Notes on Extending Prior Correction to Multinomial Logit Models

P. Swanson

June 28, 2018

Contents

1 Introduction

When modeling low event rate data with logistic regression, it is common to use stratified sampling on the target variable, Y. This could mean taking all of the cases where $Y = 1$, and a random sample of the cases where $Y = 0$ to build a training dataset. This new dataset would have the wrong marginal distribution of the target, and unadjusted logistic regression models would overpredict $P(Y = 1)$. Fortunately, this is easily remedied by **prior correction** where a correction factor is added to to the linear component of the logistic regression model. This correction factor only needs the marginal (posterior) distribution of the target variable and the "true" population (prior) distribution of the target. In practice, the priors are often estimated from the data before sampling.

This process allows us to "correct", "adjust", "calibrate", or "tune" a logistic regression model built on over-sampled training data to a prior distribution of Y. For example, in credit risk, we can use it to calibrate a logistic regression model to any pre-specified default rate.

These notes extend prior correction to multinomial logistic regression which allows us to apply the method to many commonly used discrete choice models including transition matrices and competing risks survival models.

Additionally, a method is proposed for calculating prior probabilities from an incomplete vector of pre-specified probabilities by making adjustments to the observed marginals in the model build data. For example, we may have models predicting five levels: default, payoff, upgrade, downgrade, no-change, but we may only want to calibrate a model's default rate. The method proposed below allows an analyst to specify prior probabilities for any of the levels, and automatically calculates the others in a consistent way that keeps the sum of the probabilities equal to one. Note, this method assumes the relationship beetween input(s) X and target Y are the same in the training data and the (possibly) hypothetical data used to estimate the prior probabilities.

An example of both methods is included using simulated data.

2 Multinomial Logistic Regression

Let outcome Y be a discrete random variable with $j = 1, \ldots, J$ levels and density $P(Y)$. Let J be the "reference" category. Let X be a vector of any random variables. Then the standard multinomial logistic regression model is given by

$$P(Y = j|X) = \frac{e^{X\beta_j}}{\sum_{k=1}^{J} e^{X\beta_k}} \tag{1}$$

where β_j is a vector of regression coefficients that define a linear combination of X to estimate the relationship between $P(Y = j)$ and $P(Y = J)$. Equation 1 can also be expressed as $J - 1$ binary regression models with J as the reference category [1, 2].

$$\ln\left[\frac{P(Y = j|X)}{P(Y = J|X)}\right] = X\beta_j, \quad j = 1, \ldots, J - 1 \tag{2}$$

which implies that the left-hand side of 2 is equal to the log-odds of the conditional probability, $P(Y = j|P(Y = j) \cup P(Y = J))$. From equation 2,

$$P(Y = j|X) = P(Y = J|X)e^{X\beta_j} \tag{3}$$

and since $\sum_{j=1}^{J} P(Y = j|X) = 1$,

$$\begin{aligned} P(Y = J|X) &= 1 - \sum_{j=1}^{J-1} P(Y = J|X)e^{X\beta_j} \\ 1 &= P(Y = J|X)^{-1} - \sum_{j=1}^{J-1} e^{X\beta_j} \\ P(Y = J|X) &= \frac{1}{1 + \sum_{j=1}^{J-1} e^{X\beta_j}} \end{aligned} \tag{4}$$

Then using 4 in 3

$$P(Y = j|X) = \frac{e^{X\beta_j}}{1 + \sum_{j=1}^{J-1} e^{X\beta_j}} \tag{5}$$

To see the relationship between 5 and 1 (softmax), note

$$e^{X\beta_J} = \log\left[\frac{P(Y = J|X)}{P(Y = J|X)}\right] = 1 \tag{6}$$

3 Prior Correction for Binary Logit Models

Outcome-dependent sampling refers to a sampling method applied to the dependent variable that results in different marginal probabilities between the population, $P(Y = j)$, and the sample, $P(y = j)$. It is also known as oversampling, biased sampling, choice-based sampling, and stratified sampling. When modeling rare-events data with binary logistic regression, it is common to sample all of the observations where $Y = 1$, and a simple random sample of the observations where $Y = 0$. With the appropriate statistical correction, analyses using this selection scheme can be consistent and efficient.

One common method is **prior correction** where we correct the posterior probabilities from the sample used to build a model, $P(y)$, to the prior probabilities, $P(Y)$, (assumed known) from the population. This is done by adding a correction factor to the linear component, $X\beta$ of a logistic regression model. Although its often attributed to others, King and Zeng (2001) provide a derivation for binary logit models [3]. These notes adapt their notation and extend it to the multinomial case.

3.1 General Derivation

Let X, Y be random variables with density $P(X, Y)$ and let x, y be random variables with density $P(x, y)$. Define $P(x, y)$ by subsampling such that $P(X|Y) = P(x|y)$. In the case where Y is discrete, this is equivalent to taking a stratified random sample on Y. Note, this does not imply that $P(Y)$, $P(X)$, or $P(Y|X)$ equal $P(y)$, $P(x)$, or $P(y|x)$ respectively. The goal is to use $P(y|x)$ to make inferences about $P(Y|X)$. We use the definitions of conditional probability and joint probability to find

$$\begin{aligned} P(Y|X) &= \frac{P(XY)}{P(X)} \\ &= \frac{P(X|Y)P(Y)}{P(X)} \\ &= \frac{P(xy)}{P(y)}\frac{P(Y)}{P(X)} \\ &= \frac{P(y|x)P(x)}{P(y)}\frac{P(Y)}{P(X)} \\ &= P(y|x)\left[\frac{P(Y)}{P(y)}\frac{P(x)}{P(X)}\right] \end{aligned} \tag{7}$$

The implication is that we can make inferences about $P(Y|X)$ using $P(y|x)$ and some correction factor given by the bracketed term in 7. King and Zeng (2001) then show the factor is consistent by proving convergence in distribution. Let D and d be random samples of size n from $P(X, Y)$ and $P(x, y)$, respectively. Then as $n \to \infty$,

$$P(Y|X, D) = P(X|Y, D)\frac{P(Y|D)}{P(X|D)} \xrightarrow{d} P(X|Y)\frac{P(Y)}{P(X)} = P(Y|X)$$

Note $P(y|x, d) \xrightarrow{d} P(Y|X)$, however, let $A_y = P(Y|D)/P(y|d)$ be a function of y and $B = P(x|d)/P(X|D) = \left[\sum_y P(y|x, d)A_y\right]^{-1}$ be a constant normalization factor

$$\begin{aligned} P(y|x, d)A_yB &= P(x|y, d)\frac{P(y|d)}{P(x|d)}A_yB \\ &= P(x|y, d)\frac{P(y|d)}{P(x|d)}\frac{P(Y|D)}{P(y|d)}\frac{P(x|d)}{P(X|D)} \\ &= P(x|y, d)\frac{P(Y|D)}{P(X|D)} \\ &\xrightarrow{d} \frac{P(X|Y)P(Y)}{P(X)} \\ &= P(Y|X) \end{aligned}$$

3.2 Finite Discrete Choice Models

Let Y be a discrete random variable, and specify the conditional distribution $P(Y = j|X)$ for $j = 1, \ldots, J$ for finite J. Let $P(Y = j|D) = \tau_j$, which is assumed known either from some knowledge of the true population distribution or from the observed distribution, D. Let $P(y = j|d) = \bar{y}_j$ be known or be estimated from the observed distribution, d. The correction factors are then $A_j = \tau_j/\bar{y}_j$ and $B^{-1} = \sum_{j=1}^{J} P(y = j|x, d)\tau_j/\bar{y}_j$. The sample estimate is

$$P(y = j|x, d)A_jB = \frac{P(y = j|x, d)\tau_j/\bar{y}_j}{\sum_{k=1}^{J} P(y = k|x, d)\tau_k/\bar{y}_k} \xrightarrow{d} P(Y = j|X) \tag{8}$$

3.3 Binary Models

For Binary models, there are only two levels, so for $Y \in \{0, 1\}$ we have $P(Y = 1) = \tau$, and $P(y = 1) = \bar{y}$. This implies

$$\begin{aligned} A_1 &= \tau/\bar{y} \\ A_0 &= (1 - \tau)/(1 - \bar{y}) \end{aligned}$$

and

$$B^{-1} = P(y=1|x,d)\tau/\bar{y} + (1 - P(y=1|x,d)) \left[\frac{1-\tau}{1-\bar{y}}\right]$$

which gives

$$\begin{aligned} P(y=1|x,d)A_1B &= \frac{P(y=1|x,d)\frac{\tau}{\bar{y}}}{P(y=1|x,d)\frac{\tau}{\bar{y}} + (1 - P(y=1|x,d))\frac{1-\tau}{1-\bar{y}}} \\ &= \frac{\frac{\tau}{\bar{y}}}{(\frac{\tau}{\bar{y}}) + ((P(y=1|x,d)^{-1} - 1))\frac{1-\tau}{1-\bar{y}}} \\ &= \left[1 + \left(P(y=1|x,d)^{-1} - 1\right)\left(\frac{1-\tau}{\tau}\right)\left(\frac{\bar{y}}{1-\bar{y}}\right)\right]^{-1} \end{aligned} \tag{9}$$

3.4 Logistic Regression

The logit model is $P(y=1|x,d) = 1/(1+e^{-x_i\beta})$, which we substitute into 9 to find

$$\begin{aligned} P(y=1|x,d)A_1B &= \left[1 + \left(1 + e^{-x_i\beta} - 1\right)\left(\frac{1-\tau}{\tau}\right)\left(\frac{\bar{y}}{1-\bar{y}}\right)\right]^{-1} \\ &= \left[1 + \left(\frac{1-\tau}{\tau}\right)\left(\frac{\bar{y}}{1-\bar{y}}\right)e^{-x_i\beta}\right]^{-1} \\ &= \left[1 + e^{\ln\left[\left(\frac{1-\tau}{\tau}\right)\left(\frac{\bar{y}}{1-\bar{y}}\right)\right]}e^{-x_i\beta}\right]^{-1} \\ &= \frac{1}{1 + e^{-x_i\beta + \ln\left[\left(\frac{1-\tau}{\tau}\right)\left(\frac{\bar{y}}{1-\bar{y}}\right)\right]}} \end{aligned} \tag{10}$$

which shows that for logistic regression, the posterior (modeled) probabilities can be corrected using the prior probabilities by adding the constant correction factor, $\ln\left[\left(\frac{1-\tau}{\tau}\right)\left(\frac{\bar{y}}{1-\bar{y}}\right)\right]$ to the linear component of the model.

4 Extending Prior Correction to Multinomial Logit Models

This section extends the derivation in King and Zeng (2001) to multinomial logistic regression models where $Y \in \{1, \ldots, J\}$ and J is the "reference" category. Notation is consistent with King and Zeng (2001), and the derivation picks off from Equation 8:

$$\begin{aligned} P(y=j|x,d)A_jB &= \frac{P(y=j|x,d)\tau_j/\bar{y}_j}{\sum_{k=1}^{J} P(y=k|x,d)\tau_k/\bar{y}_k} \\ &= \frac{P(y=j|x,d)A_j}{\sum_{k=1}^{J} P(y=k|x,d)A_k} \\ &= \frac{\frac{P(y=j|x,d)A_j}{P(y=J|x,d)A_J}}{\sum_{k=1}^{J}\frac{P(y=k|x,d)A_k}{P(y=J|x,d)A_J}} \end{aligned} \tag{11}$$

Since multinomial logistic regression can be expressed in terms of J binary models, $\ln\left(\frac{P(y=j|x)}{P(y=J|x)}\right) = X\beta_j$ (see Equation 2), we make the substitution

$$\begin{aligned} P(y=j|x,d)A_jB &= \frac{\frac{A_j}{A_J}e^{X\beta_j}}{\sum_{k=1}^{J}\frac{A_k}{A_J}e^{X\beta_k}} \\ &= \frac{e^{X\beta_j + \ln(A_j/A_J)}}{\sum_{k=1}^{J} e^{X\beta_k + \ln(A_k/A_J)}} \end{aligned} \tag{12}$$

In this form, the posterior multinomial probabilities for each $P(y=j|x)$ can be adjusted by applying a correction factor to each of the $j=1$ linear models that comprise the multinomial logit. This correction allows us to make

inferences about $P(Y = j|X)$ using only the model from the sampled data, d. In other words, if we have the prior probabilities, $P(Y = j)$, already defined, and we have the observed marginal distribution from the data we built a model on, $P(y = j)$, we can use these marginals to "correct" the posterior to pre-specified prior probabilities.

Note that $\ln\left(\frac{P(y=J|x)}{P(y=J|x)}\right) = X\beta_j = 0$ which means $\beta_J = 0$

5 Specifying Prior Probabilities

Prior correction was originally developed for "correcting" or "unbiasing" predictions from models built on oversampled data. In this case, $P(y = j)$ can be estimated from the oversampled data, and $P(Y = j)$ can be estimated from the whole data prior to oversampling. Alternatively, we can use some population estimates for $P(Y = j)$.

However, the method can be used to "correct" or "calibrate" our modeled probabilities to any set of pre-specified "prior" probabilities. This is useful for scaling models built on vended data to a specific bank portfolio. It can also be used to adjust a model to a different long-run average probability or to make estimates more conservative as a way of addressing uncertainty.

If the priors for each level of Y are known ahead of time, they can be used directly in 12. If, however, we have less than $J - 1$ levels which are not specified, we need a way to estimate the unspecified $P(Y = j)$. This needs to be done in a way that will preserve $\sum_{j=1}^{J} P(Y = j) = 1$.

This could be done in any number of ways. The method proposed below uses the observed posterior probabilities to determine the relationships between the non-specified priors, since those relationships provide the best available information.

5.1 Method for Specifying Prior Probabilities from an Incomplete Set

Let $Y \sim \text{Multinomial}(n = 1, p)$ with $p = (p_1, \ldots, p_J)$ as the prior probabilities for each j level of Y, and let $y \sim \text{Multinomial}(n = 1, p^*)$ with $p^* = (p_1^*, \ldots p_J^*)$ as the observed posterior probabilities from some observed data used to build a multinomial logit model. All p_j^* are known from the empirical data used to build a model. In some cases, we may only wish to pre-specify some p_j in p.

Call this subset, $S \subset p$. Let S_c be the complement of S, then we have $S \cap S_C = p$, and while each $p_j \in S_c$ is unknown, their sum is known, $\sum_{S_C} p_j = 1 - \sum_S p_j$.

If we have no other information about $p_j \in S_c$, one reasonable alternative is to use the relative proportions in the corresponding p_j^* and to scale each of these relative proportions by $1 - \sum_S p_j$ to ensure $\sum_{j=1}^{J} p_j = 1$. Then we define each $p_j \in S_c$ as

$$p_j = \frac{p_j^*}{\sum_{S_c} p_j^*}\left(1 - \sum_S p_j\right) \tag{13}$$

6 Simulated Example

To demonstrate (1) the proposed method of specifying prior probabilities from an incomplete set and (2) the application of prior correction to a multinomial target, we simulate data from a multinomial logistic regression model. Here we simulate data for $J = 3$ outcome levels.

```
#MNL sim
rm(list=ls(all=T))
library(nnet)

set.seed(42)
n = 10000
npreds = 3

# simulate multinomial logistic regression data for 3 categories
x = matrix(rnorm(n*npreds), c(n,npreds))           # random design matrix
b1 = rnorm(3)/10                                   # true coef beta_1
b2 = rnorm(3)/10                                   # true coef beta_2
p1 = exp(x%*%b1)/(1 + exp(x%*%b1) + exp(x%*%b2))   # softmax P(Y=1)
p2 = exp(x%*%b2)/(1 + exp(x%*%b1) + exp(x%*%b2))   # softmax P(Y=2)
u = runif(n, min=0, max=1)                          # uniform(0,1)
y = ifelse((p1+p2)<u, 0,                           # assign Y using probs
           ifelse(p1<u, 2, 1))
```

6.1 Fit Model

Using the simulated data fit a multinomial model

```
fit = nnet::multinom(y~x)

## # weights:  15 (8 variable)
## initial  value 10986.122887
## iter  10 value 10935.307786
## final  value 10934.942283
## converged
```

Next we can verify our model fit by comparing our estimated coefficients with our "true" coefficients. The observed fit does not perfectly match the true parameters, but they fall well within the standard errors.

```
b1          # true coefs for beta_1

## [1]  0.0680923388  0.1013496371 -0.0009105223

b2          # true coefs for beta_2

## [1]  0.05600534 -0.14785411  0.09520968

summary(fit)

## Call:
## nnet::multinom(formula = y ~ x)
##
## Coefficients:
##   (Intercept)         x1          x2          x3
## 1 0.005094711 0.05135889  0.07837396 0.007651777
## 2 0.035196728 0.04738892 -0.12971100 0.096413632
##
## Std. Errors:
##   (Intercept)         x1         x2         x3
## 1  0.02473278 0.02455379 0.02459274 0.02435932
## 2  0.02456975 0.02436168 0.02443982 0.02420750
##
## Residual Deviance: 21869.88
## AIC: 21885.88
```

6.2 Posterior Probabilities

We next calculate the quantities needed for the correction factors. The posterior probabilities are given by empirical distribution of the target.

```
## Sample Marginal Probs (Posterior Probabilities)
post_p = prop.table(table(y))
s1 = post_p['1']  # posterior probs P(y=j)
s2 = post_p['2']
sJ = 1 - s1 - s2
```

6.3 Prior Probabilities

To demonstrate the method from Section 5, we specify a prior probability only for $P(Y = 1)$, leaving the remaining levels, $P(Y = 2)$ and $P(Y = J)$ to be determined by Equation 13. In our case, we only need to specify one level, $p_2 = P(Y = 2)$, and after that the remaining level is given.

$$p_2 = \frac{p_2^*}{p_2^* + p_J^*} \times (1 - p_1)$$

```
p1 = .15
p2 = (s2/(s2+sJ))*(1-p1)    # use posteriors to inform unspecified priors
pJ = 1 - p1 - p2
prior_p = c(p1, p2, pJ)
```

6.4 Correction Factors

Once we have specified our prior and posterior probabilities, the correction factors for each binomial component of the multinomial logit are given by Equation 12.

```
cf1 = log((p1/pJ)*(sJ/s1))
cf2 = log((p2/pJ)*(sJ/s2))
```

6.5 Results

Next we calculate the average unadjusted modeled probabilities, then apply the adjustment from Equation 12, and compare the model output results to the prior and posterior probabilities specified earlier.

```
# data & estimated coefs
X = cbind(1, x)
b1 = coef(fit)[1,]  # fitted coefs from b1  ln(P(y=2)/P(y=J)) = xb1
b2 = coef(fit)[2,]  # fitted coefs from b2  ln(P(y=2)/P(y=J)) = xb2

p_y1 = exp(X%*%b1) / (1 + exp(X%*%b1) + exp(X%*%b2))
p_y2 = exp(X%*%b2) / (1 + exp(X%*%b1) + exp(X%*%b2))
p_yJ = 1 - p_y1 - p_y2
p_y1_adj = exp(X%*%b1 + cf1) / (1 + exp(X%*%b1 + cf1) + exp(X%*%b2 + cf2))
p_y2_adj = exp(X%*%b2 + cf2) / (1 + exp(X%*%b1 + cf1) + exp(X%*%b2 + cf2))
p_yJ_adj = 1 - p_y1_adj - p_y2_adj

result = data.frame(p_y1, p_y2, p_yJ, p_y1_adj, p_y2_adj, p_yJ_adj)
```

The unadjusted average modeled probabilities are equal to the posterior probabilities given by the empirical distribution of the target, Y. Note that here 0 is the reference category, $Y = j$.

```
# review results
round(colMeans(result)[1:3], 5)   # mean probs

##   p_y1   p_y2   p_yJ
## 0.3305 0.3418 0.3277

round(post_p, 5)                  # posterior probs (from empircal Y)

## y
##      0      1      2
## 0.3277 0.3305 0.3418
```

More importantly, we find the adjusted average modeled probabilities after prior correction are very close to the the priors we specified above.

```
round(colMeans(result)[4:6], 5)   # adj mean probs

## p_y1_adj p_y2_adj p_yJ_adj
##  0.15054  0.43286  0.41660

round(prior_p, 5)                 # prior probs

##               2       2
## 0.15000 0.43395 0.41605
```

7 References

References

[1] A Agresti. *Categorical Data Analysis*. 3rd ed. Wiley, 2013.

[2] David W Hosmer Jr, Stanley Lemeshow, and Rodney X Sturdivant. *Applied logistic regression*. Vol. 398. John Wiley & Sons, 2013.

[3] Gary King and Langche Zeng. "Logistic regression in rare events data". In: *Political analysis* 9.2 (2001), pp. 137–163.