MWSUG 2019 Paper RF-038

Evaluate your SCORE: Logistic Regression Prediction Comparison using the SCORE Statement

Robert G. Downer, Grand Valley State University, Allendale, MI

ABSTRACT

The SCORE statement in PROC LOGISTIC was introduced in SAS/STAT 9.0 and it is a feature that can be utilized efficiently to quickly evaluate prediction performance for new observations. Used in conjunction with the OUTMODEL and INMODEL statements, the SCORE statement can be a very beneficial aid in quickly comparing the prediction performance of multiple logistic regression models for the same test or validation observations. The concise syntax of these statements will be illustrated. Performance criterion output such as the misclassification rate will be discussed through a worked example involving multiple models of a binary response. Although some knowledge of logistic regression would be beneficial for full understanding of this paper, it is written for a general audience interested in predictive modeling.

INTRODUCTION

The SCORE statement is common to several SAS/STAT ® procedures. A recent general review of options for evaluating test observations from stored models was given by Koval (2018) and this review included possible input to PROC PLM and applications within SAS Viya. A detailed introduction to PROC PLM in SAS/STAT 9.22 was previously given by Tobias and Cai (2010) which includes a logistic regression application of classifying new observations. Lund (2017) discusses related features within PROC HPLOGISTIC including the PARTITION statement and the CHOOSE=VALIDATE option within the SELECTION statement. The purpose of this paper is to highlight the SCORE statement within PROC LOGISTIC and the concise nature of the syntax within this procedure for quick evaluation of test set observations. Evaluation of model prediction performance is the statistical objective but other features of PROC LOGISTIC are illustrated that may be of interest

TRAINING AND TEST DATA SELECTION

There are a variety of ways to randomly select from a data set to create training and test set observations for general modeling from an existing data set. For the selection of one random sample, stratified sampling and repeated sampling, PROC SURVEYSELECT is commonly used in SAS for such an objective. A variety of model applications have been discussed in SAS-based investigations involving replicated sampling (eg. bootstrapping) such as Moser and Liang (2001) and a logistic regression application can be found in Downer (2016). A comprehensive review of PROC SURVEYSELECT features can be found in Cassell (2007) while other recent sampling design details can be found in Bordenvae (2015) as well as Becker and Doyle (2016).

In this paper, PROC SURVEYSELECT is utilized to simply divide one data set into a training data set and a test data set. The OUTALL option on the PROC SURVEYSELECT line allows one to quickly separate the observations selected for either the training and test data sets (based on 0, 1 values of the automatically generated SELECTED variable in the output data set). In this paper, new data set names of these subsets of the original data are immediately used as input into runs of PROC LOGISTIC with the training set identified on the PROC LOGISTIC line and the test set utilized in the SCORE statement.

LOGISTIC REGRESSION MODEL SYNTAX

For a binary response, a logistic regression model expresses the log odds of presence versus absence p/(1-p) as a linear function of the predictor variables. The logistic regression model for predictors $X_1 \ldots X_k$ is expressed as:

$$\log\left(\frac{p}{1-p}\right) = \beta_o + \beta_1 x_1 + \ldots \beta_k x_k$$

The estimated coefficients $\hat{\beta}_1, \hat{\beta}_2, \ldots \hat{\beta}_k$ can be interpreted on the log-odds or odds scale. With indicator variables coded for categorical predictors, exponentiation of the estimated coefficient represents the odds of the response at the given level of the categorical variable versus the baseline category after adjusting for other variables included. For continuous predictors, exponentiation of the estimated coefficient $\hat{\beta}_i$ represents the estimated odds of the response for a unit change in the predictor X_i, after adjusting for other predictors.

In the PROC LOGISTIC run below with 2 predictors and no interaction, the log-odds of the binary response (residential treatment for mental health clinic patients), is modeled as a linear function of age and gender. The effect of gender on the odds of a residential treatment diagnosis is adjusted for the effect of age (and vise-versa):

```
proc logistic;
class gender/ param = glm descending;
model restreat= age gender ;
run;
```

The fitted probability $\hat{p}$ of the characteristic of interest (residential treatment in the model above) can be obtained for each observation and a cut-off can be established (for existing or for new observations not involved with modeling). An OUTPUT statement would generally be used to send the estimated probabilities for existing observations to a new output data set for further investigation. ODS graphics options such as the PLOTS = effect option can be utilized for further visual understanding of the estimated probabilities of the fitted model (See, for example, Downer and Richardson (2009)). For models with a multi-category response, estimated individual or cumulative probabilities are provided for each category and the highest estimated category probability is the default for assignment of a predicted category. In the PROC LOGISTIC syntax given above, the options PARM = GLM and DESCENDING on the CLASS statement ensure a 0,1 definition to the categorical gender variable with the estimated coefficient being fit for the variable's higher level (defined as male in this example).

SCORE STATEMENT SYNTAX

Expanding on the methodology and syntax described in the previous two sections, suppose we define the indented model of the previous section as model0 and a test set of observations called testdat0 has been previously randomly selected via PROC SURVEYSELECT (and not used in the fitted model). The remaining observations (in traindat0) were used for the modeling. The following PROC LOGISTIC run will store and output the fitted model through the OUTMODEL option on the PROC LOGISTIC line. The subsequent PROC LOGISTIC run uses this model0 information through the INMODEL option on the PROC LOGISTIC line and scores the test set observations of testdat0. The mod0out data set will contain

estimated probabilities and classification information for the newly scored observations. The FITSTAT option in the second run (using the SCORE statement) will contain classification diagnostic information including the error rate:

```
proc logistic descending data = traindat0 outmodel = mod0;
class gender/ param = glm descending;
model restreat= age gender;
run;

proc logistic descending data = traindat0 inmodel = mod0;
score data=testdat0 out = mod0out fitstat;
run;
```

The higher of the two estimated probabilities will result in a prediction of a 1 for a new observation and a 1 will be the assigned value of the automatically generated variable I_restreat (i.e. the response restreat is classified 'into' a 1). It will be assigned as a 0 otherwise. The actual true value of restreat in the test data set will be stored as the value of F_restreat. Although the error rate will also be part of the output generated from the FITSTAT option, a classification table of F_Y * I_Y generated by PROC FREQ will allow for closer examination of the false positive and false negative rates and the fractions involved.

EXAMPLE APPLICATION AND RESULTS

To illustrate the application of the SCORE statement for model comparison, the Adolescent Placement mental health data from Hosmer et al (2013) was utilized. See Hilliard (2017) for another logistic regression application using the same data set. An application of PROC SURVEYSELECT established a training data set of 408 (called adoltr) and a test data set (called adoltest) of size 100. Although a simulation was not performed, similar significance results were found for replicated samples. The original placement variable of this data set has been changed to binary with further residential treatment (restreat) as the characteristic of interest. In the test set, the true value of the response is 1 (residential treatment) in 51 of the 100 observations

The variables included in the model comparison are all within the model 1 below and are as follows: age (age at admission), custd (1 if in state custody, 0 otherwise), neuro (a categorical neurological disturbance variable of 4 levels), behav (a behavioral health score from 0 to 9 considered to be continuous), and los (length of hospitalization in days). In model 1, all predictor terms are significant at the .10 level. The least significant main effects in model 1 are neuro (p = .089) and age (p = .032). Model 2 has neuro dropped. Model 3 also has age dropped and for simplicity, the interaction of los and custd is also dropped in model 4:

```
title2 '5 main effects plus los*custd, model 1';
proc logistic descending data = adoltr outmodel = mod1;
class custd  neuro   / param = glm descending;
model restreat= age los behav neuro custd los*custd ;
run;
```

The application of the SCORE syntax for model 1 in the previous section provided the information displayed in Output 1 below (as well as other fit statistics not shown). The Error rate is the total fraction of incorrectly classified observations. The Brier Score is an alternative performance measure which (for a binary response) is defined as as a weighted squared distance of the fitted probability and observed response (of 1 or 0). Max Rescaled R-square (generalized R-square for general linear models) and AIC and other

traditional fit statistics could be utilized for comparison but these measures are not as direct in showing the classification performance accuracy at a glance.

Fit Statistics for SCORE Data					
Data Set	Total Frequency	Max-Rescaled R-Square	Error Rate	AIC	Brier Score
WORK.ADOLTEST	100	0.691305	0.1000	85.52320	0.095105

Output 1. Subset of displayed information from FITSTAT option

A cross-classified table from PROC FREQ is able to quickly utilize the automatically generated variables that are part of the output file from the OUT = option of the SCORE statement For the response restreat, the variable F_restreat is the actual true classification of the test set observation and I_restreat is the model's predicted classification. The counts in the 2 x 2 table allow one to quickly see the correctly classified observations (in the diagonal cells) and the false positive and false negatives (in the two off-diagonal cells). From the scoring of model 1, a basic PROC FREQ display is given below in Output 2. Only a total of 10 errors were made in the test set of size 100. There are 3/49 false positives (fraction .061) and 7/51 false negatives (fraction .137).

Table of F_restreat by I_restreat			
F_restreat(From : restreat)	I_restreat(Into: restreat)		
Frequency Percent Row Pct Col Pct	0	1	Total
0	46 46.00 93.88 86.79	3 3.00 6.12 6.38	49 49.00
1	7 7.00 13.73 13.21	44 44.00 86.27 93.62	51 51.00
Total	53 53.00	47 47.00	100 100.00

Output 2. PROC FREQ Display of scored test data using model 1

Since the syntax of the OUTMODEL and INMODEL statements are not complex and will be the same for reduced models, the nature of the classification errors of (indexed) reduced models can be evaluated using a suffix on the scoring input and output file names and incorporated into a macro such as the one shown in the Appendix. Such a macro could also easily be expanded and incorporated into a prediction performance simulation in which the training, test, (and possibly validation) data sets will vary for replications of the simulation and the classification results of each repetition can be stored for a more comprehensive overall evaluation of models.

For this single split of the data, summary information for the model comparison is given below in Table 1. The areas under the ROC curves for the training data were .917, .912, .909, .905 respectively for models 1 to 4 respectively. As can be seen in Table 1, significance of more predictor terms may not correspond to much improvement in classification of new observations.

Indexed Model and Predictors	Training ROC	Overall Error Rate	False Positive Rate	False Negative Rate
1) age, los, behav, neuro, custd, los*custd	.917	.10	.137	.061
2) age, los, behav, custd,los*custd	.912	.12	.176	.061
3) los,behav,custd,los*custd	.909	.12	.176	.061
4) los, behav, custd	.905	.12	.176	.061

Table 1. Prediction performance model comparison summary

The overall improvement in the classification error fraction of the test data using model 1 corresponded to the occurrence of two less false negatives as compared to the three simpler models (7/51 for model 1 as compared to 9/51 for the others). The false negative rate remained the same for all models (3/49) and it was the same three observations that were misclassified. Models 2, 3 and 4 were identical with respect to classification errors. A similar result was observed to be common in other test samples in which the error rate varied from .08 to .16. Model 4 had only 3 predictors and is a much simpler model with no interaction terms. Hence, if there is much extra time or extra overall cost in obtaining some predictor variables for new observations, these disadvantages may outweigh the benefits and a simpler prediction model may still be very effective. A validation study would be quite valuable in such an assessment and the SCORE statement will be an effective tool for such decision making.

CONCLUSION

The SCORE statement in PROC LOGISTIC is a straightforward way to evaluate the prediction accuracy of new observations. In this paper, the basic SCORE syntax and output were emphasized but repeated application of this concise statement could also be used as an efficient part of a more extensive comparison of models or methodology. The SCORE statement should be considered as part of the tool box for logistic regression applications by the occasional modeler or practicing statistician.

REFERENCES

Becker, R. and Doyle, D. (2016). Sampling in SAS ® using PROC SURVEYSELECT. SAS Global Forum Proceedings, Paper 11762.

Bordenvae, R.K. (2015) Using PROC SURVEYSELECT: Random Sampling, South East SAS Users Group Proceedings, Paper AD190.

Cassell, D.L., (2007). Don't be Loopy: Re-Sampling and Simulation the SAS® Way. SAS Global Forum Proceedings, Paper 183..

Downer, R.G. and Richardson, P.J. (2009). Illustrative Logistic Regression Examples using PROC LOGISTIC: New Features in SAS/STAT® 9.2, Pharmaceutical SAS Users Group Meeting Proceedings, Paper SP-03.

Downer, R.G. (2016). To be two or not be two, that is a LOGISTIC question, Proceedings of the 2016 Midwest SAS Users Group Meeting, Paper AA-18.

Hilliard, P.J. (2017). Using New SAS 9.4 Features for Cumulative Logit Models with Partial Proportional Odds. SAS Global Forum Proceedings, E-Poster 406.

Hosmer, D.W., Lemeshow, S. and Sturdivant, R.X. (2013) *Applied Logistic Regression*: Third Edition, New York, John Wiley & Sons.

Koval, S. (2018)' How to Score Big with SAS® Solutions: Various Ways to Score New Data with Trained Models. MidWest SAS Users Group Proceedings, Paper AA-121.

Lund, B. (2017). Logistic Model Selection with SAS® PROC's LOGISTIC, HPLOGISTIC, HPGENSELECT, Proceedings of the 2017 Midwest SAS Users Group Meeting, Paper AA-02.

Moser, E.B. and Liang, X. (2001). Bootstrapping a Multidimensional Preference Analysis, Proceedings of the South Central SAS Users Group, Paper P407.

Tobias, R and Cai, W. (2010). Introducing PROC PLM and Postfitting Analysis for Very General Linear Models in SAS/STAT® 9.22, SAS Global Forum Proceedings 2010, Paper 258, SAS Institute Inc.

CONTACT INFORMATION

Your comments and questions are valued and encouraged. Contact the author at:

Robert G. Downer
Biostatistics Director & Professor
Department of Statistics
Grand Valley State University
Allendale, MI 49401
downerr@gvsu.edu

APPENDIX: MACRO FOR COMPARISON SCORING OF INDEXED MODELS

```
%macro myscore(modnum);
      ;
      %do i = 1 %to &modnum;

          title "Model tested is model&i ";
        proc logistic inmodel = mod&i;
          score data = adoltest fitstat out = scout&i ;
        run;

            title "Misclassfication Table for model&i";
            proc freq data = scout&i;
            tables F_restreat*I_restreat;
            run;

       %end;
%mend;

%myscore(4);
```