treatment;
   model response = diagnosis treatment(diagnosis) / dist=binomial;
   estimate 'trt A vs C in comp'
            treatment(diagnosis) 1 0 -1 0 0 0 / exp;
   run;
```

Effects Coding

Effects or *Deviation from mean* coding of a predictor replaces the actual variable in the design matrix (or model matrix) with a set of variables that use values of –1, 0, or 1 to indicate the level of the original variable. The number of variables that are created is one fewer than the number of levels of the original variable, yielding one fewer parameters than levels, but equal to the number of degrees of freedom. For this reason, it is known as a *full-rank* parameterization. Reference parameterization (using the PARAM=REF option) is also a full-rank parameterization. A main effect parameter is interpreted as the deviation of the level's effect from the average effect of all the levels. This coding scheme is used by default by PROC CATMOD and PROC LOGISTIC and can be specified in these and some other procedures such as PROC GENMOD with the PARAM=EFFECT option in the CLASS statement.

Using effects coding, the model still looks like model 3b, but the design variables for diagnosis and treatment are defined differently as you can see in the following table. Notice that if you add up the rows for diagnosis (or treatments), the sum is zero. With effects coding, the parameters are constrained to sum to zero. Therefore, the estimate of the last level of an effect, A, is $\alpha_a = -(\alpha_1 + \alpha_2 + ... + \alpha_{a-1})$.

Class Level Information

Class	Value	Design Variables 1	2
diagnosis	complicated	1	
	uncomplicated	-1	
treatment	A	1	0
	B	0	1
	C	-1	-1

Estimating and Testing Odds Ratios with Effects Coding

Although the coding scheme is different, you still follow the same steps to determine the contrast coefficients. First, write the model, being sure to verify its parameters and their order from the procedure's displayed results:

$$\log(\text{Odds}_{dt}) = \mu + d\text{O} + t_1\text{A} + t_2\text{B} + g_1\text{OA} + g_2\text{OB} \qquad (3e)$$

Now write each part of the contrast in terms of the effects-coded model (3e). Use the Class Level Information table which shows the design variable settings. For treatment A in the complicated diagnosis, O = 1, A = 1, B = 0. So the log odds are:

$$\log(\text{Odds}_{\text{OA}}) = \mu + d + t_1 + g_1$$

For treatment C in the complicated diagnosis, O = 1, A = –1, B = –1. So the log odds is:

$$\log(\text{Odds}_{\text{OC}}) = \mu + d - t_1 - t_2 - g_1 - g_2$$

Subtracting these gives the difference in log odds, or equivalently, the log odds ratio:

$$\log(\text{Odds}_{\text{OA}}) - \log(\text{Odds}_{\text{OC}}) = 2t_1 + t_2 + 2g_1 + g_2$$

The following PROC LOGISTIC statements fit the effects-coded model and estimate the contrast:

```
proc logistic data=uti;
   freq count;
   class diagnosis treatment;
   model response(event='cured') = diagnosis treatment diagnosis*treatment;
   contrast 'trt A vs C in comp' treatment 2 1
            diagnosis*treatment 2 1 / estimate=both;
   run;
```

The same log odds ratio and odds ratio estimates are obtained as from the dummy-coded model. The change in coding scheme does not affect how you specify the ODDSRATIO statement. The ODDSRATIO statement used above with dummy coding provides the same results with effects coding.

These are the equivalent PROC GENMOD statements:

```
proc genmod data=uti;
   freq count;
   class diagnosis treatment / param=effect;
   model response = diagnosis treatment diagnosis*treatment / dist=binomial;
   estimate 'trt A vs C in comp' treatment 2 1
            diagnosis*treatment 2 1 / exp;
   run;
```

A More Complex Contrast with Effects Coding

Suppose you want to test whether the effect of treatment A in the complicated diagnosis is different from the average effect of the treatments in the complicated diagnosis. The null hypothesis, in terms of model 3e, is:

$$H_0: \log(\text{Odds}_{OA}) = \Sigma_j \log(\text{Odds}_{Oj})/3$$

or

$$H_0: \log(\text{Odds}_{OA}) - \Sigma_j \log(\text{Odds}_{Oj})/3 = 0$$

We saw above that the first component of the hypothesis, $\log(\text{Odds}_{OA}) = \mu + d + t_1 + g_1$. You use model 3e to expand the average treatment effect:

$$\Sigma_j \log(\text{Odds}_{Oj})/3 = [(\mu + d + t_1 + g_1) + (\mu + d + t_2 + g_2) + (\mu + d - t_1 - t_2 - g_1 - g_2)]/3 = (3\mu + 3d)/3 = \mu + d$$

You subtract this from $\log(\text{Odds}_{OA})$:

$$(\mu + d + t_1 + g_1) - (\mu + d) = t_1 + g_1$$

So the hypothesis, written in terms of the model parameters, is simply:

$$H_0: t_1 + g_1 = 0$$

The following CONTRAST statement used in PROC LOGISTIC estimates and tests this hypothesis, and produces the following output tables:

```
   contrast 'trt A vs avg trt in comp' treatment 1 0
            diagnosis*treatment 1 0 / estimate=both;
```

In PROC GENMOD, use this equivalent ESTIMATE statement:

```
   estimate 'trt A vs avg trt in comp' treatment 1 0
            diagnosis*treatment 1 0 / exp;
```

Contrast Rows Estimation and Testing Results

Contrast	Type	Row	Estimate	Standard Error	Alpha	Lower Limit	Upper Limit	Wald Chi-Square	Pr > ChiSq
trt 1 vs avg trt in comp	PARM	1	-0.1863	0.1919	0.05	-0.5624	0.1898	0.9428	0.3316
trt 1 vs avg trt in comp	EXP	1	0.8300	0.1593	0.05	0.5698	1.2090	0.9428	0.3316

The exponentiated contrast estimate, 0.83, is not really an odds ratio. After exponentiating, the denominator is not just a simple odds, but rather a geometric mean of the treatment odds. The result, while not strictly an odds ratio, is useful as a comparison of the odds of treatment A to the "average" odds of the treatments. However, this is something that cannot be estimated with the ODDSRATIO statement which only compares odds of levels of a specified variable. The LSMEANS, LSMESTIMATE, and SLICE statements cannot be used with effects coding.

Because PROC CATMOD also uses effects coding, you can use the following CONTRAST statement in that procedure to get the same results as above. The WEIGHT statement in PROC CATMOD enables you to input data summarized in cell count form.

```
proc catmod data=uti;
   weight count;
   model response = diagnosis treatment diagnosis*treatment;
   contrast 'trt 1 vs avg trt in comp' treatment 1 0
            diagnosis*treatment 1 0 / estimate=both;
   run;
```

PROC CATMOD has a feature that makes testing this kind of hypothesis even easier. You can specify *nested-by-value* effects in the MODEL statement to test the effect of one variable within a particular level of another variable. This is an extension of the nested effects that you can specify in other procedures such as GLM and LOGISTIC. For more information, see the "Generation of the Design Matrix" section in the CATMOD documentation.

In the medical example, you can use nested-by-value effects to decompose treatment*diagnosis interaction as follows:

```
proc catmod data=uti;
   weight count;
   model response = diagnosis
                    treatment(diagnosis='complicated')
```

```
                    treatment(diagnosis='uncomplicated');
   run;
```

The model effects, treatment(diagnosis='complicated') and treatment(diagnosis='uncomplicated'), are nested-by-value effects that test the effects of treatments within each of the diagnoses. In the following output, the first parameter of the treatment(diagnosis='complicated') effect tests the effect of treatment A versus the average treatment effect in the complicated diagnosis. This is exactly the contrast that was constructed earlier.

Analysis of Maximum Likelihood Estimates

Parameter		Estimate	Standard Error	Chi-Square	Pr > ChiSq
Intercept		1.6377	0.1514	116.98	<.0001
diagnosis	complicated	-0.4268	0.1514	7.95	0.0048
treat(diagn=complicated)	A	-0.1864	0.1919	0.94	0.3315
	B	1.0064	0.2329	18.67	<.0001
trea(diag=uncomplicated)	A	0.0149	0.3822	0.00	0.9689
	B	0.3150	0.3793	0.69	0.4063

Example 4: Comparing Models

The CONTRAST statement can also be used to compare competing *nested* models. Models are nested if one model results from restrictions on the parameters of the other model. The most commonly used test for comparing nested models is the likelihood ratio test, but other tests (such as Wald and score tests) can also be used. The next section illustrates using the CONTRAST statement to compare nested models. Other methods must be used to compare nonnested models and this is discussed in the section that follows.

Comparing Nested Models

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). Two logistic models are fit in this example: The first model is *saturated*, meaning that it contains all possible main effects and interactions using all available degrees of freedom. The second model is a reduced model that contains only the main effects.

The following statements create the data set and fit the saturated logistic model.

```
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
;

ods select modelfit type3;
ods output modelfit=full;
proc genmod data=detergent;
   class Softness Previous Temperature;
   freq Count;
   model Brand = Softness|Previous|Temperature / dist=binomial type3;
run;
```

The partial results shown below suggest that interactions are not needed in the model:

Criteria For Assessing Goodness Of Fit

Criterion	DF	Value	Value/DF
Deviance	996	1364.4956	1.3700
Scaled Deviance	996	1364.4956	1.3700
Pearson Chi-Square	996	1008.0000	1.0120
Scaled Pearson X2	996	1008.0000	1.0120
Log Likelihood		-682.2478	

LR Statistics For Type 3 Analysis

Source	DF	Chi-Square	Pr > ChiSq
Softness	2	0.10	0.9522

LR Statistics For Type 3 Analysis			
Source	DF	Chi-Square	Pr > ChiSq
Previous	1	22.13	<.0001
Softness*Previous	2	3.79	0.1506
Temperature	1	3.64	0.0564
Softness*Temperature	2	0.20	0.9066
Previous*Temperature	1	2.26	0.1327
Softne*Previo*Temper	2	0.74	0.6917

The simpler main-effects-only model can be fit by restricting the parameters for the interactions in the above model to zero. This simpler model is nested in the above model. These statements fit the restricted, main effects model:

```
ods select modelfit type3;
ods output modelfit=reduced;
proc genmod data=detergent;
   class Softness Previous Temperature;
   freq Count;
   model Brand = Softness Previous Temperature / dist=binomial type3;
run;
```

This partial output summarizes the main-effects model:

Criteria For Assessing Goodness Of Fit			
Criterion	DF	Value	Value/DF
Deviance	1003	1372.7236	1.3686
Scaled Deviance	1003	1372.7236	1.3686
Pearson Chi-Square	1003	1007.9360	1.0049
Scaled Pearson X2	1003	1007.9360	1.0049
Log Likelihood		-686.3618	

LR Statistics For Type 3 Analysis			
Source	DF	Chi-Square	Pr > ChiSq
Softness	2	0.22	0.8976
Previous	1	19.89	<.0001
Temperature	1	3.74	0.0532

The question is whether there is a significant difference between these two models. Stated another way, are any of the interaction parameters not equal to zero as implied by the main-effects model? This test can be done using a CONTRAST statement to jointly test the interaction parameters.

Any estimable linear combination of model parameters can be tested using the procedure's CONTRAST statement. The difficulty is constructing combinations that are estimable and that jointly test the set of interactions. This can be particularly difficult with dummy (PARAM=GLM) coding. The problem is greatly simplified using effects coding, which is available in some procedures via the PARAM=EFFECT option in the CLASS statement. The CONTRAST statement tests the hypothesis $\mathbf{L\beta=0}$, where $\mathbf{L}$ is the hypothesis matrix and $\boldsymbol{\beta}$ is the vector of model parameters. With effects coding, each row of $\mathbf{L}$ can be written to select just one interaction parameter when multiplied by $\boldsymbol{\beta}$. In the CONTRAST statement, the rows of $\mathbf{L}$ are separated by commas. The CONTRAST statement below defines seven rows in $\mathbf{L}$ for the seven interaction parameters resulting in a 7 DF test that all interaction parameters are zero.

```
ods select contrasts;
proc genmod data=detergent;
   class Softness Previous Temperature / param=effect;
   freq Count;
   model Brand = Softness|Previous|Temperature / dist=binomial;
   contrast 'lrt'
   softness*previous 1 0,
   softness*previous 0 1,
   softness*temperature 1 0,
   softness*temperature 0 1,
   previous*temperature 1,
   softness*previous*temperature 1 0,
   softness*previous*temperature 0 1;
run;
```

By default, PROC GENMOD computes a likelihood ratio test for the specified contrast. As expected, the results show that there is no significant interaction (p=0.3129) or that the reduced model fits as well as the saturated model.

Contrast Results				
Contrast	DF	Chi-Square	Pr > ChiSq	Type
lrt	7	8.23	0.3129	LR

Some procedures, like PROC LOGISTIC, produce a Wald chi-square statistic instead of a likelihood ratio statistic. PROC GENMOD produces the Wald statistic when the WALD option is used in the CONTRAST statement. The likelihood ratio and Wald statistics are asymptotically equivalent. The following statements do the model comparison using PROC LOGISTIC and the Wald test produces a very similar result.

```
ods select contrasttest;
proc logistic data=detergent;
   class Softness Previous Temperature / param=effect;
   freq Count;
```

```
   model Brand = Softness|Previous|Temperature;
   contrast 'lrt'
   softness*previous 1 0,
   softness*previous 0 1,
   softness*temperature 1 0,
   softness*temperature 0 1,
   previous*temperature 1,
   softness*previous*temperature 1 0,
   softness*previous*temperature 0 1;
run;
```

Contrast Test Results

Contrast	DF	Wald Chi-Square	Pr > ChiSq
lrt	7	8.1794	0.3170

In addition to using the CONTRAST statement, a likelihood ratio test can be constructed using the likelihood values obtained by fitting each of the two models. After fitting both models and constructing a data set with variables containing predicted values from both models, the %VUONG macro with the TEST=LR parameter provides the likelihood ratio test. See the example titled "Comparing nested models with a likelihood ratio test" which illustrates using the %VUONG macro to produce the same test as obtained above from the CONTRAST statement in PROC GENMOD.

Limitations on constructing valid LR tests

The likelihood ratio test can be used to compare any two nested models that are fit by maximum likelihood. It is not necessary that the larger model be saturated. So, this test can be used with models that are fit by many procedures such as GENMOD, LOGISTIC, MIXED, GLIMMIX, PHREG, PROBIT, and others, but there are cases with some of these procedures in which a LR test cannot be constructed:

- With any procedure, models that are not nested cannot be compared using the LR test.
- Models fit with the GENMOD or GEE procedure using the REPEATED statement are estimated using the generalized estimating equations (GEE) method and not by maximum likelihood so a LR test cannot be constructed. However, the CONTRAST statement can be used in PROC GENMOD as shown above to produce a score test of the hypothesis.
- With mixed models fit in PROC MIXED, if the models are nested in the covariance parameters and have identical fixed effects, then a LR test can be constructed using results from REML estimation (the default) or from ML estimation. Basing the test on the REML results is generally preferred. However, if the nested models do not have identical fixed effects, then results from ML estimation must be used to construct a LR test. See this note on comparing covariance structures in PROC MIXED.
- In most cases, models fit in PROC GLIMMIX using the RANDOM statement do not use a true log likelihood. When the procedure reports a log pseudo-likelihood you cannot construct a LR test to compare models. In some cases, the Laplace or quadrature estimation methods (METHOD=LAPLACE or METHOD=QUAD, first available in SAS 9.2) can be used which compute and report an approximate log likelihood making construction of a LR test possible. See this note on comparing covariance structures in PROC GLIMMIX.

Comparing Nonnested Models

Nonnested models can still be compared using information criteria such as AIC, AICC, and BIC (also called SC). These statistics are provided in most procedures using maximum likelihood estimation. Models with smaller values of these criteria are considered better models. However, no statistical tests comparing criterion values is possible.

Tests to compare nonnested models are available, but not by using CONTRAST statements as discussed above. See this sample program for discussion and examples of using the Vuong and Clarke tests to compare nonnested models. For a more detailed definition of nested and nonnested models, see the Clarke (2001) reference cited in the sample program.

Example 5: A Quadratic Logistic Model

This example shows the use of the CONTRAST and ODDSRATIO statements to compare the response at two levels of a continuous predictor when the model contains a higher-order effect. Specifically, PROC LOGISTIC is used to fit a logistic model containing effects X and X^2. The correct coefficients are determined for the CONTRAST statement to estimate two odds ratios: one for an increase of one unit in X, and the second for a two unit increase. It is shown how this can be done more easily using the ODDSRATIO and UNITS statements in PROC LOGISTIC.

Operating System and Release Information

Product Family	Product	System	SAS Release	
			Reported	Fixed*
SAS System	SAS/STAT	All	n/a	

* For software releases that are not yet generally available, the Fixed Release is the software release in which the problem is planned to be fixed.

Type: Usage Note
Priority: low
Topic: Analytics ==> Mixed Models
Analytics ==> Longitudinal Analysis
SAS Reference ==> Procedures ==> SURVEYLOGISTIC
Analytics ==> Categorical Data Analysis
SAS Reference ==> Procedures ==> SURVEYREG
SAS Reference ==> Procedures ==> MIXED
SAS Reference ==> Procedures ==> LOGISTIC
SAS Reference ==> Procedures ==> CATMOD
SAS Reference ==> Procedures ==> GENMOD

SAS Reference ==> Procedures ==> GLM
SAS Reference ==> Procedures ==> GLIMMIX
Analytics ==> Analysis of Variance
SAS Reference ==> Procedures ==> PHREG
SAS Reference ==> Procedures ==> SURVEYPHREG
Analytics ==> Regression
Analytics ==> Survey Sampling and Analysis
Analytics ==> Survival Analysis
SAS Reference ==> Procedures ==> ORTHOREG

Date Modified: 2005-12-02 05:51:25

Date Created: 2005-10-07 13:22:10