Ordered logits - models for ordered categorical outcomes

Silje Synnøve Lyder Hermansen

2026-04-19

Introduction

Today: Ordinal variables

Strategies when our outcome variable is categorical

- categorical (e.g. party, profession,. . .) $\rightarrow$ *multinomial regression*
- ordinal (e.g. attitudes towards topics. . .) $\rightarrow$ *ordinal regression*

$\Rightarrow$ *Models of choice where we model the chooser's characteristics*

What's the problem?

What is an ordered variable?

A ranked variable with unknown distance between categories.

- Often the result of binning: Close connection to latent formulation.
- We can choose how to treat it: As linear, categorical or **ordinal**.

The ordered logistic regression

- keeps the information on ordering
- but estimates the distance between categories from data

Discuss with your neighbour

What kind of ordered variables do political scientists run into?

Relationship with other models

- multinomial model: each category has its own set of regression coefficients
  - ordered model: a single set of regression coefficients
- linear models: dependent variable is continuous (distance between values is constant)
  - ordered model: latent dependent variable is continuous

Two conceptions of ordered logisitc regression

Two conceptions of ordered logisitc regression

There are two ways of understanding the ordered logit:

- Parallel regressions: useful for understanding estimation and model assessment.
- Latent variable approach: useful for interpretation.

Parallel regressions

Estimation

The model is estimated by changing the reference level in a series of parallel regression

- binomial logits comparing being below or above threshold 1, then 2...
- reported regression coefficients are "averaged over" from these models

⇒ *implications for interpretation and main model assumption*

Latent variable approach: cutpoints

Latent variable: chooser appreciation

Close to political science theories about choice utility

- unobserved: captures the chooser's (continuous) appreciation (utility) of different options
- observed: translated to choice categories

$\Rightarrow$ *a continuous, unobserved space that is sliced up by cutpoints*

Cutpoints

We rely on cutpoints to slice up the latent variable and determine outcomes

- **Binomial logistic:** One cutpoint. $\rightarrow$ Rarely estimated.
- **Ordinal logistic:** Serveral cutpoints. $\rightarrow$ Explicit.

$\Rightarrow$ *Model estimates both regression parameters (β) and cutpoints (τ).*

A series of cutpoints

You are in the category m **when the latent variable is between its two cutpoints:** $\tau_{m-1} < y^{\star} < \tau_m$

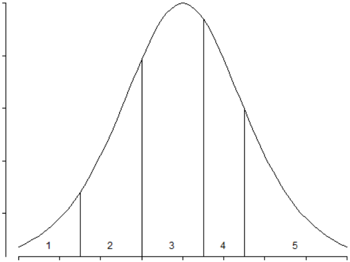


Figure 1: Slicing up a latent variable

Interpretation

Interpretation

Two types of interpretations

- **marginal effect:** change in probability of passing a threshold
  - cutpoints: function as “intercept” / “threshold”
  - slope: probability of passing any threshold (from one to next)
- **predicted outcomes:** probability / category
  - probability of being below / above a certain thresholds (cumulative probability)
  - probability of being between two thresholds (in a category)

The regression coefficients

The model calculates the odds of being lower than τ_m

- The first cutpoint (τ_0) is 0 ($-inf$): you cant be lower than the lowest.
- The last cutpoint is 1 ($+inf$): all observations are in some category.
- You end up with $m-1$ cutpoints.

The regression output

The regression output reports both β and τ

- **Regression coefficient** β is reported in relation to *upper* cutpoint of the category: $\tau_m - \beta x_i$
- **Cutpoints** serve also as intercepts.

The predicted value

The predicted probability of being in category m:

$$Pr(y_i = m) = \frac{exp(\tau_m - \beta x_i)}{1 + exp(\tau_m - \beta x_i)} - \frac{exp(\tau_{m-1} - \beta x_i)}{1 + exp(\tau_{m-1} - \beta x_i)} \quad (1)$$

An example: Attitudes towards redistribution

An example:

ESS respondents (that voted V or DF) are asked to what extent they believe the state should engage in redistribution (1 = disagree; 5 = agree).

```
#Load in data
download.file(
  url("https://siljehermansen.github.io/teaching/beyond-linear-models/kap10
  destfile = "kap10.rda"
)
df <- kap10

#Check distribution
barplot(table(df$Udjaevn))
```

An example:

ESS respondents (that voted V or DF) are asked to what extent they believe the state should engage in redistribution (1 = disagree; 5 = agree).

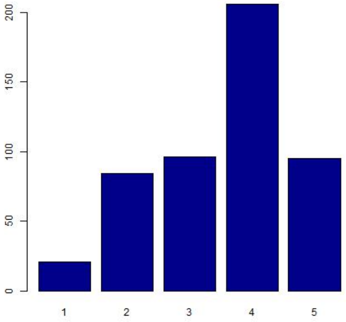


Figure 2: Attitudes towards redistribution is an ordered variable

Attitudes towards redistribution as a function of income

```
#Library for ordinal regression
library(MASS)
#Recode into ordered factor
df$Udjaevn.ord <- as.ordered(as.factor(df$Udjaevn))
#Run regression
mod.ord <- polr(Udjaevn.ord ~ Indtaegt,
                df,
                method = "logistic",
                Hess = TRUE)
summary(mod.ord)
```

Attitudes towards redistribution as a function of income

In R, we first pre-process (code category as ordered), then estimate the model.

```
## Call:
## polr(formula = Udjaevn.ord ~ Indtaegt, data = df, Hess = TRUE,
##     method = "logistic")
##
## Coefficients:
##           Value Std. Error t value
## Indtaegt 0.1153    0.03155   3.653
##
## Intercepts:
##     Value   Std. Error t value
## 1|2 -2.4186  0.2903    -8.3306
## 2|3 -0.6008  0.2179    -2.7566
## 3|4  0.3069  0.2150     1.4277
## 4|5  2.2276  0.2403     9.2686
##
## Residual Deviance: 1298.396
## AIC: 1308.396
## (51 observations deleted due to missingness)
```

We learn two things from the regression output

Regression coefficient reports effect of x on probability to be placed one category higher

- Effect in logodds: `0.115`
- We can backtransform to one unit increase in x: $(exp(\beta) - 1) \times 100$ $= 12\%$ increase in likelihood of a higher category.

$\Rightarrow$ *Hypothesis testing as in a binomial logit*

We learn two things from the regression output

We have one intercept per cutpoint

- e.g.: intercept of passing from 1 to 2 is `-2.419`
- e.g.: intercept is reported as significant (with standard errors)

$\Rightarrow$ *The model does a fair job in distinguishing between categories.*

Predicted scenarios

We interpret predicted probability by choosing one level of x and one category (two cutpoints) of y: What is the probability of m?

$$Pr(y_i = m) = \frac{exp(\tau_m - \beta x_i)}{1 + exp(\tau_m - \beta x_i)} - \frac{exp(\tau_{m-1} - \beta x_i)}{1 + exp(\tau_{m-1} - \beta x_i)} \quad (2)$$

Example

Let's choose low-income respondents (x = 1) and category 3 (diff between cutpoints 2 and 3)

```
z = mod.ord$zeta
x = 1

logodds1 <- z[3] - coefficients(mod.ord) * x
logodds2 <- z[3-1] - coefficients(mod.ord) * x
## Probabilities
p1 <- exp(logodds1)/(1 + exp(logodds1)) #3|4 or lower
p2 <- exp(logodds2)/(1 + exp(logodds2)) #2|3 or lower
## Difference between cutpoints
p1 - p2 #cat 3
```

```
##       3|4
## 0.2194954
```

An example

Predicted proportion in category

```
paste(round((p1-p2)*100),
"% of low-income respondents are predicted to answer x = 3 ('neutral')." )
```

[1] "22 % of low-income respondents are predicted to answer x = 3 ('neutral')."

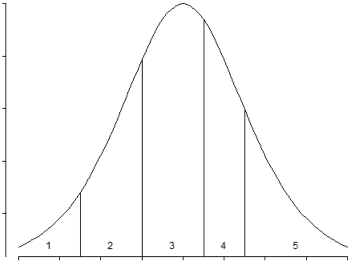
Cumulative probability

```
paste(round((p1)*100),
"% of low-income respondents are predicted to answer x = 3 ('neutral') or l
```

[1] "55 % of low-income respondents are predicted to answer x = 3 ('neutral') or lower to the question of whether they support redistribution."

Two ways of viewing the slicing

We can report the probability (e.g. `0.22`) of ending up between two cutpoints, or the *cumulative* probability (e.g. `0.55`) to be below each



point.

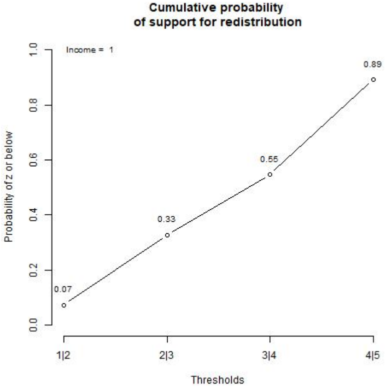
Changing scenarios

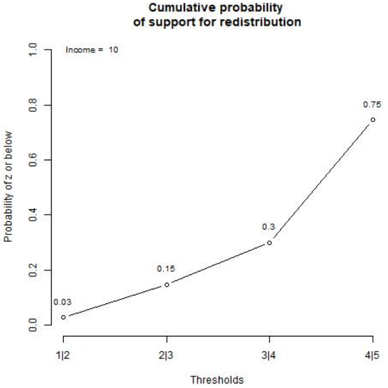
Example of how cumulative probability changes for different respondents (scenarios)

- Income is low ($x = 1$)
- Income is high ($x = 10$)

$\Rightarrow$ *thresholds / intercepts / cutpoints change*

Result


Cumulative probability
of support for redistribution
Income = 1
Probability of z or below
Thresholds
1|2
2|3
3|4
4|5
0.0
0.2
0.4
0.6
0.8
1.0
0.07
0.33
0.55
0.89


Cumulative probability
of support for redistribution
Income = 10
Probability of z or below
Thresholds
1|2
2|3
3|4
4|5
0.0
0.2
0.4
0.6
0.8
1.0
0.03
0.15
0.3
0.75

Parallel regressions approach: for assessment

Parallel regressions approach

The parallel regression approach is useful to understand how the model is estimated

- The y is recoded into $m-1$ dummy variables indicating if $y \leq m$
- Run a series of regressions where all β are fixed (i.e.: the same).

$\Rightarrow$ *This is also useful when we assess the model*

How good is my model?

The basic assumption

The basic assumption is that all parallel regressions have (about) the same regression coefficient

- Check the mean of the predictor for each value of y. Does it trend?

```
df %>%
  filter(!is.na(Udjaevn)) %>%
  group_by(Udjaevn) %>%
  reframe(mean(Indtaegt, na.rm = T))
```

```
## # A tibble: 5 x 2
##   Udjaevn `mean(Indtaegt, na.rm = T)`
##     <dbl>                       <dbl>
## 1       1                        4.8
## 2       2                        5.58
## 3       3                        5.96
## 4       4                        6.41
## 5       5                        6.75
```

- Run parallel regressions without constraint on β. Are they similar?

An example of parallel regressions

Recode into dummies

The dummies flag cases below a cumulative threshold of *outcomes*

```
##
df$ut1 <- ifelse(df$Udjaevn > 1, 1 , 0) #2 or above
df$ut2 <- ifelse(df$Udjaevn > 2, 1 , 0) #3 or above
df$ut3 <- ifelse(df$Udjaevn > 3, 1 , 0) #4 or above
df$ut4 <- ifelse(df$Udjaevn > 4, 1 , 0) #5
```

$\Rightarrow$ *The model then runs 4 regressions where* β *reports an aggregated value from all 4 coefficients (think: weigted mean).*

Run four regressions

Let's examplify with the parallel regressions without fixed β:

```
##Parallel regressions:
mod1 <- glm(ut1 ~ Indtaegt, df, family = "binomial")
mod2 <- glm(ut2 ~ Indtaegt, df, family = "binomial")
mod3 <- glm(ut3 ~ Indtaegt, df, family = "binomial")
mod4 <- glm(ut4 ~ Indtaegt, df, family = "binomial")
```

Compare coefficients from four regressions

```
##
## ==========================================================
##                          Dependent variable:
##                     --------------------------------------
##                       ut1      ut2      ut3       ut4
##                       (1)      (2)      (3)       (4)
## ----------------------------------------------------------
## Indtaegt            0.189**  0.125*** 0.110***  0.094**
##                     (0.085)  (0.041)  (0.035)   (0.045)
##
## Constant            2.048*** 0.552**   -0.270  -2.082***
##                     (0.474)  (0.260)  (0.231)   (0.319)
##
## ----------------------------------------------------------
## Observations          459      459      459       459
## Log Likelihood      -79.653  -234.669 -303.983 -217.674
## Akaike Inf. Crit.   163.306  473.338  611.967   439.348
## ==========================================================
## Note:                        *p<0.1; **p<0.05; ***p<0.01
```

Coefficient should be a weighted average from four regressions

These βs are weighted by the number of observations in each category:

```
table(df$Udjaevn)
```

```
##
##   1   2   3   4   5
##  21  84  96 206  95
```

We can plot the βs for comparison:

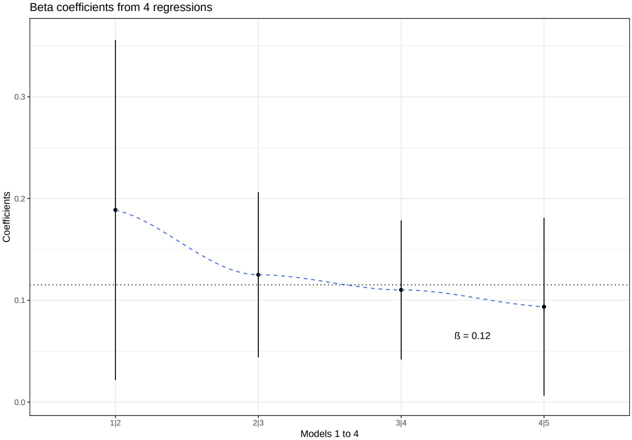
```
slope <- rbind(summary(mod1)$coefficients[2, c(1)],
               summary(mod2)$coefficients[2, c(1)],
               summary(mod3)$coefficients[2, c(1)],
               summary(mod4)$coefficients[2, c(1)])
se.slope <- rbind(summary(mod1)$coefficients[2, c(2)],
               summary(mod2)$coefficients[2, c(2)],
               summary(mod3)$coefficients[2, c(2)],
               summary(mod4)$coefficients[2, c(2)])
threshold <- c("1|2","2|3","3|4","4|5")
```

We can plot the βs for comparison:

```
data.frame(slope,
           se.slope,
           threshold) %>%
  ggplot +
  geom_point(aes(x = threshold,
                 y = slope)) +
  geom_errorbar(aes(x = threshold,
                    ymax = slope + 1.96 * se.slope,
                    ymin = slope - 1.96 * se.slope),
                width = 0) +
  geom_hline(yintercept = mod.ord$coefficients,
             lty = 3) +
  geom_text(aes(y = mod.ord$coefficients-0.05,
                x = 3.5,
                label = paste("\u03b2 =", round(mod.ord$coefficients,2))
                ),
            parse = F) +
  labs(title = "Slope coefficients from 4 regressions") +
  ylab("Coefficients") +
  xlab("Models 1 to 4")
```

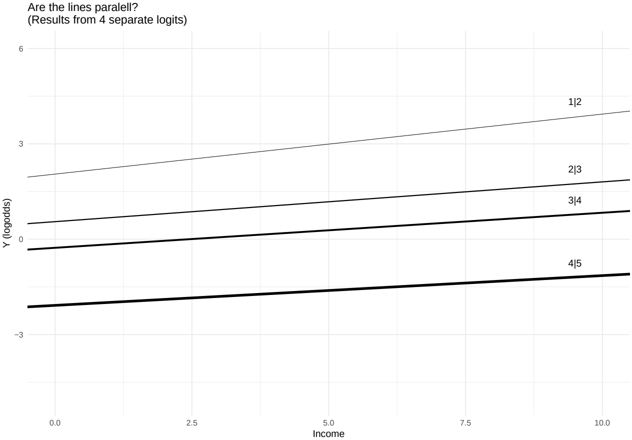
We can plot the βs for comparison:

The overall β is `0.12`. If the ordered model describes the data well, then all the unconstrained βs should resemble that description.



A visual inspection

A more visual way of checking the “parallel lines assumption” is to inspect if the regression lines are parallel.



When is it smart to run an ordered logit?

- You have few ordered categories
- The effect is approximately the same across the categories (parallel lines assumption)

What do I do if the assumption doesn't hold?

- Run an OLS/linear model:
  - if you have many categories
  - fairly equal spread of observations between categories
- Run a multinomial model:
  - i.e. estimate different β for each regression/threshold
- Run a binomial logistic model:
  - if the intermediate categories behave like a 0 or 1
  - i.e. the threshold / cutpoint are very imprecise
  - collapse cateogries

Main takeaways

Main takeaways

- model choice
  - dependent variable is ordered: ordered logistic regression
  - predictors relate to chooser
- two concepts:
  - latent variable approach: chooser's choice utility
  - estimation / comparison game: a set of parallel regressions

Specifics

- logistic regression: logodds
- cutpoints / intercepts
  - latent space is sliced up by *estimated* cutpoints
  - we use them as intercepts
- interpretation:
  - marginal effects: β reports probability of being below cutpoint (-β, not β)
  - predicted outcomes:
    - probability of falling *below* / *above* cutpoint
    - probability of falling *within* cutpoint
- main assumption:
  - parallel lines : effect of β is same for all cutpoints