Usage Note *22603:* Producing an actual-by-predicted table (confusion matrix) for a multinomial response

PROC LOGISTIC can fit a logistic or probit model to a binary or multinomial response. By default, a binary logistic model is fit to a binary response variable, and an ordinal logistic model is fit to a multinomial response variable. To fit a binary or ordinal probit model in these cases, specify the LINK=PROBIT option in the MODEL statement. To fit a nominal (unordered) logistic model to a nominal multinomial response variable, specify the LINK=GLOGIT option. Another approach is to fit a classification tree model. Beginning in SAS® 9.4 TS1M3, use the HPSPLIT procedure. See the examples in the HPSPLIT documentation.

For a binary response, the CTABLE option in the MODEL statement of PROC LOGISTIC produces actual-by-predicted classification tables for a range of cutoff values applied to the predicted event probabilities for the observations. This option is not available for multinomial responses. For binary or multinomial responses, use the PREDPROBS=INDIVIDUAL option in the OUTPUT statement of PROC LOGISTIC. This option creates a data set with separate variables containing predicted probabilities for the response levels and a variable (_INTO_) containing the predicted response category. You can also request bias-adjusted (cross validated) predicted values and predicted response categories for binary-response models by using the PREDPROBS=CROSSVALIDATE option.

With the data set from either OUTPUT statement option, you can use PROC FREQ to create a cross classification table, often called a *confusion matrix*, of the actual and predicted response variables for the data used to fit the model. Similarly, an actual by predicted table can be created for a validation data set by using the SCORE statement which also produces a data set containing predicted probability variables and a variable (I_*y*, where *y* is the name of your response variable) containing the predicted response category. Note that the validation data set must contain the observed responses in order to produce the table.

Example 1: For the original data

The following uses the example titled "Nominal Response Data: Generalized Logits Model" in the LOGISTIC documentation. The nominal multinomial response, Style, has three levels and PROC LOGISTIC is used to fit a nominal logistic model to the data. The PREDPROBS=INDIVIDUAL option saves the predicted probabilities and the predicted response level (_INTO_) in the data set PREDS. PROC FREQ displays the confusion matrix by cross classifying the actual and predicted response variables. The cell counts of the matrix are saved in data set CellCounts. The subsequent DATA step adds a variable, Match, which indicates when the actual and predicted response levels agree. The mean of Match, computed by PROC MEANS, is the proportion of observations correctly classified by the nominal logistic model.

```
proc logistic data=school;
  freq Count;
  class School Program(ref=first);
  model Style(order=data)=School Program / link=glogit;
  output out=preds predprobs=individual;
  run;
proc freq data=preds;
  table Style*_INTO_ / out=CellCounts;
  run;
data CellCounts;
  set CellCounts;
  Match=0;
  if Style=_INTO_ then Match=1;
  run;
proc means data=CellCounts mean;
  freq count;
  var Match;
  run;
```

The results show that the nominal logistic model did not classify any of the observations into the TEAM response level and that 33% of the observations were correctly classified by the model.

Frequency
Percent
Row Pct
Col Pct

Table of STYLE by _INTO_

STYLE	_INTO_(Formatted Value of the Predicted Response): class	self	Total
class	5 27.78 83.33 33.33	1 5.56 16.67 33.33	6 33.33
self	5 27.78 83.33 33.33	1 5.56 16.67 33.33	6 33.33
team	5 27.78 83.33 33.33	1 5.56 16.67 33.33	6 33.33
Total	15 83.33	3 16.67	18 100.00

The MEANS Procedure

Analysis Variable : match
Mean
0.3333333

Example 2: For original and validation data

The following uses the example titled "Scoring Data Sets" in the LOGISTIC documentation. These statements create a validation data set named NewCrops. It contains five observations from each of the crop types.

```
data NewCrops;
   input Crop $ x1-x4;
   datalines;
```

```
Clover      34  26  60  90
Corn        12  27  25  11
Corn        16  24  19  70
Corn        15  21  30  32
Corn        15  25  27  30
Corn        14  23  27  31
Cotton      42  52  58  64
Cotton      30  38  66  32
Cotton      31  43  -8  78
Cotton      37  38  -7  35
Cotton      28  33  11  49
Soybeans    20  16  19  28
Soybeans    15  19  28  11
Soybeans    21  23  23  25
Soybeans    18  21  23  24
Soybeans    16  37  23  18
Sugarbeets  18  29  19  29
Sugarbeets  43  32  29   7
Sugarbeets  21  20   1  47
Sugarbeets  18  43  18  59
Sugarbeets  32  46  27  17
;
```

In the following statements, the OUTMODEL= option saves the model information to a data set so that it can be used later to score additional data. As in Example 1, the OUTPUT scores the original data and the following steps produce the confusion matrix and the correctly-classified proportion. The SCORE statement uses the fitted model to score the NewCrops data set and saves the result in a data set named NewCropPred.

```
proc logistic data=Crops outmodel=CropModel;
  model Crop=x1-x4 / link=glogit;
  output out=preds predprobs=individual;
  score data=NewCrops out=NewCropPred;
  run;
proc freq data=preds;
  table Crop*_INTO_ / out=CellCounts;
  run;
data CellCounts;
  set CellCounts;
  Match=0;
  if Crop=_INTO_ then Match=1;
  run;
proc means data=CellCounts mean;
  freq count;
  var Match;
  run;
```

The results show that the model correctly classified approximately 53% of the observations in the original data set.

Frequency
Percent
Row Pct
Col Pct

Table of CROP by _INTO_

CROP	_INTO_(Formatted Value of the Predicted Response)					
	Clover	Corn	Cotton	Soybeans	Sugarbeets	Total
Clover	6 16.67 54.55 46.15	0 0.00 0.00 0.00	2 5.56 18.18 50.00	2 5.56 18.18 25.00	1 2.78 9.09 33.33	11 30.56
Corn	0 0.00 0.00 0.00	7 19.44 100.00 87.50	0 0.00 0.00 0.00	0 0.00 0.00 0.00	0 0.00 0.00 0.00	7 19.44
Cotton	4 11.11 66.67 30.77	0 0.00 0.00 0.00	1 2.78 16.67 25.00	1 2.78 16.67 12.50	0 0.00 0.00 0.00	6 16.67
Soybeans	1 2.78 16.67 7.69	1 2.78 16.67 12.50	1 2.78 16.67 25.00	3 8.33 50.00 37.50	0 0.00 0.00 0.00	6 16.67
Sugarbeets	2 5.56 33.33 15.38	0 0.00 0.00 0.00	0 0.00 0.00 0.00	2 5.56 33.33 25.00	2 5.56 33.33 66.67	6 16.67
Total	13 36.11	8 22.22	4 11.11	8 22.22	3 8.33	36 100.00

The MEANS Procedure

Analysis Variable : Match
Mean
0.5277778

Similarly, these statements produce the confusion matrix and correct classification proportion for the validation data set, NewCrops. Note that the variable containing the predicted response from the SCORE statement is I_Crop rather than _INTO_ as produced by the OUTPUT statement.

```
proc freq data=NewCropPred;
   table Crop*I_Crop / out=CellCounts;
   run;
data CellCounts;
   set CellCounts;
```

```
        freq count;
        var Match;
        run;
```

The results indicate that the model was able to correctly classify 52% of the observations in the validation data set.

Frequency
Percent
Row Pct
Col Pct

Table of Crop by I_CROP

Crop	I_CROP(Into: CROP) Clover	Corn	Cotton	Soybeans	Sugarbeets	Total
Clover	3 12.00 60.00 60.00	1 4.00 20.00 16.67	1 4.00 20.00 33.33	0 0.00 0.00 0.00	0 0.00 0.00 0.00	5 20.00
Corn	0 0.00 0.00 0.00	4 16.00 80.00 66.67	0 0.00 0.00 0.00	0 0.00 0.00 0.00	1 4.00 20.00 20.00	5 20.00
Cotton	0 0.00 0.00 0.00	0 0.00 0.00 0.00	2 8.00 40.00 66.67	0 0.00 0.00 0.00	3 12.00 60.00 60.00	5 20.00
Soybeans	0 0.00 0.00 0.00	1 4.00 20.00 16.67	0 0.00 0.00 0.00	4 16.00 80.00 66.67	0 0.00 0.00 0.00	5 20.00
Sugarbee	2 8.00 40.00 40.00	0 0.00 0.00 0.00	0 0.00 0.00 0.00	2 8.00 40.00 33.33	1 4.00 20.00 20.00	5 20.00
Total	5 20.00	6 24.00	3 12.00	6 24.00	5 20.00	25 100.00

The MEANS Procedure

Analysis Variable : Match
Mean
0.5200000

Should you need to score additional data sets, you can use the saved model information from the OUTMODEL= option. For example, the following statements score a data set named MoreCrops.

```
     proc logistic inmodel=CropModel;
       score data=MoreCrops out=MoreCropPred;
       run;
```

Operating System and Release Information

Product Family	Product	System	SAS Release Reported	SAS Release 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: SAS Reference ==> Procedures ==> LOGISTIC
Analytics ==> Categorical Data Analysis
Analytics ==> Regression

Date Modified: 2019-04-12 10:18:55
Date Created: 2002-12-16 10:56:39

SUPPORT

Contact Us >

Follow Us