#SASGF

VIRTUAL

SAS® GLOBAL FORUM 2021

AMERICAS | MAY 18 – 20   ASIA PACIFIC | MAY 19 – 20   EMEA | MAY 25 – 26

Weight of Evidence, Dummy Variables, and Degrees of Freedom

Bruce Lund, Statistical Trainer and Consultant, Novi Michigan

---

Independent Statistical Consulting, primarily for OneMagnify of Detroit
Established Analytics Practice for OneMagnify … and first Manager of Analytics
Customer Database Manager at Ford Motor Company
Mathematics Professor, University of New Brunswick, Canada
Mathematics PhD, Stanford University

Weight of Evidence, Dummy Variables, and Degrees of Freedom

Hello! I am Bruce Lund

This Presentation Applies to Binary Logistic Models:

It answers the question …..

- How to assign degrees of freedom to weight of evidence predictors …
- … when running Forward Selection to Minimize AIC

Goals of the Presentation

Here are the GOALS for the presentation:

Let D be Discrete (… nominal, ordinal, numeric with only a few levels)

- Explain Weight of Evidence (=WOE) coding for predictor D
- … WOE is alternative to dummy variables for D
- Explain "degrees of freedom" problem for WOE
- Propose a process to *adjust* the d.f. for WOE
- Use *adjusted* d.f. in Forward Selection of predictors (WOE and other predictors) to fit a logistic model
- Give Example. The Example also demonstrates my SAS® macro

Introduction:

Properties of Weight of Evidence (WOE)

Weight of Evidence Coding of Predictor D

D: Predictor
Y: Target (binary response)

D	Y = 0	Y = 1	% Y = 0 = b_j	% Y = 1 = g_j	WOE= $Log(g_j/b_j)$
D=D1	2	3	40.0%	42.9%	0.0690
D=D2	1	3	20.0%	42.9%	0.7621
D=D3	2	1	40.0%	14.3%	-1.0296
SUM	5	7	100%	100%	

If D = D3 then D_woe(D3) = -1.0296

WOE is widely used in Credit Risk Modeling as alternative to Dummy Variables. See Book: Siddiqi, N. 2017. *Intelligent Credit Scoring*

Dummy Variable Coding for D vs. D_woe

D: Predictor
Y: Target (binary response)

TABLE 1		
D	Y = 0	Y = 1
D=D1	2	3
D=D2	1	3
D=D3	2	1

Creating Dummies for D

```
DATA WORK; SET TABLE1;
D1 = (D = "D1"); /* 0 and 1 */
D2 = (D = "D2"); /* 0 and 1 */
RUN;
```

CLASS Statement creates the same dummies

```
PROC LOGISTIC DATA = WORK DESC;
CLASS D (param = ref);
MODEL Y = D;
RUN;
```

Alternative approach using WOE:

```
PROC LOGISTIC DATA = WORK DESC;
MODEL Y = D_woe;
RUN;
```

Side Comments … regarding Binning of D

- Modeler can enter D into a logistic model as WOE or Dummies …
- … But first: D should be "*binned*":
  - Reduces number of levels of D while maintaining predictive power
  - Simplifies model, removes outliers, improves validation on holdout

Before Binning		
D	Y = 0	Y = 1
D=D1	200	300
D=D2	100	300
D=D3	200	100
D=D4	205	95

After Binning		
D	Y = 0	Y = 1
D=D1	200	300
D=D2	100	300
D in (D3 D4)	405	195

See: Lund, B. SAS® Macros for Binning Predictors with a Binary Target, SGF 2017

Why Use WOE?

Take Discrete D and other predictors Z

➔ Model Log-Odds are <u>*Linear*</u> vs. Empirical Log-Odds (… other Z fixed)

$$\text{Log}(\ P\ /\ (1-P)\ |\ D=D_j\ ) = \text{xbeta} \ldots = \widehat{\beta_{D\_woe}} * D\_woe(D_j) + \alpha + \widehat{\beta_Z} * Z$$

where: Z are other predictors … α is intercept

Recall: $D\_woe(D_j) = \text{Log}(g_j\ /\ b_j\ |\ D=D_j)$ … = the Empirical Log-Odds

… <u>*not*</u> true for dummy variable coding of D

FROM ABOVE: If D has an ordering … (and fix Z)

… if D_woe is monotonic vs. D, then P is monotonic vs. D

… <u>*not*</u> always true for dummy variables

Why Use WOE? … continued

- Fewer parameters are added to the logistic model.
  - If D has L levels, then dummy coding adds L-1 parameters versus only 1 for WOE
- D_woe is numeric … can compare to other numeric predictors to assess collinearity
- Natural connection between WOE coding and generation of Risk Model "scorecard"
  - See Book: Siddiqi, 2017. *Intelligent Credit Scoring*

“Degrees of Freedom Problem” for WOE Predictors

Models (A) and (B) are the Same

Let discrete predictor D have L > 2 levels.

Following 2 models are equal (i.e. produce same probabilities):

(A) PROC LOGISTIC DESC; CLASS D; MODEL Y=D;

(B) PROC LOGISTIC DESC; MODEL Y=D_woe;

In Model (A) … D has L-1 degrees of freedom

Therefore, In Model (B) … D_woe must have L-1 d.f.

Adding a Predictor in Models (A), (B)

Let Z be a numeric predictor and let D have L > 2 levels

Then models (A), (B) are NOT the same (i.e. different probabilities):

(A) PROC LOGISTIC DESC; CLASS D; MODEL Y= Z D;

(B) PROC LOGISTIC DESC; MODEL Y= Z D_woe;

D (as dummies) adds L-1 d.f if added to MODEL Y= Z;

But now, what about adding D_woe to MODEL Y= Z; ... ?

... How many d.f. to assign to D_woe?

1? ... but D_woe had L-1 > 1 d.f. in earlier MODEL Y= D_woe;

L-1? ... but is L-1 too much? ... same as D in CLASS D?

ANSWER: In range: $1 \le$ d.f. $\le$ L-1 ... But where?

Does it matter how many d.f. to assign to D_woe?

- Consider FORWARD SELECTION when fitting a logistic model ...
- Suppose Z is already in the model
- Suppose the choice for the next predictor to enter is X or D_woe
  ➔ How to choose between X and D_woe?
- The d.f. for D_woe matters because it determines:
  - P-value of Chi-Sq to Enter ... Chi-Sq probability depends on d.f.
  - AIC to Enter (AIC = -2LL + 2*(d.f.)) ... depends on d.f. (... also BIC)

Normal practice is to assign 1 d.f. to D_woe in model fitting

- ... This normal practice unfair to X because ...

D_woe benefitted from heavy involvement of Y in its coding

Assigning d.f. to WOE Predictors

Model Comparison Test -- Reminder

- Model comparison test requires truly Nested models … The "Restricted" model is "nested" in the "Full" Model where …
  - The "Full" model uses every parameter in the "Restricted" model plus some additional parameters.
- Test statistic T is the difference of the "-2*Log(L)" from 2 models:

$$T = -2*\text{Log}(L)_{restricted} - (-2*\text{Log}(L)_{full})$$

- For large samples the distribution of T is chi-square with degrees of freedom = $d.f._{full} - d.f._{restricted}$
- Let t be value of T from a sample.
  - Specify α in $0 < \alpha < 1$ (e.g. $\alpha = 0.5$)
  - If $P(T > t) \geq \alpha$, then restricted and full model are "equal"

"Nesting" of WOE within CLASS

Consider discrete D, numeric Z, binary target Y

Find *max. likelihood* estimates (MLE_{WOE}) for:

PROC LOGISTIC; MODEL Y = D_woe Z; ... WOE model

Coefficients can *always* be found for CLASS model:

PROC LOGISTIC; CLASS D; MODEL Y = D Z; ... CLASS model

so that WOE model and CLASS model have *same* probabilities:

That is: $P_{CLASS}(Y=1 \mid D, Z) = P_{WOE}(Y=1 \mid D, Z)$

These Coefficients are derived by using MLE_{WOE} as starting point

BUT: These Coefficients are NOT the MLE_{CLASS}

- Generalizes to multiple WOE's and multiple Z's

This is the new "NESTING" ... math behind this slide is in paper

Assign d.f. to D_woe on Model Entry

Consider models (CLASS) and (WOE) … D discrete and Z numeric.

(CLASS) PROC LOGISTIC DESC; CLASS D; MODEL Y=D Z;

(WOE) PROC LOGISTIC DESC; MODEL Y=D_woe Z;

Full Model (CLASS) and Restricted Model (WOE) are "nested"

Consider (with some abuse) a *model comparison statistic*:

$$\text{Chi-Sq}_{C\text{-}W} = -2*\text{Log}(L)_{WOE} - (-2*\text{Log}(L)_{CLASS})$$

? $d.f._{C\text{-}W}$? ? $d.f._{WOE}$? known d.f.

… $d.f._{C\text{-}W}$ must be *assigned* somehow to comparison stat.

… then use: $d.f._{woe} = d.f._{class} - d.f._{C\text{-}W}$

Assign d.f. to D_woe on Model Entry

Consider (with some abuse) a model comparison statistic:

$$\text{Chi-Sq}_{C\text{-}W} = -2*\text{Log}(L)_{WOE} - (-2*\text{Log}(L)_{CLASS})$$

? $\text{d.f.}_{C\text{-}W}$? ? d.f._{WOE} ? known d.f.

From the sample, compute and let $t = \text{Chi-Sq}_{C\text{-}W}$

Given α, there is k d.f. so that:

$$P(\text{Chi-Sq}_{C\text{-}W} \geq t \mid k) = \alpha$$ ... k may be fractional

➔ DEFINITION: $\text{d.f.}_{C\text{-}W}$ equals k

CLASS and WOE are *just barely* statistically equal for this k

With this definition: $\text{d.f.}_{woe} = \text{d.f.}_{class} - \text{d.f.}_{C\text{-}W}$

Assign d.f. to D_woe on Model Entry

The 1 d.f. for Z and in WOE model and CLASS model cancels out …

… making d.f. for entry of D_woe to the MODEL Y = Z;

$$d.f._{D\_woe} = L\text{-}1 - d.f._{C\text{-}W}$$

Let L = 5 and suppose $d.f._{C\text{-}W}$ = 2.4, then $d.f._{D\_woe}$ = 5-1 – 2.4 = 1.6

The formula generalizes for any "Z" (multiple predictors) in a model

Forward Selection with minimum AIC,

… and adjusted d.f. for WOE predictors

MACRO to Implement FORWARD with best AIC

FORWARD SELECTION: SELECT predictor with min AIC but *adjusted* d.f. for WOE's

The adjustment uses the formula from the prior slide

- Uses PROC LOGISTIC or HPLOGISTIC (user's choice) … (also option to use BIC)

Here is example dataset Test.

- Note: classification variables C1-C5 *do not* appear in the equation for target Y
- But B1 B2 N1 N2 *do* appear in equation for target Y

```
DATA Test;
do i = 1 to 5000;
   call streaminit(12345);
   C1 = floor(10*rand('uniform'));
   C2 = floor(9*rand('uniform'));
   C3 = floor(7*rand('uniform'));
   C4 = floor(5*rand('uniform'));
   C5 = floor(3*rand('uniform'));
   B1 = 2*floor(2*rand('uniform')) - 1;
   B2 = 2*floor(2*rand('uniform')) - 1;
   N1 = rand('normal');
   N2 = rand('normal');
   U = rand('uniform'); e = log(U/(1-U));
   Y_star = 1*B1 + 2*B2 + 1*N1 + 2*N2 + e;
   Y = (Y_star > 0);
   output;
   end;
run;
```

C1-C5 are recoded as WOE

Obs	class_ variable	class_ levels
1	C1	10
2	C2	9
3	C3	7
4	C4	5
5	C5	3

- C1 - C5 are re-coded as WOE's
- Set α = 0.05
- Run MACRO with predictors:

B1 B2 N1 N2
C1_woe C2_woe C3_woe C4_woe C5_woe

```
DATA test2; SET test;
if C1 in ( 0 ) then C1_woe = -0.120296147 ;
if C1 in ( 1 ) then C1_woe = -0.0914636 ;
if C1 in ( 2 ) then C1_woe = -0.033780334 ;
if C1 in ( 3 ) then C1_woe = 0.0936619553 ;
if C1 in ( 4 ) then C1_woe = 0.0760205948 ;
if C1 in ( 5 ) then C1_woe = 0.1009817326 ;
if C1 in ( 6 ) then C1_woe = -0.033612409 ;
if C1 in ( 7 ) then C1_woe = -0.008736447 ;
if C1 in ( 8 ) then C1_woe = 0.1291250313 ;
if C1 in ( 9 ) then C1_woe = -0.104426927 ;
… etc. for C2 - C5
```

FORWARD Entry with adjusted d.f. for WOE's

- B2, N2, B1, N1 enter first (no surprise, each is included in equation for target)
- But then C4_woe (Levels = 5) is entered … essentially by chance
  … it has adjusted d.f. of 3.8 (note the fractional d.f.)
- Minimum AIC is reached at Step 5 … Note: No WOE's had d.f. adjusted to 1

step	min AIC var	min adj AIC	best model	adj-df for min	new model df
1	B2	5753.50		1.0	2.0
2	N2	4375.79		1.0	3.0
3	B1	3960.51		1.0	4.0
4	N1	3496.29		1.0	5.0
5	C4_woe	3490.01	*	3.8	8.8
6	C5_woe	3490.73		2.0	10.8
7	C2_woe	3498.45		7.7	18.5
8	C3_woe	3508.36		5.1	23.6
9	C1_woe	3521.94		7.8	31.4

COMPARE: FORWARD Entry with **1** d.f. for all WOE's

- Now we "*fool*" the macro so that C1_woe … C5_woe are *not* regarded as WOE
- B2 N2 B1 N1 enter the model first, as expected
- Min AIC is reached at Step 8 … four "woe's" entered … all unrelated to target Y

step	min AIC var	min adj AIC	best model	adj-df for min	new model df
1	B2	5753.50		1	2
2	N2	4375.79		1	3
3	B1	3960.51		1	4
4	N1	3496.29		1	5
5	C4_woe	3484.41		1	6
6	C2_woe	3478.67		1	7
7	C5_woe	3477.45		1	8
8	C1_woe	3477.40	*	1	9
9	C3_woe	3479.14		1	10

Giving 1 d.f. to the 5 woe's "damages" the model by

- adding meaningless predictors
- loss of parsimony

Comments

The Role of Alpha … a Tuning Parameter of the Macro

- Modeler should regard α as a *tuning* parameter and not so much as type I error probability for hypothesis testing
- Modeler may want to experiment with values of α before final model fitting has begun
- For fixed sample size N, if α is decreased to near 0, then adjusted d.f. for D_woe will reach L-1
- Otherwise, if α is increased to near 1, then the adjusted d.f. for D_woe will reach 1
- Chi-square test is sensitive to sample size N, and for large samples the α level should be decreased

Final Comments … 1

- The adjustment of d.f. is likely to reduce the entry of WOE predictors into the model …

  … this is because …

  - AIC = -2LL + 2*(d.f.) … (smaller is better)
  - The WOE adjustment increases d.f.

Binning increases the chances for a WOE predictor to enter the model:

➔ Reason: Predictive power is largely maintained by binning while binning reduces d.f.

Final Comments … 2

- My macro works well … BUT is computationally intensive:
  - If there are N numeric predictors and C WOE predictors, then the worst case number of (HP)LOGISTIC calls is:

$$N*(N+1)/2 + 2*N*C + C(C+1)$$

… *quadratic* growth in PROC calls v. number of predictors

- But MACRO has parameter options to *avoid* long run times.
  - Stop running when min AIC is reached
  - Predictors known to be good can be “included” (not part of FORWARD)

Example with 20 predictors and N = 5,000 runs in ~ 15 seconds

… But 200 predictors and N = 100,000 is NOT practical

… Upper limit? … maybe 60 predictors and N = 10,000

Thank you!

Contact Information
Bruce Lund
blund_data@mi.rr.com
blund.data@gmail.com