Journal Title: Categorical data analysis and multilevel modeling using R , Xing Liu

Article Title: Chapter 7. Multinomial
Article Author: Xing Liu

Volume/Issue: /
Date: 2023
Pages: 271-
NRC TN:1345677

RAPID

12/12/2023 8:29:44 AM

Borrower: RAPID:PUL
Library Name: RAPID:PUL
Email:
Odyssey: 128.112.202.139

ILL Number: -21738746

Categorical Data Analysis and Multilevel Modeling Using R

Xing Liu

Eastern Connecticut State University

SAGE

Los Angeles | London | New Delhi
Singapore | Washington DC | Melbourne

7

MULTINOMIAL LOGISTIC REGRESSION MODELS

OBJECTIVES OF THIS CHAPTER

This chapter introduces multinomial logistic regression models. It first starts with an introduction to the multinomial logistic regression model followed by a discussion of the odds and odds ratios or relative risk ratios in the model, goodness-of-fit statistics, and how to interpret parameter estimates. After a description of the research example, the data, and the sample, the multinomial logistic regression models are illustrated with the `VGAM`, `nnet`, and `mlogit` packages. R commands and output are explained in detail. This chapter focuses on fitting the multinomial logistic regression models with R, as well as on interpreting and presenting the results. After reading this chapter, you should be able to:

- Identify when multinomial logistic regression models are used.
- Fit a multinomial logistic regression model using R.
- Interpret the output.
- Compute, plot and interpret the predicted probabilities.
- Compare models using the likelihood ratio test.
- Present results in publication-quality tables using R.
- Write the results for publication.

7.1 MULTINOMIAL LOGISTIC REGRESSION MODELS: AN INTRODUCTION

The multinomial logistic regression model is used to estimate nominal response variables that have multiple unordered categories. For example, the nominal response variable in Hoffmann (2016) and Kaufman (2019) was the three-category political views including liberal, moderate, and conservative. Another example in Menard (2010) was the four-category political party affiliation including Democrat, Independent, Republican, and other. This model is a generalization of binary logistic regression when there are more than two nominal categories in a response variable. It can also be used for an ordinal response variable when the proportional odds assumption does not hold. It estimates the odds of being in a category versus the base category of a nominal variable. Although the proportional odds model compares the cumulative probabilities of being at or below a particular category and the probabilities of being above that category, the multinomial logistic regression model compares a particular category with the base category. If a nominal response variable has J levels, there are $J - 1$ comparisons between any other categories and the base category. For example, if we disregard the ordinal nature of the ordinal response variable, health status, and treat it as a nominal response variable with four categories, then we compare category 2 and category 1, category 3 and category 1, and category 4 and category 1 in the multinomial logistic model where the base category is set to be one.

The multinomial logistic model can be expressed as follows:

$$\ln\left(\frac{P(Y = j|x_1, x_2, \ldots, x_p)}{P(Y = J|x_1, x_2, \ldots, x_p)}\right) = \alpha_j + \beta_{j1}X_1 + \beta_{j2}X_2 + \ldots + \beta_{jp}X_p \tag{7.1}$$

where $j = 1, 2, \ldots, J - 1$; J is the base category, which can be any category but is generally the highest one; α_j are the intercepts; and $\beta_{j1}, \beta_{j2}, \ldots, \beta_{jp}$ are the logit coefficients for each comparison. The model estimates $J - 1$ logit coefficients for each predictor. If we set the base category to be category 1, the lowest category, then the model can be rewritten as follows:

$$\ln\left(\frac{P(Y = j|x_1, x_2, \ldots, x_p)}{P(Y = 1|x_1, x_2, \ldots, x_p)}\right) = \alpha_j + \beta_{j1}X_1 + \beta_{j2}X_2 + \ldots + \beta_{jp}X_p \tag{7.2}$$

where $j = 2, 3, \ldots, J$; and category 1 is the base category.

It can be treated as a simultaneous estimation of a series of binary logistic regression models comparing a particular category and the base category. In each binary model, being in a particular category is coded as the binary outcome of 1 and being in the base category is coded as 0. For example, when the base category is 1, the dichotomized outcome in the first binary model compares category 2 with category 1.

7.1.1 The Multinomial Distribution

The multinomial distribution is an extension of the binomial distribution when the discrete random variable is a nominal variable with more than two categories. Recall that the binomial distribution is expressed.

$$P(Y = k) = \binom{n}{k} p^k (1-p)^{n-k} \tag{7.3}$$

where $\binom{n}{k}$ is the binomial coefficient, k is the number of successes, n is the number of trials, and p is the success probability when the binary outcome is 1.

$$\binom{n}{k} = \frac{n!}{(n-k)!k!}$$

where $n!$ is n factorial or the factorial of n. $n! = n^*(n-1) \ldots 2^*1$.

When the outcome of a nominal variable has more than two categories, the probability function or the PDF is expressed as:

$$P(n_1, n_2, \ldots n_j) = \frac{n!}{n_1!n_2!\ldots n_j!} p_1^{n_1} p_2^{n_2} \ldots p_j^{n_j} \tag{7.4}$$

where j is the number of categories in the nominal variable, n_j is the number of observations for a particular category, and p_j is the probability of choosing each category. The total number of observations across all the categories, $n = n_1 + n_2 + \ldots + n_j$. In addition, the total probability across all the categories is 1. $p_1 + p_2 + \ldots + p_j = 1$. When the nominal variable has only two categories (i.e., $j = 2$), the multinomial distribution becomes the binomial distribution.

The log likelihood function for the multinomial distribution is expressed as:

$$l(p, n) = \sum_{i=1}^{j} n_i \ln p_i + \ln \frac{n!}{n_1!n_2!\ldots n_j!} \tag{7.5}$$

where $\ln \frac{n!}{n_1!n_2!\ldots n_j!}$ is a constant term and does not involve the parameter p. $\sum_{i=1}^{j} n_i \ln p_i$ is the summation term which adds the product of the number of observations for a particular category (n_i) and the log of the probability of choosing that category ($\ln p_i$). For example, if the nominal outcome has four categories, $\sum_{i=1}^{j} n_i \ln p_i = n_1 \ln p_1 + n_2 \ln p_2 + n_3 \ln p_3 + n_4 \ln p_4$.

7.1.2 Odds in Multinomial Logistic Models

The multinomial logistic model estimates the logit or log odds of being in a particular category relative to the baseline category. The odds in the multinomial logistic model can be defined as the ratio of the probability of being in a particular category to the probability of being in the base category. It is expressed as:

$$\text{Odds}(Y = j \text{ vs. } Y = J) = \frac{P(Y = j)}{P(Y = J)} \tag{7.6}$$

where j can be any categories from 1 to $J - 1$ categories.

For example, if we treat the ordinal response variable, health status, as nominal with four categories from 1 to 4, with 1 = poor, 2 = fair, 3 = good, and 4 = excellent, then we estimate three odds with category 1 as the base category: The odds of being in category 2 versus category 1, the odds of being in category 3 versus category 1, and the odds of being in category 4 versus category 1.

Specifically, odds ($Y = 2$ vs. $Y = 1$) equal the ratio of the probability of being in category 2 to the probability of being in category 1:

$$\text{Odds}(Y = 2 \text{ vs. } 1) = \frac{P(Y = 2)}{P(Y = 1)} = \frac{P(2)}{P(1)}$$

The other two odds, odds ($Y = 3$ vs. $Y = 1$) and odds ($Y = 4$ vs. $Y = 1$), are expressed as follows:

$$\text{Odds}(Y = 3 \text{ vs. } 1) = \frac{P(Y = 3)}{P(Y = 1)} = \frac{P(3)}{P(1)}$$

$$\text{Odds}(Y = 4 \text{ vs. } 1) = \frac{P(Y = 4)}{P(Y = 1)} = \frac{P(4)}{P(1)}$$

Table 7.1 presents the logits, odds, and category comparisons for the multinomial logistic regression model for the nominal response variable with four levels.

7.1.3 Odds Ratios or Relative Risk Ratios in Multinomial Logistic Regression Models

Since the multinomial logistic model can be treated as a series of binary logistic regression models estimated simultaneously with the comparison of any other categories to the base category, the logit coefficients can be interpreted in a similar way as that for the binary logistic regression. The odds ratio of being in a category j versus the baseline category J is obtained by taking the exponential of the logit coefficient β. Although the relative risk can be defined differently, the odds ratio in the multinomial logistic

TABLE 7.1 ⬢ Category Comparisons for the Multinomial Logistic Regression Model With Four Levels of Health Status (j = 1, 2, 3, 4)

Equation	Logit $P(Y = j \text{ vs. } J)$	Odds	Probability Comparisons
1	logit $P(Y = 2 \text{ vs. } 1)$	$\frac{P(Y=2)}{P(Y=1)}$	Category 2 vs. category 1
2	logit $P(Y = 3 \text{ vs. } 1)$	$\frac{P(Y=3)}{P(Y=1)}$	Category 3 vs. category 1
3	logit $P(Y = 4 \text{ vs. } 1)$	$\frac{P(Y=4)}{P(Y=1)}$	Category 4 vs. category 1

regression is also called the relative risk ratio. Hilbe (2009) preferred the use of the relative risk ratio rather than the odds ratio since the categories of the nominal response variable are independent of each other. The odds ratio or relative risk ratio in multinomial logistic regression can be interpreted as the change in the odds or the relative risk for a one-unit change in a predictor variable when holding other predictor variables constant. To obtain the multiplicative inverse or reciprocal of the odds, the odds of being in the base category versus a particular category, we exponentiate the logit coefficient with a negative sign $\exp(-\beta)$.

7.1.4 Model Fit Statistics

Same as those discussed in the previous chapters, model fit statistics, such as the log likelihood statistic, the residual deviance, the model chi-square statistic, the AIC and BIC statistics, and the pseudo R^2 statistics, can be computed for the multinomial logistic regression model. The likelihood ratio test and the AIC and BIC statistics can also be used for model comparisons.

7.1.5 Interpretation of Model Parameter Estimates

A logit coefficient in the multinomial logistic regression model is the log odds of being a particular category relative to the base category. Exponentiating the product of the logit coefficients gives us the odds ratios of being a category j versus the baseline J. The interpretation of odds ratios is similar to that of other logistic regression models. The odds ratios are the change in the predicted odds of being in a particular category compared with the base category for a one-unit increase in the predictor variable when holding other predictor variables constant.

When an OR is larger than 1, the odds of being in a particular category versus the base category increase for a one-unit increase in the predictor variable.

When an OR is less than 1, the odds of being in a particular category versus the base category decrease for a one-unit increase in the predictor variable.

An OR of 1 indicates that there is no relationship between the predictor variable and the estimated odds.

The odds of being in the base category compared with a particular category can also be estimated since they are just the reciprocal of the odds of being in a particular category versus the base category. These two odds are different in the order when comparing categories. The odds of being in a particular category versus the base category compares category j and the base category J, whereas the odds of being in the base category compared with a particular category compares categories in the reversed order, that is, the base category J versus a particular category j.

7.2 RESEARCH EXAMPLE AND DESCRIPTION OF THE DATA AND SAMPLE

We investigate the relationships between the nominal response variable, health status, and four predictor variables. Unlike other chapters, however, here the research interest focuses on using multinomial logistic regression to predict the nominal response variable. The GSS 2016 data are used for the following analyses. The following are the variables used for data analysis in this chapter:

- `healthre`: the recoded variable of health (health status) with four categories (1 = poor health, 2 = fair health, 3 = good health, and 4 = excellent health)
- `maritals`: the recoded variable of marital (marital status) with 1 = currently married and 0 = not currently married
- `educ`: the highest education completed
- `female`: recoded variable of sex with 1 = female and 0 = male
- `wrkfull`: working full time or not

7.3 FITTING A ONE-PREDICTOR MULTINOMIAL LOGISTIC REGRESSION MODEL WITH R

7.3.1 Packages and Functions for Multinomial Logistic Regression Models in R

Several packages in R can be used for fitting multinomial logistic regression models. This chapter introduces the `VGAM`, `nnet`, and `mlogit` packages with the first package as the main focus. The `vglm()` function in `VGAM` (Yee, 2010), the `multimon()` function in `nnet` (Venables & Ripley, 2002), and the `mlogit()` function in `mlogit` (Croissant, 2020) are introduced in sequence. Since the `nnet` package is part of the R base distribution, we do not need to install it. You just need to install the other two packages first by using the `install.packages()` function and then load them with the `library()` function.

7.3.2 The `vglm()` Function With the `multinomial` Family in the VGAM Package

The `vglm()` function in the `VGAM` package can be used for the analysis of multinomial logistic regression models. If the user-written `VGAM` package is not installed, you need to install it first by typing `install.packages("VGAM")` before fitting the model. Since the package has been installed in earlier chapters, we only need to load the package by typing `library(VGAM)`.

The syntax for multinomial logistic regression models is similar to that for other models using the `vglm()` function. The `multinomial` family needs to be specified for the `family` argument. For example, the command `vglm(y ~ x, family = multinomial(refLevel = 1), data = data1)` tells R to fit a multinomial logistic regression model predicting the dependent variable *y* with an independent variable *x*. The argument `family = multinomial(refLevel = 1)` tells R that it is the `multinomial` family and the reference level is the first level. If not specified, the default reference level is the highest level of the nominal response variable. For more details on how to use this command, type `help(multinomial)` in the command prompt after loading the VGAM package.

7.3.3 The Multinomial Logistic Regression Model: One-Predictor Model

The command `mulmodel1 <- vglm(healthre ~ educ, multinomial(refLevel = 1), data = chp7.mul)` tells R to fit the multinomial logistic regression model for the nominal response variable `healthre` with the predictor variable `educ`. In the `vglm()` function, the `multinomial(refLevel = 1)` argument tells us that the `multinomial` family is used to fit the model and the reference level is the first level of `healthre`. The `summary(mulmodel)` command displays the output of the fitted model.

```
> # One-predictor multinomial logistic regression model with vglm() in VGAM
> library(VGAM)
> mulmodel1 <- vglm(healthre ~ educ, multinomial(refLevel = 1), data=chp7.mul)
> summary(mulmodel1)

Call:
vglm(formula = healthre ~ educ, family = multinomial(refLevel = 1),
     data = chp7.mul)

Pearson residuals:
                      Min      1Q      Median     3Q      Max
log(mu[,2]/mu[,1])  -4.378  -0.3625  -0.3029  -0.2267  2.704
log(mu[,3]/mu[,1])  -5.234  -0.6881  -0.4458   0.8864  2.145
log(mu[,4]/mu[,1])  -5.159  -0.3677  -0.2720  -0.1541  8.317

Coefficients:
              Estimate   Std. Error   z value   Pr(>|z|)
(Intercept):1  0.17577   0.42732      0.411    0.68082
(Intercept):2 -0.78176   0.42076     -1.858    0.06318 .
(Intercept):3 -3.27536   0.48121     -6.807    1.00e-11 ***
educ:1         0.08907   0.03382      2.633    0.00845 **
educ:2         0.21724   0.03305      6.573    4.93e-11 ***
educ:3         0.33528   0.03640      9.212     < 2e-16 ***
---
Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Names of linear predictors: log(mu[,2]/mu[,1]), log(mu[,3]/mu[,1]),
log(mu[,4]/mu[,1])

Residual deviance: 4333.297 on 5613 degrees of freedom

Log-likelihood: -2166.649 on 5613 degrees of freedom

Number of Fisher scoring iterations: 5

No Hauck-Donner effect found in any of the estimates

Reference group is level 1 of the response
```

7.3.4 Interpreting the Output

The output for the multinomial logistic regression model looks similar to that for binary logistic regression models except that there are multiple binary comparisons to the reference group. As with the other logistic regression models, the R output for the multinomial logistic regression model also includes the call, the Pearson residuals, the coefficients, the number and names of the three linear predictors, the residual deviance, the log-likelihood value, and the number of iterations. In addition, it lists the reference group number.

The coefficients section displays the parameter estimates for the intercepts and the predictor variable, their standard errors, the Wald z statistics, and the associated p values.

The null hypothesis for the Wald z test is that the coefficient of the predictor variable is 0, and the alternative hypothesis is that the coefficient of the predictor variable is significantly different from 0.

Three coefficients for the predictor variable `educ` are displayed as `educ:1`, `educ:2`, and `educ: 3` since they are the parameter estimates for the three binary logistic models comparing each category with the base category. Only three binary models are estimated since the base outcome is category 1. These three equations, labeled `log(mu[,2]/mu[,1])`, `log(mu[,3]/mu[,1])`, and `log(mu[,4]/mu[,1])`, compare categories 2 with 1, categories 3 with 1, and categories 4 with 1, respectively. The estimated intercepts and logit coefficients for these three sub-models are numbered 1, 2, and 3 in the output.

Based on the parameter estimates in the output, the three equations can be expressed as:

$$\ln\left(\frac{P(Y=2)}{P(Y=1)}\right) = .176 + .089\textbf{educ}$$

$$\ln\left(\frac{P(Y=3)}{P(Y=1)}\right) = -.782 + .217\textbf{educ}$$

$$\ln\left(\frac{P(Y=4)}{P(Y=1)}\right) = -3.275 + .335\textbf{educ}$$

The first equation compares category 2 and category 1. The coefficient for `educ`, displayed as `educ:1`, $\beta = .089$, Wald $z = 2.633$. The associated p value, `Pr(>|z|)` $< .01$, so we reject the null hypothesis and conclude that `educ` is significant in predicting the log odds of being in category 2 versus category 1.

The second equation compares category 3 and category 1. The predictor variable `educ`, displayed as `educ:2`, is also significant. For `educ:2`, $\beta = .217$, Wald $z = 6.573$, $p < .001$.

The third equation compares category 4 and category 1. The predictor variable `educ`, displayed as `educ:3`, is also significant. For `educ:3`, $\beta = .335$, Wald $z = 9.212$, $p < .001$.

The Coefficients section also reports the intercepts (labeled as `(Intercept):1`, `(Intercept):2`, and `(Intercept):3`). They are the intercepts for each equation comparing a particular category with the reference group or base category. The reference group in this example is level 1 (i.e., healthre = 1). If not specified, the default is the highest outcome, but you can specify any category as the reference group.

We can extract the coefficients with `coef(mulmodel1, matrix = TRUE)` and obtain the confidence intervals with `confint(mulmodel1, matrix = TRUE)` as follows.

```
> coef(mulmodel1, matrix = TRUE)
            log(mu[,2]/mu[,1])  log(mu[,3]/mu[,1])  log(mu[,4]/mu[,1])
(Intercept)         0.17577310          -0.7817562          -3.2753578
educ                0.08906957           0.2172438           0.3352757

> confint(mulmodel1, matrix = TRUE)
                    2.5 %       97.5 %
(Intercept):1 -0.66175738   1.01330357
(Intercept):2 -1.60643404   0.04292172
(Intercept):3 -4.21851259  -2.33220303
educ:1         0.02277699   0.15536215
educ:2         0.15246679   0.28202081
educ:3         0.26393850   0.40661298
```

7.3.5 Interpreting the Odds Ratios of Being in a Particular Category Versus the Base Category for the Multinomial Logistic Regression Model

The multinomial logistic regression model estimates the logit odds of being a category relative to the baseline category. Recall that the exponentiated (β_j) is the odds ratio of being in a category j versus the baseline J for a one-unit change in a predictor variable. In this model, we define the odds ratio of being in category 2 compared with the base category 1 as OR (2, 1). Since $\beta = .089$ for `educ:1`, OR(2, 1) = $e^{(.089)} = 1.093$, which indicates that for a one-unit increase in education the odds of being in category 2 of health condition versus the base category 1 increase by a factor of 1.093. In other words, people who work full time have greater odds of being in the highest health condition (category 4) rather than being in category 1.

The odds ratio of being in category 3 versus category 1, OR(3, 1) = $e^{(.217)} = 1.242$, which indicates that for a one-unit increase in education the odds of being in category 2 of health condition versus the base category 1 increase by 24.2%. Similarly, the odds ratio of being in category 4 versus category 1, OR(4, 1) = $e^{(.335)} = 1.398$, which indicates that for a one-unit increase in education the odds of being in category 4 of health condition versus the base category 1 increase by 39.8%.

The above results can be obtained using the `exp(coef(slmodel1, matrix = TRUE))` command. We also use the `exp(confint(mulmodel1, matrix = TRUE))` command to obtain the corresponding confidence intervals. Both results are combined with the `cbind(exp(coef(mulmodel1)), exp(confint(mulmodel1)))` command.

```
> exp(coef(mulmodel1, matrix = TRUE))
            log(mu[,2]/mu[,1])  log(mu[,3]/mu[,1])  log(mu[,4]/mu[,1])
(Intercept)           1.192168           0.4576017          0.03780334
educ                  1.093157           1.2426470          1.39832590

> exp(confint(mulmodel1, matrix = TRUE))
                    2.5 %      97.5 %
(Intercept):1  0.51594383  2.75468631
(Intercept):2  0.20060168  1.04385618
(Intercept):3  0.01472052  0.09708164
educ:1         1.02303836  1.16808091
educ:2         1.16470378  1.32580631
educ:3         1.30204812  1.50172279

> cbind(exp(coef(mulmodel1)), exp(confint(mulmodel1)))
                                2.5 %      97.5 %
(Intercept):1  1.19216752  0.51594383  2.75468631
(Intercept):2  0.45760169  0.20060168  1.04385618
(Intercept):3  0.03780334  0.01472052  0.09708164
educ:1         1.09315670  1.02303836  1.16808091
educ:2         1.24264702  1.16470378  1.32580631
educ:3         1.39832590  1.30204812  1.50172279
```

7.3.6 Model Fit Statistics

Testing the Overall Model Using the Likelihood Ratio Test

To test if the overall model is significant, we fit a null model with the intercept only and compare the single-predictor model with the null model using the `lrtest()` function. The null model is fitted using the `vglm()` function with the predictor as 1 for the intercept in the model equation. The command and the output are displayed below.

```
> # Testing the overall model using the likelihood ratio test
> mulmodel0 <- vglm(healthre ~ 1, multinomial(refLevel = 1), data=chp7.mul)
> summary(mulmodel0)

Call:
vglm(formula = healthre ~ 1, family = multinomial(refLevel = 1),
    data = chp7.mul)

Pearson residuals:
                      Min       1Q   Median       3Q    Max
log(mu[,2]/mu[,1]) -2.107  -0.4351  -0.3087  -0.3087 1.6649
log(mu[,3]/mu[,1]) -2.459  -0.6678  -0.6608   0.9287 0.9287
log(mu[,4]/mu[,1]) -2.094  -0.4218  -0.3025  -0.3025 1.6997
```

```
Coefficients:
               Estimate   Std. Error   z value    Pr(>|z|)
(Intercept):1   1.28610    0.10400      12.37     <2e-16 ***
(Intercept):2   2.04715    0.09782      20.93     <2e-16 ***
(Intercept):3   1.25518    0.10436      12.03     <2e-16 ***
---
Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Names of linear predictors: log(mu[,2]/mu[,1]), log(mu[,3]/mu[,1]),
log(mu[,4]/mu[,1])

Residual deviance: 4476.435 on 5616 degrees of freedom

Log-likelihood: -2238.218 on 5616 degrees of freedom

Number of Fisher scoring iterations: 5

No Hauck-Donner effect found in any of the estimates

Reference group is level 1 of the response
```

The `lrtest(mulmodel0, mulmodel1)` command compares the log-likelihood statistics of the fitted model `mulmodel1` and the null model `mulmodel0` using the likelihood ratio test.

```
> lrtest(mulmodel0, mulmodel1)
Likelihood ratio test

Model 1: healthre ~ 1
Model 2: healthre ~ educ

   #Df    LogLik   Df     Chisq     Pr(>Chisq)
1  5616  -2238.2
2  5613  -2166.7   -3    143.14     < 2.2e-16 ***
---
Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

The null hypothesis of the test for the overall model is that the predictor variable does not contribute to the model, and the alternative hypothesis is that the one-predictor model is better than the null model with no independent variables. The likelihood ratio chi-square test statistic $LR\ \chi^2_{(3)} = 143.14$, $p < .001$, which indicates that the overall model with one predictor `educ` is significantly different from zero. Therefore, the one-predictor model provides a better fit than the null model in predicting the logit or log odds of being in a particular category relative to the base category.

Pseudo R^2

We use the `nagelkerke()` function in the `rcompanion` package (Mangiafico, 2021) to compute the pseudo R^2 statistics for the single-predictor model. We load the package first with `library(rcompanion)` and then use `nagelkerke(mulmodel1)`.

```
> # Pseudo R2 with nagelkerke()
> library(rcompanion)
> nagelkerke(mulmodel1)
$`Models`

Model: "vglm, healthre ~ educ, multinomial(refLevel = 1), chp7.mul"
Null:  "vglm, healthre ~ 1, multinomial(refLevel = 1), chp7.mul"

$Pseudo.R.squared.for.model.vs.null
                             Pseudo.R.squared
McFadden                            0.0319758
Cox and Snell (ML)                  0.0735744
Nagelkerke (Cragg and Uhler)        0.0809962

$Likelihood.ratio.test
Df.diff    LogLik.diff     Chisq       p.value
      3        -71.569    143.14    7.9595e-31

$Number.of.observations
Model: 1873
Null:  1873

$Messages
[1] "Note: For models fit with REML, these statistics are based on refitting with ML"

$Warnings
[1] "None"
```

The McFadden R^2 is .032, the Cox and Snell R^2 is .074, and the Nagelkerke R^2 is .081. The same results can be computed using the equations for these three pseudo R^2 statistics. In the R commands below, LLM1 is the log-likelihood value for the single-predictor model and LL0 is the log-likelihood value for the null model. In addition, McFadden1 is the object name for McFadden's R^2, CS1 for Cox and Snell's R^2, and NG1 for Nagelkerke's R^2.

```
> LLM1 <- logLik(mulmodel1)
> LL0 <- logLik(mulmodel0)
> McFadden1 <- 1-(LLM1/LL0)
> McFadden1
[1] 0.0319758
> CS1 <- 1-exp(2*(LL0-LLM1)/1873)
> CS1
[1] 0.07357443
> NG1 <- CS1/(1-exp(2*LL0/1873))
> NG1
[1] 0.08099622
```

AIC and BIC Statistics

The AIC and BIC statistics can also be computed from the AIC() and BIC() functions. The output is shown as follows.

```
> AIC(mulmodel1)
[1] 4345.297
> BIC(mulmodel1)
[1] 4378.509
```

7.4 FITTING A MULTIPLE-PREDICTOR MULTINOMIAL LOGISTIC REGRESSION MODEL WITH R

7.4.1 The Multinomial Logistic Regression Model: Multiple-Predictor Model

The command mulmodel2 <- vglm(healthre ~ educ + maritals + female + wrkfull, multinomial(refLevel = 1), data = chp7.mul) tells R to predict the nominal response variable healthre from the four predictor variables educ, maritals, female and wrkfull with multinomial logistic regression. In the vglm() function, the model equation is specified as healthre ~ educ + maritals + female + wrkfull; the multinomial(refLevel = 1) argument specifies that the multinomial family is used to fit the model and the reference level is the first level of healthre; and the data argument specifies data = chp7.mul. The output is shown by the summary(mulmodel2) command.

```
> #Multiple-predictor multinomial logistic regression model with vglm() in VGAM
> mulmodel2 <- vglm(healthre ~ educ + maritals + female + wrkfull, multinomial(refLevel = 1),
  data=chp7.mul)
> summary(mulmodel2)
Call:
vglm(formula = healthre ~ educ + maritals + female + wrkfull,
    family = multinomial(refLevel = 1), data = chp7.mul)

Pearson residuals:
                        Min       1Q      Median       3Q      Max
log(mu[,2]/mu[,1])   -5.057   -0.3768    -0.2919   -0.2047   3.343
log(mu[,3]/mu[,1])   -5.982   -0.6924    -0.3757    0.8763   2.003
log(mu[,4]/mu[,1])   -5.734   -0.3810    -0.2745   -0.1603   7.289

Coefficients:
                Estimate   Std. Error   z value   Pr(>|z|)
(Intercept):1    0.02270    0.45925      0.049    0.960576
(Intercept):2   -1.19945    0.45236     -2.652    0.008013 **
(Intercept):3   -3.63716    0.51091     -7.119    1.09e-12 ***
```

```
educ:1          0.08421    0.03483    2.417   0.015632 *
educ:2          0.19416    0.03403    5.706   1.16e-08 ***
educ:3          0.31509    0.03729    8.450    < 2e-16 ***
maritals:1      0.38625    0.22994    1.680   0.093000 .
maritals:2      0.68849    0.21996    3.130   0.001747 **
maritals:3      0.64602    0.23401    2.761   0.005769 **
female:1       -0.02156    0.21283   -0.101   0.919298
female:2        0.16979    0.20413    0.832   0.405546
female:3        0.15659    0.21921    0.714   0.475009
wrkfull:1       0.32714    0.23802    1.374   0.169321
wrkfull:2       0.95782    0.22646    4.230   2.34e-05 ***
wrkfull:3       0.82250    0.24018    3.424   0.000616 ***

---
Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Names of linear predictors: log(mu[,2]/mu[,1]), log(mu[,3]/mu[,1]),
log(mu[,4]/mu[,1])

Residual deviance: 4275.242 on 5604 degrees of freedom

Log-likelihood: -2137.621 on 5604 degrees of freedom

Number of Fisher scoring iterations: 6

No Hauck-Donner effect found in any of the estimates

Reference group is level 1 of the response
```

7.4.2 Interpreting R Output

The coefficients section (labeled `Coefficients:`) displays the parameter estimates for the three intercepts and the four predictor variables. Since the multinomial logistic regression model includes a series of binary logistic regression models, the table displays the parameter estimates for the three binary logistic models comparing each category versus the base category. These three equations, labeled `log(mu[,2]/mu[,1])`, `log(mu[,3]/mu[,1])`, and `log(mu[,4]/mu[,1])`, compare categories 2 with 1, categories 3 with 1, and categories 4 with 1, respectively. The estimated intercepts and logit coefficients for these three sub-models are numbered 1, 2, and 3 in the output.

Based on the parameter estimates in the output, the three equations for the model can be expressed as:

$$\ln\left(\frac{P(Y=2)}{P(Y=1)}\right) = .023 + .084\texttt{educ} + .386\texttt{maritals} - .022\texttt{female} + .327\texttt{wrkfull}$$

$$\ln\left(\frac{P(Y=3)}{P(Y=1)}\right) = -1.199 + .194\texttt{educ} + .688\texttt{maritals} + .170\texttt{female} + .958\texttt{wrkfull}$$

$$\ln\left(\frac{P(Y=4)}{P(Y=1)}\right) = -3.637 + .315\texttt{educ} + .646\texttt{maritals} + .157\texttt{female} + .823\texttt{wrkfull}$$

The first equation compares category 2 and category 1. Among the four predictor variables, only `educ` is significant, whereas the other three predictor variables `maritals`, `female`, and `wrkfull` are not significant in predicting the log odds of being in category 2 versus category 1. The coefficient for `educ`, displayed as `educ:1`, $\beta = .084$, Wald $z = 2.417$. The associated p value, `Pr(>|z|) < .05`, so we reject the null hypothesis and conclude that `educ` is significant in predicting the log odds of being in category 2 versus category1. The coefficient for `maritals`, displayed as `maritals:1`, $\beta = .386$, Wald $z = 1.680$, $p > .05$, which is not significant; the coefficient for `female`, displayed as `female:1`, $\beta = -.022$, Wald $z = -.101$, $p > .05$, which is not significant, either; the coefficient for `wrkfull`, displayed as `wrkfull:1`, $\beta = .327$, Wald $z = 1.374$, $p > .05$, which is not significant, either.

The second equation compares category 3 and category 1. The three predictor variables `educ`, `maritals`, and `wrkfull` are significant, whereas `female` is not significant. The predictor variable `educ`, displayed as `educ:2`, $\beta = .194$, Wald $z = 5.706$, $p < .001$. For the predictor variable `maritals`, displayed as `maritals:2`, $\beta = .688$, Wald $z = 3.130$. The associated p value, `P>|z| < .01`, so we reject the null hypothesis; for `wrkfull`, displayed as `wrkfull:2`, $\beta = .958$, Wald $z = 4.230$, $p < .001$; however, for `female`, displayed as `female:2`, $\beta = .170$, Wald $z = .830$, $p = .406$, so the coefficient for `female` is not significantly different from 0.

The third equation compares category 4 and category 1. The three predictor variables `educ`, `maritals`, and `wrkfull` are significant, whereas `female` is not significant. The predictor variable `educ`, displayed as `educ:3`, is also significant. For `educ:3`, $\beta = .315$, Wald $z = 8.450$, $p < .001$. For the predictor variable `maritals`, displayed as `maritals:3`, $\beta = .646$, Wald $z = 2.761$, $p < .01$, so we reject the null hypothesis; for `wrkfull`, displayed as `wrkfull:3`, $\beta = .823$, Wald $z = 3.424$, $p < .001$; however, for `female`, displayed as `female:3`, $\beta = .157$, Wald $z = .714$, $p > .05$, so the coefficient for `female` is not significantly different from 0.

We use the `coef(mulmodel2, matrix = TRUE)` command to extract the coefficients table for the three underlying binary models which compare categories 2 with 1, categories 3 with 1, and categories 4 with 1, respectively. We also use the `confint(mulmodel2, matrix = TRUE)` command to compute the corresponding confidence intervals. The output is omitted here.

We request the odds ratios of being in a particular category versus the base category and the corresponding confidence intervals using the `exp(coef(mulmodel2, matrix = TRUE))` and the `exp(confint(mulmodel2, matrix = TRUE))` commands, respectively. The results are combined with the `cbind()` function.

```
> exp(coef(mulmodel2, matrix = TRUE))
              log(mu[,2]/mu[,1])  log(mu[,3]/mu[,1])  log(mu[,4]/mu[,1])
(Intercept)            1.0229608           0.3013613          0.02632712
educ                   1.0878549           1.2142874          1.37038037
maritals               1.4714529           1.9907077          1.90794028
female                 0.9786676           1.1850530          1.16951660
wrkfull                1.3869905           2.6060030          2.27617493

> exp(confint(mulmodel2, matrix = TRUE))
                      2.5 %       97.5 %
(Intercept):1   0.415858123   2.51636007
(Intercept):2   0.124176341   0.73136819
(Intercept):3   0.009672075   0.07166171
educ:1          1.016061645   1.16472089
educ:2          1.135942132   1.29803614
educ:3          1.273795085   1.47428923
maritals:1      0.937605488   2.30925872
maritals:2      1.293540526   3.06362029
maritals:3      1.206069994   3.01826272
female:1        0.644877457   1.48522842
female:2        0.794293786   1.76804926
female:3        0.761058336   1.79719349
wrkfull:1       0.869896953   2.21146033
wrkfull:2       1.671903678   4.06198751
wrkfull:3       1.421546528   3.64460269

> cbind(exp(coef(mulmodel2)), exp(confint(mulmodel2)))
                                   2.5 %       97.5 %
(Intercept):1   1.02296079   0.415858123   2.51636007
(Intercept):2   0.30136129   0.124176341   0.73136819
(Intercept):3   0.02632712   0.009672075   0.07166171
educ:1          1.08785487   1.016061645   1.16472089
educ:2          1.21428742   1.135942132   1.29803614
educ:3          1.37038037   1.273795085   1.47428923
maritals:1      1.47145290   0.937605488   2.30925872
maritals:2      1.99070766   1.293540526   3.06362029
maritals:3      1.90794028   1.206069994   3.01826272
female:1        0.97866763   0.644877457   1.48522842
female:2        1.18505297   0.794293786   1.76804926
female:3        1.16951660   0.761058336   1.79719349
wrkfull:1       1.38699048   0.869896953   2.21146033
wrkfull:2       2.60600304   1.671903678   4.06198751
wrkfull:3       2.27617493   1.421546528   3.64460269
```

The results of the odds ratios across the three binary comparisons are summarized in Table 7.2.

7.4.3 Interpreting the Odds Ratios of Being in a Category *j* Versus the Base Category 1

The odds ratio of being in a particular category compared with the base category can be interpreted as the change in the odds of being in that category versus the base category for a one-unit increase in the predictor variable when holding other predictors constant. Recall there are $J-1$ binary comparisons for a nominal response variable with J categories.

TABLE 7.2 ● Odds Ratios for All Four Predictor Variables Across Three Comparisons ($Y = j$ vs. $Y = 1$)

Category Comparisons	$Y = 2$ vs. $Y = 1$	$Y = 3$ vs. $Y = 1$	$Y = 4$ vs. $Y = 1$
Variables	OR	OR	OR
`Educ`	1.088*	1.214**	1.370**
`Maritals`	1.471	1.991**	1.908**
`Female`	.979	1.185	1.170
`Wrkfull`	1.387	2.606**	2.276**

*$p < 0.05$
**$p < 0.01$

The odds ratio for each predictor needs to be interpreted across three comparisons. For `educ`, the odds ratios of being in category 2 versus category 1, category 3 versus category 1, and category 4 versus category 1 are 1.088, 1.214, and 1.370, respectively. The results indicate that the odds of being in category 2 versus the base category, category 3 versus the base category, and category 4 versus the base category increase by a factor of 1.088, 1.214, and 1.370, respectively, for a one-unit increase in the `educ` predictor when holding all other predictors constant.

For `maritals`, the odds ratios for the three binary comparisons (i.e., categories 2 vs. 1, 3 vs. 1, and 4 vs. 1) are 1.471, 1.991, and 1.908, respectively. OR(2,1) = 1.471, $p > .05$, which indicates that there is no relationship between `maritals` and the odds of being in category 2 versus the base category 1. The odds ratios for the other two comparisons are significant, OR(3,1) = 1.991, OR(4,1) = 1.908, which indicates that the odds of being in category 3 versus the base category and category 4 versus the base category for the married are 1.991 and 1.908 times as large as the odds for the unmarried, respectively, when holding other predictors constant.

The odds ratios for `wrkfull` can be interpreted in the similar way. The odds ratios for three binary comparisons (i.e., categories 2 vs. 1, 3 vs. 1, and 4 vs. 1) are 1.387, 2.606, and 2.276, respectively. OR(2,1) = 1.387, $p > .05$, which indicates that there is no relationship between `wrkfull` and the odds of being in category 2 versus the base category 1. The odds ratios for the other two comparisons are significant. The odds of being in category 3 versus the base category and category 4 versus the base category for those working full time are 2.606 and 2.267 times as large as the odds for those not working full time, respectively, when holding other predictors constant.

With regard to `female`, none of the odds ratios for the binary comparisons are significant. It indicates that being female does not impact the odds of being in any particular category versus the base category when holding other predictors constant.

7.4.4 Model Fit Statistics

Testing the Overall Model Using the Likelihood Ratio Test

To test if the overall model is significant, we compare the multiple-predictor model with the null model using the `lrtest()` function. The `lrtest(mulmodel0, mulmodel2)` command compares the log-likelihood statistics of the fitted model `mulmodel2` and the null model `mulmodel0` using the likelihood ratio test. The resulting output is displayed below.

```
> # Testing the overall model using the likelihood ratio test
> lrtest(mulmodel0, mulmodel2)
Likelihood ratio test

Model 1: healthre ~ 1
Model 2: healthre ~ educ + maritals + female + wrkfull
    #Df    LogLik    Df    Chisq    Pr(>Chisq)
1   5616   -2238.2
2   5604   -2137.6   -12   201.19    < 2.2e-16 ***
---
Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' '
```

The likelihood ratio test $\chi^2_{(12)} = 201.19, p < .001$, indicates that the full model with the four predictors provides a better fit than the null model with no independent variables in predicting the nominal response variable.

Pseudo R^2

The `nagelkerke(mulmodel2)` command computes the three types of pseudo R^2 statistics and the likelihood ratio test statistic for the overall multiple-predictor model. The output is omitted here. To obtain the same results, we can also compute the three types of pseudo R^2 statistics with their equations for the multiple-predictor model as follows.

```
> LLM2 <- logLik(mulmodel2)
> McFadden2 <- 1-(LLM2/LL0)
> McFadden2
[1] 0.04494503
> CS2 <- 1-exp(2*(LL0-LLM2)/1873)
> CS2
[1] 0.1018496
> NG2 <- CS2/(1-exp(2*LL0/1873))
> NG2
[1] 0.1121237
```

In the output, `LLM2` is the log-likelihood value for the multiple-predictor model and `LL0` is the log-likelihood value for the null model. The McFadden R^2 is .045, the Cox and Snell R^2 is .102, and the Nagelkerke R^2 is .112.

AIC and BIC Statistics

The AIC(mulmodel2) and BIC(mulmodel2) commands produce the AIC and BIC statistics.

```
> AIC(mulmodel2)
[1] 4305.241
> BIC(mulmodel2)
[1] 4388.271
```

The AIC and BIC statistics are 4,305.241 and 4,388.271, respectively. Recall the AIC and BIC in the single predictor model are 4,345.297 and 4,378.509, respectively. Compared with the single-variable model, both AIC and BIC indicate that the multiple-predictor model fits the data better.

7.4.5 Interpreting the Predicted Probabilities With the ggpredict() Function in the ggeffects Package

By using the ggpredict() function in the ggeffects package (Lüdecke, 2018b), we can compute the predicted probabilities for each category of the nominal response variable at specified values of the predictor variables. The command pr.mul2m <- ggpredict(mulmodel2, terms = "educ[12, 14, 16]", ci = NA) tells R to compute the predicted probabilities for each category of the nominal response variable using the ggpredict() function. The argument inside the function includes the estimated model, mulmodel2, the terms = "educ[12, 14, 16]" option, which specifies the predictor variable educ at the values of 12, 14, and 16 when holding the other predictor variables at their means, and the ci = NA option for not specifying the confidence intervals. The terms option can specify up to four variables, including the second to fourth grouping variables. Please also note that the confidence intervals can only be obtained for the cumulative probabilities, so the ci = NA option is needed there. The output is assigned to the object named pr.mul2m.

```
> library(ggeffects)
> pr.mul2m <- ggpredict(mulmodel2, terms="educ[12, 14, 16]", ci=NA)
> pr.mul2m

# Predicted probabilities of healthre

# Response Level = 1

educ   |   Predicted
-------------------
12     |   0.07
14     |   0.05
16     |   0.03
```

```
# Response Level = 2

educ   |    Predicted
--------------------
  12   |       0.26
  14   |       0.22
  16   |       0.17

# Response Level = 3

educ   |    Predicted
--------------------
  12   |       0.50
  14   |       0.51
  16   |       0.51

# Response Level = 4

educ   |    Predicted
--------------------
  12   |       0.17
  14   |       0.22
  16   |       0.28

Adjusted for:
* maritals = 0.44
*    female = 0.56
*   wrkfull = 0.47

> plot(pr.mul2m)
```

When `educ` = 12, and the other predictor variables are held at their means (`maritals` = .44, `female` = .56, and `wrkfull` = .47), the predicted probability for the response level 1 (i.e., $Y = 1$) is .07.

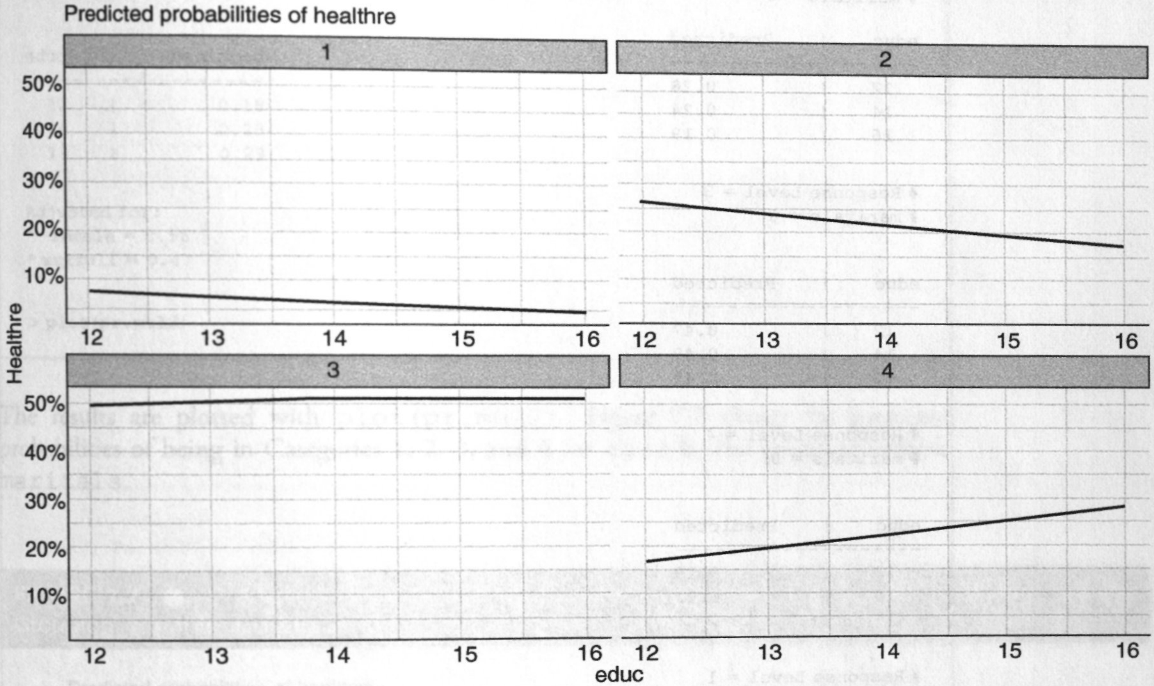
When `educ` = 14, and the other three predictor variables are held at their means, the predicted probability for $Y = 1$ is .05.

When `educ` = 16, and the other predictor variables are held at their means, the predicted probability for $Y = 1$ is .03.

The predicted probabilities for all the four response levels are plotted using `plot(pr.mul2m)`. Figure 7.1 shows the predicted probabilities of being in each category (i.e., $Y = 1$, 2, 3, and 4).

The graph shows that with the increase of the years of education, the probabilities of being in poor and fair health condition (categories 1 and 2) decrease. In other words, people with higher levels of education are less likely associated with poor and fair health conditions. In addition, with the increase of the years of education, the probabilities of being in good and excellent health condition (categories 3 and 4) increase. In other words, people with a higher level of education are more likely to be in good and excellent health condition.

FIGURE 7.1 ⬢ Predicted Probabilities for educ at 12, 14, and 16



With the terms = c("educ[12, 14, 16]", "maritals") argument, the predicted probabilities for educ at 12, 14, and 16 can be grouped by maritals. The output is assigned to the object named pr.mul2.

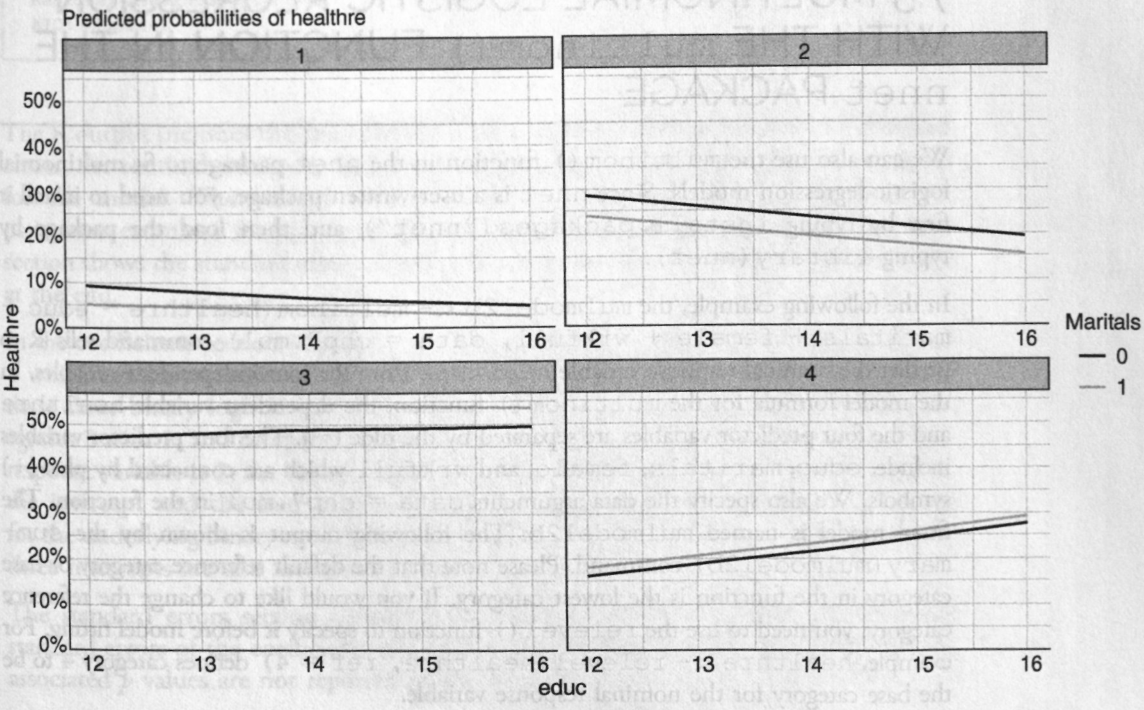
```
> pr.mul2 <- ggpredict(mulmodel2, terms=c("educ[12, 14, 16]", "maritals"), ci=NA)
> pr.mul2

# Predicted probabilities of healthre

# Response Level = 1
# maritals = 0

educ   |   Predicted
-------------------
  12   |      0.09
  14   |      0.06
  16   |      0.04
```

```
# Response Level = 2
# maritals = 0

educ   |   Predicted
--------------------
  12   |        0.28
  14   |        0.24
  16   |        0.19

# Response Level = 3
# maritals = 0

educ   |   Predicted
--------------------
  12   |        0.47
  14   |        0.49
  16   |        0.49

# Response Level = 4
# maritals = 0

educ   |   Predicted
--------------------
  12   |        0.16
  14   |        0.22
  16   |        0.28

# Response Level = 1
# maritals = 1

educ   |   Predicted
--------------------
  12   |        0.05
  14   |        0.03
  16   |        0.02

# Response Level = 2
# maritals = 1

educ   |   Predicted
--------------------
  12   |        0.24
  14   |        0.19
  16   |        0.15

# Response Level = 3
# maritals = 1

educ   |   Predicted
--------------------
  12   |        0.53
  14   |        0.54
  16   |        0.53
```

```
# Response Level = 4
# maritals = 1

educ    |  Predicted
-------------------
  12    |      0.18
  14    |      0.23
  16    |      0.29

Adjusted for:
*  female = 0.56
* wrkfull = 0.47

> plot(pr.mul2)
```

The results are plotted with plot(pr.mul2). Figure 7.2 shows the predicted probabilities of being in Categories 1, 2, 3, and 4 for educ by the grouping variable maritals.

FIGURE 7.2 ● Predicted Probabilities of Being in Categories 1, 2, 3, and 4 for educ by the Grouping Variable maritals



7.4.6 Model Comparisons Using the Likelihood Ratio Test

The likelihood ratio test, or the deviance difference test, is used to compare the multiple-predictor model and the one-predictor model. In the `lrtest(mulmodel1, mulmodel2)` syntax, `mulmodel1` and `mulmodel2` are the two models being compared. The following output is displayed.

```
> # Model comparison with the likelihood ratio test
> lrtest(mulmodel1, mulmodel2)
Likelihood ratio test


Model 1: healthre ~ educ
Model 2: healthre ~ educ + maritals + female + wrkfull
      #Df     LogLik    Df     Chisq   Pr(>Chisq)
1    5613    -2166.7
2    5604    -2137.6    -9    58.056    3.171e-09 ***
---
Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

The likelihood ratio test, $\chi^2_{(9)} = 58.056$, $p < .001$, which indicates that the full model with the four predictor variables fits the data better than the single-predictor model.

7.5 MULTINOMIAL LOGISTIC REGRESSION WITH THE `multinom()` FUNCTION IN THE `nnet` PACKAGE

We can also use the `multinom()` function in the `nnet` package to fit multinomial logistic regression models. Since `nnet` is a user-written package, you need to install it first by typing `install.packages("nnet")` and then load the package by typing `library(nnet)`.

In the following example, the `mulmodel2b <- multinom(healthre ~ educ + maritals + female + wrkfull, data = chp7.mul)` command tells R to predict the nominal response variable `healthre` from the four independent variables. In the model formula for the `multinom()` function, the dependent variable `healthre` and the four predictor variables are separated by the tilde (~). The four predictor variables include, `educ`, `maritals`, `female`, and `wrkfull` which are connected by plus (+) symbols. We also specify the data arguments `data = chp7.mul` in the function. The fitted model is named `mulmodel2b`. The following output is shown by the `summary(mulmodel2b)` command. Please note that the default reference category or base category in the function is the lowest category. If you would like to change the reference category, you need to use the `relevel()` function to specify it before model fitting. For example, `healthre <- relevel(healthre, ref = 4)` defines category 4 to be the base category for the nominal response variable.

```
> # Using multinom() in nnet
> library(nnet)
> mulmodel2b <- multinom(healthre ~ educ + maritals + female + wrkfull, data=chp7.mul)
# weights: 24 (15 variable)
initial value 2596.529338
iter 10 value 2150.935582
iter 20 value 2137.622895
final value 2137.620746
converged
> summary(mulmodel2b)
Call:
multinom(formula = healthre ~ educ + maritals + female + wrkfull,
    data = chp7.mul)

Coefficients:
  (Intercept)       educ  maritals       female   wrkfull
2  0.02279756 0.08420084 0.3861814  -0.02155419 0.3271689
3 -1.19940172 0.19415272 0.6884447   0.16981115 0.9578518
4 -3.63712240 0.31508366 0.6459960   0.15661149 0.8225368

Std. Errors:
  (Intercept)       educ  maritals    female   wrkfull
2   0.4592498 0.03483414 0.2299392 0.2128258 0.2380238
3   0.4523597 0.03402868 0.2199547 0.2041307 0.2264614
4   0.5109066 0.03729027 0.2340112 0.2192057 0.2401842


Residual Deviance: 4275.241
AIC: 4305.241
```

The R output includes the call of the model command, the coefficients, the standard errors, the residual deviance, and the AIC statistic. The call shows the R command for the model. The second section shows the coefficients table including the parameter estimates for the intercept and the four predictor variables. The third section shows the standard errors. Finally, the deviance residuals and AIC are displayed at the end.

In the coefficients section (labeled `Coefficients:`), the first column lists 2, 3, and 4 which are the categories of the nominal response variable. They are the outcomes of these three binary logistic regression models which compare categories 2 with 1, categories 3 with 1, and categories 4 with 1, respectively. For example, when the nominal response variable is 2, the binary model compares categories 2 with 1. The first row lists the intercept and the four predictor variables. Each predictor has three coefficients for the underlying binary models since the reference level is category 1. In addition, the three intercepts are also estimated for the binary models.

The standard errors section (labeled `Std. Errors:`) provides the corresponding standard errors of the coefficients above. Please note that the Wald z statistics and the associated p values are not reported in the output.

We can compute the Wald z statistics and the associated p values with the following command `z <- summary(mulmodel2b)$coefficients/summary(mulmodel2b)$standard.errors`. We first compute the Wald z statistics by dividing the coefficients to their standard errors and name the object `z`. We then compute the associated p values with the command `p <-(1-pnorm(abs(z),0,1))*2`.

```
> z <- summary(mulmodel2b)$coefficients/summary(mulmodel2b)$standard.errors
> z
   (Intercept)       educ   maritals      female   wrkfull
2   0.04964088   2.417193   1.679493  -0.1012762  1.374522
3  -2.65143354   5.705561   3.129938   0.8318747  4.229647
4  -7.11895783   8.449488   2.760535   0.7144498  3.424608

> p <- (1-pnorm(abs(z),0,1))*2
> p
   (Intercept)          educ     maritals      female       wrkfull
2 9.604086e-01  1.564071e-02  0.093055936  0.9193312  1.692796e-01
3 8.015089e-03  1.159605e-08  0.001748430  0.4054797  2.340586e-05
4 1.087352e-12  0.000000e+00  0.005770681  0.4749491  6.156865e-04
```

The logit coefficients can be extracted with `coef(mulmodel2b)`.

```
> coef(mulmodel2b)
   (Intercept)        educ   maritals       female    wrkfull
2   0.02279756  0.08420084  0.3861814  -0.02155419  0.3271689
3  -1.19940172  0.19415272  0.6884447   0.16981115  0.9578518
4  -3.63712240  0.31508366  0.6459960   0.15661149  0.8225368
```

The odds ratios can be obtained by using `exp(coef(mulmodel2b))`. We also use the `exp(confint(mulmodel2b))` command to obtain the corresponding confidence intervals.

```
> exp(coef(mulmodel2b))
   (Intercept)      educ   maritals     female   wrkfull
2   1.0230594   1.087847  1.471352  0.9786764  1.387036
3   0.3013745   1.214282  1.990617  1.1850810  2.606092
4   0.0263280   1.370374  1.907886  1.1695411  2.276267

> exp(confint(mulmodel2b))
, , 2

                 2.5 %     97.5 %
(Intercept)  0.4158979   2.516604
educ         1.0160547   1.164713
```

```
maritals        0.9375444    2.309091
female          0.6448838    1.485241
wrkfull         0.8699243    2.211535

, , 3

                    2.5 %       97.5 %
(Intercept)     0.1241816    0.7314013
educ            1.1359369    1.2980300
maritals        1.2934866    3.0634692
female          0.7943130    1.7680901
wrkfull         1.6719583    4.0621322

, , 4

                    2.5 %       97.5 %
(Intercept)     0.00967238   0.0716642
educ            1.27378919   1.4742822
maritals        1.20604029   3.0181662
female          0.76107465   1.7972304
wrkfull         1.42160204   3.6447554
```

We can use the ggpredict() function in the ggeffects package to compute the predicted probabilities for each category of the nominal response variable at the specified values of the predictor variables. The same command introduced in the previous section on the vglm() function in this chapter can be used. The output is omitted here.

7.6 MULTINOMIAL LOGISTIC REGRESSION WITH THE mlogit() FUNCTION IN THE mlogit PACKAGE

We can also use the mlogit() function in the mlogit package to fit multinomial logistic regression models. Since mlogit is a user-written package, you need to install it first by typing install.packages("mlogit") and then load the package by typing library(mlogit).

Different from the vglm() and multinom() functions, we need to follow two steps to use the mlogit() function. First, before fitting the model, we need to create the dataset in the wide format by using the mlogit.data() function so that the mlogit() function can load the data. In the command chp7 <- mlogit.data (chp7.mul, choice = "healthre", shape = "wide"), we specify the data chp7.mul, followed by the choice = "healthre" argument for the nominal response variable and the shape = "wide" argument. The new data are named chp7. Second, we run the model command. The model equation for the mlogit() function is slightly different from those for the vglm() and multinom() functions. In the following example, the mulmodel2c <- mlogit `(healthre ~ 1 | educ + maritals + female + wrkfull, reflevel = 1, data = chp7)` command tells R to predict the nominal response variable `healthre` from the intercept and four independent variables. In the model formula for the `mlogit()` function, the dependent variable `healthre` and the intercept are separated by the tilde (~); `1` is the intercept, which is separated from the predictor variables by the vertical line (|); the four predictor variables include, `educ`, `maritals`, `female`, and `wrkfull` which are connected by plus (+) symbols. The `reflevel = 1` argument defines the reference level. We also specify the data arguments `data = chp7` in the function. The fitted model is named `mulmodel2c`. The following output is shown by the `summary(mulmodel2c)` command.

```
> # Using mlogit() in mlogit
> library(mlogit)
> chp7 <- mlogit.data(chp7.mul, choice="healthre", shape="wide")
> mulmodel2c <- mlogit(healthre ~ 1 | educ + maritals + female + wrkfull, reflevel = 1,
data=chp7)
> summary(mulmodel2c)

Call:
mlogit(formula = healthre ~ 1 | educ + maritals + female + wrkfull,
     data = chp7, reflevel = 1, method = "nr")

Frequencies of alternatives:
       1           2           3           4
0.063001    0.227977    0.487987    0.221036

nr method
6 iterations, 0h:0m:1s
g'(-H)^-1g = 4.62E-05
successive function values within tolerance limits

Coefficients :
                  Estimate   Std. Error   z-value     Pr(>|z|)
2:(intercept)     0.022702     0.459249    0.0494     0.960575
3:(intercept)    -1.199445     0.452359   -2.6515     0.008013 **
4:(intercept)    -3.637155     0.510906   -7.1190    1.087e-12 ***
2:educ            0.084208     0.034834    2.4174     0.015632 *
3:educ            0.194157     0.034029    5.7057    1.159e-08 ***
4:educ            0.315088     0.037290    8.4496    < 2.2e-16 ***
2:maritals        0.386250     0.229941    1.6798     0.093001 .
3:maritals        0.688490     0.219957    3.1301     0.001747 **
4:maritals        0.646024     0.234013    2.7606     0.005769 **
2:female         -0.021563     0.212826   -0.1013     0.919298
3:female          0.169787     0.204131    0.8318     0.405546
4:female          0.156590     0.219206    0.7144     0.475009
2:wrkfull         0.327136     0.238023    1.3744     0.169322
3:wrkfull         0.957817     0.226461    4.2295    2.342e-05 ***
4:wrkfull         0.822496     0.240184    3.4244     0.000616 ***

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
Log-Likelihood: -2137.6
McFadden R^2: 0.044945
Likelihood ratio test : chisq = 201.19 (p.value = < 2.22e-16)
```

The R output includes the call of the model command, the frequencies of alternatives, the estimation method, the coefficients section, the log likelihood value, the McFadden R^2, and the likelihood ratio test statistic. The call shows the R command for the model. The second section shows the Newton-Raphson method for maximum likelihood estimation. The third section shows the coefficients table including the parameter estimates for the intercept and the four predictor variables. Finally, the log likelihood value, the McFadden R^2, and the likelihood ratio test statistic are displayed at the end.

The coefficients section (labeled `Coefficients:`) displays the parameter estimates for the intercepts and the predictor variables, their standard errors, the Wald z statistics, and the associated p values. It displays the parameter estimates for the three binary logistic models comparing each category with the base category 1. For example, the three coefficients for `educ` are displayed as `2:educ`, `3:educ`, and `4:educ`, respectively.

We use the `coef(mulmodel2c)` command to extract the logit coefficients and use the `confint(mulmodel2c)` command to obtain the confidence intervals. The output is omitted here.

The odds ratios can be obtained with `exp(coef(mulmodel2c))`. We also use the `exp(confint(mulmodel2c))` command to obtain the corresponding confidence intervals. Both results are combined with the `cbind()` function.

```
> exp(coef(mulmodel2c))
2:(intercept)  3:(intercept)  4:(intercept)     2:educ      3:educ
   1.02296127     0.30136144     0.02632714  1.08785483  1.21428738
       4:educ     2:maritals     3:maritals  4:maritals    2:female
   1.37038032     1.47145243     1.99070700  1.90793964  0.97866759
     3:female       4:female      2:wrkfull   3:wrkfull   4:wrkfull
   1.18505292     1.16951655     1.38698996  2.60600205  2.27617406

> exp(confint(mulmodel2c))
                    2.5 %      97.5 %
 2:(intercept) 0.415858299 2.51636139
 3:(intercept) 0.124176395 0.73136860
 4:(intercept) 0.009672079 0.07166175
 2:educ        1.016061603 1.16472085
 3:educ        1.135942082 1.29803610
 4:educ        1.273795029 1.47428918
 2:maritals    0.937605127 2.30925811
 3:maritals    1.293540013 3.06361947
 4:maritals    1.206069519 3.01826190
 2:female      0.644877409 1.48522840
 3:female      0.794293723 1.76804925
 4:female      0.761058277 1.79719347
 2:wrkfull     0.869896566 2.21145965
 3:wrkfull     1.671902920 4.06198627
 4:wrkfull     1.421545886 3.64460156
```

```
> cbind(exp(coef(mulmodel2c)), exp(confint(mulmodel2c)))
                                       2.5 %      97.5 %
 2:(intercept)   1.02296127   0.415858299   2.51636139
 3:(intercept)   0.30136144   0.124176395   0.73136860
 4:(intercept)   0.02632714   0.009672079   0.07166175
 2:educ          1.08785483   1.016061603   1.16472085
 3:educ          1.21428738   1.135942082   1.29803610
 4:educ          1.37038032   1.273795029   1.47428918
 2:maritals      1.47145243   0.937605127   2.30925811
 3:maritals      1.99070700   1.293540013   3.06361947
 4:maritals      1.90793964   1.206069519   3.01826190
 2:female        0.97866759   0.644877409   1.48522840
 3:female        1.18505292   0.794293723   1.76804925
 4:female        1.16951655   0.761058277   1.79719347
 2:wrkfull       1.38698996   0.869896566   2.21145965
 3:wrkfull       2.60600205   1.671902920   4.06198627
 4:wrkfull       2.27617406   1.421545886   3.64460156
```

The `ggpredict()` function in the `ggeffects` package also works with the `mlogit()` function. The output is omitted here.

7.7 MAKING PUBLICATION-QUALITY TABLES

7.7.1 Presenting the Results of the vglm Models Using the texreg Package

The `stargazer()` function currently cannot directly produce the results table from the `vglm` models, so we use the `screenreg()` and `htmlreg()` functions from the `texreg` package (Leifeld, 2013). Since the package has been installed in preceding chapters, we only need to load the package by typing `library(texreg)`.

After we use the `vglm()` function to fit the single-predictor model `mulmodel1` and the multiple-predictor model `mulmodel2`, we create a table containing the results of the both model with the following command: `screenreg(list(mulmodel1, mulmodel2))`. In the `screenreg()` function, we specify the two model objects to be presented with the `list()` function. The output is a plain text table.

```
> # Presenting the results of the multinomial logistic Models using the texreg
package
> library(texreg)
Version:     1.37.5
Date:        2020-06-17
Author:      Philip Leifeld (University of Essex)

> screenreg(list(mulmodel1, mulmodel2))
```

```
==================================================
                    Model 1           Model 2
--------------------------------------------------
(Intercept):1         0.18              0.02
                     (0.43)            (0.46)
(Intercept):2        -0.78             -1.20 **
                     (0.42)            (0.45)
(Intercept):3        -3.28 ***         -3.64 ***
                     (0.48)            (0.51)
educ:1                0.09 **           0.08 *
                     (0.03)            (0.03)
educ:2                0.22 ***          0.19 ***
                     (0.03)            (0.03)
educ:3                0.34 ***          0.32 ***
                     (0.04)            (0.04)
maritals:1                              0.39
                                       (0.23)
maritals:2                              0.69 **
                                       (0.22)
maritals:3                              0.65 **
                                       (0.23)
female:1                               -0.02
                                       (0.21)
female:2                                0.17
                                       (0.20)
female:3                                0.16
                                       (0.22)
wrkfull:1                               0.33
                                       (0.24)
wrkfull:2                               0.96 ***
                                       (0.23)
wrkfull:3                               0.82 ***
                                       (0.24)
--------------------------------------------------
Log Likelihood     -2166.65          -2137.62
DF                  5613              5604
Num. obs.           5619              5619
==================================================
*** p < 0.001; ** p < 0.01; * p < 0.05
> htmlreg(list(mulmodel1, mulmodel2), file="chap7mul.doc", doctype=TRUE,
html.tag=TRUE, head.tag=TRUE)
The table was written to the file 'chap7mul.doc'.
```

We can also use the `htmlreg()` function to create a regression table for the estimated results and save it to a Microsoft Word file named `chap7mul.doc` with the following command: `htmlreg(list(mulmodel1, mulmodel2), file = "chap7mul.doc", doctype = TRUE, html.tag = TRUE, head.tag = TRUE)`. It automatically produces Table 7.3, as shown here in its original format, presenting the results of both the single-predictor and the multiple-predictor multinomial logistic regression models.

TABLE 7.3 ● Results of the Multinomial Logistic Regression Models: Single-Predictor Model and Multiple-Predictor Models (Shown in Original Format Generated by R)

	Model 1	Model 2
(Intercept):1	0.18	0.02
	(0.43)	(0.46)
(Intercept):2	−0.78	−1.20**
	(0.42)	(0.45)
(Intercept):3	−3.28***	−3.64***
	(0.48)	(0.51)
educ:1	0.09**	0.08*
	(0.03)	(0.03)
educ:2	0.22***	0.19***
educ:3	(0.03)	(0.03)
	0.34***	0.32***
maritals:1	(0.04)	(0.04)
		0.39
maritals:2		(0.23)
		0.69**
maritals:3		(0.22)
		0.65**
female:1		(0.23)
		−0.02
female:2		(0.21)
		0.17
female:3		(0.20)
		0.16
wrkfull:1		(0.22)
		0.33

Table 7.3 *(Continued)*

	Model 1	Model 2
wrkfull:2		(0.24)
		0.96***
wrkfull:3		(0.23)
		0.82***
Log Likelihood		(0.24)
	−2166.65	−2137.62
DF	5613	5604
Num. obs.	5619	5619

***$p < 0.001$
**$p < 0.01$
*$p < 0.05$

7.8 REPORTING THE RESULTS

Reporting the results for multinomial logistic regression is similar to that used for binary logistic regression. The following are the generic guidelines for reporting the results. You may need to adjust your writing since your discipline or journals may have different requirements.

First, describe the multinomial logistic regression model, the nominal response variable and the independent variables, and your research hypothesis or the purpose of your study. Include a couple of sentences explaining why this model is appropriate for the analysis.

Second, if available, report the likelihood ratio test statistic for the model and the associated p value, followed by the interpretation on whether the fitted model is better than the null model. If more than one model is developed, then compare models using likelihood ratio test statistics and/or the AIC and BIC statistics.

Third, report the parameter estimates for the predictor variables, their standard errors, the associated p values in a table. Since a multinomial logistic regression model includes $J - 1$ binary comparisons, label them in the table. If more than one model is fitted, then the results of all the competing models need to be presented in a table. In addition, report the odds ratios or relative risk ratios for each predictor in the table or text and interpret the results. The following is an example of summarizing the results for the multinomial logistic regression model illustrated previously.

The multinomial logistic regression analysis was conducted to predict the ordinal outcome variable, health status, from a set of predictor variables, such as marital status, years of education, gender, and working status. Although the multinomial logistic regression model is normally used to estimate the nominal response variables, it is an alternative to estimate ordinal response variables when the proportional odds assumption is violated.

The likelihood ratio test for the fitted model $\chi^2_{(12)} = 151.28$, $p < .001$, which indicated that the four-predictor model provided a better fit than the null model with no independent variables in predicting the logit of being in any other category of health status compared with being in the base category (i.e., poor health).

Table 7.2 displays the parameter estimates for the three binary logistic models comparing each category with the base category since the multinomial logistic regression model is treated as a series of binary logistic regression models. These three equations compare categories 2 with 1, 3 with 1, and 4 with 1, respectively.

For `educ`, the odds ratios of being in category 2 versus category 1, category 3 versus category 1, and category 4 versus category 1 are 1.088, 1.214, and 1.370, respectively. The results indicate that the odds of being in category 2 versus the base category, category 3 versus the base category, and category 4 versus the base category increase by a factor of 1.088, 1.214, and 1.370, respectively, for a one-unit increase in the `educ` predictor when holding all other predictors constant.

For `maritals`, the odds ratios for the three binary comparisons (i.e., categories 2 vs. 1, 3 vs. 1, and 4 vs. 1) are 1.471, 1.991, and 1.908, respectively. $OR(2,1) = 1.471$, $p > .05$, which indicates that there is no relationship between `maritals` and the odds of being in category 2 versus the base category 1. The odds ratios for the other two comparisons are significant, $OR(3,1) = 1.991$, $OR(4,1) = 1.908$, which indicates that the odds of being in category 3 versus the base category and category 4 versus the base category for the married are 1.991 and 1.908 times as large as the odds for the unmarried, respectively, when holding other predictors constant.

The odds ratios for `wrkfull` can be interpreted in the similar way. The odds ratios for three binary comparisons (i.e., categories 2 vs. 1, 3 vs. 1, and 4 vs. 1) are 1.387, 2.606, and 2.276, respectively. $OR(2,1) = 1.387$, $p > .05$, which indicates that there is no relationship between `wrkfull` and the odds of being in category 2 versus the base category 1. The odds ratios for the other two comparisons are significant. The odds of being in category 3 versus the base category and category 4 versus the base category for those working full time are 2.606 and 2.267 times as large as the odds for those not working full time, respectively, when holding the other predictors constant.

With regard to `female`, none of the odds ratios for the binary comparisons are significant. It indicates that being female does not impact the odds of being in any particular category versus the base category when holding other predictors constant.

7.9 SUMMARY OF R COMMANDS IN THIS CHAPTER

```
# Chap 7 R Script
# Remove all objects
rm(list = ls(all = TRUE))

# The following user-written packages need to be installed first by using install.packages(" ") and then
by loading it with library()
# library(VGAM)          # It is already installed for Chapter 4
# library(rcompanion)    # It is already installed for Chapter 3
# library(ggeffects)     # It is already installed for Chapter 2
# library(texreg)        # It is already installed for Chapter 4
# library(nnet)
# library(mlogit)

# Import the GSS 2016 data
library(foreign)
chp7.mul <- read.dta("C:/CDA/gss2016.dta")
chp7.mul$healthre <- factor(chp7.mul$healthre)
chp7.mul$educ <- as.numeric(chp7.mul$educ)
chp7.mul$wrkfull <- as.numeric(chp7.mul$wrkfull)
chp7.mul$maritals <- as.numeric(chp7.mul$maritals)
attach(chp7.mul)

# One-predictor multinomial logistic regression model with vglm() in VGAM
library(VGAM)
mulmodel1 <- vglm(healthre ~ educ, multinomial(refLevel = 1), data=chp7.mul)
summary(mulmodel1)
coef(mulmodel1, matrix = TRUE)
confint(mulmodel1, matrix = TRUE)
exp(coef(mulmodel1, matrix = TRUE))
exp(confint(mulmodel1, matrix = TRUE))
cbind(exp(coef(mulmodel1)), exp(confint(mulmodel1)))
```

```
# Testing the overall model using the likelihood ratio test
mulmodel0 <- vglm(healthre ~ 1, multinomial(refLevel = 1), data=chp7.mul)
summary(mulmodel0)
lrtest(mulmodel0, mulmodel1)

# Pseudo R2 with nagelkerke()
library(rcompanion)
nagelkerke(mulmodel1)

# Pseudo R2 with equations
LLM1 <- logLik(mulmodel1)
LL0 <- logLik(mulmodel0)
McFadden1 <- 1-(LLM1/LL0)
McFadden1
CS1 <- 1-exp(2*(LL0-LLM1)/1873)
CS1
NG1 <- CS1/(1-exp(2*LL0/1873))
NG1

# AIC and BIC Statistics
AIC(mulmodel1)
BIC(mulmodel1)

# Multiple-predictor multinomial logistic regression model with vglm() in VGAM
mulmodel2 <- vglm(healthre ~ educ + maritals + female + wrkfull, multinomial(refLevel = 1),
data=chp7.mul)
summary(mulmodel2)
coef(mulmodel2, matrix = TRUE)
confint(mulmodel2, matrix = TRUE)
exp(coef(mulmodel2, matrix = TRUE))
exp(confint(mulmodel2, matrix = TRUE))
cbind(exp(coef(mulmodel2)), exp(confint(mulmodel2)))

# Testing the overall model using the likelihood ratio test
lrtest(mulmodel0, mulmodel2)

# Pseudo R2 with nagelkerke()
nagelkerke(mulmodel2)

# Pseudo R2 with equations
LLM2 <- logLik(mulmodel2)
McFadden2 <- 1-(LLM2/LL0)
McFadden2
CS2 <- 1-exp(2*(LL0-LLM2)/1873)
CS2
NG2 <- CS2/(1-exp(2*LL0/1873))
NG2

# AIC and BIC Statistics
AIC(mulmodel2)
BIC(mulmodel2)
```

```
# Model comparison with the likelihood ratio test
lrtest(mulmodel1, mulmodel2)

# Predicted probabilities with ggpredict() in ggeffects
library(ggeffects)
pr.mul2m <- ggpredict(mulmodel2, terms="educ[12, 14, 16]", ci=NA)
pr.mul2m
plot(pr.mul2m)

pr.mul2 <- ggpredict(mulmodel2, terms=c("educ[12, 14, 16]", "maritals"), ci=NA)
pr.mul2
plot(pr.mul2)

# Presenting the results of the multinomial logistic Models using the texreg package
library(texreg)
screenreg(list(mulmodel1, mulmodel2))
htmlreg(list(mulmodel1, mulmodel2), file="chap7mul.doc", doctype=TRUE, html.tag=TRUE, head.tag=TRUE)

# Using multinom() in nnet
library(nnet)
mulmodel2b <- multinom(healthre ~ educ + maritals + female + wrkfull, data=chp7.mul)
summary(mulmodel2b)
coef(mulmodel2b)
z <- summary(mulmodel2b)$coefficients/summary(mulmodel2b)$standard.errors
z
p <- (1-pnorm(abs(z),0,1))*2
p
exp(coef(mulmodel2b))
exp(confint(mulmodel2b))

# Using mlogit() in mlogit
library(mlogit)
chp7 <- mlogit.data(chp7.mul, choice="healthre", shape="wide")
mulmodel2c <- mlogit(healthre ~ 1 | educ + maritals + female + wrkfull, reflevel = 1, data=chp7)
summary(mulmodel2c)
coef(mulmodel2c)
confint(mulmodel2c)
exp(coef(mulmodel2c))
exp(confint(mulmodel2c))
cbind(exp(coef(mulmodel2c)), exp(confint(mulmodel2c)))

detach(chp7.mul)
```

Glossary

The multinomial distribution is an extension of the binomial distribution when the discrete random variable is a nominal variable with more than two categories.

The multinomial logistic regression model is used to estimate nominal response variables with multiple unordered categories. This model is a generalization of binary logistic regression when there are more than two categories in a response variable.

The odds in the multinomial logistic model can be defined as the ratio of the probability of being in a particular category to the probability of being in the base category.

Exercises

Use the GSS 2016 data available at **https://edge.sagepub.com/liu1e** for the following problems.

1. Conduct an analysis for a multinomial logistic regression model and estimate the ordinal response variable `happy` from the four predictor variables, `sex`, `educ`, and `satfin`. Choose category 3 (i.e., not too happy) as the referent category.
2. Interpret the likelihood ratio test for the overall model.
3. In the regression table, identify the logit coefficients for the predictor variable `satfin` across two binary comparisons. Are they both statistically significant? What categories are they comparing?
4. Compute the odds ratios for the predictor variable `satfin` across two binary comparisons.
5. Interpret the relative risk ratios/odd ratios of `satfin`.
6. Based on the parameter estimates in the output, write the two equations for the model.
7. Make a publication-quality table containing the estimated logit.
8. Write a report to summarize the results from the output.

8

POISSON REGRESSION MODELS

OBJECTIVES OF THIS CHAPTER

This chapter introduces Poisson regression models. It first starts with an introduction to the Poisson regression model followed by a discussion of the incidence rates and incidence rate ratios in the model, goodness-of-fit statistics, and how to interpret parameter estimates. After a description of the research example, the data, and the sample, a one-predictor Poisson regression model and a multiple-predictor Poisson regression model are illustrated with the `glm()` function in R. The `vglm()` function in the `VGAM` package is also used to fit the multiple-predictor model. R commands and output are explained in detail. This chapter focuses on fitting the Poisson regression models with R, as well as on interpreting and presenting the results. After reading this chapter, you should be able to:

- Identify when Poisson regression models are used.
- Fit a Poisson regression model using R.
- Interpret the output.
- Interpret the incidence rate ratios and marginal effects.
- Compute, plot, and interpret the predicted counts.
- Compare models using the likelihood ratio test.
- Present results in publication-quality tables using R.
- Write the results for publication.