Title

stata.com

cmclogit — Conditional logit (McFadden's) choice model

Description

`cmclogit` fits McFadden's choice model, which is a specific case of the more general conditional logistic regression model fit by `clogit`.

The command requires multiple observations for each case (representing one individual or decision maker), where each observation represents an alternative that may be chosen. `cmclogit` allows two types of independent variables: alternative-specific variables, which vary across both cases and alternatives, and case-specific variables, which vary across only cases.

Quick start

McFadden's choice model of `y` on alternative-specific variable `x1` using `cmset` data

```
cmclogit y x1
```

As above, and add indicators for levels of `x2`, which are constant within case

```
cmclogit y x1, casevars(i.x2)
```

As above, but omit alternative-specific intercepts

```
cmclogit y x1, casevars(i.x2) noconstant
```

Include only case-specific covariates (equivalent to `mlogit` when data are balanced)

```
cmclogit y, casevars(i.x2 x3)
```

Menu

Statistics > Choice models > Conditional logit (McFadden's choice) model

Syntax

```
cmclogit depvar [indepvars] [if] [in] [weight] [, options]
```

depvar equal to 1 identifies the chosen alternatives, whereas a 0 indicates the alternatives that were not chosen. There can be only one chosen alternative for each case.

options	Description
Model	
`casevars(`*varlist*`)`	case-specific variables
`basealternative(`*# | lbl | str*`)`	set base alternative
`noconstant`	suppress alternative-specific constant terms
`altwise`	use alternativewise deletion instead of casewise deletion
`offset(`*varname*`)`	include *varname* in model with coefficient constrained to 1
`constraints(`*constraints*`)`	apply specified linear constraints
SE/Robust	
`vce(`*vcetype*`)`	*vcetype* may be `oim`, `robust`, `cluster` *clustvar*, `bootstrap`, or `jackknife`
Reporting	
`level(#)`	set confidence level; default is `level(95)`
`or`	report odds ratios and relative-risk ratios
`noheader`	do not display the header on the coefficient table
`nocnsreport`	do not display constraints
display_options	control columns and column formats, row spacing, line width, display of omitted variables and base and empty cells, and factor-variable labeling
Maximization	
maximize_options	control the maximization process; seldom used
`collinear`	keep collinear variables
`coeflegend`	display legend instead of statistics

You must `cmset` your data before using `cmclogit`; see [CM] **cmset**.

indepvars and *varlist* may contain factor variables; see [U] **11.4.3 Factor variables**.

`bootstrap`, `by`, `fp`, `jackknife`, and `statsby` are allowed; see [U] **11.1.10 Prefix commands**.

Weights are not allowed with the `bootstrap` prefix; see [R] **bootstrap**.

`fweights`, `iweights`, and `pweights` are allowed (see [U] **11.1.6 weight**), but they are interpreted to apply to cases as a whole, not to individual observations. See *Use of weights* in [R] **clogit**.

`collinear` and `coeflegend` do not appear in the dialog box.

See [U] **20 Estimation and postestimation commands** for more capabilities of estimation commands.

Options

Model

`casevars(`*varlist*`)` specifies the case-specific numeric variables. These are variables that are constant for each case. If there are a maximum of J alternatives, there will be $J - 1$ sets of coefficients associated with each *casevar*.

`basealternative(#|lbl|str)` sets the alternative that normalizes the level of utility. The base alternative may be specified as a number when the alternatives variable is numeric, as a label when it is numeric and has a value label, or as a string when it is a string variable. The default is the alternative with the highest frequency of being chosen. This option is ignored if neither alternative-specific constants nor case-specific variables are specified.

If `vce(bootstrap)` or `vce(jackknife)` is specified, you must specify the base alternative. This is to ensure that the same model is fit with each call to `cmclogit`.

`noconstant` suppresses the $J-1$ alternative-specific constant terms.

`altwise` specifies that alternativewise deletion be used when omitting observations because of missing values in your variables. The default is to use casewise deletion; that is, the entire group of observations making up a case is omitted if any missing values are encountered. This option does not apply to observations that are excluded by the `if` or `in` qualifier or the `by` prefix; these observations are always handled alternativewise regardless of whether `altwise` is specified.

`offset(`*varname*`)`, `constraints(`*numlist*|*matname*`)`; see [R] **Estimation options**.

SE/Robust

`vce(`*vcetype*`)` specifies the type of standard error reported, which includes types that are derived from asymptotic theory (`oim`), that are robust to some kinds of misspecification (`robust`), that allow for intragroup correlation (`cluster` *clustvar*), and that use bootstrap or jackknife methods (`bootstrap`, `jackknife`); see [R] ***vce_option***.

Reporting

`level(#)`; see [R] **Estimation options**.

`or` reports the estimated coefficients transformed to odds ratios for alternative-specific variables and relative-risk ratios for case-specific variables. That is, e^b rather than b is reported. Standard errors and confidence intervals are transformed accordingly. This option affects how results are displayed, not how they are estimated. `or` may be specified at estimation or when replaying previously estimated results.

`noheader` prevents the coefficient table header from being displayed.

`nocnsreport`; see [R] **Estimation options**.

display_options: `noci`, `nopvalues`, `noomitted`, `vsquish`, `noemptycells`, `baselevels`, `allbaselevels`, `nofvlabel`, `fvwrap(#)`, `fvwrapon(`*style*`)`, `cformat(%`*fmt*`)`, `pformat(%`*fmt*`)`, `sformat(%`*fmt*`)`, and `nolstretch`; see [R] **Estimation options**.

Maximization

maximize_options: `difficult`, `technique(`*algorithm_spec*`)`, `iterate(#)`, `[no]log`, `trace`, `gradient`, `showstep`, `hessian`, `showtolerance`, `tolerance(#)`, `ltolerance(#)`, `nrtolerance(#)`, `nonrtolerance`, and `from(`*init_specs*`)`; see [R] **Maximize**. These options are seldom used.

`technique(bhhh)` is not allowed.

The initial estimates must be specified as `from(`*matname* `[, copy])`, where *matname* is the matrix containing the initial estimates and the `copy` option specifies that only the position of each element in *matname* is relevant. If `copy` is not specified, the column stripe of *matname* identifies the estimates.

The following options are available with cmclogit but are not shown in the dialog box:

collinear, coeflegend; see [R] **Estimation options**.

Remarks and examples

stata.com

cmclogit fits McFadden's choice model (McFadden 1974). For a brief introduction, see Greene (2018, sec. 18.2) or Cameron and Trivedi (2010, sec. 15.5).

cmclogit requires data in long form. For each individual (or decision maker), there are multiple Stata observations, one for each of the alternatives the individual could have chosen. We call the group of Stata observations for an individual a "case". Each case represents a single statistical observation although it comprises multiple Stata observations. See [CM] **Intro 2**.

Independent variables for McFadden's choice model come in two forms: alternative specific and case specific. Alternative-specific variables vary across cases and within cases by alternative. Case-specific variables are constant within cases.

We index the set of unordered alternatives by $1, 2, \ldots, J$. Let y_{ij}, $j = 1, \ldots, J$, be an indicator variable for the alternative chosen by the ith individual (case). That is, $y_{ij} = 1$ if individual i chose alternative j, and $y_{ij} = 0$ otherwise.

The sets of possible alternatives across individuals, also known as choice sets, can be unbalanced. That is, choice sets can vary by case. For example, individual 1 could have choice set $\{1, 2, 3\}$, and individual 2 could have choice set $\{1, 2, 4\}$. For individual 1, the 4th alternative was unavailable to be chosen, and for individual 2, the 3rd alternative was unavailable. We take $1, 2, \ldots, J$ to represent all possible alternatives taken across all individuals.

Assume that we have p alternative-specific variables so that for case i we have a $J \times p$ data matrix $\mathbf{X}_i$. Further, assume that we have q case-specific variables so that we also have a $1 \times q$ data vector $\mathbf{z}_i$ for case i. Our random utility model can be expressed as

$$\mathbf{u}_i = \mathbf{X}_i\boldsymbol{\beta} + (\mathbf{z}_i\mathbf{A})' + \boldsymbol{\epsilon}_i$$

where $\mathbf{u}_i$ is the utility for case i, $\boldsymbol{\beta}$ is a $p \times 1$ vector of alternative-specific regression coefficients, and $\mathbf{A} = (\boldsymbol{\alpha}_1, \ldots, \boldsymbol{\alpha}_J)$ is a $q \times J$ matrix of case-specific regression coefficients. The elements of the $J \times 1$ vector $\boldsymbol{\epsilon}_i$ are independent type I (Gumbel-type) extreme-value random variables with mean γ (the Euler–Mascheroni constant, approximately 0.577) and variance $\pi^2/6$.

We must fix one of the $\boldsymbol{\alpha}_j$ to be the zero vector to normalize the location. We set $\boldsymbol{\alpha}_k = \mathbf{0}$, where k is specified by the basealternative() option. The vector $\mathbf{u}_i$ quantifies the utility that the individual gains from the J alternatives. If alternative j is unavailable to the ith individual, we omit the jth row in $\mathbf{u}_i$. The alternative chosen by individual i is the one that maximizes utility.

McFadden's choice model is a specific instance of conditional logistic regression. You can use clogit to obtain the same estimates as cmclogit by specifying the case ID variable used to cmset your data as the group() variable in clogit. Your case-specific variables in casevars() must be interacted with each alternative, excluding the interaction associated with the base alternative. The alternatives variable used to produce this interaction is the alternatives variable used with cmset. These interactions are included in the clogit estimation along with the alternative-specific variables. cmclogit does this for you. See *Duplicating cmclogit using clogit* in [CM] **Intro 5** for an example that uses clogit to reproduce the results from cmclogit.

Before you can fit McFadden's choice model using cmclogit, you must first cmset your data to specify which variables in your dataset identify the cases and the alternatives; see [CM] **cmset** for information on this command.

Example 1: Consumer car choice data

We have fictitious data on 885 consumers and their choice of automobile. Each consumer chose among an American, Japanese, European, or Korean car (variable `car`). We want to explore the relationship between the choice of car and the consumer's gender (variable `gender`) and income (variable `income` in thousands of dollars). We also have the number of dealerships of each nationality in the consumer's community (variable `dealers`), which we want to include as a regressor.

The variable `dealers` is an alternative-specific variable, and `gender` and `income` are case-specific variables. Each consumer's chosen car is indicated by the variable `purchase`, a 0/1 variable.

Let's list some of the data.

```
. use https://www.stata-press.com/data/r16/carchoice
(Car choice data)
. list consumerid car purchase dealers gender income
> if consumerid <= 4, sepby(consumerid) abbr(10)
```

	consumerid	car	purchase	dealers	gender	income
1.	1	American	1	9	Male	46.7
2.	1	Japanese	0	11	Male	46.7
3.	1	European	0	5	Male	46.7
4.	1	Korean	0	1	Male	46.7
5.	2	American	1	10	Male	26.1
6.	2	Japanese	0	7	Male	26.1
7.	2	European	0	2	Male	26.1
8.	2	Korean	0	1	Male	26.1
9.	3	American	0	8	Male	32.7
10.	3	Japanese	1	6	Male	32.7
11.	3	European	0	2	Male	32.7
12.	4	American	1	5	Female	49.2
13.	4	Japanese	0	4	Female	49.2
14.	4	European	0	3	Female	49.2

We see that the first consumer, a male earning $46,700 per year, chose to purchase an American car. The third consumer purchased a Japanese car. The third and fourth consumers do not have the choice of a Korean car as a possible alternative because there are no Korean automobile dealerships in their communities.

Before we can run a `cm` estimation command, we must `cmset` our data. The first argument to `cmset` is the case ID variable, which must be numeric. For these data, it is the variable `consumerid`, which identifies individual consumers. The alternatives variable identifies the alternatives that could have been chosen. In this instance, it is the variable `car`, which gives the nationality of car, American, Japanese, European, or Korean.

```
. cmset consumerid car
note: alternatives are unbalanced across choice sets; choice sets of different
      sizes found
            caseid variable:  consumerid
      alternatives variable:  car
```

The message from `cmset` tells us that the choice sets for these data are unbalanced (which we already knew from the data listing). The `cmchoiceset` command will display the choice sets:

```
. cmchoiceset
Tabulation of choice-set possibilities
```

Choice set	Freq.	Percent	Cum.
1 2 3	380	42.94	42.94
1 2 3 4	505	57.06	100.00
Total	885	100.00	

```
Total is number of cases.
```

The numeric variable **car** is labeled so that 1 = American, 2 = Japanese, 3 = European, and 4 = Korean. We see that 43% of the consumers do not have a Korean car dealership in their communities, and this alternative is not considered available to them. All consumers in these data have American, Japanese, and European dealerships locally, and everyone has these alternatives in their choice sets.

We now fit our model.

```
. cmclogit purchase dealers, casevars(i.gender income)
Iteration 0:   log likelihood = -959.21405
Iteration 1:   log likelihood = -948.48587
Iteration 2:   log likelihood =  -948.1217
Iteration 3:   log likelihood = -948.12096
Iteration 4:   log likelihood = -948.12096

Conditional logit choice model                  Number of obs      =      3,075
Case ID variable: consumerid                    Number of cases    =        862

Alternatives variable: car                      Alts per case: min =          3
                                                               avg =        3.6
                                                               max =          4

                                                  Wald chi2(7)     =      51.03
Log likelihood = -948.12096                       Prob > chi2      =     0.0000
```

purchase	Coef.	Std. Err.	z	P>|z|	[95% Conf.	Interval]
car						
dealers	.0448082	.0262818	1.70	0.088	-.0067032	.0963196
American	(base alternative)					
Japanese						
gender						
Male	-.379326	.1712399	-2.22	0.027	-.71495	-.0437021
income	.0154978	.0065145	2.38	0.017	.0027296	.0282659
_cons	-.4787261	.331378	-1.44	0.149	-1.128215	.1707628
European						
gender						
Male	.653345	.2647694	2.47	0.014	.1344065	1.172283
income	.0343647	.0080286	4.28	0.000	.0186289	.0501006
_cons	-2.839606	.461613	-6.15	0.000	-3.744351	-1.934861
Korean						
gender						
Male	.0679233	.4464535	0.15	0.879	-.8071094	.942956
income	-.0377716	.0158434	-2.38	0.017	-.068824	-.0067191
_cons	.0511728	.8033048	0.06	0.949	-1.523276	1.625621

Displaying the results as odds ratios and relative-risk ratios makes interpretation easier.

```
. cmclogit, or noheader
```

purchase	Odds Ratio	Std. Err.	z	P>|z|	[95% Conf.	Interval]
car						
dealers	1.045827	.0274862	1.70	0.088	.9933192	1.101111
American	(base alternative)					
Japanese						
gender						
Male	.6843225	.1171833	-2.22	0.027	.4892166	.9572391
income	1.015618	.0066162	2.38	0.017	1.002733	1.028669
_cons	.6195721	.2053126	-1.44	0.149	.3236104	1.186209
European						
gender						
Male	1.921959	.5088759	2.47	0.014	1.143858	3.229358
income	1.034962	.0083093	4.28	0.000	1.018804	1.051377
_cons	.0584487	.0269807	-6.15	0.000	.023651	.1444443
Korean						
gender						
Male	1.070283	.4778316	0.15	0.879	.4461458	2.56756
income	.9629329	.0152561	-2.38	0.017	.9334909	.9933034
_cons	1.052505	.8454821	0.06	0.949	.2179966	5.081575

```
Note: Exponentiated coefficients represent odds ratios for
      alternative-specific variables (first equation) and relative-risk ratios
      for case-specific variables.
Note: _cons estimates baseline relative risk for each outcome.
```

These results indicate that males are less likely to pick a Japanese car over an American car than females (relative-risk ratio 0.68) but males are more likely to choose a European car over an American car than females (relative-risk ratio 1.9). Persons with higher incomes are more likely to purchase a Japanese or European car over an American car but less likely to purchase a Korean car over an American one.

How would increasing the number of dealerships for a certain nationality of car affect the likelihood of more people buying that car? Using `margins` after `cmclogit` can answer this question based on the model fit.

Let's make the question more precise: How would the probability of a person selecting a European car change if one additional European dealership was opened in each community? If this probability increases (as we expect it to), the increase must come at the expense of American, Japanese, or Korean cars. Which one of these is affected the most?

We type `margins`, specifying the options `at(dealers=generate(dealers))` and `at(dealers=generate(dealers+1))` to estimate the probabilities of selecting each nationality of car with the current number of dealerships and with one additional dealership in each community. We add the `contrast(atcontrast(r))` option to estimate the differences between these two sets of probabilities. By including the `alternative(European)` option, we request that these differences are for a change in alternative `European` (the value label for this alternative).

```
. margins, at(dealers=generate(dealers)) at(dealers=generate(dealers+1))
> contrast(atcontrast(r)) alternative(European)
Contrasts of predictive margins                 Number of obs     =      3,075
Model VCE    : OIM

Expression   : Pr(car|1 selected), predict()
Alternative  : European

1._at        : dealers         = dealers
2._at        : dealers         = dealers+1
```

	df	chi2	P>chi2
_at@_outcome			
(2 vs 1) American	1	2.81	0.0937
(2 vs 1) Japanese	1	2.80	0.0940
(2 vs 1) European	1	2.82	0.0934
(2 vs 1) Korean	1	2.52	0.1121
Joint	3	2.84	0.4177

	Delta-method Contrast	Std. Err.	[95% Conf.	Interval]
_at@_outcome				
(2 vs 1) American	-.0028946	.0017268	-.0062791	.0004899
(2 vs 1) Japanese	-.0024619	.0014701	-.0053434	.0004195
(2 vs 1) European	.0056244	.0033521	-.0009456	.0121944
(2 vs 1) Korean	-.0002679	.0001686	-.0005983	.0000625

We see that adding a European dealership changes the probability of someone purchasing a European car by 0.0056. This increase comes at the expense of American cars slightly more than Japanese cars. The probability of someone purchasing an American car decreases by 0.0029 per European dealership increased, and the probability of someone purchasing a Japanese car decreases by 0.0025. The probability of buying a Korean car is barely changed, only a tiny decrease of 0.0003.

See [CM] **Intro 1** and [CM] **margins** for more on using `margins` after `cmclogit`.

◁

▷ Example 2: Changing the base alternative

In the preceding example, the base alternative category was American cars, which was chosen by default because we did not specify the option `basealternative()`. The default base category is the alternative with the highest frequency of being chosen. To set the base category to Japanese cars, we specify `basealternative(Japanese)`.

```
. cmclogit purchase dealers, casevars(i.gender income)
> basealternative(Japanese) or

Iteration 0:   log likelihood = -961.18687
Iteration 1:   log likelihood = -948.53711
Iteration 2:   log likelihood = -948.12131
Iteration 3:   log likelihood = -948.12096
Iteration 4:   log likelihood = -948.12096

Conditional logit choice model                  Number of obs      =      3,075
Case ID variable: consumerid                    Number of cases    =        862

Alternatives variable: car                      Alts per case: min =          3
                                                               avg =        3.6
                                                               max =          4

                                                   Wald chi2(7)    =      51.03
Log likelihood = -948.12096                        Prob > chi2     =     0.0000
```

purchase	Odds Ratio	Std. Err.	z	P>|z|	[95% Conf.	Interval]
car						
dealers	1.045827	.0274862	1.70	0.088	.9933192	1.101111
American						
gender						
Male	1.461299	.2502327	2.22	0.027	1.044671	2.044084
income	.9846217	.0064143	-2.38	0.017	.9721298	.9972741
_cons	1.614017	.5348498	1.44	0.149	.8430215	3.090136
Japanese	(base alternative)					
European						
gender						
Male	2.808557	.7404989	3.92	0.000	1.675163	4.708792
income	1.019046	.0083438	2.30	0.021	1.002823	1.035532
_cons	.0943372	.0429738	-5.18	0.000	.0386306	.2303745
Korean						
gender						
Male	1.564004	.698962	1.00	0.317	.6513754	3.755299
income	.9481246	.0152149	-3.32	0.001	.918768	.9784192
_cons	1.698761	1.365008	0.66	0.510	.3516937	8.205401

```
Note: Exponentiated coefficients represent odds ratios for
      alternative-specific variables (first equation) and relative-risk ratios
      for case-specific variables.
Note: _cons estimates baseline relative risk for each outcome.
```

With the default base alternative of American cars, it was hard to make comparisons involving the choice of European and Korean cars relative to Japanese cars. The differences are now easy to see. ◁

▷ Example 3: altwise handling of missing values

The `altwise` option changes how `cmclogit` handles missing values. By default, missing values are handled casewise, meaning that any missing value in any observation composing the case causes the entire case to be omitted from the estimation sample. This applies to missing values in the alternative-specific and case-specific variables, the dependent variable, the alternatives variable, and in the weights if any.

If we only want to omit only observations with missing values and not the entire case, we specify the option `altwise`. We refit the model in example 1 using `altwise`. We also specify the option `basealternative(American)` so that the base alternative is the same as it was when we did casewise deletion. (When `altwise` deletion is done, the most frequent alternative is Japanese, and if we did not specify `basealternative()`, Japanese would be used by default as the base alternative.)

```
. cmclogit purchase dealers, casevars(i.gender income) altwise or
> basealt(American)
note: variable dealers has 1 case that is not alternative-specific; there is
      no within-case variability

Iteration 0:   log likelihood =  -976.1027
Iteration 1:   log likelihood = -965.37952
Iteration 2:   log likelihood = -965.01714
Iteration 3:   log likelihood =  -965.0164
Iteration 4:   log likelihood =  -965.0164

Conditional logit choice model                  Number of obs      =      3,137
Case ID variable: consumerid                    Number of cases    =        885

Alternatives variable: car                      Alts per case: min =          2
                                                               avg =        3.5
                                                               max =          4

                                                   Wald chi2(7)    =      54.18
Log likelihood =  -965.0164                        Prob > chi2     =     0.0000
```

purchase	Odds Ratio	Std. Err.	z	P>|z|	[95% Conf.	Interval]
car						
dealers	1.05095	.0272151	1.92	0.055	.9989405	1.105668
American	(base alternative)					
Japanese						
gender						
Male	.6866621	.1165921	-2.21	0.027	.4922809	.9577963
income	1.01551	.0065797	2.38	0.018	1.002696	1.028488
_cons	.6287248	.2070286	-1.41	0.159	.3297418	1.198801
European						
gender						
Male	1.997394	.5266466	2.62	0.009	1.191324	3.348863
income	1.035968	.0082453	4.44	0.000	1.019933	1.052255
_cons	.0558718	.0255968	-6.30	0.000	.0227629	.1371379
Korean						
gender						
Male	1.066616	.4762214	0.14	0.885	.4445952	2.55889
income	.9628352	.0153454	-2.38	0.017	.9332236	.9933864
_cons	1.054358	.8492452	0.07	0.948	.2174589	5.112091

```
Note: Exponentiated coefficients represent odds ratios for
      alternative-specific variables (first equation) and relative-risk ratios
      for case-specific variables.
Note: _cons estimates baseline relative risk for each outcome.
```

Results are similar to the model fit in example 1. That estimation sample had 862 cases; this one has 885, a difference of 23 cases.

We suspect that handling missing values alternativewise changes the choice sets. To see the choice sets used in the estimation, we type cmchoiceset with an if restriction to the estimation sample.

```
. cmchoiceset if e(sample) == 1
Tabulation of choice-set possibilities
```

Choice set	Freq.	Percent	Cum.
1 2	2	0.23	0.23
1 2 3	378	42.71	42.94
1 2 3 4	489	55.25	98.19
1 2 4	4	0.45	98.64
1 3	2	0.23	98.87
1 3 4	2	0.23	99.10
2 3	3	0.34	99.44
2 3 4	5	0.56	100.00
Total	885	100.00	

```
Total is number of cases.
```

When missing values were handled casewise, there were only two distinct choice sets: $\{1, 2, 3\}$ and $\{1, 2, 3, 4\}$. Handling the missing values alternativewise gives six new choice sets, albeit each with low frequency.

Handling missing values casewise never creates new choice sets. Handling missing values with altwise almost always changes the choice sets used in the estimation. You should be aware of the consequences. For instance, a dataset with balanced choice sets will typically become unbalanced when missing values are handled alternativewise.

The cmsample command can help you to see exactly what observations and cases are dropped, whether you use the casewise default or altwise.

◁

▷ Example 4: Multiple choices per case

Let us continue with our fictitious car choice dataset but expand it so that it also contains data on the purchase of a second car. Here is what the data look like now:

```
. use https://www.stata-press.com/data/r16/carchoice_panel, clear
(Car choice panel data)
. list consumerid carnumber car purchase
> if inlist(consumerid, 6, 7), sepby(consumerid carnumber) abbr(10)
```

	consumerid	carnumber	car	purchase
18.	6	1	American	1
19.	6	1	Japanese	0
20.	6	1	European	0
21.	7	1	American	0
22.	7	1	Japanese	0
23.	7	1	European	1
24.	7	1	Korean	0
25.	7	2	American	0
26.	7	2	Japanese	0
27.	7	2	European	0
28.	7	2	Korean	1

The person with consumerid = 7 has two cars, the first a European car and the second a Korean car. The person with consumerid = 6 has only one car, an American one.

How do we model these data?

The random utility model for McFadden's choice model yields only one chosen alternative per case: that with the greatest utility. Because the utility function is continuous, ties are theoretically impossible. See *Methods and formulas*. Choice models for rank-ordered data allow for multiple alternatives to be chosen and allow for tied ranks; for more information, see [CM] **Intro 6**.

Train (2009, sec. 2.2) notes that the set of alternatives can always be made mutually exclusive by considering the choice of two alternatives as a separate alternative. For example, with one or two choices allowed from alternatives A, B, and C, the set of alternatives is A only, B only, C only, A and B, A and C, and B and C, a total of six alternatives.

We could do this with our expanded car choice data. But this would mean a model with 14 alternatives. There are four nationalities of cars. So the alternatives are only one car of one of these nationalities (four possibilities), two cars of the same nationality (four possibilities), and two cars of different nationalities (six possibilities). With so many possibilities, there are concerns both about statistical power and about ease of model interpretation.

There is another way to view our expanded car choice data. The stated design of the fictitious data collection was to take the most recent car purchased, and if another car was purchased in the previous five years by anyone in the household, then to collect data on that car as well. So these data are panel data with information from two time points.

The variable `carnumber` will do as a time variable, and we can `cmset` the data as panel choice data:

```
. cmset consumerid carnumber car
panel data: panels consumerid and time carnumber
note: case identifier _caseid generated from consumerid carnumber
note: panel by alternatives identifier _panelaltid generated from consumerid
      car
note: alternatives are unbalanced across choice sets; choice sets of different
      sizes found
                caseid variable:  _caseid
          alternatives variable:  car
      panel by alternatives variable:  _panelaltid (unbalanced)
                time variable:  carnumber, 1 to 2
                        delta:  1 unit
note: data have been xtset
```

See [CM] **Intro 7** and [CM] **cmset** for more information on `cmset`ting panel choice data.

Once we have `cmset` the data, we can run `cmclogit`, issuing the same command line we used for the dataset with only one car per person.

```
. cmclogit purchase dealers, casevars(i.gender income) or basealt(American)
note: data were cmset as panel data, and the default vcetype for panel data is
      vce(cluster consumerid); see cmclogit

Iteration 0:   log pseudolikelihood = -1236.6065
Iteration 1:   log pseudolikelihood = -1221.0412
Iteration 2:   log pseudolikelihood = -1220.8605
Iteration 3:   log pseudolikelihood = -1220.8604

Conditional logit choice model                  Number of obs      =      3,728
Case ID variable: _caseid                       Number of cases    =       1045

Alternatives variable: car                      Alts per case: min =          3
                                                               avg =        3.6
                                                               max =          4

                                                   Wald chi2(7)    =      42.76
Log pseudolikelihood = -1220.8604                  Prob > chi2     =     0.0000

                          (Std. Err. adjusted for 862 clusters in consumerid)
```

purchase	Odds Ratio	Robust Std. Err.	z	P>|z|	[95% Conf.	Interval]
car						
dealers	1.020878	.0236626	0.89	0.373	.9755374	1.068325
American	(base alternative)					
Japanese						
gender						
Male	.6751564	.1104886	-2.40	0.016	.4898986	.9304704
income	1.01464	.0060839	2.42	0.015	1.002785	1.026634
_cons	.518233	.1608041	-2.12	0.034	.2821	.9520221
European						
gender						
Male	1.590534	.3411024	2.16	0.030	1.044711	2.421529
income	1.032775	.0074815	4.45	0.000	1.018216	1.047543
_cons	.074066	.0306088	-6.30	0.000	.0329494	.1664908
Korean						
gender						
Male	1.133754	.3022134	0.47	0.638	.6723919	1.911679
income	.9938	.0101097	-0.61	0.541	.9741815	1.013814
_cons	.5676166	.3002472	-1.07	0.284	.2012808	1.600692

```
Note: Exponentiated coefficients represent odds ratios for
      alternative-specific variables (first equation) and relative-risk ratios
      for case-specific variables.
Note: _cons estimates baseline relative risk for each outcome.
```

Note that `cmclogit` knew the data were panel data with some individuals having more than one case, and it automatically set the variance estimator used to `vce(cluster consumerid)`, where `consumerid` is the ID for individuals. If, for whatever reason, you wish to use another variance estimator, you can set `vce()` explicitly and `cmclogit` will respect your choice. See [R] ***vce_option*** for details on the available choices for `vce()`.

◁

Stored results

cmclogit stores the following in e():

Scalars

e(N)	number of observations
e(N_case)	number of cases
e(N_ic)	N for Bayesian information criterion (BIC)
e(N_clust)	number of clusters
e(k)	number of parameters
e(k_alt)	number of alternatives
e(k_indvars)	number of alternative-specific variables
e(k_casevars)	number of case-specific variables
e(k_eq)	number of equations in e(b)
e(k_eq_model)	number of equations in overall model test
e(df_m)	model degrees of freedom
e(ll)	log likelihood
e(const)	constant indicator
e(i_base)	base alternative index
e(chi2)	χ^2
e(p)	p-value for model test
e(alt_min)	minimum number of alternatives
e(alt_avg)	average number of alternatives
e(alt_max)	maximum number of alternatives
e(rank)	rank of e(V)
e(ic)	number of iterations
e(rc)	return code
e(converged)	1 if converged, 0 otherwise

Macros

e(cmd)	cmclogit
e(cmdline)	command as typed
e(depvar)	name of dependent variable
e(caseid)	name of case ID variable
e(altvar)	name of alternatives variable
e(alteqs)	alternative equation names
e(alt#)	alternative labels
e(wtype)	weight type
e(wexp)	weight expression
e(marktype)	casewise or altwise, type of markout
e(key_N_ic)	cases, key for N for Bayesian information criterion (BIC)
e(title)	title in estimation output
e(clustvar)	name of cluster variable
e(offset)	linear offset variable
e(chi2type)	Wald, type of model χ^2 test
e(vce)	*vcetype* specified in vce()
e(vcetype)	title used to label Std. Err.
e(opt)	type of optimization
e(which)	max or min; whether optimizer is to perform maximization or minimization
e(ml_method)	type of ml method
e(user)	name of likelihood-evaluator program
e(technique)	maximization technique
e(datasignature)	the checksum
e(datasignaturevars)	variables used in calculation of checksum
e(properties)	b V
e(predict)	program used to implement predict
e(marginsok)	predictions allowed by margins
e(marginsnotok)	predictions disallowed by margins
e(asbalanced)	factor variables fvset as asbalanced
e(asobserved)	factor variables fvset as asobserved

Matrices

e(b)	coefficient vector
e(stats)	alternative statistics
e(altvals)	alternative values
e(altfreq)	alternative frequencies
e(alt_casevars)	indicators for estimated case-specific coefficients—e(k_alt)×e(k_casevars)
e(Cns)	constraints matrix
e(ilog)	iteration log (up to 20 iterations)
e(gradient)	gradient vector
e(V)	variance–covariance matrix of the estimators
e(V_modelbased)	model-based variance

Functions

e(sample)	marks estimation sample

In addition to the above, the following is stored in r():

Matrices

r(table)	matrix containing the coefficients with their standard errors, test statistics, p-values, and confidence intervals

Note that results stored in r() are updated when the command is replayed and will be replaced when any r-class command is run after the estimation command.

Methods and formulas

In this model, we have a set of unordered alternatives indexed by $1, 2, \ldots, J$. Let y_{ij}, $j = 1, \ldots, J$, be an indicator variable for the alternative chosen by the ith individual (case). That is, $y_{ij} = 1$ if individual i chose alternative j, and $y_{ij} = 0$ otherwise.

The independent variables come in two forms: alternative specific and case specific. Alternative-specific variables vary among the alternatives and the cases, and case-specific variables vary only among cases. Assume that we have p alternative-specific variables so that for case i we have a $J \times p$ matrix, $\mathbf{X}_i$. Assume that we have q case-specific variables so that we have a $1 \times q$ vector $\mathbf{z}_i$ for case i.

The deterministic component of the random utility model can then be expressed as

$$\begin{aligned}\boldsymbol{\eta}_i &= \mathbf{X}_i\boldsymbol{\beta} + (\mathbf{z}_i\mathbf{A})' \\ &= \mathbf{X}_i\boldsymbol{\beta} + (\mathbf{z}_i \otimes \mathbf{I}_J)\,\mathrm{vec}(\mathbf{A}') \\ &= (\mathbf{X}_i,\ \mathbf{z}_i \otimes \mathbf{I}_J)\left\{\begin{array}{c}\boldsymbol{\beta} \\ \mathrm{vec}(\mathbf{A}')\end{array}\right\} \\ &= \mathbf{X}_i^*\boldsymbol{\beta}^*\end{aligned}$$

As before, $\boldsymbol{\beta}$ is a $p \times 1$ vector of alternative-specific regression coefficients, and $\mathbf{A} = (\boldsymbol{\alpha}_1, \ldots, \boldsymbol{\alpha}_J)$ is a $q \times J$ matrix of case-specific regression coefficients. We must set one of the $\boldsymbol{\alpha}_j$ to zero to normalize the location. Here $\mathbf{I}_J$ is the $J \times J$ identity matrix, vec(·) is the vector function that creates a vector from a matrix by placing each column of the matrix on top of the other (see [M-5] **vec( )**), and $\otimes$ is the Kronecker product (see [M-2] **op_kronecker**).

We have rewritten the linear equation so that it is a form that can be used by clogit, namely, $\mathbf{X}_i^*\boldsymbol{\beta}^*$, where

$$\mathbf{X}_i^* = (\mathbf{X}_i,\ \mathbf{z}_i \otimes \mathbf{I}_J)$$

$$\boldsymbol{\beta}^* = \left\{\begin{array}{c}\boldsymbol{\beta} \\ \mathrm{vec}(\mathbf{A}')\end{array}\right\}$$

With this in mind, see *Methods and formulas* in [R] **clogit** for the computational details of the conditional logit model.

This command supports the clustered version of the Huber/White/sandwich estimator of the variance using `vce(robust)` and `vce(cluster` *clustvar*`)`. See [P] **_robust**, particularly *Maximum likelihood estimators* and *Methods and formulas*. Specifying `vce(robust)` is equivalent to specifying `vce(cluster` *caseid*`)`, where *caseid* is the variable that identifies the cases.

Daniel Little McFadden (1937–) was born in North Carolina. He studied physics, psychology, and economics at the University of Minnesota and has taught economics at Pittsburgh, Berkeley, MIT, and the University of Southern California. His contributions to logit models were triggered by a student's project on freeway routing decisions, and his work consistently links economic theory and applied problems. In 2000, he shared the Nobel Prize in Economics with James J. Heckman.

References

Cameron, A. C., and P. K. Trivedi. 2010. *Microeconometrics Using Stata*. Rev. ed. College Station, TX: Stata Press.

Greene, W. H. 2018. *Econometric Analysis*. 8th ed. New York: Pearson.

McFadden, D. L. 1974. Conditional logit analysis of qualitative choice behavior. In *Frontiers in Econometrics*, ed. P. Zarembka, 105–142. New York: Academic Press.

Train, K. E. 2009. *Discrete Choice Methods with Simulation*. 2nd ed. New York: Cambridge University Press.

Also see

[CM] **cmclogit postestimation** — Postestimation tools for cmclogit

[CM] **cmmixlogit** — Mixed logit choice model

[CM] **cmmprobit** — Multinomial probit choice model

[CM] **cmset** — Declare data to be choice model data

[CM] **margins** — Adjusted predictions, predictive margins, and marginal effects

[CM] **nlogit** — Nested logit regression

[R] **clogit** — Conditional (fixed-effects) logistic regression

[R] **mlogit** — Multinomial (polytomous) logistic regression

[U] **20 Estimation and postestimation commands**