Multinomial logistic regression is used to model nominal outcome variables, in which the log odds of the outcomes are modeled as a linear combination of the predictor variables.

This page uses the following packages. Make sure that you can load them before trying to run the examples on this page. If you do not have a package installed, run: **install.packages("packagename")**, or if you see the version is out of date, run: **update.packages()**.

```
require(foreign)
require(nnet)
require(ggplot2)
require(reshape2)
```

Please note: The purpose of this page is to show how to use various data analysis commands. It does not cover all aspects of the research process which researchers are expected to do. In particular, it does not cover data cleaning and checking, verification of assumptions, model diagnostics or potential follow-up analyses.

Examples of multinomial logistic regression

Example 1. People's occupational choices might be influenced by their parents' occupations and their own education level. We can study the relationship of one's occupation choice with education level and father's occupation. The occupational choices will be the outcome variable which consists of categories of occupations.

Example 2. A biologist may be interested in food choices that alligators make. Adult alligators might have different preferences from young ones. The outcome variable here will be the types of food, and the predictor variables might be size of the alligators and other environmental variables.

Example 3. Entering high school students make program choices among general program, vocational program and academic program. Their choice might be modeled using their writing score and their social economic status.

Description of the data

For our data analysis example, we will expand the third example using the **hsbdemo** data set. Let's first read in the data.

```
ml <- read.dta("https://stats.idre.ucla.edu/stat/data/hsbdemo.dta")
```

The data set contains variables on 200 students. The outcome variable is **prog**, program type. The predictor variables are social economic status, **ses**, a three-level categorical variable and writing score, **write**, a continuous variable. Let's start with getting some descriptive statistics of the variables of interest.

```
with(ml, table(ses, prog))
```

```
##          prog
## ses       general academic vocation
##    low          16       19       12
##    middle       20       44       31
##    high          9       42        7
```

```
with(ml, do.call(rbind, tapply(write, prog, function(x) c(M = mean(x), SD = sd(x)))))
```

```
##               M     SD
## general   51.33 9.398
## academic  56.26 7.943
## vocation  46.76 9.319
```

Analysis methods you might consider

- Multinomial logistic regression, the focus of this page.
- Multinomial probit regression, similar to multinomial logistic regression with independent normal error terms.
- Multiple-group discriminant function analysis. A multivariate method for multinomial outcome variables
- Multiple logistic regression analyses, one for each pair of outcomes: One problem with this approach is that each analysis is potentially run on a different sample. The other problem is that without constraining the logistic models, we can end up with the probability of choosing all possible outcome categories greater than 1.
- Collapsing number of categories to two and then doing a logistic regression: This approach suffers from loss of information and changes the original research questions to very different ones.
- Ordinal logistic regression: If the outcome variable is truly ordered and if it also satisfies the assumption of proportional odds, then switching to ordinal logistic regression will make the model more parsimonious.
- Alternative-specific multinomial probit regression, which allows different error structures therefore allows to relax the IIA assumption. This requires that the data structure be choice-specific.
- Nested logit model, another way to relax the IIA assumption, also requires the data structure be choice-specific.

Multinomial logistic regression

Below we use the **multinom** function from the **nnet** package to estimate a multinomial logistic regression model. There are other functions in other R packages capable of multinomial regression. We chose the **multinom** function because it does not require the data to be reshaped (as the **mlogit** package does) and to mirror the

example code found in Hilbe's *Logistic Regression Models*.

First, we need to choose the level of our outcome that we wish to use as our baseline and specify this in the `relevel` function. Then, we run our model using `multinom`. The `multinom` package does not include p-value calculation for the regression coefficients, so we calculate p-values using Wald tests (here z-tests).

```
ml$prog2 <- relevel(ml$prog, ref = "academic")
test <- multinom(prog2 ~ ses + write, data = ml)
```

```
## # weights:  15 (8 variable)
## initial  value 219.722458
## iter  10 value 179.982880
## final  value 179.981726
## converged
```

```
summary(test)
```

```
## Call:
## multinom(formula = prog2 ~ ses + write, data = ml)
##
## Coefficients:
##           (Intercept) sesmiddle seshigh    write
## general         2.852   -0.5333 -1.1628 -0.05793
## vocation        5.218    0.2914 -0.9827 -0.11360
##
## Std. Errors:
##           (Intercept) sesmiddle seshigh    write
## general         1.166    0.4437  0.5142 0.02141
## vocation        1.164    0.4764  0.5956 0.02222
##
## Residual Deviance: 360
## AIC: 376
```

```
z <- summary(test)$coefficients/summary(test)$standard.errors
z
```

```
##           (Intercept) sesmiddle seshigh  write
## general         2.445   -1.2018  -2.261 -2.706
## vocation        4.485    0.6117  -1.650 -5.113
```

```
# 2-tailed z test
p <- (1 - pnorm(abs(z), 0, 1)) * 2
p
```

```
##            (Intercept) sesmiddle seshigh    write
## general     1.448e-02    0.2294 0.02374 6.819e-03
## vocation    7.299e-06    0.5408 0.09895 3.176e-07
```

- We first see that some output is generated by running the model, even though we are assigning the model to a new R object. This model-running output includes some iteration history and includes the final negative log-likelihood 179.981726. This value multiplied by two is then seen in the model summary as the <mark>Residual Devianc</mark>e and it can be used in comparisons of nested models, but we won't show an example of comparing models on this page.

- The model summary output has a block of coefficients and a block of standard errors. Each of these blocks has one row of values corresponding to a model equation. Focusing on the block of coefficients, we can look at the first row comparing **prog** = **"general"** to our baseline **prog** = **"academic"** and the second row comparing **prog** = **"vocation"** to our baseline **prog** = **"academic"**. If we consider our coefficients from the first row to be b_1 and our coefficients from the second row to be b_2, we can write our model equations:

$$ln\left(\frac{P(prog = general)}{P(prog = academic)}\right) = b_{10} + b_{11}(ses = 2) + b_{12}(ses = 3) + b_{13}write$$

$$ln\left(\frac{P(prog = vocation)}{P(prog = academic)}\right) = b_{20} + b_{21}(ses = 2) + b_{22}(ses = 3) + b_{23}write$$

- b_{13} A one-unit increase in the variable **write** is associated with the decrease in the log odds of being in general program vs. academic program in the amount of .058 .

- b_{23} A one-unit increase in the variable **write** is associated with the decrease in the log odds of being in vocation program vs. academic program. in the amount of .1136 .

- b_{12} The log odds of being in general program vs. in academic program will decrease by 1.163 if moving from **ses="low"** to **ses="high"**.

- b_{11} The log odds of being in general program vs. in academic program will decrease by 0.533 if moving from **ses="low"**to **ses="middle"**, although this coefficient is not significant.

- b_{22} The log odds of being in vocation program vs. in academic program will decrease by 0.983 if moving from **ses="low"** to **ses="high"**.

- b_{21} The log odds of being in vocation program vs. in academic program will increase by 0.291 if moving from **ses="low"** to **ses="middle"**, although this coefficient is not significant.

The ratio of the probability of choosing one outcome category over the probability of choosing the baseline

The ratio of the probability of choosing one outcome category over the probability of choosing the baseline category is often referred as relative risk (and it is sometimes referred to as *odds*, described in the regression parameters above). The relative risk is the right-hand side linear equation exponentiated, leading to the fact that the exponentiated regression coefficients are relative risk ratios for a unit change in the predictor variable. We can exponentiate the coefficients from our model to see these risk ratios.

```
## extract the coefficients from the model and exponentiate
exp(coef(test))
```

```
##          (Intercept) sesmiddle seshigh  write
## general        17.33    0.5867  0.3126 0.9437
## vocation      184.61    1.3383  0.3743 0.8926
```

- The relative risk ratio for a one-unit increase in the variable **write** is .9437 for being in general program vs. academic program.
- The relative risk ratio switching from **ses** = 1 to 3 is .3126 for being in general program vs. academic program.

You can also use predicted probabilities to help you understand the model. You can calculate predicted probabilities for each of our outcome levels using the **fitted** function. We can start by generating the predicted probabilities for the observations in our dataset and viewing the first few rows

```
head(pp <- fitted(test))
```

```
##    academic general vocation
## 1    0.1483  0.3382   0.5135
## 2    0.1202  0.1806   0.6992
## 3    0.4187  0.2368   0.3445
## 4    0.1727  0.3508   0.4765
## 5    0.1001  0.1689   0.7309
## 6    0.3534  0.2378   0.4088
```

Next, if we want to examine the changes in predicted probability associated with one of our two variables, we can create small datasets varying one variable while holding the other constant. We will first do this holding **write** at its mean and examining the predicted probabilities for each level of **ses**.

```
dses <- data.frame(ses = c("low", "middle", "high"), write = mean(ml$write))
predict(test, newdata = dses, "probs")
```

```
##    academic general vocation
## 1    0.4397  0.3582   0.2021
## 2    0.4777  0.2283   0.2939
## 3    0.7009  0.1785   0.1206
```

Another way to understand the model using the predicted probabilities is to look at the averaged predicted probabilities for different values of the continuous predictor variable **write** within each level of **ses**.

```
dwrite <- data.frame(ses = rep(c("low", "middle", "high"), each = 41), write = rep(c(30:70),
    3))

## store the predicted probabilities for each value of ses and write
pp.write <- cbind(dwrite, predict(test, newdata = dwrite, type = "probs", se = TRUE))

## calculate the mean probabilities within each level of ses
by(pp.write[, 3:5], pp.write$ses, colMeans)
```

```
## pp.write$ses: high
## academic  general vocation
##    0.6164   0.1808   0.2028
## ----------------------------------------------------------
## pp.write$ses: low
## academic  general vocation
##    0.3973   0.3278   0.2749
## ----------------------------------------------------------
## pp.write$ses: middle
## academic  general vocation
##    0.4256   0.2011   0.3733
```

Sometimes, a couple of plots can convey a good deal amount of information. Using the predictions we generated for the pp.write object above, we can plot the predicted probabilities against the writing score by the level of **ses** for different levels of the outcome variable.

```
## melt data set to long for ggplot2
lpp <- melt(pp.write, id.vars = c("ses", "write"), value.name = "probability")
head(lpp)   # view first few rows
```

```
##     ses write variable probability
## 1 low    30 academic      0.09844
## 2 low    31 academic      0.10717
## 3 low    32 academic      0.11650
```

```
## 4 low     33 academic       0.12646
## 5 low     34 academic       0.13705
## 6 low     35 academic       0.14828
```

```
## plot predicted probabilities across write values for each level of ses
## facetted by program type
ggplot(lpp, aes(x = write, y = probability, colour = ses)) + geom_line() + facet_grid(variable ~
    ., scales = "free")
```
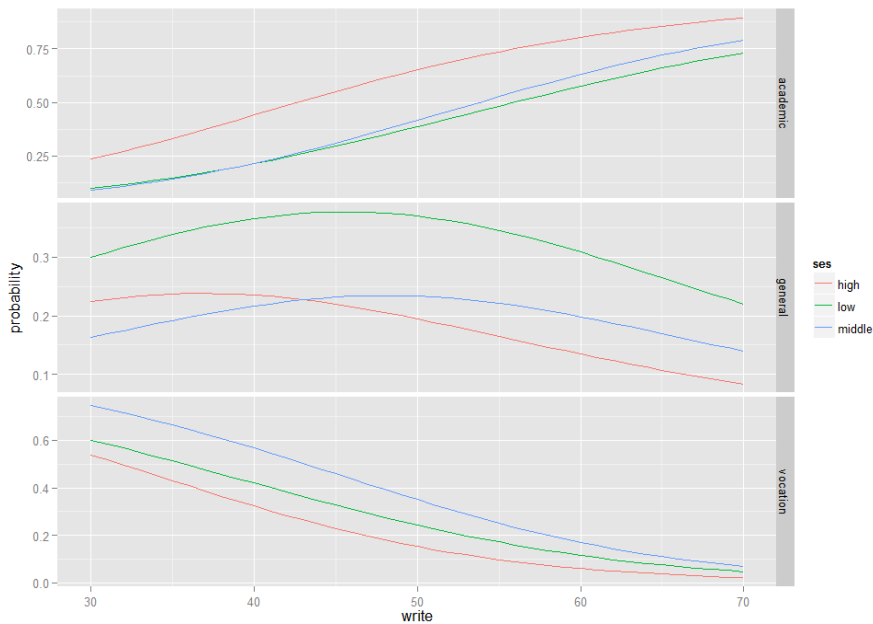

Things to consider

- The Independence of Irrelevant Alternatives (IIA) assumption: Roughly, the IIA assumption means that adding or deleting alternative outcome categories does not affect the odds among the remaining outcomes. There are alternative modeling methods, such as alternative-specific multinomial probit model, or nested logit model to relax the IIA assumption.

- Diagnostics and model fit: Unlike logistic regression where there are many statistics for performing model diagnostics, it is not as straightforward to do diagnostics with multinomial logistic regression models. For the purpose of detecting outliers or influential data points, one can run separate logit models and use the diagnostics tools on each model.

- Sample size: Multinomial regression uses a maximum likelihood estimation method, it requires a large sample size. It also uses multiple equations. This implies that it requires an even larger sample size than ordinal or binary logistic regression.

- Complete or quasi-complete separation: Complete separation means that the outcome variable separate a predictor variable completely, leading perfect prediction by the predictor variable.

- Perfect prediction means that only one value of a predictor variable is associated with only one value of the response variable. But you can tell from the output of the regression coefficients that something is wrong. You can then do a two-way tabulation of the outcome variable with the problematic variable to confirm this and then rerun the model without the problematic variable.
- Empty cells or small cells: You should check for empty or small cells by doing a cross-tabulation between categorical predictors and the outcome variable. If a cell has very few cases (a small cell), the model may become unstable or it might not even run at all.

See also

- Applied Logistic Regression (Second Edition) (/examples/alr2/) by David Hosmer and Stanley Lemeshow
- An Introduction to Categorical Data Analysis (/examples/icda/) by Alan Agresti
- Logistic Regression Models by Joseph M. Hilbe

Click here to report an error on this page or leave a comment

How to cite this page (https://stats.idre.ucla.edu/other/mult-pkg/faq/general/faq-how-do-i-cite-web-pages-and-programs-from-the-ucla-statistical-consulting-group/)

HOME (/)   CONTACT (/contact)