Screening, Binning, Transforming Predictors for a Generalized Logit Model

by Bruce Lund
Statistical Modeling Consultant and Trainer, Novi, MI
blund_data@mi.rr.com or blund.data@gmail.com

Slides for Today will be posted to MiSUG Site

Terminology

Most books use the term “multinomial logit” for the Model of today’s talk.

… SAS® uses “generalized logit” instead.

We’ll use the SAS terminology.

Goals for Today

- Discuss Screening, Binning, Transforming of Predictors for Binary Logit Model
  - Create a dataset called BINARY with target Y with 2 levels
  - Apply my SAS macros for screening, binning, transforming X's from BINARY
- Extend Screening, Binning, Transforming of Predictors to Generalized Logit
  - Create a dataset called GL with target Y with 3 levels
  - Apply my SAS macros for screening, binning, and transforming X's from GL

SAS code for datasets BINARY and GL are given at the end of the slides.

Send me an email to obtain my macros. But, they remain "beta" versions.

… I'll talk for about 20 minutes and leave some time for Q/A

Binary Logistic Regression Model

- In dataset BINARY … there is 0/1 Target Y, Classification X1 (5 levels), Numeric X2
  - Let $P(Y=0 \mid X1\ X2) = p$
  - Here is a Binary Logistic Model:

$$LOG(p/(1-p)) = \alpha + \beta_1 X1_{Dum1} + \beta_2 X1_{Dum2} + \beta_3 X1_{Dum3} + \beta_4 X1_{Dum4} + \eta X2 = xbeta$$

p/(1-p) is called “Odds”, LOG(p/(1-p)) is called “Log-Odds”, and Right-Side is “xbeta

- X1 is expressed via 4 dummy variables (e.g. $X1_{Dum1}$ = (X1 = “01”))
- But no term for the 5th level of X1 … it is the reference level for X1
- From LOG(p/(1-p)) = xbeta … Solve for p = exp(xbeta)/(1 + exp(xbeta))
- Given a sample dataset, PROC (HP)LOGISTIC can estimate the coefficients α, β_1-β_4, η

Go to NEXT SLIDE

Screening and Transforming Predictors for Binary Logit

BUT, Questions:

A. Can X1 and X2 pass a "screener"
   - Are they strong enough to be included in a model ??

B. If "YES",
   - Can X1 and/or X2 be "transformed" to achieve a better model … a better fit ??

Screening of classification predictors for Binary Logistic

- Several ways to "screen" (eliminate) classification predictors like X1
  - Information Value is widely used … (see later slide)
  - Alternatively, the Likelihood Ratio Chi-Square statistic can be used … see below

```
PROC FREQ DATA=BINARY;
TABLES Y * X1 / CHISQ; run;
```

Statistic	DF	Value	Prob
Likelihood Ratio Chi-Square	4	33.364	<.0001

Likelihood Ratio Chi-Square (LRCS)

- For large n, LRCS is chi-square with L-1 d.f. where X has L levels. (Here X1 has L=5)
- If Likelihood Ratio Chi-Square (LRCS) = 0, then no predictive power … i.e. X is not useful
- Large LRCS gives a small right-tail "Prob" … implying X has power to predict Y
- … X1 appears to have power with Prob <.0001

IV of classification predictor X1 for Binary Logit

X1	Y = 0	Y = 1	Col % Y=0 "b_k"	Col % Y=1 "g_k"	$\log(g_k/b_k)$ = X1_woe	D $=(g_k - b_k)$	D * X1_woe
01	156	65	19.50%	32.50%	-0.511	-13.00%	0.0664
02	135	52	16.88%	26.00%	-0.432	-9.13%	0.0394
03	164	31	20.50%	15.50%	0.280	5.00%	0.0140
04	168	28	21.00%	14.00%	0.405	7.00%	0.0284
05	177	24	22.13%	12.00%	0.612	10.13%	0.0619
SUM	800	200	100%	100%		IV =	0.2102

IV Range	Interpretation
IV < 0.02	"Not Predictive"
IV in [0.02 to 0.1)	"Weak"
IV in [0.1 to 0.3)	"Medium"
IV $\geq$ 0.3	"Strong"

Siddiqi (2017, p. 179) *Intelligent Credit Scoring*

Binning of classification predictors for Binary Logit

- A predictor, like X1, that "passes the screeners" should be "binned"

  Binning:
  - Reduces number of levels of X1 while maintaining (most of) predictive power
  - Eliminates low frequency levels of X1 by collapsing with other levels ("clean-up")
- Binning Algorithm does Binning while preserving "Predictive Power" as measured by one of:
  - **Information Value**

    or
  - **LRCS** (Note: equivalent to "entropy" for the purpose of binning)

- If IV is used, the algorithm maximizes **Information Value** at each step of binning

  or, if LRCS is used, then …
- the binning algorithm maximizes **LRCS** at each step of binning

(Yet, it is true that suboptimal final solutions can be produced by these binning algorithms)

Binning of classification X1 for Binary Logit … continued

- I have an SAS macro called %NOD_BIN for binning predictors like X1
- Using LRSC at each step: Two levels of X1 are combined so as to maximize LRSC
  - For predictor X1 … a case can be made for stopping at three BINs
  - … then X1 is transformed to have three Levels {01, 02}, {03, 04}, {05}

Bins	IV	LRSC	L1	L2	L3	L4	L5
5	0.2102	33.36	01	02	03	04	05
4	0.2094	33.24	01+02	03	04	05	
3	0.2080	33.04	01+02	03+04	05		
2	0.1997	31.92	01+02	03+04+05			

- Using IV the binning of X1 is identical to using LRSC (although this is not always true)

Bins	IV	LRSC	L1	L2	L3	L4	L5
5	0.2102	33.36	01	02	03	04	05
4	0.2094	33.24	01+02	03	04	05	
3	0.2080	33.04	01+02	03+04	05		
2	0.1997	31.92	01+02	03+04+05			

Transforming Continuous Predictor X for Binary Logistic

Function Selection Procedure (FSP) was introduced in the 1990's to find transformations
See Royston and Sauerbrei (2008) *Multivariable Model Building*

If X ≥ 1, then good … otherwise translate X so that min(X)=1

Consider the "fractional polynomials":

$G1=X^{-2}$, $G2=X^{-1}$, $G3=X^{-0.5}$, $G4=\log(X)$, $G5=X^{0.5}$, $G6=X^2$, $G7=X^3$, $G8=X$ (un-transformed)

FSP picks one of the following four decisions:

1. Screen out (eliminate) X
2. Specifies X, un-transformed
3. Specifies one of the G's (excluding G8=X)
4. Specifies one of 36 pairs:
   a) two G's (28 pairs) … or …
   b) Gi and Gi*Log(X) for i = 1 to 8 (8 pairs)

See Royston and Sauerbrei for explanation of (1) to (4)

For example: decision "4" might be (a) $X^{0.5}$ and X^2 or (b) X^{-1} and X^{-1}*Log(X)

Transforming Continuous, apply to X2

I have a macro %FSP_8LR_Glogit to run FSP

Applying FSP to predictor X2 from BINARY:

- A requirement of FSP is the min(X2) ≥ 1
  - First translate X2 by +3.70612 … this makes min(X2) = 1
- Then, in this example, FSP finds these transforms (decision "4")

  $G2=(X2)^{-1}$ $G7=(X2)^{3}$ …

Translate X2 = X2+3.70612;

Run PROC LOGISTIC; MODEL Y = G2 G7 <other X's>;

Go to NEXT SLIDE

Final Binary Logit Model

```
DATA BINARY_T; SET BINARY;
if X1 in ( "01" "02" ) then X1_bin = 1;
if X1 in ( "03" "04" ) then X1_bin = 2;
if X1 in ( "05" ) then X1_bin = 3;
X2 = X2+3.70612;
G2=X2**(-1);
G7=X2**3;
run;
PROC LOGISTIC DATA= BINARY_T;
CLASS X1_bin;
MODEL Y = X1_bin G2 G7;
run;
```

Analysis of Maximum Likelihood Estimates				
Parameter		DF	Estimate	Pr > ChiSq
Intercept		1	-3.547	0.0003
X1_bin	1	1	-0.637	<.0001
X1_bin	2	1	0.190	0.1336
G2		1	12.336	<.0001
G7		1	0.026	<.0001

Model is improved with these transformations

Using a Validation Dataset:

Transformed Model gives Better Fit than Un-Transformed Model

- Higher c-Stat,
- Lower Average Squared Error

MODEL	c-Stat	AVG SQ ERROR
Un-Transformed (CLASS X1; MODEL Y=X1 X2;)	0.596	0.1597
Transformed (CLASS X1_bin; MODEL Y=X1_bin G2 G7;)	0.663	0.1542

Generalized Logit vs. Cumulative Logit

Generalized Logit: Target Y has more than 2 levels and they are not (meaningfully) ordered:

- Fast Food Chains: 1= McDonald's, 2= Wendy's, 3= Burger King
- Disease: 1= respiratory, 2= digestive, 3= circulatory, 4= no disease
- ?? Track Events: 1= 100 meters, 2= 400 meters, 3= 1500 meters, 4= marathon

If levels of Y have a meaningful ordering, then correct model is the Cumulative Logit Model

- Severity of Illness: 1= No Illness, 2= Mild, 3= Severe

For more than 2 levels, the Cumulative Logit and Generalized Logit are fundamentally different

Go to NEXT SLIDE

The Generalized Logit

- Dataset GL has:
  - Target Y with three unordered levels 1, 2, 3
  - Classification X1 (5 levels) and Numeric X2
- Let P(Y=1 | X1 X2) = p1, P(Y=2 | X1 X2) = p2, P(Y=3 | X1 X2) = p3
- Now (arbitrability) pick **Y=3** as the reference level (p3 is in denominator of odds)
- … the Odds of outcome 1 to outcome 3 are p1/p3
- … the Odds of outcome 2 to outcome 3 are p2/p3
- General Logistic Model is given in terms of Log-Odds for two response equations:

$$\mathrm{LOG}(p1/p3) = \alpha_1 + \beta_{1,1}X1_{\mathrm{Dum1}} + \beta_{1,2}X1_{\mathrm{Dum2}} + \beta_{1,3}X1_{\mathrm{Dum3}} + \beta_{1,4}X1_{\mathrm{Dum4}} + \eta_1\, X2$$

$$\mathrm{LOG}(p2/p3) = \alpha_2 + \beta_{2,1}X1_{\mathrm{Dum1}} + \beta_{2,2}X1_{\mathrm{Dum2}} + \beta_{2,3}X1_{\mathrm{Dum3}} + \beta_{2,4}X1_{\mathrm{Dum4}} + \eta_2\, X2$$

- There are 3-1 = 2 *distinct* coefficients for each predictor … 1 for each response equation
- Reference level Y=3 seems to play a distinctive role.
- But choice of reference level does not affect PROBABILITIES or MODEL Fit
- Coefficients depend on choice of reference but dependence this is not meaningful

The Generalized Logit

BUT, Questions:

A. Can X1 and X2 pass a “screener”
   - Are these strong enough to be included in model ??

B. If YES,
   - Can X1 and/or X2 be “transformed” to achieve a better model ??

LRCS is used as a screener

A generalized **IV** is an alternative for screening … requires a definition … next slides

LRCS of X1 for Generalized Logit

I have a macro that computes LRCS for multiple classification X's in one macro call
Here, only X1 is used.

%MULTI_LOGIT_SCREEN_1(GL, Y, X1,);

Var_Name	Levels	Log_L_Intercept	Log_Likelihood	LRCS	df	Pr>ChiSq
X1	5	-826.050	-809.284	33.5320	8	.000049374

LRCS = 33.5320 with Pr>ChiSq < .0001
... implying X1 has power to predict Y

Go to NEXT SLIDE

IV for X1 for Generalized Logit

"Generalized **IV**" requires a "reference level for Y" having special characteristics

The reference level should have these characteristics:

- Large count vs. counts for other outcome levels
- Unique meaning … distinguished from other outcome levels

- Can't use "Generalized IV" unless willing to designate a special level as reference.

E.g. Disease Status: 1= respiratory, 2= digestive, 3= circulatory, **4= no disease**

E.g. Shopping for Vehicle: 1= cash, 2= loan, 3= lease, **4= did not acquire**

E.g. Type of High School: 1= parochial, 2= private, **3= public**

Go to NEXT SLIDE

IV for X1 for Generalized Logit

Let Y=3 be reference for target Y in dataset GL and non-reference outcomes be 1 and 2

The generalized IV will measure how well X1 distinguishes "3" from "1" and "3" from "2"

Here is the Process:

Two usual IV's computed:

- IV is computed only for observations where Y=1 or Y=3 … IV(1,3)
- IV is computed only for observations where Y=2 or Y=3 … IV(2,3)

Next: These two IV's are combined in 3 ways … See next slide.

Go to NEXT SLIDE

Generalized IV for X1

IV Range	Interpretation
IV < 0.02	"Not Predictive"
IV in [0.02 to 0.1)	"Weak"
IV in [0.1 to 0.3)	"Medium"
IV $\geq$ 0.3	"Strong"

Compute IV only for observations where Y=1 or Y=3 … IV(1,3) = 0.1113
Likewise, compute IV for observations where Y=2 or Y=3 … IV(2,3) = 0.2541 (details not shown)

X1	Y = 1	Y = 3	Col % Y=1 "b_k"	Col % Y=3 "g_k"	Log(g_k/b_k) = X_woe	D =($g_k - b_k$)	D * X_woe
01	59	32	29.95%	27.83%	0.074	2.12%	0.0016
01	39	31	19.80%	26.96%	-0.309	-7.16%	0.0221
03	27	21	13.71%	18.26%	-0.287	-4.56%	0.0131
04	38	22	19.29%	19.13%	0.008	0.16%	0.0000
05	34	9	17.26%	7.83%	0.791	9.43%	0.0746
SUM	197	115	100.00%	100.00%		IV =	0.1113

Based on Siddiqi's table, MAX_IV = 0.2541 seems to justify keeping X1

How to use IV … adopt one of these 3 measures:

(1) AVG_IV = (0.1113 + 0.2541)/2 = 0.1827 … average strength across all levels
(2) MIN_IV = MIN(0.1113, 0.2541) = 0.1113 … if large, then all levels strong.
(3) MAX_IV = MAX(0.1113, 0.2541) = 0.2541 … if large, then at least one level is strong.

Let's focus on MAX_IV … if MAX_IV is large, then at least one level v. reference is "strong"

After passing the screening, perform Binning of X1

I have an SAS macro called %MULTI_LOGIT_BIN for binning X1

Binning was set to maximize "MAX_IV" at each Step. (But could use Avg_IV, MIN_IV, or LRCS)

A case can be made for stopping at BIN=3. Then X1 has levels {01, 02}, {03, 04}, {05}

step	collapseX	LRCS	Avg_IV	MIN_IV	MAX_IV	IV_1	IV_2
5		33.53	0.183	0.111	0.254	0.111	0.254
4	01+02	30.95	0.173	0.093	0.254	0.093	0.254
3	03+04	30.36	0.169	0.085	0.252	0.085	0.252
2	01_02+03_04	19.13	0.144	0.085	0.203	0.085	0.203

step	__BIN_1	__BIN_2	__BIN_3	__BIN_4
4	01+02	03	04	05
3	01_02	03+04	05	
2	01_02+03_04	05		

BIN Code
if X1 in ("01","02") then X1_bin = 1;
if X1 in ("03","04") then X1_bin = 2;
if X1 in ("05") then X1_bin = 3;

The choice of MAX_IV AVG_IV MIN_IV LRCS can lead to different binning results

Transforming of X2 for Generalized Logit

Macro %FSP_8LR_Glogit runs FSP on X2

- First translate X2 by +4.50119 … this makes min(X2)=1
- Now FSP picks one of the following four decisions:

1. Screen out (eliminate) X
2. Specifies X, un-transformed
3. Specifies one of the G's (excluding G8=X)
4. Specifies a pair of G's (28 combinations) or a pair: Gi and Gi*Log(X) for i = 1 to 8

In this example, FSP selects decision "**4**"

Translate X2 = X2+4.50119 and use G8=X2 G7=$(X2)^3$ in MODEL

Go to NEXT SLIDE

Final Generalized Logit Model

```
DATA GL_T; SET GL;
if X1 in ( "01","02" ) then X1_bin = 1;
if X1 in ( "03","04" ) then X1_bin = 2;
if X1 in ( "05" ) then X1_bin = 3;
X2 = X2+4.50119;
G8=X2;
G7=X2**3;
run;
PROC LOGISTIC DATA= GL_T;
CLASS X1_bin;
MODEL Y = X1_bin G8 G7
/ LINK=GLOGIT ;
run;
```

Analysis of Maximum Likelihood Estimates					
Parameter		Y	DF	Estimate	Pr > ChiSq
Intercept		1	1	-3.340	0.1626
Intercept		2	1	9.571	<.0001
X1_bin	1	1	1	-0.283	0.1114
X1_bin	1	2	1	-0.689	<.0001
X1_bin	2	1	1	-0.317	0.0942
X1_bin	2	2	1	-0.155	0.3544
G8		1	1	1.465	0.066
G8		2	1	-2.601	<.0001
G7		1	1	-0.026	0.0419
G7		2	1	0.041	<.0001

(added) GL Model is improved with these transformations

Using a Validation Dataset:

Transformed Model gives Better Fit than Un-Transformed Model

- Average Probability of Y = y given Target Level Y = y
- Lower Average Squared Error

	Average probability given target level (higher is better)			
MODEL	Average P(Y=1 | Y=1)	Average P(Y=2 | Y=2)	Average P(Y=3 | Y=3)	AVG SQ ERROR
Un-Transformed (CLASS X1; MODEL Y=X1 X2;)	0.203	0.697	0.129	0.4450
Transformed (CLASS X1_bin; MODEL Y=X1_bin G8 G7;)	0.246	0.718	0.141	0.4226

Added Slides

Final Generalized Logit Model … with Equalslopes option

```
PROC LOGISTIC DATA= GL_T;
CLASS X1_bin;
MODEL Y = X1_bin G8 G7
/ LINK=GLOGIT Equalslopes=(X1_bin G8 G7);
run;
```

Parameter		Y	DF	Estimate	Pr > ChiSq
Intercept		1	1	6.5313	0.0001
Intercept		2	1	7.7819	<.0001
X1_bin	1		1	-0.5817	0.0002
X1_bin	2		1	-0.1993	0.2250
G8			1	-1.9618	0.0004
G7			1	0.0302	0.0005

Still 2 response equations but coefficient of G7 is 0.0302 for BOTH

- Same parameter value for a predictor across all outcomes.
- This is a Strong assumption.
- … A topic for another day.

Dataset BINARY (and validation dataset BINARY_V)

```
Data BINARY BINARY_V;
do i = 1 to 2000;
temp = floor(5*ranuni(3)) - 3;
X1 = PUT(temp+4,Z2.0);
X2 = rannor(4);
xbeta = (1*X1 + 2*(X2**2)) + 5*rannor(1);
P1 = exp(xbeta) / (1 + exp(xbeta));
P2 = 1 - P1;
random = ranuni(6);
if random < P1 then Y = 0;
else Y = 1;
If i <= 1000 then OUTPUT BINARY;
else OUTPUT BINARY_V;
end;
run;
```

```
DATA BINARY2 BINARY2_V; SET BINARY BINARY_V;
if X1 in ( "01","02" ) then X1_B = 01 ;
if X1 in ( "03","04" ) then X1_B = 02 ;
if X1 in ( "05" ) then X1_B = 03 ;
X2_T = X2 +3.70612;
G2=X2_T**(-1); G7=X2_T**3;
if _N_ <= 1000 then OUTPUT BINARY2;
else OUTPUT BINARY2_V;
run;
/* Un-Transformed */
PROC LOGISTIC DATA=BINARY;
CLASS X1;
MODEL Y = X1 X2;
SCORE DATA=BINARY_V FITSTAT;
/* Transformed */
PROC LOGISTIC DATA=BINARY2;
CLASS X1_B;
MODEL Y = X1_B G2 G7;
SCORE DATA=BINARY2_V FITSTAT;
run;
```

Dataset GL

```
Data GL;
do i = 1 to 1000;
temp = floor(5*ranuni(3)) - 3;
X1 = PUT(temp+4,Z2.0);
X2 = rannor(4);
xbeta1 = (0.1*X1 - 2*(X2**2)) + 5*rannor(1); ;
xbeta2 = (1*X1 + 2*(X2**2)) + 5*rannor(1); ;
P1 = exp(xbeta1) / (1 + exp(xbeta1) + exp(xbeta2));
P2 = exp(xbeta2) / (1 + exp(xbeta1) + exp(xbeta2));
P3 = 1 - P1 - P2;
random = ranuni(6);
if random < P1 then Y = 1;
else if P1 <= random < P1 + P2 then Y = 2;
else Y = 3;
output;
end;
run;
```

Create Validate Dataset for GL

```
Data GL_Validate;
do i = 1 to 1000;
temp = floor(5*ranuni(13)) - 3;
X1 = PUT(temp+4,Z2.0);
X2 = rannor(14);
xbeta1 = (0.1*X1 - 2*(X2**2)) + 5*rannor(11); ;
xbeta2 = (1*X1 + 2*(X2**2)) + 5*rannor(11); ;
P1 = exp(xbeta1) / (1 + exp(xbeta1) + exp(xbeta2));
P2 = exp(xbeta2) / (1 + exp(xbeta1) + exp(xbeta2));
P3 = 1 - P1 - P2;
random = ranuni(16);
if random < P1 then Y = 1;
else if P1 <= random < P1 + P2 then Y = 2;
else Y = 3;
output;
end;
run;
DATA GL_T_Validate; SET GL_Validate;
if X1 in ( "01","02" ) then X1_bin = 1;
if X1 in ( "03","04" ) then X1_bin = 2;
if X1 in ( "05" ) then X1_bin = 3;
X2 = X2+4.50119;
G8=X2;
G7=X2**3;
run;
```

Fit untransformed vs. transformed GL … measure on validation

```
DATA GL_T; SET GL;
if X1 in ( "01","02" ) then X1_bin = 1;
if X1 in ( "03","04" ) then X1_bin = 2;
if X1 in ( "05" ) then X1_bin = 3;
X2 = X2+4.50119;
G8=X2;
G7=X2**3;
run;
PROC LOGISTIC DATA= GL;
CLASS X1;
MODEL Y = X1 X2
/ LINK=GLOGIT ;
SCORE DATA=GL_Validate OUT = GL_Scored FITSTAT;
run;
PROC MEANS DATA=GL_Scored mean; CLASS Y;
Var P_1 P_2 P_3;
run;
PROC LOGISTIC DATA= GL_T;
CLASS X1_bin;
MODEL Y = G8 G7 X1_bin
/ LINK=GLOGIT ;
SCORE DATA=GL_T_Validate OUT = GL_T_Scored FITSTAT;
run;
PROC MEANS DATA=GL_T_Scored mean; CLASS Y;
Var P_1 P_2 P_3;
run;
```