How does Mplus calculate the standardized coefficients based on a logistic regression?

Authored by
stats writer

July 1, 2024

RECOMMENDED CITATION

stats writer (2024). *How does Mplus calculate the standardized coefficients based on a logistic regression?*. PSYCHOLOGICAL SCALES. Retrieved from https://scales.arabpsychology.com/?p=165027

Mplus is a statistical modeling software that utilizes logistic regression to analyze the relationship between a set of independent variables and a binary outcome variable. In order to assess the strength and direction of this relationship, Mplus calculates standardized coefficients that represent the effect size of each independent variable on the outcome variable.

To calculate the standardized coefficients, Mplus first standardizes all variables by subtracting the mean and dividing by the standard deviation. This ensures that all variables are on the same scale and allows for direct comparison of their effects.

Next, Mplus uses an iterative process to estimate the coefficients that maximize the likelihood of the observed data. This is known as maximum likelihood estimation. The resulting coefficients are then transformed into standardized coefficients by dividing them by the standard deviation of the corresponding independent variable.

The standardized coefficients produced by Mplus provide a standardized measure of the effect size of each independent variable on the outcome variable, making it easier to compare the relative importance of different variables in predicting the outcome. This allows for a more meaningful interpretation of the logistic regression results.

How does Mplus calculate the standardized coefficients based on a logistic regression? | Mplus FAQ

The following example shows the output in Mplus, as well as how to reproduce it using Stata. For this example we will use the same dataset we used for our logit regression data analysis example. You can download the dataset for Mplus here: logit.dat. The model we specify for this example includes four variables, three predictors and one outcome. We use Graduate Record Exam scores (gre), undergraduate

grade point average (gpa),
and prestige of the undergraduate program (topnotch) to predict that whether an
applicant is admitted to graduate school. The Mplus input for this
model is:

data: file is logit.dat;

variable: names are admit gre topnotch gpa;
categorical = admit;

analysis:
type = general;
estimator = ml;
! need to use estimator = ml to make this a logistic model;

model: admit on gre topnotch gpa;

output: stand;

Below are the results from the model described above. Note that Mplus produces
two types of standardized coefficients "Std" which are

in the fifth column of
the output shown below,
and "StdXY" which are in the sixth column. The Std column contains coefficients standardized using the variance of continuous latent variables.
Because all of the variables in this model are manifest (i.e. observed) the
coefficients in this column are identical to those in the column of regular
coefficients (i.e. the "Estimates" column). The StdXY column contains the
coefficients standardized using the variance of the background and/or outcome
variables, in addition to the variance of continuous latent variables.

MODEL RESULTS

Estimates S.E. Est./S.E. Std StdYX

ADMIT ON
GRE 0.002 0.001 2.314 0.002 0.152
TOPNOTCH 0.437 0.292 1.498 0.437 0.086
GPA 0.668 0.325 2.052 0.668 0.135

Thresholds
ADMIT$1 4.601 1.096 4.196 4.601 2.439

Now, from the latent variable point of view, there is a latent variable behind the observed dichotomous variable and this latent variable is the true outcome variable. In other word, the logistic regression is simply modeling the latent variable using the linear relationship:

$$
y^{*} = \beta_0 + \beta_1 * GRE + \beta_2 * TOPNOTCH + \beta_3 * GPA
$$

Notice that there is no random residual term here. Instead, we assume that

$$
y^{*} - (\beta_0 + \beta_1 * GRE + \beta_2 * TOPNOTCH + \beta_3 * GPA)
$$

obeys the standard logistic distribution. Therefore, the variance of (y^{*}) is the sum of variance of the linear prediction plus the variance of standard logistic distribution, which is (frac{pi^2}{3}), that is (Var(y^{*}) = Var(Xbeta) +frac{pi^2}{3}). This is the formula that Mplus uses to calculate the variance for the outcome variable.

Now we are ready to replicate the results from Mplus in Stata. The first bold line below opens the dataset, and the second runs the logistic regression model in Stata. Note that the raw coefficients from Stata and Mplus are within rounding error of each other, this should be the case, since we are running the same model. We have also run fitstat to display many fit indices including the variance for (y^{*}).

use https://stats.idre.ucla.edu/stat/stata/dae/logit.dta, clear
logit admit gre topnotch gpa, nolog

Logistic regression Number of obs = 400
LR chi2(3) = 21.85
Prob > chi2 = 0.0001
Log likelihood = -239.06481 Pseudo R2 = 0.0437

--

admit | Coef. Std. Err. z P>|z|

-------------+--

gre | .0024768 .0010702 2.31 0.021 .0003792 .0045744
topnotch | .4372236 .2918532 1.50 0.134 -.1347983 1.009245
gpa | .6675556 .3252593 2.05 0.040 .0300592 1.305052
_cons | -4.600814 1.096379 -4.20 0.000 -6.749678 -2.451949

--

fitstat

Measures of Fit for logit of admit

Log-Lik Intercept Only: -249.988 Log-Lik Full Model: -239.065
D(396): 478.130 LR(3): 21.847
Prob > LR: 0.000

McFadden's R2: 0.044 McFadden's Adj R2: 0.028
ML (Cox-Snell) R2: 0.053 Cragg-Uhler(Nagelkerke) R2: 0.074
McKelvey & Zavoina's R2: 0.075 Efron's R2: 0.052
Variance of y*: 3.558 Variance of error: 3.290
Count R2: 0.683 Adj Count R2: 0.000
AIC: 1.215 AIC*n: 486.130
BIC: -1894.490 BIC': -3.873
BIC used by Stata: 502.095 AIC used by Stata: 486.130

How does fitstat compute the variance of (y^{*})? We have explained earlier
that (Var(y^{*}) = Var(Xbeta) +frac{pi^2}{3}) and now let's check if this is the case.

predict xb, xb
sum xb

Variable | Obs Mean Std. Dev. Min Max
-------------+---
xb | 400 -.8111861 .5180669 -2.166729 .4880949

return list

scalars:

r(N) = 400

r(sum_w) = 400

r(mean) = -.8111860970774433

r(Var) = .2683933174379701

r(sd) = .5180669044032538

r(min) = -2.166728973388672

r(max) = .4880948960781097

r(sum) = -324.4744388309773

display r(Var) + (_pi^2)/3

3.5582615

As you can see, they match very nicely. Now we are ready to calculate a standardized coefficient.
This is also called "full-standardization" since it requires both the
outcome variable and the predictor variable to be standardized. As always, we will need three pieces of information, the standard deviation of (y^{*}), the standard
deviation of the predictor variable for which we want to create a standardized
coefficient, and the raw coefficient for that predictor

variable.

To
obtain the standard deviation for the linear predictor, we will create a local
macro variable based on what have calculated above, this is the first line
of code below. Next we
summarize the predictor variable for which we want to create a standardized coefficient,
in this case gre, and save the standard deviation to a local macro
variable called "xstd." Since Stata
automatically stores the coefficients from the last regression we ran, we can
access the coefficient for gre by typing _b. Now we are ready to actually calculate the standardized coefficients. The second to
last command below creates a new local macro called "gre_std" and sets it equal
to the standardized coefficient for gre (i.e. _b*`xstd'/`ystd').
The last command shown below tells Stata to display the contents of "gre_std"

which is the standardized coefficient for the relationship between gre
and the log odds of y. This value is approximately 0.1516, looking at the Mplus output above, we see that the standardized
coefficient (StdYX) for male is also estimated to be 0.152 by Mplus.

```
local ystd=sqrt(r(Var)+(_pi^2)/3)
sum gre

Variable | Obs Mean Std. Dev. Min Max
-------------+--------------------------------------------------------
gre | 400 587.7 115.5165 220 800

local xstd = r(sd)
local gre_std = _b*`xstd'/`ystd'
display "`gre_std'"
.1516774659729085
```

The commands and output below show the same process for the other two predictor variables
in the model.

```
sum topnotch
```

```
Variable | Obs Mean Std. Dev. Min Max
-------------+---------------------------------------------------------
topnotch | 400 .1625 .3693709 0 1

local xstd = r(sd)
local topnotch_std = _b*`xstd'/`ystd'
display "`topnotch_std'"
.0856144885799177

sum gpa

Variable | Obs Mean Std. Dev. Min Max
-------------+---------------------------------------------------------
gpa | 400 3.3899 .3805668 2.26 4

local xstd = r(sd)
local gpa_std = _b*`xstd'/`ystd'
display "`gpa_std'"
.1346788501438455
```

Cautions, Flies in the Ointment

See Also