# Multinomial Logistic Regression with R

- The ratio of the probability of choosing one outcome category over the probability of choosing the baseline category is often referred as **relative risk**
- It is also sometimes referred as **odds**

# Multinomial Logistic Regression with R

- 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 (next slide).

# Multinomial Logistic Regression with R

*Extract the coefficients from the model then 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
```

# Multinomial Regression with R

- The relative risk ratio for a one-unit increase in the variable **write** is 0.9437 for being in general program vs. academic program.
- The relative risk ratio switching from **ses** $= 1$ to 3 is 0.3126 for being in general program vs. academic program.

## Multinomial Regression with R

- 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

# Multinomial Regression with R

```
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
```

## Prediction

- Suppose 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**.
- Three Cases

# Multinomial Logistic Regression with R

```
dses <- data.frame(ses = c("low", "middle", "h
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
```

# Multinomial Logistic Regression with R

- 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**.

# Multinomial Logistic Regression with R

Store the predicted probabilities for each value of ses and write

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

```
pp.write <- cbind(dwrite,
     predict(test, newdata = dwrite,
     type = "probs", se = TRUE))
```

# Multinomial Logistic Regression with R

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
```

# Multinomial Regression with R

- 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.

# Multinomial Logistic Regression with R

remark: 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, co
    ., 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.

# Multinomial Regression with R

**Things to consider**

- 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.

# Multinomial Regression with R

## Things to consider

- 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.

**Things to consider**

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

## Things to consider

- 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.

**Things to consider**

- 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.