Title

Intro 6 — Models for rank-ordered alternatives

Description Remarks and examples References Also see

Description

This introduction covers the commands cmroprobit and cmrologit. These estimation commands each fit choice models for rank-ordered alternatives. That is, models in which each decision maker ranks alternatives from a finite set of available alternatives.

Remarks and examples

stata.com

Remarks are presented under the following headings:

Overview of CM commands for rank-ordered alternatives

Stata has two commands designed for fitting choice models for rank-ordered alternatives. Below, we give you a brief overview of the models fit by these commands.

cmroprobit fits an extension of the multinomial probit choice model for rank-ordered alternatives. It allows both alternative-specific and case-specific predictors. It does not assume IIA; instead, it models the correlation of errors across alternatives. If you are not familiar with IIA, see *Overview of CM commands for discrete choices* in [CM] **Intro 5** and see [CM] **Intro 8**. cmroprobit allows tied ranks, but computation time increases with the number of ties, so in practice, it works best when there are only a small number of ties.

cmrologit fits a choice model for rank-ordered alternatives for the case in which alternatives are not explicitly identified. That is, there is no variable specifying the alternatives. Alternatives are known only by their characteristics as given by a set of alternative-specific variables. All predictors must be alternative-specific variables. This model assumes IIA is true. It allows tied ranks. It also allows a subset of alternatives to be evaluated but not given a rank—equivalent to making them tied for the lowest possible rank.

cmroprobit: Probit regression for rank-ordered alternatives

cmroprobit is similar to cmmprobit. Both are probit regression models, and both likelihoods are computed using simulated integration. Covariance structures are specified in exactly the same manner in each of the commands. Indeed, every option for cmmprobit works with cmroprobit and does the same thing. The difference is, of course, that cmroprobit has ranked alternatives as outcomes and cmmprobit requires a single choice of one alternative.

cmmprobit is described in the introduction [CM] **Intro 5** and in its manual entry [CM] **cmmprobit**. You should read the discussion of cmmprobit in the earlier introduction if you have not already done so.

Just like cmmprobit, cmroprobit models the random-error term of the utility using a multivariate normal distribution and allows the user to specify many different parameterizations for the covariance matrix.

Here is an example using cmroprobit. We use data from the Wisconsin Longitudinal Study cited by Long and Freese (2014, 477). This is a study involving high school graduates who were asked to rank their preferences of four job characteristics: esteem, variety, autonomy, and security.

The case-specific covariates are female (1 if female and 0 if male) and score, a score on a general mental ability test measured in standard deviations. The dataset also includes variables high and low, which indicate whether the respondent's current job is high or low in esteem, variety, autonomy, and security. These two variables together yield three possible ratings for the characteristics of his or her current job—high, low, or neither. From the high and low variables, we create a new variable currentjob that we include as an alternative-specific variable in our model.

The alternatives were ranked (variable rank) such that 1 is the most preferred alternative and 4 is the least, and respondents were allowed to have ties in their rankings. A variable noties indicates those persons who did not have any ties in their rankings.

Here we prepare our data and list them for three respondents:

```
. use https://www.stata-press.com/data/r17/wlsrank
(1992 Wisconsin Longitudinal Study data on job values)

. keep if noties
(11,244 observations deleted)

. generate currentjob = 1 if low==1
(1,304 missing values generated)

. replace currentjob = 2 if low==0 & high==0
(805 real changes made)

. replace currentjob = 3 if high==1
(499 real changes made)

. label define current 1 "Low" 2 "Neither" 3 "High"

. label values currentjob current

. list id jobchar rank female score currentjob in 1/12, sepby(id)
```

	id	jobchar	rank	female	score	curren~b
1.	13	Esteem	4	Male	.3246512	Low
2.	13	Variety	2	Male	.3246512	High
3.	13	Autonomy	1	Male	.3246512	Neither
4.	13	Security	3	Male	.3246512	Low
5.	19	Esteem	3	Female	.0492111	Neither
6.	19	Variety	2	Female	.0492111	Neither
7.	19	Autonomy	4	Female	.0492111	Neither
8.	19	Security	1	Female	.0492111	High
9.	22	Esteem	4	Female	1.426412	High
10.	22	Variety	1	Female	1.426412	Neither
11.	22	Autonomy	2	Female	1.426412	High
12.	22	Security	3	Female	1.426412	Neither

Note that we kept only the data without ties for our analysis. cmroprobit handles ties by evaluating the likelihoods of all possible ways of breaking a tie. For example, suppose someone reported ranks $(1, 1, 3, 4)$, where the two 1s indicate a tie. For this person, cmroprobit computes likelihoods for ranks $(1, 2, 3, 4)$ and $(2, 1, 3, 4)$. If there is a 3-way tie among the ranks, 6 different likelihoods are computed. If there is a 4-way tie among the ranks, 24 different likelihoods are computed. If there are ties in your data, cmroprobit will be slower than if there were no ties. For this example, we drop the cases with ties in these data, just to make it run faster. Running the example with ties takes about 7 times longer, adjusting for the number of cases omitted (which is substantial because 87% of respondents had ties among their rankings).

Before we can run cmroprobit, we must cmset our data:

```
. cmset id jobchar

     Case ID variable: id
Alternatives variable: jobchar
```

We can now fit our model. We specify the option reverse because by default a bigger rank indicates a preferred alternative, whereas in these data, it is the opposite—a smaller rank indicates a preferred alternative.

```
. cmroprobit rank i.currentjob, casevars(i.female score) reverse structural
note: variable 2.currentjob has 69 cases that are not alternative-specific;
      there is no within-case variability.
note: variable 3.currentjob has 107 cases that are not alternative-specific;
      there is no within-case variability.
Iteration 0:   log simulated-likelihood = -1102.9667
Iteration 1:   log simulated-likelihood = -1095.0846  (backed up)
Iteration 2:   log simulated-likelihood = -1091.2528  (backed up)
Iteration 3:   log simulated-likelihood = -1086.9763  (backed up)
Iteration 4:   log simulated-likelihood = -1086.8123  (backed up)
Iteration 5:   log simulated-likelihood = -1086.3018  (backed up)
Iteration 6:   log simulated-likelihood = -1086.0779
Iteration 7:   log simulated-likelihood = -1085.7839
Iteration 8:   log simulated-likelihood = -1085.4865  (backed up)
Iteration 9:   log simulated-likelihood = -1084.9575
Iteration 10:  log simulated-likelihood = -1084.0493
Iteration 11:  log simulated-likelihood = -1083.9447
Iteration 12:  log simulated-likelihood = -1083.1608
Iteration 13:  log simulated-likelihood =  -1082.959
Iteration 14:  log simulated-likelihood = -1082.7606
Iteration 15:  log simulated-likelihood = -1081.3702
Iteration 16:  log simulated-likelihood = -1080.8372
Iteration 17:  log simulated-likelihood = -1080.6969
Iteration 18:  log simulated-likelihood = -1080.5508
Iteration 19:  log simulated-likelihood = -1079.9994
Iteration 20:  log simulated-likelihood = -1079.9774
Iteration 21:  log simulated-likelihood = -1079.9741
Iteration 22:  log simulated-likelihood = -1079.9734
Iteration 23:  log simulated-likelihood = -1079.9733
Iteration 24:  log simulated-likelihood = -1079.9733
Reparameterizing to correlation metric and refining estimates
Iteration 0:   log simulated-likelihood = -1079.9733
Iteration 1:   log simulated-likelihood = -1079.9733
```

```
Rank-ordered probit choice model                Number of obs      =      1,660
Case ID variable: id                            Number of cases    =        415

Alternatives variable: jobchar                  Alts per case: min =          4
                                                               avg =        4.0
                                                               max =          4

Integration sequence:      Hammersley
Integration points:               642              Wald chi2(8)    =      33.92
Log simulated-likelihood = -1079.9733              Prob > chi2     =     0.0000

------------------------------------------------------------------------------
        rank | Coefficient  Std. err.      z    P>|z|     [95% conf. interval]
-------------+----------------------------------------------------------------
jobchar      |
  currentjob |
     Neither |   .0694754   .1092521     0.64   0.525    -.1446549    .2836056
        High |     .44359   .1216771     3.65   0.000     .2051073    .6820727
-------------+----------------------------------------------------------------
Esteem       |  (base alternative)
-------------+----------------------------------------------------------------
Variety      |
      female |
      Female |   .1354483   .1843801     0.73   0.463    -.2259301    .4968266
       score |   .1407149   .0977656     1.44   0.150    -.0509022    .3323319
       _cons |   1.734451   .1449841    11.96   0.000     1.450288    2.018615
-------------+----------------------------------------------------------------
Autonomy     |
      female |
      Female |   .2562946   .1645936     1.56   0.119    -.0663029    .5788921
       score |    .189966   .0873585     2.17   0.030     .0187466    .3611854
       _cons |   .7007339   .1203077     5.82   0.000     .4649352    .9365326
-------------+----------------------------------------------------------------
Security     |
      female |
      Female |   .2326753   .2055824     1.13   0.258    -.1702589    .6356095
       score |  -.1779948   .1101965    -1.62   0.106     -.393976    .0379865
       _cons |   1.343435   .1598743     8.40   0.000     1.030088    1.656783
-------------+----------------------------------------------------------------
   /lnsigma3 |  -.1088538   .1629293    -0.67   0.504    -.4281894    .2104817
   /lnsigma4 |   .3181601    .115562     2.75   0.006     .0916627    .5446575
-------------+----------------------------------------------------------------
  /atanhr3_2 |  -.1607581   .2035115    -0.79   0.430    -.5596332    .2381171
  /atanhr4_2 |  -.2718915   .1700903    -1.60   0.110    -.6052624    .0614793
  /atanhr4_3 |  -.3839589   .2473491    -1.55   0.121    -.8687541    .1008364
-------------+----------------------------------------------------------------
      sigma1 |          1  (base alternative)
      sigma2 |          1  (scale alternative)
      sigma3 |   .8968615    .146125                       .651688    1.234273
      sigma4 |   1.374596   .1588511                      1.095995    1.724018
-------------+----------------------------------------------------------------
     rho3_2 |  -.1593874   .1983414                     -.5077053    .2337165
      rho4_2 |   -.265384    .158111                     -.5407836    .0614019
      rho4_3 |  -.3661406   .2141897                     -.7007406     .100496
------------------------------------------------------------------------------
(jobchar=Esteem is the alternative normalizing location)
(jobchar=Variety is the alternative normalizing scale)
```

We also specified the option `structural` to fit a variance–covariance parameterization based on the full 4×4 variance–covariance matrix. The postestimation commands, `estat covariance` and `estat correlation`, show the estimated covariance and correlations matrices, respectively.

```
. estat covariance
```

	Esteem	Variety	Autonomy	Security
Esteem	1			
Variety	0	1		
Autonomy	0	-.1429484	.8043605	
Security	0	-.3647959	-.4513864	1.889515

```
. estat correlation
```

	Esteem	Variety	Autonomy	Security
Esteem	1.0000			
Variety	0.0000	1.0000		
Autonomy	0.0000	-0.1594	1.0000	
Security	0.0000	-0.2654	-0.3661	1.0000

By default, `cmroprobit` fits a covariance matrix parameterized by differences between alternatives, which yields a 3×3 matrix. See *Covariance structures* in [CM] **cmmprobit** for details.

Expected choice probabilities (the margins command) after cmroprobit

As with the other `cm` estimators, running `margins` afterward can help us understand the model results.

```
. margins

Predictive margins                                     Number of obs = 1,660
Model VCE: OIM

Expression: Pr(jobchar), predict(pr1)
```

	Margin	Delta-method std. err.	z	P>|z|	[95% conf.	interval]
_outcome						
Esteem	.0228205	.0059232	3.85	0.000	.0112112	.0344297
Variety	.4503611	.0238804	18.86	0.000	.4035564	.4971658
Autonomy	.1461409	.0165112	8.85	0.000	.1137795	.1785022
Security	.3806896	.0232362	16.38	0.000	.3351474	.4262318

Typing `margins` without any options after `cmroprobit` gives the average predicted probability that a particular outcome is the highest-ranked choice. This is indicated on the output by `predict(pr1)`. Also note that the probabilities sum to one. Here variety has the greatest probability (45%) of being the characteristic ranked first. Security has the second greatest probability (38%) of being ranked first.

The above probabilities from `margins` are the expected probabilities of ranking these characteristics as most important based on our model and the characteristics of the individuals in this sample. We can also ask questions about what our model tells would happen if covariates change. For instance, what would we expect these probabilities to be if everyone ranked his or her current job as high in security? To answer this, we use the option `alternative(security)` with `margins` to select the security alternative, and we set it to the `High` ranking by specifying `3.currentjob`.

```
. margins 3.currentjob, alternative(Security)

Predictive margins                                Number of obs = 1,660
Model VCE: OIM

Expression:  Pr(jobchar), predict(pr1)
Alternative: Security
```

	Margin	Delta-method std. err.	z	P>|z|	[95% conf.	interval]
_outcome# currentjob						
Esteem#High	.0196645	.0053179	3.70	0.000	.0092416	.0300874
Variety#High	.4184274	.0252003	16.60	0.000	.3690358	.467819
Autonomy # High	.1342234	.0160335	8.37	0.000	.1027982	.1656485
Security # High	.4276949	.02543	16.82	0.000	.3778529	.4775369

Based on this model, we expect that 43% of individuals whose current job is high in security would rank security as being the most important job characteristic.

See [CM] **Intro 1** and [CM] **margins** for details and more examples of running `margins` after `cm` estimators.

cmrologit: Logistic regression for rank-ordered alternatives

`cmrologit` fits a rank-ordered logistic regression model for choice data (Beggs, Cardell, and Hausman 1981). This model is also known as the Plackett–Luce model (Marden 1995), as the exploded logit model (Punj and Staelin 1978), and as the choice-based method of conjoint analysis (Hair et al. 2010).

The model fit by `cmrologit` is unique among the `cm` estimators in that the model does not have explicitly identified alternatives. (`cmmixlogit` and `cmxtmixlogit` can fit models without identified alternatives as well, but for these commands having no alternatives is optional.) Independent variables for `cmrologit` must all be alternative specific; that is, they must vary within case. Any purely case-specific variables will be dropped from the estimation.

Like `cmroprobit`, `cmrologit` allows the ranks to be tied. However, `cmrologit` uses a different computational method, and computation with ties is speedy.

Here is an example using `cmrologit`. We have data on human resource managers ranking their perceived desirability of fictitious applicants. The dataset has 29 cases, each consisting of 10 applicants that are ranked. The variable `caseid` identifies the cases, and we list the observations for `caseid == 7`.

```
. use https://www.stata-press.com/data/r17/evignet, clear
(Vignet study employer prefs (Inge de Wolf 2000))
. list pref female age grades edufit workexp if caseid == 7, noobs
```

pref	female	age	grades	edufit	workexp
0	yes	28	A/B	no	none
0	no	25	C/D	yes	one year
0	no	25	C/D	yes	none
0	yes	25	C/D	no	internship
1	no	25	C/D	yes	one year
2	no	25	A/B	yes	none
3	yes	25	A/B	yes	one year
4	yes	25	A/B	yes	none
5	no	25	A/B	yes	internship
6	yes	28	A/B	yes	one year

The variable `pref` contains the managers' ranks, with 6 indicating the applicant they rank the highest and 0 indicating the four least desirable applicants (tied for last). Predictors for the outcome include `female`, `age`, `grades`, `edufit` (whether the applicant's education fits the job requirements), and `workexp` (work experience). Note that all of these variables represent characteristics of the applicants. There are no variables representing any of the traits of the managers doing the ranking; such variables would be case specific and would be dropped from the model.

As with all `cm` commands, the data must be `cmset`. But here there is no alternatives variable, so we use the `noalternatives` option and only have to specify the case ID variable.

```
. cmset caseid, noalternatives
     Case ID variable: caseid
Alternatives variable: <none>
```

We now fit our model. We include the `baselevels` option to show the base levels of the factor variables to make the output more understandable. (The option `baselevels` can be used with all estimation commands that allow factor variables; see **[U] 11.4.3 Factor variables**.)

```
. cmrologit pref i.female i.age i.grades i.edufit i.workexp, baselevels
Iteration 0:   log likelihood = -342.28088
Iteration 1:   log likelihood = -300.81224
Iteration 2:   log likelihood =  -300.2559
Iteration 3:   log likelihood =  -300.2549
Iteration 4:   log likelihood =  -300.2549
Refining estimates:
Iteration 0:   log likelihood =  -300.2549

Rank-ordered logit choice model                 Number of obs     =        290
Case ID variable: caseid                        Number of cases   =         29

Ties adjustment: exactm                         Obs per case:
                                                              min =         10
                                                              avg =      10.00
                                                              max =         10

                                                LR chi2(7)        =      84.05
Log likelihood = -300.2549                      Prob > chi2       =     0.0000

------------------------------------------------------------------------------
        pref | Coefficient  Std. err.      z    P>|z|     [95% conf. interval]
-------------+----------------------------------------------------------------
      female |
          no |          0  (base)
         yes |   -.053564   .1676711    -0.32   0.749    -.3821933    .2750654
             |
         age |
          22 |          0  (base)
          25 |      .2227   .2529694     0.88   0.379    -.2731109    .7185109
          28 |  -.0374488   .2692834    -0.14   0.889    -.5652347    .4903371
             |
      grades |
         C/D |          0  (base)
         A/B |    1.07066   .1877038     5.70   0.000     .7027673    1.438553
             |
      edufit |
          no |          0  (base)
         yes |   .4591287    .174471     2.63   0.008     .1171718    .8010855
             |
     workexp |
        none |          0  (base)
  internship |   .6644342   .2613156     2.54   0.011      .152265    1.176603
    one year |   1.463883    .245367     5.97   0.000     .9829721    1.944793
------------------------------------------------------------------------------
```

`grades`, `edufit`, and `workexp` are all significant predictors of the ranked outcomes.

The maximum likelihood estimates from `cmrologit` are obtained as the maximum partial-likelihood estimates of an appropriately specified Cox regression model for waiting time; see [ST] **stcox**. A higher ranking of an alternative is formally equivalent to a higher hazard rate of failure. A higher stated preference has a shorter waiting time until failure. `cmrologit` uses `stcox` to fit the rank-ordered logit model based on such a specification of the data in Cox terms. See *Methods and formulas* in [CM] **cmrologit**.

References

Beggs, S., S. Cardell, and J. A. Hausman. 1981. Assessing the potential demand for electric cars. *Journal of Econometrics* 17: 1–19. https://doi.org/10.1016/0304-4076(81)90056-7.

Hair, J. F., Jr., W. C. Black, B. J. Babin, and R. E. Anderson. 2010. *Multivariate Data Analysis*. 7th ed. Upper Saddle River, NJ: Pearson.

Long, J. S., and J. Freese. 2014. *Regression Models for Categorical Dependent Variables Using Stata*. 3rd ed. College Station, TX: Stata Press.

Marden, J. I. 1995. *Analyzing and Modeling Rank Data*. London: Chapman & Hall.

Punj, G. N., and R. Staelin. 1978. The choice process for graduate business schools. *Journal of Marketing Research* 15: 588–598. https://doi.org/10.1177/002224377801500408.

Also see

[CM] **Intro 1** — Interpretation of choice models

[CM] **Intro 2** — Data layout

[CM] **Intro 3** — Descriptive statistics

[CM] **Intro 4** — Estimation commands

[CM] **cmrologit** — Rank-ordered logit choice model

[CM] **cmroprobit** — Rank-ordered probit choice model