**Paper ###-2021**

# Weight of Evidence, Dummy Variables, and Degrees Of Freedom

Bruce Lund, Statistical Consultant and Trainer, Novi MI

## ABSTRACT

Predictive models with a binary target are often fitted by logistic regression. An important step in using logistic regression is transforming of predictors before the model fitting stage. In credit risk modeling and direct marketing modeling, predictors are often transformed by weight of evidence (WOE) coding. This approach applies to discrete predictors, whether nominal, ordinal, or numeric. It also applies to continuous numeric predictors after such a predictor has been reduced to discrete ranges ("fine classing"). A widely used alternative to WOE coding is dummy variable coding. Let C be a discrete predictor and C_woe be its WOE coding. A model where C_woe is the *only* predictor has the same probabilities as the model with C appearing in CLASS C as the only predictor. Hence, the degrees of freedom of C_woe is L-1 where C has L levels. But if there are additional predictors in the model, it is unclear how to assign degrees of freedom to C_woe when considering the entry of C_woe into the model. Does C_woe have 1 degree of freedom, L-1 degrees of freedom, or something in between? This ambiguity affects the usage of predictor selection methods based on p-value significance, AIC, or BIC. In this paper it is shown that a model with predictor C_woe and other predictors <X> can be thought of as "nested" within the model with CLASS C and predictors C and <X>. It is this nesting property which suggests a process to assign degrees of freedom to C_woe when entering a model. This process enables the use of forward selection to select predictors for entry where the d.f. for WOE predictors are adjusted (not simply assigned 1 d.f.). Lastly, an algorithm is provided that applies forward selection with adjusted d.f. for WOE predictors to choose the logistic model that gives minimum AIC.

## INTRODUCTION

Predictive models with a binary target are often fitted by logistic regression. [1] An important step in using logistic regression is the transforming of predictors before the model fitting stage. In credit risk modeling and direct marketing modeling, predictors are transformed by weight of evidence (WOE) coding. The books by Siddiqi (2019), Finlay (2010), and Thomas (2009) show the usage of weight of evidence coding for credit risk modeling.

This WOE approach applies to discrete predictors, whether nominal, ordinal, or numeric. It also applies to continuous numeric predictors after the predictor has been reduced to discrete ranges ("fine classing").

A widely used alternative to WOE coding is dummy variable coding. Let C be a discrete predictor and C_woe be its WOE coding. A model where C_woe is the *only* predictor has the same probabilities as the model with C appearing in CLASS C as the only predictor. Hence, the degrees of freedom of C_woe is L-1 where C has L levels. But if there are additional predictors in the model, it is unclear how to assign degrees of freedom to C_woe when considering the entry of C_woe into the model. Does C_woe have 1 degree of freedom, L-1 degrees of freedom, or something in between? This ambiguity affects the usage of predictor selection methods based on p-value significance, AIC, or BIC. [2]

In this paper it is shown that a model with predictor C_woe and other predictors <X> can be thought of as "nested" within the model with CLASS C and predictors C and <X>. It is this nesting property which suggests a process to assign degrees of freedom to C_woe when entering a model.

This process enables the use of forward selection to select predictors for entry into the model where the d.f. for WOE predictors are adjusted (not simply assigned 1 d.f.).

Lastly, an algorithm is provided that applies forward selection with adjusted d.f. for WOE predictors to choose the logistic model that gives minimum AIC. I have a SAS macro which implements the algorithm.

---

[1] In this paper it is assumed that a logistic model does not have complete or quasi-complete separation. After models with separation are excluded, the logistic model has a unique maximum likelihood estimate. For discussion of separation, see Allison (2012, ch. 3).

[2] AIC = -2*Log(L) + 2*K where K = 1 + d.f. of predictors in model. BIC = -2*Log(L) + log(N)*K with sample size N.

## DEFINITION AND FORMULA FOR WEIGHT OR EVIDENCE CODING

In **Table 1** the weight of evidence transformation (or coding) of predictor C is illustrated. The right-hand column gives the value "WOE(cj)" of the weight of evidence transformation for C = $c_j$.

| C | Y = 0 "Bj" | Y = 1 "Gj" | Col % Y=0 "bj" | Col % Y=1 "gj" | **WOE(ci)= Log(gj/bj)** |
|---|---|---|---|---|---|
| c1 | 2 | 1 | $B_1$ / B = 0.250 | $G_1$ / G = 0.125 | -0.69315 |
| c2 | 1 | 1 | $B_2$ / B = 0.125 | $G_2$ / G = 0.125 | 0.00000 |
| c3 | 5 | 6 | $B_3$ / B = 0.625 | $G_3$ / G = 0.750 | 0.18232 |
| SUM | B=8 | G=8 | | | |

**Table 1**. **Weight Of Evidence Transformation of C**

The weight of evidence transformation is given here as a formula:

$$\text{If } C = c_j \text{ then } C\_woe(c_j) = \log((G_j / G) / (B_j / B)) = \log(\, g_j / b_j \,)$$

Where:

$g_j$ is the column percentage in the $j^{th}$ row of Y=1, $b_j$ is the column percentage in the $j^{th}$ row of Y=0, G is the total count of Y=1 and B is the total count of Y=0.

If either $g_j$ or $b_j$ is zero, then C_woe is not defined.

There is not a tidy formula for computing the values for C_woe during processing of observations in a DATA Step. This is because the count of all occurrences of Y=1 and of Y=0 must first be made for each level C = $c_j$ as well as the count of totals for Y=1 and for Y=0. Instead, a PROC SUMMARY is needed:

```
PROC SUMMARY data = Table1;
CLASS C Y;
TYPES C*Y Y;
OUTPUT OUT = WORK;
```

Next, a DATA Step is needed to process data set `WORK` to compute the weight of evidence coding. The DATA Step logic must check for zero cells and, if none, then compute weight of evidence coding of C.[3] [4]

There are packages in R that provide a simple way to find weight of evidence coding. One such package is called "InformationValue". The R language code given below computes the WOE coding for the data of **Table 1**. This package was created by S. Prabhakaran (2016).

```
install.packages("InformationValue")
library("InformationValue")
# Data Used in Table 1 - it is weighted
weighted <- read.table(text="C Y W
c1 0 2
c1 1 1
c2 0 1
c2 1 1
c3 0 5
c3 1 6", header=TRUE)
# expand without weight
raw <- weighted[rep(1:nrow(weighted), weighted[["W"]]), ]
raw$C <- as.factor(raw$C)
C_woe <- WOETable(C=raw$C, Y=raw$Y)
print(C_woe)
```

[3] SAS macros for computing WOE, as a by-product of variable binning, are given in Lund (2017). Since the publication of this 2017 paper, the macros have been enhanced. Contact the author for latest version of the macros.
[4] PROC HPBIN can compute the weight of evidence for numeric variables X. See parameter NUMBIN to specify the number of levels of X and see Example 4.5 in the documentation for PROC HPBIN.

Weight of Evidence Codes Computed by R Package

| | CAT | GOODS | BADS | TOTAL | PCT_G | PCT_B | WOE | IV |
|---|---|---|---|---|---|---|---|---|
| 1 | c1 | 1 | 2 | 3 | 0.125 | 0.250 | -0.6931472 | 0.08664340 |
| 2 | c2 | 1 | 1 | 2 | 0.125 | 0.125 | 0.0000000 | 0.00000000 |
| 3 | c3 | 6 | 5 | 11 | 0.750 | 0.625 | 0.1823216 | 0.02279019 |

**Table 2. Weight Of Evidence Coding of C produced by R Information Value Package**

## WHEN X IS ORDERED AND X_WOE IS MONOTONIC VS. X

If X is ordered (but not necessarily numeric) and if the relationship between X and X_woe is monotonic, then the relationships between:

(i) X and Gj / $B_j$

as well as

(ii) X and Gj / (Gj + $B_j$)

are also monotonic.

The converse to these statements is true as well.

A proof is given below:

If Log (gj/bj) ≤ Log($gj_{+1}$/$bj_{+1}$) then Log (Gj/Bj) - Log(G/B) ≤ Log($Gj_{+1}$/$Bj_{+1}$) - Log(G/B)
Subtracting Log(G/B) from each side gives: Log (Gj/Bj) ≤ Log($Gj_{+1}$/$Bj_{+1}$)
Taking exponents gives: Gj/Bj ≤ $Gj_{+1}$/$Bj_{+1}$ … (i)
Equivalently: Gj$Bj_{+1}$ ≤ $Gj_{+1}$Bj
Adding Gj$Gj_{+1}$ to each side gives: Gj$Bj_{+1}$ + Gj$Gj_{+1}$ ≤ $Gj_{+1}$Bj + Gj$Gj_{+1}$
Equivalently, Gj($Bj_{+1}$ + $Gj_{+1}$) ≤ $Gj_{+1}$(Bj + Gj)
Finally: Gj/(Gj+Bj) ≤ $Gj_{+1}$/($Gj_{+1}$+$Bj_{+1}$) … (ii)

These steps are valid in reverse order.

Likewise, if there is a curvilinear relationship between X and X_woe, then a similar relationship also holds between X and Gj / Bj as well as X and Gj / (Gj + Bj).

## CLASS STATEMENT AND DUMMY VARIABLE CODING

Suppose C has L levels and C appears in a CLASS statement in a logistic model:

PROC LOGISTIC descending; CLASS C (PARAM=ref); MODEL Y = C <other predictors>;

The statement CLASS C with (PARAM=ref) has the effect of creating dummy variables for the lowest (in natural sort order) of the L-1 levels of C. The effect of `PARAM=ref` is to set to zero the implied coefficient of the dummy variable for C = $c_L$.

Let dummy variables be created from C and be denoted by $D_j$ = (C = $c_j$) for j = 1 to L-1

The equivalent dummy variable MODEL can be expressed, in terms of log-odds as:

Log (P / (1-P)) = α + $\sum_{j=1}^{L-1} \beta_j * D_j$ + <other predictors> … "CLASS model"

The SAS code is

PROC LOGISTIC descending; MODEL Y = $D_1$ - $D_{L-1}$ <other predictors>;

## WEIGHT OF EVIDENCE and DUMMY VARIABLE CODING

Weight of evidence recoding of C is an alternative to using dummy variable coding for entering the NOD predictor C into a logistic model.

With C_woe the model becomes:

PROC LOGISTIC descending; MODEL Y = C_woe <other predictors>;

The weight of evidence model can be expressed, in terms of log-odds as:

$$\text{Log } (P/(1-P)) = \alpha + \beta * C\_woe + \text{<other predictors>; … “WOE model”}$$

## THE CASE OF NO “OTHER PREDICTORS”

As is well known, Log Likelihood, abbreviated by “Log(L)”, is a measure of fit of a logistic model and the maximum likelihood estimate (MLE) of the predictors for the logistic model maximizes Log(L).

In the exceptional case where there are no “other predictors” in the logistic model, the two models (“CLASS” and “WOE”) are the same (they have the same probabilities) as explained below.

The maximum likelihood estimators for the WOE model are given here:

$$\alpha = \log(G/B) \text{ and } \beta = 1 \text{ … (see Appendix for discussion.)}$$

Using the MLE’s the probability for $C=c_j$ is easily seen to be (here, via log-odds):

$$\text{Log}(P / (1-P) \mid C=c_j) = \log(Gj / Bj)$$

The maximum likelihood estimators for the CLASS model are given below:

$$\alpha = \log(G_L / B_L) \text{ and } \beta_j = \log(Gj / Bj) - \log(G_L / B_L) \text{ … (see Appendix for discussion.)}$$

Using the MLE’s the probabilities for $C=c_j$ are, again, seen to be (here, via log-odds):

$$\text{Log}(P / (1-P) \mid C=c_j) = \log(Gj / Bj)$$

## THE CASE OF “OTHER PREDICTORS” AND “NESTING”

The model with WOE predictors as well as other predictors is “nested” (in a sense to be explained) within the model with the corresponding CLASS predictors and the same “other predictors”. The term “nested” is used here in a non-standard manner. Here is the usage:

For the CLASS model there exist values for the coefficients that give the same probabilities as the probabilities from the MLE solution for the WOE model. These coefficients for the CLASS model are not the MLE’s. There are solutions with greater log likelihood (or at least equal) for the CLASS model when evaluated at its MLE. That is, $\text{Log(L)}_{\text{WOE(MLE)}} \leq \text{Log(L)}_{\text{CLASS(MLE)}}$.

Here is a proof for the special class of one classification predictor C and one numeric predictor X. A more general case is discussed in the Appendix.[5]

Consider binary target Y, classification predictor C with four levels c1, c2, c3, c4, and one numeric predictor X. Let C_woe give the weight of evidence coding of C.

Consider the WOE model:

PROC LOGISTIC descending; MODEL Y = C_woe X;

Let $\alpha 0$, $\beta 0$, $\lambda 0$ denote the MLE coefficients for intercept, coefficient of C_woe, and coefficient of X respectively. The WOE values for C will be abbreviated by w1= C_woe(c1), …, w4= C_woe(c4).

Let dummy variables Dc1, Dc2, Dc3 be created for C=c1, C=c2, C=c3 by:

Dc1 = (C= “c1”), etc.

Consider the CLASS model:

PROC LOGISTIC descending; MODEL Y = Dc1 Dc2 Dc3 X;

[5] In the generalization discussed in the Appendix it is assumed that WOE predictors do not appear in interactions with each other or with other predictors.

It will now be shown that coefficients α1, β11, β12, β13, λ1 for the intercept, Dc1, Dc2, Dc3, and X can be found for the CLASS model so that the CLASS model is the same model (same probabilities) as the WOE model where the WOE model is evaluated at its MLEs.

The sought-after CLASS model coefficients α1, β11, β12, β13, λ1 must satisfy equations shown below with respect to the WOE model at its MLE solution given by α0, β0, λ0.

| | WOE at MLE | | CLASS |
|---|---|---|---|
| When C=c4 and X=0: | α0 + w4* β0 | = | α1 |
| When C=c4 and X=1: | α0 + w4* β0 + λ0 | = | α1 + λ1 |

This implies that λ1 = λ0 and α1 = α0 + w4* β0

These equation must also hold:

| | | | |
|---|---|---|---|
| When C=c1 and X=0: | α0 + w1* β0 | = | α1 + β11 |
| When C=c2 and X=0: | α0 + w2* β0 | = | α1 + β12 |
| When C=c3 and X=0: | α0 + w3* β0 | = | α1 + β13 |

The equations above imply that

β11 = α0 + w1* β0 - α1 = α0 + w1* β0 - α0 - w4* β0 = w1* β0 - w4* β0

with similar equations for β12 and β13

The coefficients for CLASS have been solved in terms of the given coefficients for the MLE solution for the WOE model. Therefore, the WOE model is "nested" in the CLASS model.

The SAS code below gives an example for this simple case:

First a data set is created with classification variable C and numeric variable X. The target is Y.

```
DATA test;
do i = 1 to 500;
   z = rannor(1);
   C = (-2 <= z <= 2) * z;
   C = floor(C);
   X = ranuni(1) - .5;
   Y_star = C - 0.2*X + 2*rannor(1);
   Y = (Y_star > 0);
   output;
   end;
run;
```

Using a PROC SUMMARY and a DATA Step, the weight of evidence coding for C is obtained and inserted into the DATA Step below:

```
DATA test2; set test;
if C in ( -2 ) then C_woe = -1.557080788 ;
if C in ( -1 ) then C_woe = -0.430782916 ;
if C in (  0 ) then C_woe = 0.320521773 ;
if C in (  1 ) then C_woe = 1.259833467 ;
Dc1 = (C=-2);
Dc2 = (C=-1);
Dc3 = (C= 0);
Dc4 = (C= 1);
run;
```

Then PROC LOGISTIC fits the WOE model. The option "itprint" displays the coefficients as the fitting algorithm converges to the MLE solution.

```
PROC LOGISTIC data= test2 desc;
model Y= C_woe X / itprint;
score data= test2 out= woe;
run;
```

| Maximum Likelihood Iteration History | | | | | |
|---|---|---|---|---|---|
| Iter | Ridge | -2 Log L | Intercept | C_woe | X |
| 0 | 0 | 674.600231 | -0.388826 | 0 | 0 |
| 1 | 0 | 616.649727 | -0.336143 | 0.868390 | -0.004264 |
| 2 | 0 | 615.547512 | -0.386416 | 0.993267 | -0.019273 |
| 3 | 0 | 615.544706 | -0.388778 | 1.000243 | -0.020309 |

**Table 3. Maximum Likelihood Iterations**

The MLE solution is used as the initial parameter values for fitting the CLASS model. These initial coefficients for the intercept and X are copied directly (-0.388778 and -0.020309). The coefficient of C_woe which will be denoted by “beta0” remains to be used to define the initial coefficients for the three dummy variables in the CLASS model (corresponding to the four levels of C)

The initial coefficients for the three dummy variables involve the beta0 coefficient and the weight of evidence values w1, w2, w3, and w4 for C_woe. Here is the formula for beta11 (with similar formulas for beta12 and beta13):

beta11 = beta0*(w1 - w4) = 1.000243 * (-1.557080788 - 1.259833467) = -2.81760.

These initial coefficient estimates are saved in a DATA step called “initial” where the corresponding variable names Dc1, Dc2, Dc3 must be used for the coefficients beta11, beta12, beta13.

```
DATA initial;
beta1= 1.000243;
correction1 = (1.259833467)*beta1;
Dc1 =  -1.557080788*beta1 - correction1;
Dc2 =  -0.430782916*beta1 - correction1;
Dc3 =   0.320521773*beta1 - correction1;
X = -0.020309;
intercept= -0.388778 + correction1;
output;
run;
PROC PRINT data=initial;
run;
```

| Obs | beta0 | correction1 | Dc1 | Dc2 | Dc3 | X | intercept |
|---|---|---|---|---|---|---|---|
| 1 | 1.00024 | 1.26014 | -2.81760 | -1.69103 | -0.93954 | -0.020309 | 0.87136 |

These initial estimates are used in the 0th iteration of PROC LOGISTIC, below, by the statement `INEST= initial.` No iterations are run for the CLASS model when `maxiter= 0` and model-fit uses the initial coefficient estimates. The same -2*Log(L) of 615.544 is found as for the WOE model.

```
PROC LOGISTIC data = test2 desc INEST = initial;
model Y = Dc1 Dc2 Dc3 X  / maxiter = 0 itprint;
score data = test2 out = class;
run;
```

| Maximum Likelihood Iteration History | | | | | | | |
|---|---|---|---|---|---|---|---|
| Iter | Ridge | -2 Log L | Intercept | Dc1 | Dc2 | Dc3 | X |
| 0 | 0 | 615.544706 | 0.871362 | -2.817599 | -1.691027 | -0.939540 | -0.020309 |

Comparing output data sets `woe` and `class` it is seen that data set `both` (below) is empty.

```
DATA both; merge woe(rename=(P_1 = woe_P_1))
                 class(rename=(P_1 = class_P_1)); by i;
if abs(woe_P_1 - class_P_1) > .0001 then output;
run;
```

Let the log likelihood for the class model be $Log(L)_{CLASS}$ and the log likelihood for weight of evidence be $Log(L)_{WOE}$. Then, $Log(L)_{WOE} \leq Log(L)_{CLASS}$.

With an abuse of terminology, the results above show the WOE model is "***nested***" in the CLASS model.

## WHY IS WEIGHT OF EVIDENCE EVEN CONSIDERED?

Why is the usage of C_woe considered in view of the greater fit from using CLASS C? Several reasons are presented below.

### MODEL LOG-ODDS ARE LINEAR VS. EMPIRICAL LOG-ODDS OF PREDICTOR C

The WOE coding of a predictor C gives the modeler more control over how this predictor impacts the predictions of the model as explained below:

The log-odds in the WOE model are given by this equation (provided C_woe is not part of an interaction with another predictor):

$$Log( P / (1-P) \mid C=c_k ) = xbeta = \widehat{\beta_{C\_woe}} * C\_woe(c_k) + \alpha + \widehat{\beta_Z}*Z$$

where Z gives the terms in xbeta for the other predictors and α is the intercept

Therefore, for fixed values of Z, the model log-odds are linearly related to the weight of evidence.

Now, remembering that $Log(g_j/b_j) = X\_woe(c_j)$ and expanding $Log(gj/bj) = Log(Gj/Bj) - Log (G/B)$ gives:

$$Log( P / (1-P) \mid C=c_j ) = xbeta = \widehat{\beta_{C\_woe}} * Log( Gj/Bj ) + \alpha - Log (G/B) + \widehat{\beta_Z}*Z$$

The empirical log-odds of success based solely on $C = c_j$ are:

$$Log(Gj/Bj) = Log ( [Gj/T_j] / [Bj/T_j] ) \text{ where } T_j = Gj + Bj$$

Thus, for fixed values of Z, the model log-odds are linearly related to the empirical log-odds.

### IF C_WOE IS MONOTONIC VS. C, THEN MODEL LOG-ODDS ARE MONOTONIC VS. C

Suppose C is ordinal and C_woe is monotonic with respect to C. Then the formulas developed above show there is a monotonic relationship between C and Log( P / (1-P) ), where P is computed for values of C with other predictors being held fixed.

### MODEL LOG-ODDS ARE NOT NECESSARILY MONOTONIC VS. C FOR CLASS MODEL

In the case of dummy variable coding with L-1 fitted coefficients, the effect of C on Log( P / (1-P) ) is given through $\sum_{j=i}^{L-1} \hat{c}_j * Dcj$ where Dcj = (C = $c_j$) while other predictors are held fixed. Here, $\hat{c}_j$ is the coefficient of dummy variable Dcj, with reference level coding applied to the largest level of C.

The dummy variable relationship between C and $\sum_{j=i}^{L-1} \hat{c}_j * Dcj$ need not be monotonic despite having a monotonic relationship between C and C_woe. An example is given in a later section of the paper.

### OTHER VIRTUES OF WEIGHT OF EVIDENCE CODING INCLUDE:

- Fewer parameters are added to the logistic model. If C has L levels, then dummy coding adds L-1 parameters versus only 1 for WOE. Reduction in the number of parameters could be considerable.
- C_woe is numeric and can be compared with other numeric predictors to assess collinearity.

WOE is widely used in credit risk modeling since there is a natural connection between WOE coding and the generation of a "scorecard".[6]

[6] See: Siddiqi (2017)

## BUT PROBLEMS WITH WOE IN PREDICTOR SELECTION WHEN MODEL FITTING

A disadvantage of WOE coding is that the degrees of freedom for C_woe, when being entered into a logistic model, are unknown. Here is an explanation.

Let C have L > 2 levels. Then the following two models are the same (i.e. produce the same probabilities):

(A) PROC LOGISTIC DESCENDING; CLASS C; MODEL Y=C;

(B) PROC LOGISTIC DESCENDING; MODEL Y=C_woe;

Model (A) uses L-1 degrees of freedom. Therefore, Model (B) must also use L-1 d.f.

Suppose numeric predictors W and Z are added to models (A) and (B) to form models (A2) and (B2):

(A2) PROC LOGISTIC DESCENDING; CLASS C; MODEL Y = C W Z;

(B2) PROC LOGISTIC DESCENDING; MODEL Y = C_woe W Z;

Then $Log(L)_{A2} \geq Log(L)_{B2}$. The degrees of freedom for Model (A2) equals L-1 + 2. The degrees of freedom for Model (B2) are undetermined but lie between 3 and L-1 + 2. It is difficult to accept that the d.f. of Model (B2) would only be 3 after noting that Model (B) has L-1 d.f. where L could be much greater than 2. On the other hand, to fully load Model (B2) with L-1 + 2 d.f. seems wrong in cases where $Log(L)_{A2}$ is much larger than $Log(L)_{B2}$ and, therefore, (A2) and (B2) are not, at all, the same models.

Of course, the problem in assigning d.f. to Model (B2) is due to the problem in assigning d.f. to C_woe as it enters a model already having W and Z. Should C_woe have 1 d.f. or L-1 d.f. or something else?

The d.f. assignment is important when considering the use of p-values in predictor selection methods (e.g. stepwise p-value based) and also for predictor selection methods based on minimum BIC and AIC (as are provided by PROC HPLOGISTIC).

None of the SAS procedures allow for d.f. adjustment for WOE predictors. Of course, more fundamentally, it has been unclear how to make such an adjustment.

I have not seen a discussion of how to assign degrees of freedom for WOE predictors in modeling applications. I assume that, in practice, C_woe is simply regarded as having 1 d.f. in conformance with its usage in PROC LOGISTIC and PROC HPLOGISTIC. In situations with many predictors, is the use of the 1 d.f. assignment a reasonable simplifying assumption? Research on the question is needed. Some insights are given in the discussions which follow.

## UNCORRELATED VARIABLES AND D.F. FOR WOE CODED PREDICTORS

What rules or guidelines are there for assigning degrees of freedom to a weight of evidence predictor when fitting a model with other predictors? I wondered whether correlation among predictors might influence the d.f. assigned to C_woe in a logistic model. Specifically, if C_woe is almost uncorrelated with X, do these two models (1) and (2) have roughly equal log likelihood?

(1) MODEL Y = C_woe X; and (2) CLASS C; MODEL Y = C X;

If so, then degrees of freedom assigned to C_woe in the first model could be L-1 where C has L levels.

I constructed an example where C has L = 3 levels and C_woe was almost uncorrected to numeric predictor X. In the example below the correlation between C_woe and X is 0.02796. Meanwhile, the “WOE model” has -2*Log(L) of 3657.523 while the “CLASS model” had -2*Log(L) of 3494.334.

```
DATA test;
do C = 1 to 3;
  if C = 1 then do;
    do i = 1 to 1000;
      if ranuni(1) < .40 then Y = 0; else Y = 1;
      X = rannor(1);
      output;
      end;
```

```
      end;
    if C = 3 then do;
      do i = 1 to 1000;
        if ranuni(1) < .60 then Y = 0; else Y = 1;
        X = rannor(1);
        output;
        end;
      end;
    if C = 2 then do;
      do i = 1 to 1000;
        if ranuni(1) < .50 then Y = 0; else Y = 1;
        if Y = 1 then X = 3; else X = rannor(1);
        output;
        end;
      end;
   end;
run;
DATA test2; set test;
if C = 1 then C_woe=  0.418371747;
if C = 2 then C_woe=  0.03600749;
if C = 3 then C_woe= -0.455277827;
run;
/* WOE Model */
PROC LOGISTIC data=test2 desc;
model y = X C_woe ;
run;
/* CLASS Model */
PROC LOGISTIC data=test2 desc;
class C (PARAM=ref ref="3");
model Y = C X;
run;
```

The CLASS model, with -2*Log(L) of 3494.334 would appear to provide significantly better fit than the WOE model with -2*Log(L) of 3657.523. But can these models be compared by a statistical test?

With some abuse of the model comparison method the WOE and CLASS models are regarded as nested and compared under the assumption that C_woe has 1 d.f. The model comparison chi-square statistic is 3657.523 - 3494.334 = 163.189 with 1 d.f. The null hypothesis of equal models is strongly rejected.

But how to compare these models for C_woe having 2 d.f.? This question is taken up in a later section.

### FIRST, A FURTHER USAGE OF THE EXAMPLE

Another usage of the CLASS model of the example is to show that the model log-odds are not monotonic versus C (ordering 1, 2, 3). The dummy coefficients for C are shown below. The reference level coding gives zero for the coefficient for C = 3. The dummy coefficients (1.0179, -0.7866, 0) are not monotonic versus C. This implies a non-monotonic relationship between Log (P / (1-P)) and C, for fixed X.

| Analysis of Maximum Likelihood Estimates | | | | | | |
|---|---|---|---|---|---|---|
| Parameter | | DF | Estimate | Standard Error | Wald Chi-Square | Pr > ChiSq |
| Intercept | | 1 | -0.5048 | 0.0691 | 53.4029 | <.0001 |
| C | 1 | 1 | 1.0179 | 0.0982 | 107.4346 | <.0001 |
| C | 2 | 1 | -0.7866 | 0.1203 | 42.7533 | <.0001 |
| X | | 1 | 0.8175 | 0.0387 | 445.3974 | <.0001 |

**Table 4. Coefficients of Dummy Variables formed for Predictor C**

## HOW TO ACCOUNT FOR D.F. FOR C_WOE IN SELECTION=FORWARD MODEL FITTING?

Suppose a logistic model is fit by "forward selection" where the selected predictor is the one giving smallest AIC. Assume numeric Z has already been selected. Let the candidate predictors be numeric X and weight of evidence C_woe where C has 4 levels.

Logistic procedures, (PROC LOGISTIC or PROC HPLOGISTIC), regard C_woe as having 1 degree of freedom. Suppose AIC values associated with *entry* of X or C_woe (with 1 d.f.) are given below. Using minimum AIC criterion, the C_woe is selected next.

| Predictor | -2*Log(L) | AIC |
|---|---|---|
| C_woe | **100.0** | **106.0** |
| X | 101.0 | 107.0 |

**Table 5. -2*Log(L) and AIC for Hypothetical Predictors C_woe and X**

But C_woe involves a pre-modeling transformation of C to C_woe which packs a lot of information into C_woe. What might be a "more fair" assignment of d.f. to C_woe than a default of d.f. 1? For this purpose a "model comparison test" is used which is based on "nesting" of model with C_woe within the model with CLASS C, as mentioned earlier.

The usual model comparison test requires truly nested models where each model has degrees of freedom with integer values. The test statistic T is the difference of the "-2*Log(L)" from the models:

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

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

To assign d.f. to C_woe, the statistic T is used again, but this time for CLASS and WOE models:

$$T = -2*\text{Log}(L)_{\text{woe}} - (-2*\text{Log}(L)_{\text{class}})$$

But now $\text{d.f.}_{\text{woe}}$ is unknown. For the purpose of assigning d.f. to model WOE, T will be regarded as chi-square with $\text{d.f.}_T = \text{d.f.}_{\text{woe}} - \text{d.f.}_{\text{class}}$. If, by some means, $\text{d.f.}_T$ could be assigned, then $\text{d.f.}_{\text{woe}}$ is known.

> **Loose Definition**: The $\text{d.f.}_T$ is declared to be the minimum d.f. value so that $P(T > t \mid \text{d.f.}) > \alpha$, with fractional values allowed, but where $\text{d.f.}_T$ is restricted to the interval [0, L-2]. This is the minimum d.f. that makes the models statistically equal (for α). Note that $P(T > t \mid \text{d.f.})$ is an increasing function of d.f.

For example, if $-2*\text{Log}(L)_{\text{woe}}$ is very nearly equal to $-2*\text{Log}(L)_{\text{class}}$, then C_woe carries the same information as C as a CLASS variable. So $\text{d.f.}_T$ of the definition is ~ 0. Then L-1 d.f. is assigned to C_woe. But, if $-2*\text{Log}(L)_{\text{woe}}$ is very much greater than $-2*\text{Log}(L)_{\text{class}}$, then 1 d.f. is assigned to C_woe, corresponding to $\text{d.f.}_T$ of L-2. For all other cases, the d.f. for C_woe is L-1 - $\text{d.f.}_T$. This is the broad idea. Now the operational approach is given by the procedure below called **FSAA**.

## FSAA: FORWARD SELECTION WITH ADJUSTED AIC … WITH AN EXAMPLE

Referring to **Table 5**, this FSAA process is applied to decide if C_woe or X will enter into the model.

Step 1: Enter C as a CLASS predictor (with 3 d.f.) to the logistic model already having Z.
Suppose $-2*\text{Log}(L)_{\text{CLASS}}$ for this model is 96.

Step 2: Compute t = $-2*\text{Log}(L)_{\text{C\_woe}} + 2*\text{Log}(L)_{\text{CLASS}}$. Here, $-2*\text{Log}(L)_{\text{C\_woe}}$ is the result of entering C_woe to the model already having Z. Then t = 100 - 96 = 4.

Step 3: t will be regarded as a value from a chi-square statistic T with k d.f. where $k > 0$, allowing even fractional values.[7] Now let $k_{\min}$ be the minimum $k > 0$ for which $P(T > t \mid k) > \alpha$. Let $\alpha = 5\%$. To find $k_{\min}$, $P(T > t \mid k)$ is computed for a sequence of $\{k_j\}_1^J$ beginning with $k_1 = 0.1$ and ending at $k_J = L-2$. The sequence includes at least 0.1 and integer values 1, …, L-2, but could include additional fractional values.

[7] Cumulative chi-square probability $P(T < t)$ for k d.f. is given by SAS function cdf('CHISQ', t, k), for any $k > 0$.

The computations stop at the first k where P(T > t | k) > α. (Note that f(k) = P(T > t | k) is an increasing function.) Conventions are imposed that if P(T > t | 0.1) > α, then $k_{min}$ is reset to 0 and if P(T > t | L-2) ≤ α, then $k_{min}$ is reset to L-2.

In this example, the sequence $\{k_j\}_1^J$ contains three members: {0.1, 1.0, 2.0}. The results of Steps 1 - 3 are summarized in the table where it is seen that $k_{min}$ = 2.

| α = 5% and t = -2*Log(L)$_{C\_woe}$ + 2*Log(L)$_{CLASS}$ = 100-96 = 4 | | |
|---|---|---|
| k | P(T > t \| k) | Exceeds α ? |
| 0.1 (*) | 0.0026 | No |
| 1 | 0.0455 | No |
| 2 (**) | 0.1353 | Yes |

(*) If k = 0.1, then re-assign k = 0
(**) If "Exceeds α" is "No" in all rows, then k = bottom row (here = 2)

Step 4: The degrees of freedom assigned to C_woe (with Z in the model) is determined by the formula:

d.f. for C_woe: (L-1) - $k_{min}$ = 3 - 2 = 1.

Now the two models (one with CLASS C) and (other with C_woe) give the same model, statistically.

CONCLUSION: The AIC for C_woe remains equal to 106, and C_woe is selected for entry.

Now consider a NEW CASE:

If -2*Log(L)$_{CLASS}$ = 99.7, the calculations in the table below assign $k_{min}$ = 0.

| α = 5% and t = -2*Log(L)$_{C\_woe}$ + 2*Log(L)$_{CLASS}$ = 100-99.7 = 0.3 | | |
|---|---|---|
| k | P(T > t \| k) | Exceeds α ? |
| 0.1 (*) | 0.0722 | Yes |
| 1 | 0.5839 | |
| 2 (**) | 0.8607 | |

(*) If k = 0.1, then re-assign k = 0;
(**) If "Exceeds α" is "No" in all rows, then k = bottom row value

Degrees of freedom for C_woe become 3 - 0 = 3. The new AIC for C_woe is 100 + 2*(2 + 3) = 110. Predictor X, having lower AIC of 107, is selected for entry.

**COMMENTS**

- The chi-square test is sensitive to sample size N, and for large samples the α significance level should be decreased. Alternatively, the BIC = -2*Log(L) + log(N)*(K+1) might be used.
- The sequence $\{k_j\}_1^J$ could have many values provided the first is 0.1 and the last is L-2. For this example, k might be generated by: Do k = 0.1 to 2 by 0.1;
- Implementation of the process suggested here requires a complex and computationally intensive SAS macro program. In particular, there is no SAS procedure that neatly supports this process.[8]

[8] The PROC HPLOGISTIC code below might appear to provide a short-cut for FSAA processing:

```
PROC HPLOGISTIC DATA = WORK;
   MODEL Y = X Z C_woe;
   SELECTION METHOD=FORWARD (SELECT=AIC CHOOSE=AIC STOP=AIC) DETAILS=ALL;
PROC HPLOGISTIC DATA = WORK;
   CLASS C; MODEL Y = X Z C;
   SELECTION METHOD=FORWARD (SELECT=AIC CHOOSE=AIC STOP=AIC) DETAILS=ALL;
run;
```

Due to DETAILS=ALL the AIC for each candidate variable is reported at each step. The AIC's for C and C_woe might be compared in each step. But AIC is approximated in the HPLOGISTIC report "Candidate Entry and Removal Details". The approximation error is large enough to invalidate the approach of adjusting d.f. for WOE predictors. Aside from the approximation error problem, the mechanics of using this code for FSAA become overly complex.

## AN EXAMPLE OF FSAA

PROC HPLOGISTIC documentation includes an example data set called `getStarted`.[9] It has 100 observations with binary target Y, one character variable C with 10 levels, and numeric X1 - X10. This data set is used in the example, however, only predictors X2, X8, X10, and C will be considered.

But first, predictor C will be "binned" to reduce the number of levels of C to 7 bins (i.e. levels). The binning algorithm maximizes information value (IV) at each step in the reduction from 10 levels down to 7 bins. For more details on binning see Lund (2017).[10] After binning, C_woe is created as the weight of evidence recoding of C.

```
DATA getStarted; length C $5; set getStarted;
if C in ( "A","F" ) then C_woe = -0.809318612 ;
if C in ( "B","I","H" ) then C_woe = 0.1069721196 ;
if C in ( "C" ) then C_woe = 0.6177977433 ;
if C in ( "D" ) then C_woe = -0.403853504 ;
if C in ( "E" ) then C_woe = -1.145790849 ;
if C in ( "G" ) then C_woe = -0.703958097 ;
if C in ( "J" ) then C_woe = 1.4932664807 ;
/* collapse levels of C */
if C in ( "A","F") then C = "A_F" ;
if C in ( "B","I","H" ) then C = "B_I_H" ;
run;
```

To reduce lengthy reports, SAS code is displayed and then summary results are given. The reader can run the SAS PROC LOGISTIC code to obtain the full reports. PROC HPLOGISTIC could also be used.

Step 1:

```
PROC LOGISTIC DATA= getStarted desc;
   MODEL Y = X2;
PROC LOGISTIC DATA= getStarted desc;
   MODEL Y = X8;
PROC LOGISTIC DATA= getStarted desc;
   MODEL Y = X10;
PROC LOGISTIC DATA= getStarted desc;
   MODEL Y = C_woe;
run;
```

PROC LOGISTIC treats C_woe as having 1 d.f. The FSAA process needs to consider an adjustment. With no other predictors in the model, (i) CLASS C; MODEL Y=C; and (ii) MODEL Y=C_woe are the same models. Therefore, C_woe is given 6 d.f. As shown in **Table 6**, X8 gives the minimum adjusted AIC and is entered in Step 1. Without the adjustment, C_woe would have been selected.

| Predictors in Model: Intercept | | | | |
|---|---|---|---|---|
| d.f. = | 1 | | | |
| AIC = | 125.820 | | | |
| For entry | -2*Log(L) | Adj. d.f. | Model d.f. | Adj. AIC |
| X2 | 119.804 | 1 | 2 | 123.804 |
| X8 | 119.462 | 1 | 2 | 123.462 |
| X10 | 123.229 | 1 | 2 | 127.229 |
| C_woe | 111.531 | 6 | 7 | 125.531 |

**Table 6. First Step in Forward Selection by Minimum Adjusted AIC**

[9] https://support.sas.com/documentation/cdl/en/stathpug/66410/HTML/default/viewer.htm#stathpug_hplogistic_gettingstarted01.htm

[10] The macro %NOD_BIN from Lund (2017) was used for binning. In truth, the binning process should continue past 7 to 4 or even 3 bins. But the 7 bin solution was selected for purpose of creating a good example to illustrate FSAA.

Step 2: Which is the next predictor to be entered? The PROC LOGISTIC models need to be run:

```
PROC LOGISTIC DATA= getStarted desc;
   MODEL Y = X8 X2;
PROC LOGISTIC DATA= getStarted desc;
   MODEL Y = X8 X10;
PROC LOGISTIC DATA= getStarted desc;
   MODEL Y = X8 C_woe;
PROC LOGISTIC DATA= getStarted desc;
   CLASS C; MODEL Y = X8 C;
run;
```

The degrees of freedom for C_woe for entry into the model need to the computed. C_woe is assigned L-1 - k = 6 - 0 = 6 d.f.

| $\alpha$ = 5% and t = -2*Log(L)$_{C\_woe}$ + 2*Log(L)$_{CLASS}$ = 106.243 - 106.119 = 0.124 | | |
|---|---|---|
| k (*) | P(T > t \| k) | Exceeds $\alpha$ ? |
| 0.1 | 0.1087 | Yes |
| 1 | 0.7247 | |
| etc. | etc. | |

(*) If k = 0.1, then re-assign k = 0

As shown in **Table 7**, X2 gives the minimum adjusted AIC and this AIC is smaller than the AIC (123.462) at Step 1. X2 is entered in Step 2.

| Step 2: Predictors in Model: Intercept, X8 | | | | |
|---|---|---|---|---|
| d.f. = | 2 | | | |
| AIC = | 123.462 | | | |
| For entry | -2*Log(L) | Adj. d.f. | Model d.f. | Adj. AIC |
| X2 | 114.396 | 1 | 3 | 120.396 |
| X10 | 119.056 | 1 | 3 | 125.056 |
| C_woe | 106.243 | 6 | 8 | 122.243 |

**Table 7.**

Step 3: Which is the next predictor, if any, to be entered?

```
PROC LOGISTIC data = getStarted desc;
model Y = X8 X2 X10;
PROC LOGISTIC data = getStarted desc;
model Y = X8 X2 C_woe;
PROC LOGISTIC data = getStarted desc;
class C; model Y = X8 X2 C;
run;
```

The degrees of freedom for C_woe for entry into the model, again, need to the computed. Again, C_woe is assigned L-1 - k = 6 - 0 = 6 d.f.

| $\alpha$ = 5% and t = -2*Log(L)$_{C\_woe}$ + 2*Log(L)$_{CLASS}$ = 100.577 - 100.294 = 0.283 | | |
|---|---|---|
| k (*) | P(T > t \| k) | Exceeds $\alpha$ ? |
| 0.1 | 0.0745 | Yes |
| 1 | 0.5947 | |
| etc. | etc. | |

(*) If k = 0.1, then re-assign k = 0.

As shown in **Table 8**, C_woe gives the minimum adjusted AIC of 118.577 and this AIC is smaller than the AIC (120.396) at Step 2. C_woe is entered in Step 3.

| Step 3: Predictors in Model: Intercept, X8, X2 | | | | |
|---|---|---|---|---|
| d.f. = | 3 | | | |
| AIC = | 120.396 | | | |
| For entry | -2*Log(L) | Adj. d.f. | Model d.f. | Adj. AIC |
| X10 | 114.113 | 1 | 4 | 122.113 |
| C_woe | 100.577 | 6 | 9 | 118.577 |

**Table 8.**

Now the question is whether X10, the only remaining predictor, will enter the model. The adjusted AIC for X10 must be less than the AIC (118.577) of Step 3. The following PROC LOGISTIC must be run.

```
PROC LOGISTIC data = getStarted desc;
model Y = X8 X2 C_woe X10;
run;
```

In **Table 9** it is noted that C_woe has added 6 d.f. to the Model d.f.

| Step 4: Predictors in Model: Intercept, X8, X2, C_woe | | | | |
|---|---|---|---|---|
| d.f. = | 9 | | | |
| AIC = | 118.577 | | | |
| For entry | -2*Log(L) | Adj. d.f. | Model d.f. | Adj. AIC |
| X10 | 99.175 | 1 | 10 | 119.175 |

**Table 9.**

Adjusted AIC after entry of X10 (119.175) is not less than the model AIC (118.577) from Step 3, and X10 is not entered.

**COMMENTS:**

- In the Example, each of the adjustments of the degrees of freedom for C_woe was to 6 d.f. But in general, the adjustments for WOE coding can go from 1 (no adjustment) to L-1. The choice of α is important in this adjustment process, and α should be decreased for large samples.
- The FSAA process is likely to reduce the entry of WOE predictors. This is simply because the AIC penalty term may be increased.
- It is possible FSAA will add more predictors to the model. Here is the rationale: If a strong WOE is, instead, given 1 d.f., this predictor would be entered early and would bring about an outsized decrease in -2*Log(L) and in its associated AIC (penalized by only 1 d.f.). This might cause an "artificial" minimum model AIC, and additional predictors would be excluded.
- Binning of discrete predictors (nominal or ordinal with few levels) is important to simplify a model. In the case of FSAA, binning will increase the chances of WOE predictors entering the model because predictive power is largely maintained while reduced d.f. makes the adjustment penalty smaller.
- Forward selection was discussed here but the process can be reframed for Backward or Stepwise.

**QUESTIONS REMAINING:**

- Would FSAA models validate better on a validation sample versus the corresponding model fitted by the convention approach of not adjusting d.f. for WOE predictors but using AIC for forward selection?
- In practical applications, would models (with WOE predictors) fit by FSAA be much different than other logistic models which are fit by different selection methods?

**SAS MACRO AVAILABLE**

The author has a SAS macro for the FSAA process, available upon request. See a discussion of this macro in the **Appendix**.

## REFERENCES

Allison, P.D. (2012), *Logistic Regression Using SAS: Theory and Application 2nd Ed.,* SAS Institute Inc.

Finlay, S. (2010). *Credit Scoring, Response Modelling and Insurance Rating*, Palgrave Macmillan.

Hosmer D., Lemeshow S., Sturdivant R. (2013). *Applied Logistic Regression, 3rd Ed.,* John Wiley & Sons.

Lund, B. (2017). SAS® Macros for Binning Predictors with a Binary Target, SAS Global Forum 2017

Prabhakaran, S. (2016). Information Value, http://r-statistics.co/Information-Value-With-R.html.
Also see https://cran.r-project.org/web/packages/InformationValue/index.html.

Siddiqi, N. (2017). *Intelligent Credit Scoring*, 2nd edition, Hoboken, NJ, John Wiley & Sons, Inc.

Thomas, L. (2009), *Consumer Credit Models*, Oxford University Press, Oxford, UK.

## CONTACT INFORMATION

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

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



## APPENDICES

### MLE FOR: MODEL Y= C_woe; and CLASS C; MODEL Y= C;

**For MODEL Y = C_woe**

The likelihood equations for solving for MLE's are shown below.[11]

For the intercept: $\sum_{i=1}^{n}[y_i - P\mathbf{x}_i] = 0$ … or … $\sum_{i=1}^{n} y_i = \sum_{i=1}^{n} P\mathbf{x}_i$

where there are n observations, y is a 0/1 target, **x** is a vector of predictors, and P**x** is the model probability at **x**

For a predictor z (from among the **x**): $\sum_{i=1}^{n} z_i[y_i - P\mathbf{x}_i] = 0$ … or … $\sum_{i=1}^{n} z_i y_i = \sum_{i=1}^{n} z_i P\mathbf{x}_i$

Given that an MLE solution exists for the WOE Model, it is the unique solution to the likelihood functions. The task is to show that α = log(G/B) and β = 1 satisfy the likelihood equations.

First, show that α = log(G/B) and β = 1 solves the intercept equation:

Let $G_j$ equal the number of Goods (y=1) where C = $c_j$ and $B_j$ equal the number of bad's. Let $N_j = G_j + B_j$
Let $G = G_1 + \ldots + G_L$ and $B = B_1 + \ldots + B_L$. The LHS of the likelihood equation simplifies to $G_1 + \ldots + G_L$

Now consider P**x** evaluated by C = $c_j$ with the MLE estimates.

Then $\exp(\alpha + \beta * C\_woe(c_j)) = ( G/B ) * ( G_j/B_j ) / ( G/B ) = G_j/B_j$

For the $j^{th}$ level of C the RHS contributes the term $N_j * ( G_j/B_j ) / (1 + G_j/B_j ) = G_j$

Collecting terms on the RHS gives $G_1 + \ldots + G_L$

Second, by following similar calculations, it is shown that α = log(G/B) and β = 1 solve the likelihood equation for predictor C_woe.

**For CLASS C; MODEL Y = C;**

The arguments follow that patterns given above.

[11] Hosmer, et al. (2013, p. 37)

## WEIGHT OF EVIDENCE MODEL IS "NESTED" IN CLASS MODEL

This is an example of "nesting" for two WOE predictors and two numeric predictors. The WOE predictors do not appear in any interactions. First a data set TEST is created.

```
DATA TEST;
do i = 1 to 500;
   z = rannor(1);
   C1 = (-2 <= z <= 2) * z;
   C1 = floor(C1);
   C2 = floor(5*ranuni(1));
   X1 = ranpoi(1, 2);
   X2 = ranuni(1);
   Y_star = C1 + 0.2*C2 + X1 + 0.1*X2 + 0.3*rannor(1);
   Y = (Y_star > 1);
   output;
   end;
run;
/* compute WOE for C1 and C2 (off line) */
/* code C1_woe and C2_woe */
DATA test2; set test;
if C1 in (  0 ) then C1_woe = 0.7537248551 ;
if C1 in (  1 ) then C1_woe = 3.2496813411 ;
if C1 in ( -1 ) then C1_woe = -0.24808805 ;
if C1 in ( -2 ) then C1_woe = -1.740455431 ;
if C2 in ( 0 ) then C2_woe = -0.568030985 ;
if C2 in ( 1 ) then C2_woe = 0.0135637098 ;
if C2 in ( 2 ) then C2_woe = 0.2221084617 ;
if C2 in ( 3 ) then C2_woe = 0.2792668755 ;
if C2 in ( 4 ) then C2_woe = 0.1431347012 ;
C11D = (C1=0);
C12D = (C1=1);
C13D = (C1=-1);
C14D = (C1=-2);
C21D = (C2=0);
C22D = (C2=1);
C23D = (C2=2);
C24D = (C2=3);
C25D = (C2=4);
run;
/* Find MLE estimators for WOE model */
PROC LOGISTIC data = test2 desc;
model y = C1_woe C2_woe X1 X2 / itprint;
score data = test2 out = woe;
run;
/* Use MLE from WOE to initialize parameters for CLASS model */
DATA initial;
beta1=2.897408;
beta2=2.738215 ;
correction1 = (-1.740455431)*beta1;
correction2 = 0.1431347012*beta2;
correction = (-1.740455431)*beta1 + 0.1431347012*beta2;
C11D =  0.7537248551*beta1 - correction1;
C12D =  3.2496813411*beta1 - correction1;
C13D =  -0.24808805*beta1  - correction1;
C21D =  -0.568030985*beta2 - correction2;
C22D =  0.0135637098*beta2 - correction2;
C23D =  0.2221084617*beta2 - correction2;
```

```
C24D =  0.2792668755*beta2 - correction2;
X1 = 3.7041;
X2 = -0.1797;
intercept= -4.2096 + correction;
output;
run;
/* Initial parameters and no iterations to show CLASS = WOE */
PROC LOGISTIC data = test2 desc INEST= initial;
model y = C11D C12D C13D C21D C22D C23D C24D X1 X2 / maxiter= 0 itprint;
score data = test2 out = class;
run;
```

The reader may show that the logistic probabilities in data sets “woe” and “class” are equal.

## MACRO %FSAA

The macro call for the Example.

```
%FSAA(
dataset = getStarted, /* data set which contains predictors and target */
steps = 4, /* integer >= 1 */
stop = , /* FIRST_MIN | space */
target = Y, /* binary variable for PROC (HP)LOGISTIC */
model_df = 1 , /* 1 if intercept only, else consider &include_var */
include_num = , /* numeric var's in model at start of Forward | space */
include_class = , /* class var's in model at start of Forward | space */
num_var = X2 X8 X10, /* numeric var's as candidates to enter */
class_var = C, /* classification var’s (no suffix “_woe”) num or char */
use_woe = C_woe, /* woe from class_var's by adding suffix "_woe", e.g. if
   C_woe is in &use_woe, then C must be in &class_var */
alpha = .05, /* numeric from open interval (0,1) */
verbose = , /* YES for more detail | space */
HP = , /* HP for HPLOGISTIC | space (gives LOGISTIC) */
RUN_TITLE = /* Title for run, NO commas allowed */
);
 /* Parameters required: dataset, steps, target, model_df, alpha */
 /* Parameters num_var and class_var cannot both be spaces */
```

### Summary Report

| Obs | step | min AIC var | min adj AIC | best model | adj-df for min | new model df | new included var |
|---|---|---|---|---|---|---|---|
| 1 | 1 | X8 | 123.462 | | 1 | 2 | X8 |
| 2 | 2 | X2 | 120.396 | | 1 | 3 | X8 X2 |
| 3 | 3 | C_woe | 118.577 | * | 6 | 9 | X8 X2 C_woe |
| 4 | 4 | X10 | 119.175 | | 1 | 10 | X8 X2 C_woe X10 |

### Predictors not in the model

| Obs | step | num_var | class_var |
|---|---|---|---|
| 1 | 1 | X2 X10 | C |
| 2 | 2 | X10 | C |
| 3 | 3 | X10 | |
| 4 | 4 | | |

Comments:

- User may choose to make `C` a candidate for entry by including `C` in `class_var` and omitting `C_woe` in `use_woe`. Otherwise, including `C` in `class_var` and `C_woe` in `use_woe` makes `C_woe` the candidate for entry.
- `include_class` and `class_var` can include numeric and character variables (used in CLASS)
- `stop`: The value `FIRST_MIN` stops forward selection when the first minimum (perhaps local) in AIC is reached.
- If `model_df = 1` without regard to “includes”, the Forward selection is unchanged but AIC is wrong.

FSAA is computationally intensive. If there are N variables in &num_var and C variables in &class_var, then the worst case number of (HP)LOGISTIC calls (without early out by &stop or &steps) is:

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