Preparing Interaction Variables for Logistic Regression

Bruce Lund, Magnify Analytics Solutions, a Division of Marketing Associates, Detroit, MI

ABSTRACT

Interactions between two (or more) variables often add predictive power to a binary logistic regression model beyond what the original variables offer alone. In the simplest case, if X1 and X2 are zero-one valued variables, then their interaction variable is X1_X2 = X1*X2. However, X1_X2, in combination with X1 and X2, use 3 degrees of freedom. A nominal variable X_C with four values can be defined from X1 and X2 with values X_C = compress(X1 || X2). Perhaps a collapsing of the four levels of X_C to three values (having 2 d.f.) would provide nearly as much predictive power as the saturated model X1, X2, X1_X2 while providing more predictive power than X1, X2 alone. In this paper this question is answered for interactions of nominal or numeric X1 and X2, each with 2 or more levels. First, the user creates X_C. Then a "best-collapse" algorithm optimally collapses the levels of X_C until a stopping point is reached that provides a trade-off between degrees of freedom and predictive power. All data processing was performed using Base SAS®.

INTRODUCTION

We begin the paper with a simple example to motivate the following sections. Consider two nominal binary predictors X1 and X2, a binary target Y, and a frequency variable W as shown in the data step below:

EXAMPLE 1 (hypothetical data)

```
data interact;
length X1 X2 $1;
input Y X1 $ X2 $ W;
datalines;
0 A 1 4
1 A 1 6
0 B 1 8
1 B 1 4
0 A 2 2
1 A 2 5
0 B 2 3
1 B 2 9
;
data interact2; set interact;
length Xc $2;
Xc = X1||X2;

proc print data = interact2;
```

```
Obs      X1      X2      Y      W      Xc
 1       A       1       0      4      A1
 2       A       1       1      6      A1
 3       B       1       0      8      B1
 4       B       1       1      4      B1
 5       A       2       0      2      A2
 6       A       2       1      5      A2
 7       B       2       0      3      B2
 8       B       2       1      9      B2
```

Main Effects Model: The value of -2 * log-likelihood (i.e. -2 * Log L) of the main effects model is **51.446**. It is obtained by running:

```
proc logistic data = interact2; class x1 x2; model y = x1 x2; freq w;
```

Saturated Model: The saturated model gives -2 * Log L **= 50.608**. It is obtained by running the code below.

```
proc logistic data = interact2; class Xc; model y = Xc; freq w;
```

This model is equivalent to running:

```
proc logistic data = interact2; class X1 X2; model y = X1 | X2 @2; freq w;
```

Another Model - The best collapse of X_C with 2 d.f.

Consider the variable X_{c_best} formed from collapsing A2 and B2 as shown in the data step creating Interact3. We will show that X_{c_best} is the best collapse (that is, having minimum -2 * Log L) of X_C with 2 degrees of freedom.

```
data interact3; set interact2; length Xc_best $5;
if Xc in ("A2" "B2") then Xc_best = "A2+B2";
else Xc_best = Xc;

proc logistic data = interact3; class Xc_best; model y = Xc_best; freq w;
```

X_{C_best} gives -2 * Log L = **50.637**.

There are six distinct ways to collapse X_C to a variable with 2 d.f. as shown in **Table 1**. X_{c_best} is seen to be the best.

Table 1 – Collapsed Levels from Example 1 with 2 d.f. [1]

Levels	-2 * log L	
A1+A2, B1, B2	50.847	
A1+B1, A2, B2	52.188	
A1+B2, A2, B1	51.174	
A2+B1, A1, B2	53.243	
A2+B2, A1, B1	**50.637**	← Best
B1+B2, A1, A2	54.940	

X_{c_best} with 2 d.f. has a value of -2 * Log L which is between the value -2 * Log L of the main effects model with 2 d.f. and the saturated model with 3 d.f.[2] One can conclude that X_{c_best} is superior to the main effects model.

Additionally, there are 7 ways to collapse the levels of X_C to a variable with 1 d.f. One of these, {A1+A2+B2, B1} gives -2 * Log L = **51.200** which is better than the -2 * Log L = **51.446** from the Main Effects Model.

Definition: A variable that results from concatenating X1 and X2 via X_C = compress(X1 || X2) followed by collapsing of one or more levels of X_C will be called an **Interaction Variable of X1 and X2**. This is not the standard usage of "interaction" but hopefully will be viewed by the reader as an appropriate extension.

X_{c_best} Defined: Generally, "X_{c_best}" will refer to the collapse of X_c having minimum -2 * Log L for a given number of degrees of freedom.

THE GENERAL CASE

Goal: Given X1 and X2 with K1 and K2 levels respectively, create X_C = compress(X1 || X2) and find an Interaction Variable for use in PROC LOGISTIC so that:

> Upon stopping the collapsing, the X_{C_best} has no more d.f. than the main effects but has greater Log Likelihood.[3] [4]

There are (K1*K2)*((K1*K2)-1)/2 distinct ways to collapse X_C to a variable with (K1*K2)-1 levels (and (K1*K2)-2 d.f.) when X1 has K1 levels and X2 has K2 levels. The number of possible collapses increases greatly when considering also the collapses with fewer than (K1*K2)-1 levels. An exhaustive manual search of all collapses is not practical.

[1] For example: A=drive slow, B=drive fast, 1=not drinking, 2=drinking, Y=0: no accident, Y=1: accident. It seems natural in this case to collapse A2 and B2 while keeping A1 and B1 separate.

[2] Although the main effects model has 2 d.f., it cannot be obtained by collapsing X_C.

[3] For practical use, the values of K1 and K2 should be modest in value, perhaps K1*K2 ≤ 40

[4] Comparison with the saturated model is not very useful when K1 and K2 exceed 2 since the d.f. used would be unacceptable.

In the remainder of the paper we will discuss a macro named **%BEST_COLLAPSE**. This is a macro whose purpose is to provide a tool kit for collapsing (binning) a predictor variable (numeric or character) for use in binary logistic regression (PROC LOGISTIC). The full parameter set for %BEST_COLLAPSE is discussed in a later section. This macro was presented by Lund and Brotherton (2013).

When %BEST_COLLAPSE is applied to interactions of X1 and X2, it provides a fast and easy-to-use method to collapse the levels of X_C in an optimal manner as discussed below. The modeler can select a stopping point for collapsing and compare the log likelihood for the collapsed variable to the log likelihood of the main effects model.

RELATED WORK

Doug Thompson presented a paper at MWSUG 2012 where he discussed several methods of constructing interactions (conventional definition) and then compared the effectiveness of these methods when they were used in fitting a logistic regression model.

%BEST_COLLAPSE PARAMETERS

The user has the choice of two **METHODS**, either Log Likelihood (LL) or Information Value (IV), as the criterion for selecting which two levels of a predictor X to collapse at each step. The best-collapse algorithm finds the pair of levels to collapse that maximize LL or IV versus all other "eligible" choices of pairs.[5] [6]

Pairs of levels that are eligible for collapse are determined by selecting the **MODE** of ALL pairs or ADJACENT (in the ordering of X) pairs.

Parameter Definitions of %BEST_COLLAPSE:

DATASET: A dataset name - either one or two levels
X: Character or numeric variable which can have MISSING values. Missing values are ignored in all calculations.
Y: Binary Target which is numeric and must have values 0 and 1 without MISSING values.
W: Numeric frequency variable which has values which are positive integer values. (If there is no weight variable in DATASET, a weight variable must be created in Dataset with a constant value of 1.)
METHOD: IV or LL (**Information Value** or **Log Likelihood** [7])
For METHOD = IV the criterion for selecting two eligible levels to collapse is to maximize the IV. The levels that are eligible for collapse are determined by the MODE parameter.
For METHOD = LL the criterion for selecting two eligible levels to collapse is to maximize the Log Likelihood. The levels that are eligible for collapse are determined by the MODE parameter.
MODE: A or J
For MODE = A **all pairs** of levels are compared when collapsing
For MODE = J only **adjacent pairs** of levels are compared when collapsing (in the ordering of X)
VERBOSE: If YES, then the entire history of collapsing is displayed in the SUMMARY REPORT. Otherwise, this history is not displayed in the SUMMARY REPORT.
LL_STAT: If YES, the LL for the Model, -2 * Log L, and the Likelihood Ratio Chi Square Probability are displayed.

Since both IV and LL compute a logarithm, all X * Y cells in the DATASET must have **non-zero counts**.

%BEST_COLLAPSE uses only PROC MEANS, PROC APPEND, and DATA STEP processing. The SAS code for the macro is given in the Appendix of this paper.

In this paper the %BEST_COLLAPSE parameters of **METHOD = LL** and **MODE = A** are used for the collapsing of an interaction variable X_C = compress(X1||X2). Using MODE = J would only be appropriate if the ordering of X_C was meaningful.

[5] Stratified Sampling of Y: In the case of LL, I have no example to show that collapsing results could be different for stratified sampling of Y with X_k as the strata (e.g. 100% of 1's and 10% of 0's by strata) versus not sampling. But I have no proof to rule this out. Stratified sampling, as above, would not affect the collapsing results using IV.

[6] In this paper the phrases "maximize Log Likelihood" and "minimize - 2 * Log Likelihood" are used interchangeably.

[7] See Appendix for methodology

OTHER METHODS OF COLLAPSING A PREDICTOR WITH BINARY TARGET

Clustering

A method of collapsing nominal predictors (using any-pairs collapsing) is based on clustering of levels using SAS PROC CLUSTER. This method selects the pair for collapsing which maximizes the Pearson chi-square. A stopping criterion is defined by selecting the iteration which produces the minimum chi-square statistic probability (right tail probability) of association between the target and the collapsed predictor. [8]

The clustering method is illustrated by Manahan (2006) who provides SAS macro code. Additional code is needed to apply the chi-square probabilities. See Manahan (2006) for other references.

Decision Tree

The predictor X can be nominal or ordinal. The leaf nodes that are the result of the splitting process define the collapsed levels. A stopping criterion must be specified.

Further discussion of a particular decision tree process is given at the end of this paper.

%BEST_COLLAPSE APPLIED TO EXAMPLE 1

Macro call: `%BEST_COLLAPSE(interact2, Xc, Y, W, LL, A, YES, YES);`

There are four Reports produced by %BEST_COLLAPSE. Two are discussed here. The third and fourth are not discussed in this paper.

1) The **COLLAPSE STEP** reports show the detail of collapsing of X_C step by step.

2) The **SUMMARY** report gives statistics for the result of each step including -2 * Log L, IV, and X_STAT where:

   - IV is Information Value statistic.
   - X_STAT is the model "c" (or AUC) for the model: PROC LOGISTIC; CLASS Xc; MODEL Y = Xc;

   Both IV and X_STAT are helpful in determining a stopping point for the collapsing.

The history of collapsing, step-by-step, is given if VERBOSE = YES in the macro call.

COLLAPSE STEP REPORTS

There is one report for each step in the collapsing of X_C.

Table 2A4
Dataset= interact2, Predictor= Xc, Target= Y, Method= LL, Mode= A
Collapse Step: Levels = 4

Obs	Xc	_TYPE_	G	B
			(Y=1)	(Y=0)
1		0	24	17
2	A1	1	6	4
3	A2	1	5	2
4	B1	1	4	8
5	B2	1	9	3

[8] SAS course notes "Predictive Modeling Using Logistic Regression" (2007).

Table 2A3
Dataset= interact2, Predictor= Xc, Target= Y, Method= LL, Mode= A
Collapse Step: Levels = 3

Obs	Xc	_TYPE_	G	B
			(Y=1)	(Y=0)
1		0	24	17
2	A1	1	6	4
3	A2+B2	1	14	5
4	B1	1	4	8

"A2+B2" shows that A2 and B2 have been collapsed

Table 2A2 is similar to the above tables and is not shown.

SUMMARY REPORT

Table 2B
Dataset= interact2, Predictor= Xc, Target= Y, Method= LL, Mode= A
Summary Report (partial list of columns)

k	-2*Log L	IV	X_STAT	L1	L2	L3	L4
4	50.6084	0.51783	0.68750	A1	A2	B1	B2
3	50.6373	0.51441	0.68382	A1	A2+B2	B1	
2	51.2002	0.45335	0.65196	A1+A2+B2	B1		

EXAMPLE 2: %BEST_COLLAPSE APPLIED TO MULTI-LEVEL X1 AND X2

"DEMO1" and "DEMO2" are used in a model to predict a customer's satisfaction with an automotive retail outlet. DEMO1 gives age ranges. DEMO2 gives educational attainment. DEMO1 has 6 levels and DEMO2 has 4 and these variables are regarded as ordinal.[9]

"Satisfaction" is coded as a binary variable Y with 1 = satisfied and 0 = not satisfied.

There are 6,241 observations in the analysis data set called DEMO_SAT. See **Table 3** below for counts.

%BEST_COLLAPSE will be used to create an interaction variable from DEMO1 and DEMO2. Although DEMO1 and DEMO2 are ordered, their concatenation Xc = DEMO1 || DEMO2 is not ordered.[10]

Some Preliminaries: Before running %BEST_COLLAPSE three tables are given. A frequency count of DEMO1 * DEMO2 is given in **Table 3**. **Table 4** gives the count of Y = 1 in each cell.

Table 3 – Counts by Grid Cell

DEMO1	DEMO2				
	1	2	3	4	Total
B	76	189	321	102	688
C	136	287	418	152	993
D	263	451	538	219	1471
E	298	564	550	243	1655
F	290	350	265	202	1107
G	114	95	76	42	327
Total	1177	1936	2168	960	6241

Table 4 – Counts of Satisfied Responses by Grid Cell

DEMO1	DEMO2				
	1	2	3	4	Total
B	41	113	193	50	397
C	109	224	292	104	729
D	208	326	384	160	1078
E	233	421	422	207	1283
F	248	275	205	181	909
G	97	74	62	33	266
Total	953	1454	1594	740	4741

[9] DEMO1: under 35, 35-44, 45-54, 55-64, 65-74, 75 and up. DEMO2: High School Grad or less, Some College/Trade School, College Degree, Post College Graduate
[10] For example, "age 35-44 || some-college/trade school" is not greater than or less than "age 45-54 || high-school grad or less"

Table 5 shows the percentage of Y=1 in each cell of the DEMO1 * DEMO2 grid. The color coding in **Table 5** shows there is not a simple pattern for finding cells with high or low density of Y = 1.

The highest percentages (red) are found in F1, G1, E4, F4. The lowest percentages (blue) are generally up and to the right in the grid.

Table 5 DEMO1-DEMO2 Grid [11]

DEMO1	DEMO2				
	1	2	3	4	Total
B	76.3%	70.9%	71.3%	53.9%	69.2%
C	80.1%	78.0%	69.9%	68.4%	73.4%
D	79.1%	72.3%	71.4%	73.1%	73.3%
E	78.2%	74.6%	76.7%	85.2%	77.5%
F	85.5%	78.6%	77.4%	89.6%	82.1%
G	85.1%	77.9%	81.6%	78.6%	81.3%
Total	81.0%	75.1%	73.5%	77.1%	76.0%

The usefulness of **Table 5** depends on the fact that DEMO1 and DEMO2 are ordinal. If both of DEMO1 and DEMO2 were nominal, the table would be informative but statements such as "up and to the right" would have no meaning.

Main Effects Model: The first step was to provide a baseline -2 * Log L from the Main Effects Model for comparison to interactions. The fit of the Main Effects Model with DEMO1 and DEMO2 as CLASS variables is shown below. The Main Effects Model gives 2 * Log L = 6810.203 and both DEMO1 and DEMO2 are significant predictors.

PROC LOGISTIC DATA = DEMO_SAT; CLASS DEMO1 DEMO2; MODEL Y = DEMO1 DEMO2; [12]

```
(partial output)

         Model Fit Statistics
              Intercept    Intercept and
Criterion          Only       Covariates
-2 Log L       6883.555         6810.203

         Type 3 Analysis of Effects
                            Wald
Effect          DF    Chi-Square    Pr > ChiSq
DEMO1            5       47.0306        <.0001
DEMO2            3       15.2170        0.0016
```

The Challenge: Can an interaction variable with no more than 8 d.f. be found by %BEST_COLLAPSE with -2 * Log L smaller than the **6810.203** from the Main Effects Model?

RUNNING %BEST_COLLAPSE

%BEST_COLLAPSE was run on X_C = DEMO1 || DEMO2 as shown:

```
data interact; set Demo_Sat;
length Xc $2;
Xc = DEMO1 || DEMO2;

%BEST_COLLAPSE(interact, Xc, Y, W, LL, A, NO, YES);
```

The results are shown in the SUMMARY report given in **Table 6**.

[11] Tables 3 and 4 were ODS output from PROC FREQ to Excel. Table 5 was created by a manual Excel manipulation using Tables 3 and 4.

[12] DEMO1 and DEMO2 might be recoded as numeric and used in PROC LOGISTIC DATA = DEMO_SAT; DEMO2; MODEL Y = DEMO1 DEMO2; This imposes an unrealistic interval scale on DEMO2 and requires the selection of a representative age from each age range including the open-end ranges.

Table 6

Dataset= DEMO_SAT, Predictor= Xc, Target= Y, Method= LL, Mode= A

Summary Report (some columns are omitted)

k	-2 * Log L	IV	X_STAT
24	6763.62	**0.10857**	0.58527
23	6763.62	0.10857	0.58527
22	6763.62	0.10857	0.58526
21	6763.62	0.10857	0.58526
20	6763.62	0.10857	0.58526
19	6763.63	0.10856	0.58524
18	6763.63	0.10856	0.58523
17	6763.65	0.10854	0.58518
16	6763.67	0.10852	0.58515
15	6763.69	0.10850	0.58511
14	6763.74	0.10846	0.58506
13	6763.78	0.10842	0.58498
12	6763.85	0.10836	0.58497
11	6763.94	0.10827	0.58475
10	6764.05	0.10818	0.58469
9	6764.37	0.10792	0.58407
8	6764.93	0.10740	0.58362
7	6765.87	0.10657	0.58283
6	6766.96	0.10566	0.58125
5	6769.49	0.10261	0.58086
4	**6772.66**	**0.09977**	**0.57656**
3	6785.62	0.08913	0.57372
2	6819.21	0.06029	0.53864

As stated by Siddiqi (2006) page 81, an IV value of **0.10857** (for the saturated model) is just within the range that Siddiqi designated as “medium strength” (per Siddiqi: 0.1 to 0.3).

The main effects model used 8 degrees of freedom and produced -2 * Log L of **6810.20**. Each of k = 3, …, 9 (with d.f. 2, …, 8) provides -2 * Log L for Xc_{C_best} which is lower than the main effect benchmark of **6810.20**.

HOW TO SELECT k:

The selection of a stopping point “k” is somewhat subjective. The modeler wants predictive power as measured by log-likelihood, IV, and X_STAT but also the pattern of cells within a level of Xc_{C_best} across the DEMO1-DEMO2 grid (**Table 5**) should be coherent.[13] Specifically, the cells collapsed together in a level should be connected and clustered within the DEMO1-DEMO2 grid.

This led to the selection of k = 4. **Tables 7 and 8** show the levels of Xc_{C_best} for k = 4 (**Table 7**) and the pattern of the cells within these levels across the DEMO1-DEMO2 grid (**Table 8**). Although the cells in the fourth level E4+G1+F1+F4 are disconnected, we think we have a behavioral rationale for this pattern.

The "price" for selecting k = 4 was a lower IV statistic than for selecting, for example, k = 9. But selecting k = 4 provided a savings of 5 degrees of freedom, a coherency in the grid pattern, and still an out-performance of the main effects model.

Table 7

Dataset= DEMO_SAT, Predictor= Xc, Target= Y, Method= LL, Mode= A

Collapse Step: Levels = 4

Xc_best	Sat.	Not Sat.	Sat. Rate
	Y = 1	Y = 0	%(Y=1)
TOTAL	4741	1500	76.0%
B1+E3+F3+E2+C1+G3+C2+G2+E1+D1+F2+G4	2324	678	77.4%
B2+B3+D3+D2+D4+C3+C4	1629	659	71.2%
B4	55	47	53.9%
E4+G1+F1+F4	733	116	86.3%

[13] The orderings of DEMO1 and DEMO2 provide the basis for determining "coherency".

Table 8 DEMO1-DEMO2 Grid – Color Coding of Cells in Each Level

DEMO1	DEMO2				
	1	2	3	4	Total
B	76.3%	70.9%	71.3%	53.9%	69.2%
C	80.1%	78.0%	69.9%	68.4%	73.4%
D	79.1%	72.3%	71.4%	73.1%	73.3%
E	78.2%	74.6%	76.7%	85.2%	77.5%
F	85.5%	78.6%	77.4%	89.6%	82.1%
G	85.1%	77.9%	81.6%	78.6%	81.3%
Total	81.0%	75.1%	73.5%	77.1%	76.0%

If the modeler has available a validation sample whose only purpose is to confirm predictor variable preparation, then the satisfaction rates from X_{c_best} for k=4 from the training sample can be compared to the same rates from the validation sample. If the rates are similar, then the preparation of the predictor variable is validated.

If there is no validation sample, then the modeler would proceed to include X_{c_best} among the group of variables being considered for selection for the logistic regression model.

THERE IS THE REQUIREMENT FOR JUDGMENT BY THE MODELER

In this paper the creation of tables like **Table 8** for k = 3, …, 9 was manual and their interpretation was subjective.

BUT THERE IS A PROBLEM

An optimal collapse of X_C at level k can lead to a sub-optimal collapse at level k-1. This, in fact, is the case for EXAMPLE 2. A **better k = 4** solution is given in **Table 9A**. Cell **E2** moved from row 1 in **Table 7** (%BEST_COLLAPSE solution) to row 2 in **Table 9A**.

Table 9A – Better Solution for k = 4

Xc_best	Sat.	Not Sat.	Sat. Rate
	Y = 1	Y = 0	%(Y=1)
TOTAL	4741	1500	76.0%
G3+C1+D1+F2+G4+E1+C2+G2+F3+E3+B1	1903	535	78.1%
B2+B3+C3+C4+D2+D3+D4+**E2**	2050	802	71.9%
B4	55	47	53.9%
F4+F1+E4+G1	733	116	86.3%

Table 9B – Better Solution for k = 4 (since -2 * Log L in Table 9B is less than in Table 9C)
Summary Report

k	-2 * Log L	IV	X_STAT
4	**6772.38**	0.10014	0.57672

Table 9C – Results for k = 4 from Table 6
Summary Report

k	-2 * Log L	IV	X_STAT
4	**6772.66**	0.09977	0.57656

However, the differences between **Table 9B** and **Table 9C** in the values of -2 * Log L, IV, and X_STAT are small enough to be ignored.

We determined that the k = 4 collapse was not optimal by comparing the results of %BEST_COLLAPSE with the results of splitting X_C using a Decision Tree as discussed below.

DECISION TREES

JMP® [14] has a decision tree called PARTITION. In the case of a single predictor X and a nominal binary target Y, the entropy criterion (denoted by **G^2** in JMP output) is used to determine where to split. Here, the entropy criterion for splitting is equivalent to Log Likelihood criterion for collapsing.[15]

The determination that %BEST_COLLAPSE was not optimal at k = 4 was made by running JMP PARTITION on X_C and then comparing the "leaves" after 3 splits to the %BEST_COLLAPSE for k = 4 levels.

Collapsing is stepwise top-down (starting with terminal leaves) and partitioning is stepwise bottoms-up (starting at the trunk). Despite both using Log Likelihood as the collapsing / splitting criterion, results of these processes may not be the same. In fact, the splitting process by PARTITION using entropy also may become sub-optimal.[16]

WHAT TO DO?

If the collapsing process ends after a few steps, the opportunity that a collapse occurred that led to sub-optimality is small. If the collapsing is extensive, as in Example 2, there is more chance that the collapsing process becomes sub-optimal. The difference between ideal and achieved solutions may be negligible but the magnitude of this difference would be unknown to the user when using %BEST_COLLAPSE.

However, the user does know the values -2 * Log L, IV, and X_STAT and can compare the achieved -2 * Log L to the -2 * Log L from the main effects model. These are solid criteria by which to judge the usefulness of an interaction variable.

Additionally, if the user has JMP available, then PARTITION can be run using entropy as the splitting criterion. The user can inspect the first split. If the cells in the left and right branches are the same as the cells from %BEST_COLLAPSE levels for k = 2, then %BEST_COLLAPSE was optimal, at least, at the final step.[17]

See Lund and Raimi (2012) and Lund and Brotherton (2013) for related discussions.

REFERENCES

Lund, B. and Brotherton, D. (2013). "Information Value Statistic", *MWSUG 2013, Proceedings,* Midwest SAS Users Group, Inc., paper AA-14.

Lund, B. and Raimi, S. (2012). "Collapsing Levels of Predictor Variables for Logistic Regression and Weight of Evidence Coding", *MWSUG 2012, Proceedings,* Midwest SAS Users Group, Inc., Paper SA-03.

Manahan, C. (2006). "Comparison of Data Preparation Methods for Use in Model Development with SAS Enterprise Miner", *Proceedings of the 31th Annual SAS Users Group International Conference*, Paper 079-31.

Siddiqi, N. (2006). *Credit Risk Scorecards,* Hoboken, NJ: John Wiley & Sons, Inc.

Thompson, D. (2012). "Methods for Interaction Detection in Predictive Modeling Using SAS", *MWSUG 2012, Proceedings,* Midwest SAS Users Group, Inc., Paper SA-01.

[14] See jmp.com. In this paper JMP version 9 was used.

[15] $G\hat{}2_{right}$ + $G\hat{}2_{left}$ = -2 * Log L where -2 * Log L is computed for the binary variable S that is "1" for right and "0" for left in the logistic regression: PROC LOGISTIC; CLASS S; MODEL Y = S; FREQ W;

[16] For EXAMPLE 2 the k=22 collapse from %BEST_COLLAPSE is F2+G4, B3+D3 and 20 other single cells for -2*log L = 6763.62 The corresponding split from JMP PARTITION is F2+G4, B4+D3 and 20 other single cells for -2*log L = 6775.15. Both %BEST_COLLAPSE and PARTITION selected F2+G4 and 22 other single cells for k=23.

[17] It is possible that %BEST_COLLAPSE and PARTITION could become sub-optimal at an intermediate step but return to optimality by the final step. This is the case, for example, for PARTITION when going from k=22 (sub-optimal) to k=23 (optimal).

CONTACT INFORMATION

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

Bruce Lund
Marketing Associates, LLC
777 Woodward Ave, Suite 500
Detroit, MI, 48226
blund@twmi.rr.com and blund@marketingassociates.com

All code in this paper is provided by Marketing Associates, LLC. "as is" without warranty of any kind, either express or implied, including but not limited to the implied warranties of merchantability and fitness for a particular purpose. Recipients acknowledge and agree that Marketing Associates shall not be liable for any damages whatsoever arising out of their use of this material. In addition, Marketing Associates will provide no support for the materials contained herein.

Appendix

%BEST_COLLAPSE METHODOLOGY FOR LOG LIKELIHOOD

Let G_k be the count of records with Y = 1 where $X = X_k$. Let B_k be the count of records with Y = 0 where $X = X_k$.
The Log Likelihood of X and Y is given by $LL = \sum_{k=1}^{K} (G_k * \log(G_k/(G_k + B_k)) + B_k * \log(B_k/(G_k + B_k)))$

The k^{th} term of LL will be defined as shown:

$$LL_k = G_k * \log(G_k/(G_k + B_k)) + B_k * \log(B_k/(G_k + B_k))$$

If the i^{th} and j^{th} levels of X are collapsed, then the new LL includes this term:

$$LL_{i\_j} = (G_i + G_j) * \log((G_i + G_j)/(G_i + G_j + B_i + B_j)) + (B_i + B_j) * \log((B_i + B_j)/(G_i + G_j + B_i + B_j))$$

Among eligible pairs (i,j) the %BEST_COLLAPSE algorithm finds the (i,j) pair that minimizes the expression D where:

$$D = LL_i + LL_j - LL_{i\_j}$$

%BEST_COLLAPSE SAS MACRO

```
%MACRO BEST_COLLAPSE(DATASET, X, Y, W, METHOD, MODE, VERBOSE, LL_STAT);
* Best Collapse Version 6a;

options ls=230 nocenter;

/* !!! WARNING: There is No Input Data Checking in this Program !!! */

/* DATASET is a dataset name - either one or two levels */
/* X (Predictor) is a numeric or character variable which can have MISSING values */
   /* Missing values of X are ignored in all calculations */
   /*  "___x_Char" is RESERVED.  Do not use ___x_char as name of predictor */
/* Y (Target) has values 0 and 1 without MISSING values */
/* W (Freq) has values which are positive integers.  It represents a FREQUENCY variable */
/* METHOD is IV or LL */
   /* For METHOD = IV the collapsing maximizes IV
      For METHOD = LL the collapsing maximizing Log likelihood */
/* MODE is A or J */
```

```
   /*    For MODE = A all pairs of levels are compared when collapsing IV
      For MODE = J only adjacent pairs of levels are compared when collapsing IV */
/* VERBOSE = YES is used to display the entire history of collapsing in the SUMMARY REPORT */
   /* Otherwise this history is not displayed in the SUMMARY REPORT */
/* LL_STAT = YES is used to display Log Likelihood for Model and Likelihood Ratio Chi Square
Probability */

/* It is required that ALL cell counts in the X-Y Frequency Table are positive */
/* The Program ENDS if there is a zero cell and prints "ZERO CELL DETECTED" */

%global num_levels;
%global STOP;
%global LL_inter;

%IF &METHOD NE LL
%THEN
%DO;
   %IF &METHOD NE IV
   %THEN
   %DO;
      %PUT INVALID SUBSTITUTION METHOD = &METHOD;
      %PUT ENDING EXECUTION;
      %GOTO EXIT;
      %END;
   %END;

proc means data = &DATASET noprint; class &X; var &Y; freq &W;
types () &X;
output out = mean_out_0
sum = y;
run;

%let STOP = NO;

data mean_in; set mean_out_0 nobs = num_levels;
   length ___x_char $75;
   LABEL ___x_char = "&X";
   keep ___x_char G B;
   ___x_char = trim(&X);
   B = _freq_ - y;
   G = y;
   if _n_ = 1 then call symput('num_levels',num_levels - 1); /*Subtracts 1 for _TYPE_=0*/
   if _n_ = 1 then call symput('num_levels_minus1',num_levels - 2);
   if _n_ = 1
   then
   do;
      LL_Inter = B*log(B/_freq_) + G*log(G/_freq_);
      call symput('LL_inter',LL_inter);
      end;
   if G = 0 or B = 0 then call symput("STOP","YES");
   if _type_ = 1 then output;
run;

   %IF &STOP = YES %THEN
   %DO;
      %PUT ZERO CELL DETECTED;
      %PUT ENDING EXECUTION;
      %GOTO EXIT;
      %END;

%MACRO BEST_COLLAPSE_LEVELS(NUM_LEVELS_R);

proc means data = mean_in noprint; class ___x_char; var G B;
output out = mean_out(keep = ___x_char G B _type_)
sum = G B;
run;

proc print data = mean_out label;
title1
"Dataset= &DATASET, Predictor= &X, Target= &Y, Method= &METHOD, Mode= &MODE, RUN ON &SYSDATE
&SYSTIME";
```

```
title2 " ";
title3 "Collapse Step: Levels = &num_levels_r";
run;

data
   denorm&num_levels_r
   mean_in(keep = ___x_char G B)
   %IF ("%UPCASE(&MODE)" = "J" AND &num_levels_r = &num_levels)
   %THEN %DO;
      Split(keep = split1 - split%cmpres(&num_levels_minus1))
      %END;
      ;
   set mean_out end = eof;

   length L1 - L&num_levels_r $75;
   length ___x_char $75;
   array Gx{*} G1 - G&num_levels_r;
   array Bx{*} B1 - B&num_levels_r;
   array LEVELx{*} $ L1 - L&num_levels_r;

   array Splitx{*} Split1 - Split%cmpres(&num_levels_minus1);
   retain G_total B_Total k collapsing_to IV LL_Model LRCS LR_Chi_Sq_Prob;
   retain G1 - G&num_levels_r B1 - B&num_levels_r L1 - L&num_levels_r;
   if _type_ = 0
   then
   do;
      G_total = G;
      B_total = B;
      k = 0;
      IV = 0;
      LL_Model = 0;
      end;
   if _type_ = 1
   then
   do;
      k + 1;
      collapsing_to = k - 1;
      Gx{k} = G;
      Bx{k} = B;
      LEVELx{k} = trim(left(___x_char));
      IV = IV + (G/G_total - B/B_total)*log((G/G_total) / (B/B_total));
      LL_Model = LL_Model + G * log(G/(G+B)) + B * log(B/(G+B));
      end;

   if eof
   then
   do;
      Minus2_LL = -2*LL_Model;
      LRCS = -2 * (&LL_inter - LL_Model);
      LR_Chi_Sq_Prob = 1 - PROBCHI(LRCS,k-1);
      LABEL Minus2_LL = "-2*Log L";
      LABEL LR_Chi_Sq_Prob = "Prob(x > LR_Chi_Sq)";
      LABEL LRCS = "Lik-Ratio Chi_Sq";
      LABEL LL_Model = "LL for Model";

      %IF "%UPCASE(&MODE)" = "J" AND "%UPCASE(&METHOD") = "IV" AND &num_levels_r = &num_levels
      %THEN
      %DO;
         do r = 1 to &num_levels_r - 1;
            SUM_G_left = 0; SUM_B_left = 0;
            do s = 1 to r;
               SUM_G_left = SUM_G_left + Gx{s}/G_total;
               SUM_B_left = SUM_B_left + Bx{s}/B_total;
               end;
            SUM_G_right = 0; SUM_B_right = 0;
            do s = r+1 to &num_levels_r;
               SUM_G_right = SUM_G_right + Gx{s}/G_total;
               SUM_B_right = SUM_B_right + Bx{s}/B_total;
               end;
            Splitx{r} = (SUM_G_left - SUM_B_left)      * log(SUM_G_left / SUM_B_left) +
                     (SUM_G_right - SUM_B_right)       * log(SUM_G_right / SUM_B_right);
            end;
         OUTPUT Split;
         %END;
      %IF "%UPCASE(&MODE)" = "J" AND "%UPCASE(&METHOD") = "LL" AND &num_levels_r = &num_levels
      %THEN
      %DO;
         do r = 1 to &num_levels_r - 1;
            SUM_G_left = 0; SUM_B_left = 0;
            do s = 1 to r;
               SUM_G_left = SUM_G_left + Gx{s};
               SUM_B_left = SUM_B_left + Bx{s};
               end;
            SUM_G_right = 0; SUM_B_right = 0;
            do s = r+1 to &num_levels_r;
               SUM_G_right = SUM_G_right + Gx{s};
               SUM_B_right = SUM_B_right + Bx{s};
               end;
            Splitx{r} = SUM_G_left*log(SUM_G_left/(SUM_G_left+SUM_B_left)) +
                     SUM_B_left*log(SUM_B_left/(SUM_G_left+SUM_B_left)) +
                     SUM_G_right*log(SUM_G_right/(SUM_G_right+SUM_B_right)) +
                     SUM_B_right*log(SUM_B_right/(SUM_G_right+SUM_B_right));
            end;
         OUTPUT Split;
         %END;

      min_C = 99999999;
      X_STAT = 0;
      C_STAT = 0;
      do i = 1 to &num_levels_r - 1;
         %IF "%UPCASE(&MODE)" = "A" %THEN %DO; do j = i+1 to &num_levels_r; %END;
         %IF "%UPCASE(&MODE)" = "J" %THEN %DO; do j = i+1 to i+1; %END;
            %IF &METHOD = LL
            %THEN
            %DO;
               L_i = Gx{i}*log(Gx{i}/(Gx{i}+Bx{i})) + Bx{i}*log(Bx{i}/(Gx{i}+Bx{i}));
               L_j = Gx{j}*log(Gx{j}/(Gx{j}+Bx{j})) + Bx{j}*log(Bx{j}/(Gx{j}+Bx{j}));
               C_ij = L_i + L_j -
                  ( (Gx{i}+Gx{j})*log((Gx{i}+Gx{j})/(Gx{i}+Gx{j}+Bx{i}+Bx{j})) +
                  (Bx{i}+Bx{j})*log((Bx{i}+Bx{j})/(Gx{i}+Gx{j}+Bx{i}+Bx{j})) );
               if C_ij <= min_C
               then
               do;
                  i_index = i;
                  j_index = j;
                  min_C = C_ij;
               %END;
            %ELSE %IF &METHOD = IV
            %THEN
            %DO;
               L_i = ( Gx{i}/G_total - Bx{i}/B_total ) *
                     log( (Gx{i}/G_total) / (Bx{i}/B_total) );
               L_j = ( Gx{j}/G_total - Bx{j}/B_total ) *
                     log( (Gx{j}/G_total) / (Bx{j}/B_total) );
               C_ij = L_i + L_j -
                     ( (Gx{i} + Gx{j})/G_total - (Bx{i} + Bx{j})/B_total ) *
                     log( ((Gx{i} + Gx{j})/G_total) / ((Bx{i} + Bx{j})/B_total) );
               if C_ij <= min_C
               then
               do;
                  i_index = i;
                  j_index = j;
               min_C = C_ij;
               %END;

               if &num_levels_r >= 3
               then
               do;
                  LO = log((Gx{i}*Bx{j})/(Gx{j}*Bx{i}));
                  LO_SD = sqrt(1/Gx{i} + 1/Gx{j} + 1/Bx{i} + 1/Bx{j});
                  LOplus2SD = LO + 2*LO_SD;
                  LOminus2SD = LO - 2*LO_SD;
```

```
                 end;

              end;
           END; /* END OF J loop */

        do j = i+1 to &num_levels_r;
           C_STAT = C_STAT  + Bx{I}*Gx{J};
           X_STAT = X_STAT  + ABS(Bx{i}*Gx{j} - Gx{i}*Bx{j});

           end; /* END OF: J loop */
        end; /* END OF: I loop */

      do i = 1 to &num_levels_r;
        C_STAT  = C_STAT + .5*Bx{I}*Gx{I};
        END;

      C_PAIR  = B_TOTAL * G_TOTAL;
      C_STAT  = MAX( C_STAT  / C_PAIR,  1 - C_STAT  / C_PAIR );
      X_STAT  = .5 * (X_STAT  / C_PAIR  + 1);

      OUTPUT denorm&num_levels_r;

      do i = 1 to &num_levels_r;
        if i = i_index or i = j_index
           then ___x_char = compress(LEVELx{i_index}||"+"||LEVELx{j_index});
         else ___x_char = LEVELx{i};
        G = Gx{i};
        B = Bx{i};
        OUTPUT mean_in;
        end;

      end; /* END OF: if eof then do */
run;
proc append base = denorm data = denorm&num_levels_r force nowarn;
run;
%MEND;

%MACRO INTER;
proc delete data = denorm;
run;
%do k = &num_levels %to 2 %by - 1;
   %BEST_COLLAPSE_LEVELS(&k);
   %end;
proc print data = denorm noobs label;
var K
%IF &LL_STAT = YES %THEN LL_Model Minus2_LL LRCS LR_Chi_Sq_Prob;
IV X_STAT C_STAT
%IF &VERBOSE = YES %THEN L1 - L%cmpres(&num_levels); ;
format IV X_STAT C_STAT 8.5;
title2 " ";
title3 "Summary Report";
run;
%IF ("&MODE" = "J")
%THEN %DO;
   proc print data = Split;
   title2 " ";
   title3 "Final Step Binary Splits for MODE = J";
   %END;
run;

proc print data = denorm noobs;
var K  IV X_STAT collapsing_to LO LO_SD LOminus2SD LOplus2SD;
format IV X_STAT LO LO_SD LOminus2SD LOplus2SD 8.5;
title2 " ";
title3 "Log-odds with 95% CI";
run;
%MEND;
   %INTER;
%EXIT: %MEND;
```