

Ordered logistic regression 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 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 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 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 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 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 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.

Ordered logistic regression R

- ▶ Finally, we see the residual deviance, $-2 * \text{Log Likelihood}$ of the model as well as the AIC.
- ▶ Both the deviance and AIC are useful for model comparison.
- ▶ 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  
(cetable <- 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