Screening, Transforming, and Fitting Predictors for Cumulative Logit Model

Bruce Lund
Independent Consultant, Novi, MI

This is a presentation of a paper from SAS Global Forum 2019

Cum Logit has Target Variable with Ordinal Levels

Examples of Ordinal (Ordered) Target

• Long-term bond credit ratings assigned by S&P: AAA, AA, A, BBB, BB, B, CCC, CC, C, D
• Medical diagnosis stage: stage 1, stage 2, stage 3, stage 4
• Satisfaction survey: Poor, Fair, Good
Screening, Transforming, and Fitting Predictors for Cumulative Logit Model

Topics for Today

A. Describe the Cumulative Logit Model

B. Screening (keep or drop) “NOD” predictors for cum logit model
   • NOD = Nominal, Ordinal, Discrete … few levels but more than 2

C. Binning and transforming of NOD predictors

D. Transforming Continuous predictors (“many” numeric levels)

E. Selecting predictors for fitting cum logit models:
   • PROC LOGISTIC, PROC HPLOGISTIC, PROC HPGENSELECT

Contact me for SAS macros discussed today.
Cumulative Logit Model

In my examples, Target Y has J=3 *ordered* levels, but can be any J ≥ 3

Let Target Y have 3 levels A < B < C and Predictors X1 and X2

Then one form of the Cumulative Logit Model is given by:

- \( \log \left( \frac{p_A}{p_B + p_C} \right) = \alpha_A + \beta X_1 + \lambda X_2 \) ... response eq. for A
- \( \log \left( \frac{p_A + p_B}{p_C} \right) = \alpha_B + \beta X_1 + \lambda X_2 \) ... response eq. for B

Predictors have “Equalslopes” (same coefficient in both equations)

This is *Proportional Odds (PO) cum logit model.*

Coefficients found by maximum likelihood estimation - like binary case

-- No error term assumed nor required in this formulation of Cum Logit

-- The Latent Variable derivation of Cum Logit has an error term (not discussed)
Comments

Proportional Odds (PO) assumption *may be wrong*.

PROC LOGISTIC has Test for “Proportional Odds” (see later slide).

If test fails, then consider *Partial PO (PPO)* Model (next slide).

CUM LOGIT is the usual binary logistic if Target has 2 levels.

Some of Today’s topics can be used for binary logistic models.
Partial PO Cumulative Logit Model (PPO)

Target has 3 levels (A, B, C) and Predictors X1 and X2

Then an example of PPO Cum Logit Model is:

- Log \( \left( \frac{p_A}{p_B + p_C} \right) \) = \( \alpha_A + \beta_A \times X1 + \lambda \times X2 \) ... response equation for A
- Log \( \left( \frac{p_A + p_B}{p_C} \right) \) = \( \alpha_B + \beta_B \times X1 + \lambda \times X2 \) ... response equation for B

Here, \( \beta_A \) and \( \beta_B \) are unequal. Not so for \( \lambda \).

PPO allows designated predictors to have unequal coefficients

PPO is implemented in PROC LOGISTIC by “UNEQUALSLOPES” Statement (see later slide)
PPO has issues

PPO model can produce negative probabilities
– Can occur during model fitting
– When scoring external data (especially if there are extreme values of X’s)

This topic is discussed by Richard Williams (2008) -- see my paper for reference

Williams makes three points (which I will summarize):

• “Probabilities might go negative in unlikely or impossible X ranges, e.g. when years of education is negative” ... Unrealistic Data
• “Seems most problematic with small samples, complicated models, analyses where the data are being spread very thin” ... Ill-Structured Modeling Situations
• “Multiple tests with 10s of thousands of cases typically resulted in only 0 to 3 negative predicted probabilities.” ... Doesn’t happen much in practice
Example: Cumulative Logit PO Model (target has 3 levels)

```sas
DATA Test;
X1=1; X2=3; Y="A"; output;
X1=1; X2=3; Y="B"; output;
X1=1; X2=3; Y="C"; output;
X1=1; X2=3; Y="A"; output;
X1=2; X2=2; Y="A"; output;
X1=2; X2=3; Y="C"; output;
X1=2; X2=3; Y="C"; output;
X1=2; X2=2; Y="A"; output;
X1=2; X2=3; Y="B"; output;
X1=3; X2=3; Y="C"; output;
X1=3; X2=3; Y="A"; output;
X1=3; X2=3; Y="A"; output;
X1=3; X2=4; Y="C"; output;
X1=3; X2=4; Y="B"; output;
run;

PROC LOGISTIC
DATA=Test;
MODEL Y = X1 X2;
run;
```

**Analysis of Maximum Likelihood Estimates**

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Y</th>
<th>DF</th>
<th>Estimate</th>
<th>Standard Error</th>
<th>Chi-Square</th>
<th>Pr &gt; ChiSq</th>
</tr>
</thead>
<tbody>
<tr>
<td>Intercept</td>
<td>A</td>
<td>1</td>
<td>0.9310</td>
<td>2.9117</td>
<td>0.1022</td>
<td>0.7492</td>
</tr>
<tr>
<td>Intercept</td>
<td>B</td>
<td>1</td>
<td>1.8225</td>
<td>2.9422</td>
<td>0.3837</td>
<td>0.5356</td>
</tr>
<tr>
<td>X1</td>
<td></td>
<td>1</td>
<td>-0.1074</td>
<td>0.6618</td>
<td>0.0264</td>
<td>0.8710</td>
</tr>
<tr>
<td>X2</td>
<td></td>
<td>1</td>
<td>-0.4273</td>
<td>1.0043</td>
<td>0.1810</td>
<td>0.6705</td>
</tr>
</tbody>
</table>

**Score Test for the Proportional Odds Assumption**

<table>
<thead>
<tr>
<th>Chi-Square</th>
<th>DF</th>
<th>Pr &gt; ChiSq</th>
</tr>
</thead>
<tbody>
<tr>
<td>4.7855</td>
<td>2</td>
<td>0.0914 (borderline reject)</td>
</tr>
</tbody>
</table>

Intercept A for 1st equation.  
Intercept B for 2nd equation.  
X1 or X2 has unequalslopes?  

8
Example: PPO Cumulative Logit Model

DATA Test;
X1=1; X2=3; Y="A"; output;
X1=1; X2=3; Y="B"; output;
X1=1; X2=3; Y="C"; output;
X1=1; X2=3; Y="A"; output;
X1=2; X2=2; Y="A"; output;
X1=2; X2=3; Y="C"; output;
X1=2; X2=3; Y="C"; output;
X1=2; X2=2; Y="C"; output;
X1=2; X2=3; Y="C"; output;
X1=3; X2=3; Y="B"; output;
X1=3; X2=3; Y="C"; output;
X1=3; X2=3; Y="C"; output;
X1=3; X2=4; Y="A"; output;
X1=3; X2=4; Y="C"; output;
X1=3; X2=4; Y="B"; output;
run;

Y has 3 levels ➔ 2 response equations

PROC LOGISTIC DATA = Test;
MODEL Y = X1 X2 / UNEQUALSLOPES = (X1);
run;

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Y</th>
<th>DF</th>
<th>Estimate</th>
<th>Standard Error</th>
<th>Chi-Square</th>
<th>Pr &gt; ChiSq</th>
</tr>
</thead>
<tbody>
<tr>
<td>Intercept</td>
<td>A</td>
<td>1</td>
<td>0.8248</td>
<td>3.0117</td>
<td>0.0750</td>
<td>0.7842</td>
</tr>
<tr>
<td>Intercept</td>
<td>B</td>
<td>1</td>
<td>1.8812</td>
<td>2.9737</td>
<td>0.4002</td>
<td>0.5270</td>
</tr>
<tr>
<td>X1</td>
<td>A</td>
<td>1</td>
<td>-0.0733</td>
<td>0.7244</td>
<td>0.0102</td>
<td>0.9194</td>
</tr>
<tr>
<td>X1</td>
<td>B</td>
<td>1</td>
<td>-0.1535</td>
<td>0.7601</td>
<td>0.0408</td>
<td>0.8399</td>
</tr>
<tr>
<td>X2</td>
<td></td>
<td>1</td>
<td>-0.4145</td>
<td>1.0251</td>
<td>0.1635</td>
<td>0.6860</td>
</tr>
</tbody>
</table>
Brief discussion of NOD predictors

If X is numeric with many levels or is binary (2 levels), then X can go directly into a cum logit model: PROC LOGISTIC; MODEL Y = X;

Now consider X with “few”, but more than 2 levels ...

This is what I call a “\textbf{NOD}” predictor

Nominal \{blue, green, brown\}, Ordinal: \{poor, fair, good\}, Discrete \{0, 1, 2, 3\}

Why not put a discrete X in MODEL Y = X? ... Can do.

But often “0” means something special. E.g. X= number of arrests. X=0 is much different then X=1, 2, 3. Maybe X=0 should have a “Dummy”. Maybe also “1” ... soon, may as well consider CLASS X.
Topic: Screening NOD

- Efficient screening of NOD predictors for cum logit model
  - Maybe dozens, hundreds of NOD predictors

- In Paper two macros are discussed:
  - `%CUM_LOGIT SCREEN_1` (fully discussed)
  - `%CUM_LOGIT SCREEN_2` (briefly discussed)
Model “c” for Cum Logit in PROC LOGISTIC

Measures model performance ... 0.5 to 1.0. Higher is better.

Let target have levels \( k = 1, 2, 3 \)

For each observation:

• Let Probabilities be \( p_k \) for \( k = 1, 2, 3 \)
• Compute “mean score” as \( \text{Mscore} = \sum_{k=1}^{3} p_k \times (k - 1) \)
  
e.g. If \( p_2 = 0.4 \) and \( p_3 = 0.1 \), then \( \text{Mscore} = 0.4 + 2 \times 0.1 = 0.6 \)

• NOW: Same Idea as Binary Case
  
  ❖ IP = “Informative Pairs” of obs \( (r, s) \) where Targets \( Y_r \neq Y_s \)
  ❖ If \( Y_r > Y_s \) and \( \text{Mscore}_r > \text{Mscore}_s \), then CONCORDANT
  ❖ If \( Y_r > Y_s \) and \( \text{Mscore}_r < \text{Mscore}_s \), then DISCORDANT
  ❖ Else TIE .... And **Model c** = \( \{\text{CONCORDANT} + 0.5 \times \text{TIE}\} / \text{IP} \)

\[ \text{Model}_c = \max(1 - \text{Model}_c, \text{Model}_c) \]
I cannot find a paper on Model c for Cum Logit

Is “model c” useful as a measure of predictive accuracy?

**PROC LOGISTIC;  MODEL Y = X;**

<table>
<thead>
<tr>
<th>X\Y</th>
<th>A</th>
<th>B</th>
<th>C</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>90</td>
<td>7</td>
<td>3</td>
</tr>
<tr>
<td>2</td>
<td>5</td>
<td>90</td>
<td>5</td>
</tr>
<tr>
<td>3</td>
<td>3</td>
<td>7</td>
<td>90</td>
</tr>
</tbody>
</table>

LRCS = 392.4  
Model c = 0.931

<table>
<thead>
<tr>
<th>X\Y</th>
<th>A</th>
<th>B</th>
<th>C</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>60</td>
<td>30</td>
<td>10</td>
</tr>
<tr>
<td>2</td>
<td>20</td>
<td>60</td>
<td>20</td>
</tr>
<tr>
<td>3</td>
<td>20</td>
<td>20</td>
<td>60</td>
</tr>
</tbody>
</table>

LRCS = 97.9  
Model c = 0.742

<table>
<thead>
<tr>
<th>X\Y</th>
<th>A</th>
<th>B</th>
<th>C</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>40</td>
<td>30</td>
<td>30</td>
</tr>
<tr>
<td>2</td>
<td>30</td>
<td>40</td>
<td>30</td>
</tr>
<tr>
<td>3</td>
<td>30</td>
<td>30</td>
<td>40</td>
</tr>
</tbody>
</table>

LRCS = 3.08  
Model c = 0.545

<table>
<thead>
<tr>
<th>X\Y</th>
<th>A</th>
<th>B</th>
<th>C</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>34</td>
<td>33</td>
<td>33</td>
</tr>
<tr>
<td>2</td>
<td>33</td>
<td>34</td>
<td>33</td>
</tr>
<tr>
<td>3</td>
<td>33</td>
<td>33</td>
<td>34</td>
</tr>
</tbody>
</table>

LRCS = 0.03  
Model c = 0.504

From various simulations: Model c measures predictive accuracy
**Saturated PPO Cum Logit Model with 1 NOD Predictor**

```plaintext
PROC LOGISTIC DATA=Test;
CLASS X; ★
MODEL Y = X / ★
UNEQUALSLOPES = (X); ★
```

Why Look at Saturated Model with X?

- If X and Y independent, then X adds no information about Y versus Model with an intercept-only

- Saturated model gives ways to measure deviation from independence
  - Likelihood ratio chi-square (LRCS) ... the right tail probability
  - Model c

- These measures allow predictors to be **screened**.
  - If Saturated X is weak on both LRCS / Model c, then eliminate X

- If X passes screening, use of X in Model may not be “saturated”:
  - Unequalslopes may not be needed, X might transformed or binned
Saturated PPO Cum Logit Model with 1 NOD Predictor

PROC LOGISTIC DATA=Test;
CLASS X;
MODEL Y = X /
UNEQUALSLOPES = (X);

But do not need PROC LOGISTIC to compute output from Logistic

Compute in Data Step!

Then, with “some” programming, the same Data Step can be used to efficiently compute Model c and LRCS for many X’s ... leads to my SAS macro

Model c = 0.635

<table>
<thead>
<tr>
<th></th>
<th>Chi-Square</th>
<th>DF</th>
<th>Pr &gt; ChiSq</th>
</tr>
</thead>
<tbody>
<tr>
<td>Like. Ratio</td>
<td>1.3368</td>
<td>4</td>
<td>0.8551</td>
</tr>
</tbody>
</table>

MSUG 2019
%CUM_LOGIT_SCREEN_1 (Dataset, Target, Input, Sort);

For Target with \( \geq 2 \) levels this macro computes:

- Likelihood Ratio Chi-Sq for *saturated* model
- Model “c” for the *saturated* model

All this is done in a *DATA Step* for many X’s

Useful for Binary Logistic

%CUM_LOGIT_SCREEN_1 (Test2, Y, X1 X2, Model_c);
Y target (3 levels)
X1 numeric (3 levels), X2 character (3 levels)
Sort results by descending Model c

DATA Test2;
X1=1; X2='3'; Y="A"; output;
X1=1; X2='3'; Y="B"; output;
X1=1; X2='3'; Y="C"; output;
X1=1; X2='3'; Y="A"; output;
X1=2; X2='2'; Y="A"; output;
X1=2; X2='2'; Y="B"; output;
X1=2; X2='2'; Y="C"; output;
X1=2; X2='3'; Y="A"; output;
X1=2; X2='3'; Y="B"; output;
X1=2; X2='3'; Y="C"; output;
X1=2; X2='3'; Y="C"; output;
X1=2; X2='3'; Y="C"; output;
X1=2; X2='3'; Y="C"; output;
X1=3; X2='3'; Y="A"; output;
X1=3; X2='3'; Y="A"; output;
X1=3; X2='3'; Y="A"; output;
X1=3; X2='3'; Y="B"; output;
X1=3; X2='4'; Y="C"; output;
X1=3; X2='4'; Y="B"; output;
run;
%CUM_LOGIT_SCREEN_1 (Test2, Y, X1 X2, Model_c);

<table>
<thead>
<tr>
<th>Var_name</th>
<th>Levels</th>
<th>LRCS</th>
<th>Pr &gt; ChiSq</th>
<th>Model c</th>
</tr>
</thead>
<tbody>
<tr>
<td>X1</td>
<td>3</td>
<td>1.337</td>
<td>0.855</td>
<td>0.6349</td>
</tr>
<tr>
<td>X2</td>
<td>3</td>
<td>3.063</td>
<td><strong>0.547</strong></td>
<td>0.5556</td>
</tr>
</tbody>
</table>

- **Pr>ChiSq of LRCS**: Lower is better
- **Model c**: Higher is better ... what is “poor” Model c?
- **RANK** the X’s by **Pr>ChiSq and Model c**
  - Similar not identical RANKINGS
  - Look for X with high rank on BOTH
- **Should use with %CUM_LOGIT_SCREEN_2** (mentioned later)

See PAPER for extensive example of SCREEN 1 & 2
Topic: Binning and Transforming of NOD Predictors

- Binning and Transforming of NOD predictors

But First, review for Binary Target:
- Weight-of-evidence (WOE)
- Information Value (IV)
### Information Value and WOE for BINARY

<table>
<thead>
<tr>
<th>X</th>
<th>Y = 0 “B_k”</th>
<th>Y = 1 “G_k”</th>
<th>Col % Y=0 “b_k”</th>
<th>Col % Y=1 “g_k”</th>
<th>Log(g_k/b_k) = X_woe</th>
<th>g_k - b_k</th>
<th>IV Terms (g_k - b_k) * Log(g_k/b_k)</th>
</tr>
</thead>
<tbody>
<tr>
<td>X1</td>
<td>2</td>
<td>1</td>
<td>25.0%</td>
<td>12.5%</td>
<td>-0.69315</td>
<td>-0.125</td>
<td>0.08664</td>
</tr>
<tr>
<td>X2</td>
<td>1</td>
<td>1</td>
<td>12.5%</td>
<td>12.5%</td>
<td>0.00000</td>
<td>0</td>
<td>0.00000</td>
</tr>
<tr>
<td>X3</td>
<td>5</td>
<td>6</td>
<td>62.5%</td>
<td>75.0%</td>
<td>0.18232</td>
<td>0.125</td>
<td>0.02279</td>
</tr>
<tr>
<td>SUM</td>
<td>8</td>
<td>8</td>
<td>100%</td>
<td>100%</td>
<td></td>
<td></td>
<td>IV = 0.10943</td>
</tr>
</tbody>
</table>

- Weight of Evidence of X is \( X_{woe} \)

### IV Range and Interpretation

<table>
<thead>
<tr>
<th>IV Range</th>
<th>Interpretation</th>
</tr>
</thead>
<tbody>
<tr>
<td>IV &lt; 0.02</td>
<td>“Not Predictive”</td>
</tr>
<tr>
<td>IV in [0.02 to 0.1)</td>
<td>“Weak”</td>
</tr>
<tr>
<td>IV in [0.1 to 0.3)</td>
<td>“Medium”</td>
</tr>
<tr>
<td>IV ≥ 0.3</td>
<td>“Strong”</td>
</tr>
</tbody>
</table>

Siddiqi, N. (2017). *Intelligent Credit Scoring*
WOE’s, IV’s for Cum Logit

Two Binary “splits” of Target (levels A, B, C) ... (A vs. BC, AB vs. C)

WOE’s, IV’s for each “split”

<table>
<thead>
<tr>
<th></th>
<th>A</th>
<th>B</th>
<th>C</th>
</tr>
</thead>
<tbody>
<tr>
<td>X</td>
<td>1</td>
<td>4</td>
<td>1</td>
</tr>
<tr>
<td>2</td>
<td>3</td>
<td>1</td>
<td>3</td>
</tr>
<tr>
<td>3</td>
<td>1</td>
<td>2</td>
<td>1</td>
</tr>
</tbody>
</table>

Binary: A vs. BC
WOE1_X = \(0.811\)

Binary: AB vs. C
WOE2_X = \(-0.170\)

IV_1 = \(0.441\)
IV_2 = \(0.327\)

CUM_LOGIT_SCREEN_2:
Reports IV of X for each SPLIT
Binning

Binning of NOD predictors
  Reduces number of levels of X

Reasons:
  ➢ Achieve Parsimony (fewer degrees of freedom)
  ➢ Find Logical relationships (e.g. monotonicity)
  ➢ Combine a low frequency level with another level

But BIN without giving up (much) predictive power
### The DATA: BACKACHE (*)

- Gives age of pregnant women and **Severity** of backache experienced
- Severity has three levels: A, B, and C with “A” being least severe.
- 9 Levels for Age_group

<table>
<thead>
<tr>
<th>Age_Group</th>
<th>A</th>
<th>B</th>
<th>C</th>
</tr>
</thead>
<tbody>
<tr>
<td>15to19</td>
<td>10</td>
<td>5</td>
<td>2</td>
</tr>
<tr>
<td>20to22</td>
<td>20</td>
<td>12</td>
<td>2</td>
</tr>
<tr>
<td>23to24</td>
<td>19</td>
<td>8</td>
<td>3</td>
</tr>
<tr>
<td>25to26</td>
<td>15</td>
<td>13</td>
<td>4</td>
</tr>
<tr>
<td>27to28</td>
<td>8</td>
<td>7</td>
<td>2</td>
</tr>
<tr>
<td>29to30</td>
<td>6</td>
<td>7</td>
<td>3</td>
</tr>
<tr>
<td>31to32</td>
<td>4</td>
<td>5</td>
<td>3</td>
</tr>
<tr>
<td>33to35</td>
<td>5</td>
<td>1</td>
<td>4</td>
</tr>
<tr>
<td>36andUP</td>
<td>6</td>
<td>2</td>
<td>4</td>
</tr>
<tr>
<td><strong>Total</strong></td>
<td>93</td>
<td>60</td>
<td>27</td>
</tr>
</tbody>
</table>

(*) “BACKACHE IN PREGNANCY” data set in Chatfield (1995, Exercise D.2)
%CUMLOGIT_BIN

**DATASET**: Data set to be processed

**TARGET**: Target with numeric or character levels

**X**: Predictor (numeric or character)

**W**: A frequency variable

**MODE**: A or J:
Defines the eligible pairs of levels of predictor X for collapsing. **A** = any pair; **J** = pairs with adjacent levels

**METHOD**: IV or LL:
Defines rules for selecting an eligible pair to collapse.

**IV** = Sum of binary split IV’s. = IV1 + IV2

**LL** = -2*log(likelihood)

... LL is computed for **saturated** model

---

<table>
<thead>
<tr>
<th>A</th>
<th>J</th>
<th>Age_Group</th>
</tr>
</thead>
<tbody>
<tr>
<td>OK</td>
<td>NO</td>
<td>15to19</td>
</tr>
<tr>
<td></td>
<td></td>
<td>20to22</td>
</tr>
<tr>
<td>NO</td>
<td></td>
<td>23to24</td>
</tr>
<tr>
<td></td>
<td></td>
<td>25to26</td>
</tr>
<tr>
<td>OK</td>
<td></td>
<td>27to28</td>
</tr>
<tr>
<td>OK</td>
<td></td>
<td>29to30</td>
</tr>
<tr>
<td></td>
<td></td>
<td>31to32</td>
</tr>
<tr>
<td></td>
<td></td>
<td>33to35</td>
</tr>
<tr>
<td>OK</td>
<td></td>
<td>36andUP</td>
</tr>
</tbody>
</table>

“A” & “IV” AND STEP 1
THEY ARE 36 PAIRS = \( \binom{9}{2} \)
FIND THE 1 WITH MAX “IV”
### %CUMLOGIT_BIN Results

<table>
<thead>
<tr>
<th>Bins</th>
<th>-2*LL</th>
<th>(Total) IV</th>
<th>IV_1</th>
<th>IV_2</th>
</tr>
</thead>
<tbody>
<tr>
<td>9</td>
<td>339.5</td>
<td>0.614</td>
<td>0.138</td>
<td>0.476</td>
</tr>
<tr>
<td>8</td>
<td>339.5</td>
<td>0.613</td>
<td>0.138</td>
<td>0.476</td>
</tr>
<tr>
<td>7</td>
<td>339.6</td>
<td>0.609</td>
<td>0.135</td>
<td>0.474</td>
</tr>
<tr>
<td>6</td>
<td>339.9</td>
<td>0.605</td>
<td>0.135</td>
<td>0.470</td>
</tr>
<tr>
<td>5</td>
<td>340.0</td>
<td><strong>0.598</strong></td>
<td>0.134</td>
<td>0.464</td>
</tr>
<tr>
<td>4</td>
<td>341.0</td>
<td>0.561</td>
<td>0.133</td>
<td>0.429</td>
</tr>
<tr>
<td>3</td>
<td>342.4</td>
<td>0.493</td>
<td>0.111</td>
<td>0.381</td>
</tr>
<tr>
<td>2</td>
<td>350.4</td>
<td>0.324</td>
<td>0.103</td>
<td>0.221</td>
</tr>
</tbody>
</table>

**MODE = A**

**METHOD = IV [ i.e. (TOTAL) IV ]**

Start with all 9 levels, then ...

Collapse to 8 (details not shown), Etc.,

... when to STOP?

Stopping at BIN=5

IV = **0.598**

Avoids large drop at BIN=4

---

**BIN1**

15to19_23to24

**BIN2**

20to22

**BIN3**

25to26_27to28

**BIN4**

29to30_31to32

**BIN5**

33to35_36andUP
After Binning ... Transforming

Two Decisions for a (possibly binned) NOD predictor

1. WOE’s or DUMMIES (creates different models)
   If target has J levels, then J-1 WOE’s
   If X has L levels, then L-1 dummies

2. EQUAL or UNEQUALSLOPES or In-Between

UNEQUALSLOPES comes at a “price”:

Consider dummies:

If X has L levels, Target has J levels, then unequalslopes gives:

Number of Parameters = (J-1) * (L-1)

See Lund (2017) SESUG for more discussion
Topic: Transforming Continuous Predictors

Suppose there is predictor X for cum logit model

- How should X be used in Model?
  - $X^2$, $\log(X)$, $1/X$, or something more complicated?

This discussion is also useful for binary logistic!
Function Selection Procedure and %FSP_8LR

FSP was developed in the 1990’s. Bio-medical applications. *Multivariate Model-building* (2008) by Royston and Sauerbrei

FSP: For continuous predictor $X$ for *Cumulative Logit* model:

- Selects a final transform of $X$ ... (44 transforms are checked)
  Or
- Eliminates $X$ from further consideration as a predictor
- My macro %FSP_8LR implements FSP for PO
  – *Multiple* $X$’s can be processed efficiently: %FSP_8LR(DATA, Y, X1 - X50)
- %FSP_8LR_PPO implements FSP for PPO (not discussed today)
FSP: Looks for the best transformation of $X$

- First, translate $X$ to make $\min(X)=1$. If $\min(X) \geq 1$, then leave as is.
- Form the *Fractional Polynomials* (FP):
  
  $X^p$ for $p$ in $S = \{-2, -1, -0.5, 0, 0.5, 1, 2, 3\}$ where “0” = $\log(x)$

  There are 8 $p$’s.

- Create two groups of transforms: FP1 and FP2

  **8 FP1:** $g(X,p) = \beta_0 + \beta_1 X^p$

  **36 FP2:**

  $g(X,p_1,p_2) = \beta_0 + \beta_1 X^{p_1} + \beta_2 X^{p_2}$  \hspace{1cm} p_1 \neq p_2 \ldots 28$

  $g(X,p_1,p_1) = \beta_0 + \beta_1 X^{p_1} + \beta_2 X^{p_1} \log(X)$  \hspace{1cm} p_1 = p_2 \ldots 8$
FSP: Looks for the best transformation of X

Selection: Fit each of the 8 FP1 and 36 FP2 models by logistic regression ... 44 models in total!
(But my macro \%FSP_8LR runs only 8 times)

- *FP1 Solution* is one highest log likelihood among the 8 models
- *FP2 Solution* is one highest log likelihood among the 36 models
Code Generates an Cum Logit PO Example
This code is in paper - DON’T LOOK AT CODE

- Predictor X (levels: 1 to 16)
- Use X to create “FP2” transform
  - 0.2*LOG(X) - 0.5*X^-1
  - With PO (equalslopes)
  - Also random error
- Then more code and finally, ...
- … produces Y with 3 levels

Now, apply %FSP_8LR to this data ... next slide

```sas
%LET ERROR = 0.01;
%LET SLOPE1 = 0.2;
%LET SLOPE2 = -0.5;
%LET P_Seed = 5;
%MACRO SIM(NUM);
%DO Seed = 1 %TO &NUM;
  DATA Work_&Seed;
  do i = 1 to 8000;
    X = mod(i, 16) + 1;
    rannorx = rannor(&Seed);
    T = exp(0 + &SLOPE1*LOG(X) + &SLOPE2*(1/X) + &ERROR*rannorx);
    U = exp(1 + &SLOPE1*LOG(X) + &SLOPE2*(1/X) + &ERROR*rannorx);
    PA = 1 - 1/(1 + T);
    PB = 1/(1 + T) - 1/(1 + U);
    PC = 1 - (PA + PB);
    /* Assign Target Values to match model probabilities */
    R = ranuni(&P_Seed);
    if R < PA then Y = "A";
    else if R < (PA + PB) then Y = "B";
    else Y = "C";
    output;
  end;
run;
%MEND;
%SIM(1);
```
### 3 Steps for Selection of Transform

**Summary Report from %FSP_8LR(WORK, Y, X)**

<table>
<thead>
<tr>
<th>TEST</th>
<th>-2*Log(L)</th>
<th>TEST -STAT</th>
<th>df</th>
<th>P-VALUE</th>
<th>Trans 1</th>
<th>Trans2</th>
</tr>
</thead>
<tbody>
<tr>
<td>Eliminate X (intercept only)</td>
<td>15824.5</td>
<td>172.0</td>
<td>4</td>
<td>0.000</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Use Linear X</td>
<td>15709.3</td>
<td>56.9</td>
<td>3</td>
<td>0.000</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Use FP1 ... or ...</td>
<td>15654.3</td>
<td>1.9</td>
<td>2</td>
<td>0.387</td>
<td></td>
<td>p=-0.5</td>
</tr>
<tr>
<td>Use FP2</td>
<td>15652.4</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>p=-2</td>
</tr>
</tbody>
</table>

**STEP 1:** Test for $H_0$: “Eliminate X” vs. $H_A$: Go to STEP 2

15824.5 - 15652.4 = 172.0 ... Chi-Square with 4 d.f.

Why 4 d.f.? ... 2 for exponent and 2 for coefficient

... Rejects “Eliminate X”
Three Steps for Selection of Transform

<table>
<thead>
<tr>
<th>TEST</th>
<th>-2*Log(L)</th>
<th>TEST STAT</th>
<th>d f</th>
<th>P-VALUE</th>
<th>Trans 1</th>
<th>Trans 2</th>
</tr>
</thead>
<tbody>
<tr>
<td>Eliminate X</td>
<td>15824.5</td>
<td>172.0</td>
<td>4</td>
<td>0.000</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Use Linear X</td>
<td>15709.3</td>
<td>56.9</td>
<td>3</td>
<td>0.000</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Use FP1 ... or ...</td>
<td>15654.3</td>
<td>1.9</td>
<td>2</td>
<td>0.387</td>
<td>p=-0.5</td>
<td></td>
</tr>
<tr>
<td>Use FP2</td>
<td>15652.4</td>
<td></td>
<td></td>
<td></td>
<td>p=-2</td>
<td>log</td>
</tr>
</tbody>
</table>

STEP 2: $H_0$: Use X (linear) vs. $H_A$: Go to STEP 3

... $15709.3 - 15654.4 = 56.9$ ... P-Value = 0 ... Reject $H_0$: Use X (linear)

Go to Step 3
### Three Steps for Selection of Transform

<table>
<thead>
<tr>
<th>TEST</th>
<th>-2*Log(L)</th>
<th>TEST_STAT</th>
<th>d</th>
<th>P-VALUE</th>
<th>Trans 1</th>
<th>Trans 2</th>
</tr>
</thead>
<tbody>
<tr>
<td>Eliminate X</td>
<td>15824.5</td>
<td>172.0</td>
<td>4</td>
<td>0.000</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Use Linear X</td>
<td>15709.3</td>
<td>56.9</td>
<td>3</td>
<td>0.000</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Use FP1 ... or ...</td>
<td>15654.3</td>
<td>1.9</td>
<td>2</td>
<td>0.387</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Use FP2</td>
<td>15652.4</td>
<td></td>
<td></td>
<td>p=-0.5</td>
<td></td>
<td>log</td>
</tr>
</tbody>
</table>

**STEP 3:** $H_0$: Use FP1 Solution vs. $H_A$: Use FP2 Solution

**P-Value = 0.387** ... Accept $H_0$

... Final solution is $X^{-0.5}$
Testing the Proportional Odds (PO) Assumption
Must a PPO Model be Considered ... Answer: “Borderline”

<table>
<thead>
<tr>
<th>TEST</th>
<th>-2*Log(L)</th>
<th>TEST_STAT</th>
<th>df</th>
<th>P-VALUE</th>
<th>Trans 1</th>
<th>Trans 2</th>
<th>ChiSq_PO</th>
<th>df_PO</th>
<th>Prob ChiSq_PO</th>
</tr>
</thead>
<tbody>
<tr>
<td>Eliminate X</td>
<td>15824.5</td>
<td>172.05</td>
<td>4</td>
<td>0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Use Linear</td>
<td>15709.3</td>
<td>56.86</td>
<td>3</td>
<td>0</td>
<td></td>
<td></td>
<td>1.934</td>
<td>1</td>
<td>0.164</td>
</tr>
<tr>
<td><strong>Use FP1</strong></td>
<td>15654.3</td>
<td>1.90</td>
<td>2</td>
<td>0.387</td>
<td>p=-0.5</td>
<td></td>
<td>3.112</td>
<td>1</td>
<td><strong>0.078</strong></td>
</tr>
<tr>
<td>Use FP2</td>
<td>15652.4</td>
<td></td>
<td></td>
<td>p=-2</td>
<td>log</td>
<td></td>
<td>2.479</td>
<td>2</td>
<td>0.290</td>
</tr>
</tbody>
</table>

Borderline test result
Perhaps “unequalslopes” for X^{-0.5} ?
Topic: Selecting Predictors for Model

- Selecting predictors for fitting cum logit models
  - PROC LOGISTIC
  - PROC HPLOGISTIC
  - PROC HPGENSELECT
Fitting the Cumulative Logit Models (PO and PPO)

After screening, binning, transforming:
• There may be many candidate predictors
• A predictor SELECTION method is needed for model fitting

PROC LOGISTIC:
Only procedure with UNEQUALSLOPES (added in version 12.1)
• If no UNEQUALSLOPES statement, then:
  ▪ All SELECTION options apply to PO Cum Logit (stepwise, forward, etc.)
• If UNEQUALSLOPES is used, then:
  ▪ All SELECTION, except SELECTION = SCORE (i.e. Best Subsets)
DATA WORK1;
  Do ID = 1 to 5000;
    random = ranuni(1);
    If random < 0.5 then Y = "A";
    else if random < 0.8 then Y = "B";
    else Y = "C";
    X1 = floor(ranuni(1)*5) * random;
    X2 = rannor(1) * random;
    X3 = ranuni(10) * random;
    X4 = X3*ranuni(10);
    output;
  end;
run;
PROC LOGISTIC DATA= WORK1;
  MODEL Y= X1  X2  X3  X4 / SELECTION= FORWARD SLE= .05 UNEQUALSLOPES EQUALSLOPES;
run;

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Y</th>
<th>DF</th>
<th>Estimate</th>
<th>Pr &gt; ChiSq</th>
</tr>
</thead>
<tbody>
<tr>
<td>Intercept</td>
<td>A</td>
<td>1</td>
<td>2.5651</td>
<td>&lt;.0001</td>
</tr>
<tr>
<td>Intercept</td>
<td>B</td>
<td>1</td>
<td>3.4682</td>
<td>&lt;.0001</td>
</tr>
<tr>
<td>U_X1</td>
<td>A</td>
<td>1</td>
<td>-1.1298</td>
<td>&lt;.0001</td>
</tr>
<tr>
<td>U_X1</td>
<td>B</td>
<td>1</td>
<td>-0.6621</td>
<td>&lt;.0001</td>
</tr>
<tr>
<td>U_X3</td>
<td>A</td>
<td>1</td>
<td>-7.0304</td>
<td>&lt;.0001</td>
</tr>
<tr>
<td>U_X3</td>
<td>B</td>
<td>1</td>
<td>-3.9414</td>
<td>&lt;.0001</td>
</tr>
</tbody>
</table>

See P. Hilliard (2017) SGF paper

X2 and X4 are not selected. X1 and X3 are selected with unequalslopes
HPLOGISTIC/HPGENSELECT fitting Cum Logit PO
Remember “Evils” of Forward/Stepwise/Backward based on SIGNIFICANCE !!!

HPLOGISTIC & HPGENSELECT support only Cumulative Logit PO

ALL predictor SELECT= options apply to Cumulative Logit PO
e.g. HPLOGISTIC (SBC, AIC), HPGENSELECT (LASSO)

**PROC HPLOGISTIC** DATA = WORK1;
MODEL Y = X1 X2 X3 X4;
SELECTION METHOD = FORWARD (SELECT=AIC CHOOSE=AIC STOP=NONE);
run;

**PROC HPGENSELECT** DATA = WORK1 LASSOSTEPS= 40 LASSORHO= 0.8 ;
MODEL Y = X1 X2 X3 X4 / DISTRIBUTION= BINARY; /* BINARY for CUM LOGIT! */
SELECTION METHOD = LASSO (CHOOSE=AIC STOP=NONE);
run;
Fitting the Cumulative Logit Models for PPO

Can HPLOGISTIC / HPGENSELECT be tricked in running PPO?

This would allow advanced SELECTION methods (AIC, SBC, LASSO) to be used for PPO models.

Yes, a data coding “trick” can make this work !!

But there are some issues (see Paper for discussion)

A robust testing plan is needed to determine limitations.

This DATA CODING TRICK is given in Stokes, Davis, Koch (2000) Categorical Data Analysis, 2nd ed. P. 533
A Description of the Trick

Start with Data Set WORK1.

With $Y$ with 3 Levels = 0, 1, 2

Create a new Data Set WORK2

For each obs. from WORK1:

Output 2 obs. to WORK2

Count them: $\text{SPLIT}$ values 0 to 1

Same X’s on each output obs.

IF $\text{SPLIT} \geq Y$ THEN $Y_2 = 0$ /* BINARY */
ELSE IF $\text{SPLIT} < Y$ THEN $Y_2 = 1$
CUM LOGIT MODEL (PPO for $X_1$)

/* #1 */
PROC LOGISTIC
DATA = WORK1;
MODEL Y = $X_1\ X_2\ X_3\ X_4$ /
UNEQUALSLOPES = $X_1$;

/* #2 */
PROC HPLOGISTIC
DATA = WORK2; /*TRICKED*/
CLASS Split;
MODEL Y2 =
Split $X_1\ X_2\ X_3\ X_4\ X_1\^\text{Split}$;

X1 and $X_1\^\text{Split}$ ➔ “unequalslopes” for $X_1$

#1 & #2 have similar coefficients
But not the Same!

<table>
<thead>
<tr>
<th>MODEL #2</th>
<th>Estimate</th>
</tr>
</thead>
<tbody>
<tr>
<td>Intercept</td>
<td>3.933</td>
</tr>
<tr>
<td>SPLIT 0</td>
<td>-1.728</td>
</tr>
<tr>
<td>SPLIT 1</td>
<td>0</td>
</tr>
<tr>
<td>$X_1$</td>
<td>-0.642</td>
</tr>
<tr>
<td>$X_2$</td>
<td>0.031</td>
</tr>
<tr>
<td>$X_3$</td>
<td>-5.286</td>
</tr>
<tr>
<td>$X_4$</td>
<td>-0.065</td>
</tr>
<tr>
<td>$X_1^\text{Split 0}$</td>
<td>-0.483</td>
</tr>
<tr>
<td>$X_1^\text{Split 1}$</td>
<td>0</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th>MODEL #1</th>
<th>MODEL #2</th>
</tr>
</thead>
<tbody>
<tr>
<td>Intercept A</td>
<td>2.2504</td>
<td>2.2051</td>
</tr>
<tr>
<td>Intercept B</td>
<td>3.9821</td>
<td>3.9326</td>
</tr>
<tr>
<td>$X_1\ A$</td>
<td>-1.1611</td>
<td>-1.1240</td>
</tr>
<tr>
<td>$X_1\ B$</td>
<td>-0.6728</td>
<td>-0.6415</td>
</tr>
<tr>
<td>$X_2$</td>
<td>0.0332</td>
<td>0.0308</td>
</tr>
<tr>
<td>$X_3$</td>
<td>-5.1511</td>
<td>-5.2855</td>
</tr>
<tr>
<td>$X_4$</td>
<td>-0.0427</td>
<td>-0.0646</td>
</tr>
</tbody>
</table>

Convert to Model #1
Best of “Both Worlds” - using HPLOGISTIC & LOGISTIC for PPO

★★★ • Recode DATA using TRICK. Choose predictors (suppose X1) for unequalslopes
★★★ • Run HPLOGISTIC (here ... FORWARD and SELECT=AIC).
★★★ • Re-fit **Selected** Predictors by PROC LOGISTIC with UNEQUALSLOPES (if needed)

<table>
<thead>
<tr>
<th>Selection Summary</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step</strong></td>
</tr>
<tr>
<td>0</td>
</tr>
<tr>
<td>1</td>
</tr>
<tr>
<td>2</td>
</tr>
<tr>
<td>3</td>
</tr>
<tr>
<td>4</td>
</tr>
<tr>
<td>5</td>
</tr>
</tbody>
</table>

PROC HPLOGISTIC DATA= WORK2;
CLASS SPLIT /PARAM = ref;
MODEL Y2= SPLIT X1 X2 X3 X4 X1*SPLIT / INCLUDE=1;
SELECTION METHOD= FORWARD SELECT=AIC
CHOOSE=AIC STOP=NONE);

PROC LOGISTIC DATA= WORK1;
MODEL Y= X1 X3 /*X2 X4*/
/ UNEQUALSLOPES= (X1);
Bruce Lund
Contact: blund_data@mi.rr.com