Ordered logistic regression with R

Ordered logit model

- The Ordered (or Ordinal) logit model (also ordered logistic regression or proportional odds model), is a regression model for ordinal dependent variables.
- For example, questions on a survey answered by a choice among "poor", "fair", "good", "very good", and "excellent".
- The purpose of the analysis is to see how well that response can be predicted by the responses to other questions, some of which may be quantitative

Ordered logistic regression with R

Ordered logit model

-
- It can be thought of as an extension of the logistic regression model that applies to dichotomous dependent variables, allowing for more than two (ordered) response categories.
- The model only applies to data that meet the proportional odds assumption

Ordered logistic regression with R

polr

- In this section we will use the `polr` command (from the MASS package) to estimate an ordered logistic regression model.
- The command name comes from **proportional odds logistic regression**, due to the the **proportional odds assumption** in the model.

Ordered logistic regression with R

`polr`

- `polr` uses the standard formula interface in R for specifying a regression model with outcome followed by predictors.
- We will also specify `Hess=TRUE` to have the model return the observed information matrix from optimization (called the Hessian) which is used to get standard errors.

Ordered logistic regression with R

```
## fit ordered logit model and store results 'm'
m <- polr(apply ~ pared +
          public + gpa, data = dat, Hess=TRUE)

## view a summary of the model
summary(m)
## Call:
## polr(formula = apply ~ pared +
             public + gpa, data = dat,
             Hess = TRUE)
```

Ordered logistic regression with R

```
Coefficients:
          Value Std. Error t value
pared   1.0477      0.266   3.942
public -0.0588      0.298  -0.197
gpa     0.6159      0.261   2.363

Intercepts:
                            Value  Std. Error t value
unlikely|somewhat likely     2.204  0.780       2.827
somewhat likely|very likely  4.299  0.804       5.345
```

Ordered logistic regression with R

1. The "Call", what type of model we ran, what options we specified, etc.
2. The usual regression output coefficient table including the value of each coefficient, standard errors, and $t-$value, which is simply the ratio of the coefficient to its standard error.
   (Remark: There is no significance test by default.)

Ordered logistic regression with R

3. We then have the estimates for the two intercepts (which are sometimes called **cutpoints**).
4. The intercepts indicate where the latent variable is cut to make the three groups that we observe in our data.

Ordered logistic regression with R

In the ordered logit model, there is an observed ordinal variable, Y. Y, in turn, is a function of another latent variable, Y*, that is not measured.

a. In the ordered logit model, there is a continuous, unmeasured latent variable Y*, whose values determine what the observed ordinal variable Y equals.
b. The continuous latent variable Y* has various threshold (or cutoff) points.

Ordered logistic regression with R

Your value on the observed ordinal variable Y depends on whether or not you have crossed a particular threshold. For example, when $M = 3$

- $Yi = 1$ if Y^*i is $\leq$ CP1
- $Yi = 2$ if CP1 $\leq Y^*i \leq$ CP2
- $Yi = 3$ id $Y^*i \geq$ CP2

Ordered logistic regression with R

- Note that this latent variable is continuous. In general, these are not used in the interpretation of the results.
- The cutpoints are closely related to thresholds, which are reported by other statistical packages.

Model Diagnostics

- We see the residual deviance, -2 * Log Likelihood of the model as well as the AIC.
- Both the deviance and AIC are useful for model comparison.
- Of, course, some people are not satisfied without a $p-$value.
- One way to calculate a $p-$value in this case is by comparing the $t-$value against the standard normal distribution, like a $z-$test.

Ordered logistic regression R

- Of course this is only true with infinite degrees of freedom, but is reasonably approximated by large samples, becoming increasingly biased as sample size decreases.
- First we store the coefficient table, then calculate the $p-$values and combine back with the table.

Ordered logistic regression R

```
# store table
(ctable <- coef(summary(m)))
                              Value Std. Error t value
 pared                      1.04769     0.2658  3.9418
 public                    -0.05879     0.2979 -0.1974
 gpa                        0.61594     0.2606  2.3632
 unlikely|somewhat likely   2.20391     0.7795  2.8272
 somewhat likely|very likely 4.29936     0.8043  5.3453
```

Ordered Logistic regression R

```
# calculate and store p values
p <- pnorm(abs(ctable[, "t value"]),
      lower.tail = FALSE) * 2

# Combined table
(ctable <- cbind(ctable, "p value" = p))
                     Value Std. Error t value   p value
 pared             1.04769     0.2658  3.9418 8.087e-05
 public           -0.05879     0.2979 -0.1974 8.435e-01
 gpa               0.61594     0.2606  2.3632 1.812e-02
 unli..|some..     2.20391     0.7795  2.8272 4.696e-03
 some..|very..     4.29936     0.8043  5.3453 9.027e-08
```