

Ordered logistic regression R

Turning our attention to the predictions with public as a predictor variable, we see that when public is set to "no" the difference in predictions for apply greater than or equal to two, versus apply greater than or equal to three is about 2.14 ($-0.204 - -2.345 = 2.141$). When public is set to "yes" the difference between the coefficients is about 1.37 ($-0.175 - -1.547 = 1.372$).

Ordered logistic regression R

The differences in the distance between the two sets of coefficients (2.14 vs. 1.37) may suggest that the parallel slopes assumption does not hold for the predictor public. That would indicate that the effect of attending a public versus private school is different for the transition from "unlikely" to "somewhat likely" and "somewhat likely" to "very likely."

Ordered logistic regression R

The plot command below tells R that the object we wish to plot is `s`. The command `which=1:3` is a list of values indicating levels of `y` should be included in the plot. If your dependent variable had more than three levels you would need to change the 3 to the number of categories (e.g. 4 for a four category variable, even if it is numbered 0, 1, 2, 3). The command `pch=1:3` selects the markers to use, and is optional, as are `xlab='logit'` which labels the x-axis, and `main=' '` which sets the main label for the graph to blank.

Ordered logistic regression R

If the proportional odds assumption holds, for each predictor variable, distance between the symbols for each set of categories of the dependent variable, should remain similar. To help demonstrate this, we normalized all the first set of coefficients to be zero so there is a common reference point.

Ordered logistic regression R

Looking at the coefficients for the variable pared we see that the distance between the two sets of coefficients is similar. In contrast, the distances between the estimates for public are different (i.e. the markers are much further apart on the second line than on the first), suggesting that the proportional odds assumption may not hold.

Ordered logistic regression R

Ordered logistic regression R

```
plot(s, which=1:3, pch=1:3, xlab='logit', ma
```

Ordered logistic regression R

Plot viewing proportional odds assumption

Once we are done assessing whether the assumptions of our model hold, we can obtain predicted probabilities, which are usually easier to understand than either the coefficients or the odds ratios. For example, we can vary gpa for each level of pared and public and calculate the probability of being in each category of apply. We do this by creating a new dataset of all the values to use for prediction.

Ordered logistic regression R

```
newdat <- data.frame(  
  pared = rep(0:1, 200),  
  public = rep(0:1, each = 200),  
  gpa = rep(seq(from = 1.9, to = 4, length.o  
  
newdat <- cbind(newdat, predict(m, newdat, t
```

Ordered logistic regression R

```
##show first few rows
```

```
head(newdat)
```

##	pared	public	gpa	unlikely	somewhat	li
## 1	0	0	1.900	0.7376		0.
## 2	1	0	1.921	0.4932		0.
## 3	0	0	1.942	0.7325		0.
## 4	1	0	1.964	0.4867		0.
## 5	0	0	1.985	0.7274		0.
## 6	1	0	2.006	0.4802		0.

Ordered logistic regression R

Now we can reshape the data long with the `reshape2` package and plot all of the predicted probabilities for the different conditions. We plot the predicted probabilities, connected with a line, coloured by level of the outcome, apply, and faceted by level of `pared` and `public`. We also use a custom label function, to add clearer labels showing what each column and row of the plot represent.

Ordered logistic regression R

```
lnewdat <- melt(newdat, id.vars = c("pared",
  variable.name = "Level", value.name="Probabilit
### view first few rows
head(lnewdat)
###      pared public    gpa    Level Probabilit
### 1         0        0 1.900 unlikely      0.737
### 2         1        0 1.921 unlikely      0.493
### 3         0        0 1.942 unlikely      0.732
### 4         1        0 1.964 unlikely      0.486
### 5         0        0 1.985 unlikely      0.727
### 6         1        0 2.006 unlikely      0.480
```

Ordered logistic regression R

```
ggplot(lnewdat, aes(x = gpa, y = Probability
  geom_line() +
  facet_grid(pared ~ public, scales="free",
    labeller=function(x, y) sprintf("%s = %d
```

Ordered logistic regression R

Plot of predicted probabilities for each group Things to consider Perfect prediction: Perfect prediction means that one value of a predictor variable is associated with only one value of the response variable. If this happens, Stata will usually issue a note at the top of the output and will drop the cases so that the model can run.

Ordered logistic regression R

Sample size: Both ordered logistic and ordered probit, using maximum likelihood estimates, require sufficient sample size. How big is big is a topic of some debate, but they almost always require more cases than OLS regression. Empty cells or small cells: You should check for empty or small cells by doing a crosstab between categorical predictors and the outcome variable. If a cell has very few cases, the model may become unstable or it might not run at all.

Ordered logistic regression R

Pseudo-R-squared: There is no exact analog of the R-squared found in OLS. There are many versions of pseudo-R-squares. Please see Long and Freese 2005 for more details and explanations of various pseudo-R-squares. **Diagnostics:** Doing diagnostics for non-linear models is difficult, and ordered logit/probit models are even more difficult than binary models.