The Stata Journal (2023)
23, Number 4, pp. 1020–1044 DOI: 10.1177/1536867X231212437

Data management and techniques for best–worst discrete choice experiments

Farahnaz Islam
Department of Data Science–Biostatistics
Tempus Labs
Chicago, IL
farahnaz.islam@tempus.com

James F. Thrasher
Department of Health Promotion, Education & Behavior
University of South Carolina
Columbia, SC
thrasher@mailbox.sc.edu

Feifei Xiao
Department of Biostatistics
University of Florida
Gainesville, FL
feifeixiao@ufl.edu

Robert R. Moran
Department of Epidemiology & Biostatistics
University of South Carolina
Columbia, SC
rrmoran@mailbox.sc.edu

James W. Hardin
Department of Epidemiology & Biostatistics
University of South Carolina
Columbia, SC
jhardin@sc.edu

Abstract. In this article, we present software that is suitable for use with Stata's choice modeling suite of commands, which begin with `cm`. Within the context of choice models, we focus on best–worst data. In such data, respondents are presented a set of choices and are required to select a best and a worst choice from among the alternatives. Optionally, respondents may indicate an opt-out choice, in which no best or worst choice exists in the choice set. Such data are simplified versions of experiments in which respondents rank all the choices. Once best–worst data are collected, there are specific types of data expansions that analysts use to take advantage of both explicit and implicit information. The commands described in this article support data expansion and model estimation.

Keywords: st0735, cm_expand, cm_bwpairs, cm_bwsumm, cm_bestworst, choice models, postestimation, attributes, discrete choice experiments, best–worst, maxdiff choice models

1 Introduction

A discrete choice experiment (DCE) is a quantitative technique used to elicit preferences of individuals in hypothetical scenarios. This technique allows researchers to understand how individuals value characteristics of a product or service and tradeoffs that individuals are willing to make between these characteristics. Standard DCEs ask individuals to select the most beneficial or "best" choice from a set of given alternatives, also referred to as a "first–best DCE". Modern approaches to DCEs involve asking individuals to make multiple selections from one choice set, thereby increasing the precision of estimates and statistical power (Louviere, Flynn, and Marley 2015; Huls et al. 2022).

"Best–worst DCEs" are one such modern approach, wherein individuals select the best and worst choices from a choice set of at least three alternatives. These experiments have been used in a wide range of disciplines, such as healthcare (Flynn et al. 2007), social care (Potoglou et al. 2011), marketing (Louviere et al. 2013), transport (Teffo, Earl, and Zuidgeest 2019), and environmental economics (Scarpa et al. 2011). Best–worst DCEs have been proposed to reduce an individual's cognitive burden compared with a full-ranked DCE. Furthermore, best–worst data can be expanded to add implicit information to the original explicit information as introduced by Lancsar, Fiebig, and Hole (2017). The authors of the cited article do a good job of introducing the overall topic and provide a lucid introduction to the issues with best–worst data. However, these sources lack specific instructions for individual researchers to create expanded datasets using any of the three software programs (Stata, Nlogit, and Biogeme) that are highlighted across these articles.

In this article, we present a new data management command to create and manage expanded datasets within Stata's choice modeling suite of commands (`help cm`). Additionally, we demonstrate how to run estimation commands on the expanded data and what is gained by including the implicit information in the analyses. We also present commands to fit maxdiff choice models for best–worst data. In section 2, we review the layout of best–worst datasets and other choice model data. Then, in section 3, we present syntax for the new commands, followed by examples in section 4.

2 Data

To use Stata's `cm` suite and the commands developed herein, users must first organize the data in long form; Stata users can find further information in the help files and examples for the `reshape` command. In long form, best–worst data are characterized by a collection of observations (the choice set) for which decisions about best and worst are indicated for each observation (choice). Our new commands require an indicator variable for the best choice and another indicator variable for the worst choice.

To motivate the discussion of expanded data, let's first illustrate the necessary data expansion of data suitable for rank-ordered logistic regression into data suitable for conditional logistic regression. We consider a simple dataset in which three different persons identified by `caseid` are each shown a choice set of four alternatives denoted by the `alt` variable. Each alternative is defined by different levels of several characteristics, known as attributes. These attributes and their levels are denoted by the `X1` and `X2` variables. Each person's ranks for his or her set of alternatives are saved as `1` = worst to `4` = best in the `rank` variable.

```
. list, abbrev(10) sepby(caseid)
```

	caseid	set	rank	alt	x1	x2
1.	100	1	1	4	1	1
2.	100	1	2	2	0	1
3.	100	1	3	3	0	0
4.	100	1	4	1	1	0
5.	101	1	1	1	3	0
6.	101	1	2	2	0	1
7.	101	1	3	3	2	1
8.	101	1	4	4	1	2
9.	102	1	1	2	1	1
10.	102	1	2	1	1	1
11.	102	1	3	3	0	1
12.	102	1	4	4	1	0

Following the random-utility model, the utility that person i derives from choosing alternative j in choice set s is given by

$$U_{isj} = V_{isj} + \epsilon_{isj}$$

where V_{isj} is the systematic component of the utility and ϵ_{isj} is the random-error term. We can further distinguish the systematic component as

$$V_{isj} = \alpha_j + \mathbf{X}_{isj}^{\mathrm{T}}\boldsymbol{\beta} + \mathbf{Z}_i^{\mathrm{T}}\boldsymbol{\gamma}_j$$

where $\mathbf{X}_{isj}$ is a vector of alternate-specific covariates; $\mathbf{Z}_i$ is a vector of person-specific covariates; and α, $\boldsymbol{\beta}$, and $\boldsymbol{\gamma}$ are parameters to be estimated.

The probability that person i chose alternative j from set s is given by

$$P_{isj} = P(y_{is} = j) = \frac{\exp(\lambda V_{isj})}{\sum_{\ell=1}^{j} \exp(\lambda V_{is\ell})}$$

Let's focus on the information for the first person listed above. This particular person ranked alternative 4 as the worst, alternative 2 as the second worst, alternative 3 as the second best, and alternative 1 as the best. We could analyze these data using the following sequence of commands to estimate associations for these fully ranked data.

```
. cmset caseid set alt
note: case identifier _caseid generated from caseid and set.
note: panel by alternatives identifier _panelaltid generated from caseid and
      alt.
                   Panel data: Panels caseid and time set
             Case ID variable: _caseid
        Alternatives variable: alt
Panel by alternatives variable: _panelaltid (strongly balanced)
                Time variable: set, 1 to 1
                        Delta: 1 unit
Note: Data have been xtset.
. cmrologit rank x1 x2, nolog
note: data were cmset as panel data, and the default vcetype for panel data is
      vce(cluster caseid); see cmrologit.
Rank-ordered logit choice model                 Number of obs     =         12
Case ID variable: _caseid                       Number of cases   =          3
Ties adjustment: No ties in data                Obs per case:
                                                              min =          4
                                                              avg =       4.00
                                                              max =          4
                                                Wald chi2(2)      =      64.84
Log pseudolikelihood = -8.860207                Prob > chi2       =     0.0000
                                 (Std. err. adjusted for 3 clusters in caseid)
------------------------------------------------------------------------------
             |               Robust
        rank | Coefficient  std. err.      z    P>|z|     [95% conf. interval]
-------------+----------------------------------------------------------------
          x1 |  -.5377402   .5345549    -1.01   0.314    -1.585449    .5099682
          x2 |  -.5926888   2.127354    -0.28   0.781    -4.762226    3.576848
------------------------------------------------------------------------------
```

In this approach, the probability of the ranks given by the first person is the product of the conditional probabilities of the sequence of ranks describing how a person might choose the worst alternative from a continuously shrinking set of choices.

$$\frac{\exp(\lambda\mathbf{V}_{114})}{\sum_{\ell=\{1,2,3,4\}}\exp(\lambda\mathbf{V}_{11\ell})}\frac{\exp(\lambda\mathbf{V}_{112})}{\sum_{\ell=\{1,2,3\}}\exp(\lambda\mathbf{V}_{11\ell})}\frac{\exp(\lambda\mathbf{V}_{113})}{\sum_{\ell=\{1,3\}}\exp(\lambda\mathbf{V}_{11\ell})}\frac{\exp(\lambda\mathbf{V}_{111})}{\sum_{\ell=\{1\}}\exp(\lambda\mathbf{V}_{11\ell})}$$

The contribution of these rankings to the total log likelihood is given by the sum of the logs of the terms.

We can expand the fully ranked data into a presentation that assumes that a person selects the best option out of all alternatives and then continuously reduces the choice set by removing the best choice and again choosing the best from the remaining alternatives. Under that assumption, the data for `caseid` = 100 look like this:

```
. list if caseid==100, abbrev(10) sepby(excaseid)

     ┌──────────────────────────────────────────────────────────┐
     │ set   alt   x1   x2   best   excaseid   caseid   rank │
     ├──────────────────────────────────────────────────────────┤
  1. │   1     4    1    1      0          1      100      1 │
  2. │   1     2    0    1      0          1      100      2 │
  3. │   1     3    0    0      0          1      100      3 │
  4. │   1     1    1    0      1          1      100      4 │
     ├──────────────────────────────────────────────────────────┤
 13. │   1     4    1    1      0          2      100      1 │
 14. │   1     2    0    1      0          2      100      2 │
 15. │   1     3    0    0      1          2      100      3 │
     ├──────────────────────────────────────────────────────────┤
 22. │   1     4    1    1      0          3      100      1 │
 23. │   1     2    0    1      1          3      100      2 │
     ├──────────────────────────────────────────────────────────┤
 28. │   1     4    1    1      1          4      100      1 │
     └──────────────────────────────────────────────────────────┘
```

Note that we have created a new variable, `excaseid`, to denote the expanded choice sets within the choice set of the original `caseid`. We can then run a conditional logistic regression on the expanded data to get the exact same result from the rank-ordered logistic regression:

```
. quietly cmset caseid excaseid alt

. cmclogit best x1 x2, nolog noconstant
note: data were cmset as panel data, and the default vcetype for panel data is
      vce(cluster caseid); see cmclogit.
note: 3 cases dropped because they have only one alternative.
note: variable x1 has 1 case that is not alternative-specific; there is no
      within-case variability.
note: variable x2 has 3 cases that are not alternative-specific; there is no
      within-case variability.

Conditional logit choice model                 Number of obs      =         27
Case ID variable: _caseid                      Number of cases    =          9

Alternatives variable: alt                     Alts per case: min =          2
                                                              avg =        3.0
                                                              max =          4

                                                  Wald chi2(2)    =      64.84
Log pseudolikelihood = -8.8602072                 Prob > chi2     =     0.0000

                                  (Std. err. adjusted for 3 clusters in caseid)
───────────────────────────────────────────────────────────────────────────────
             │               Robust
        best │ Coefficient  std. err.      z    P>|z|     [95% conf. interval]
─────────────┼─────────────────────────────────────────────────────────────────
alt          │
          x1 │  -.5377402   .5345549    -1.01   0.314    -1.585449    .5099682
          x2 │  -.5926888   2.127354    -0.28   0.781    -4.762226    3.576848
───────────────────────────────────────────────────────────────────────────────
```

The coefficients in these two approaches are related to best choice or increasing ranks. Note that we could reverse the ranks in the original dataset. If we do that, we can expand the data assuming that a person selects the worst option out of all alternatives and then continuously reduces the choice set by removing the worst choice and again choosing the worst from the remaining alternatives. The coefficients in the models under that approach would relate to worst choice or decreasing ranks.

We can apply a similar technique for datasets with only partially ranked choices, such as the popular best–worst data. For the sake of exposition, let's alter a dataset that is used in Stata's choice modeling documentation (StataCorp 2023). In the data section of that documentation, there is a summary of the information collected from a number of car consumers. Here the consumers are identifiable by the `consumerid` variable, and each consumer can consider up to four cars distinguished by the country of manufacture (`car`). The car among the choices that was selected is recorded in the `purchase` indicator variable. The data are in the long format with up to four rows of data for each consumer that indicate car-specific (`dealers`) and consumer-specific (`gender`, `income`) information about each of the cars from which the consumer made a selection identified by the `purchase` variable being set to 1. For illustration, imagine that this dataset now also includes a variable indicating the car that the consumer least liked (`least`).

```
. use https://www.stata-press.com/data/r18/carchoice, clear
(Car choice data)
. generate byte least = 0
. replace least = 1 in 3
(1 real change made)
. list consumerid purchase least car if consumerid==1,
> sepby(consumerid) abbrev(10)
```

	consumerid	purchase	least	car
1.	1	1	0	American
2.	1	0	0	Japanese
3.	1	0	1	European
4.	1	0	0	Korean

Let's focus on the information for the first consumer listed above. This particular consumer chose American as the best when presented with the choice set {American, Japanese, European, Korean}. This is the explicit information gained from the experiment. However, the result of this choice in this choice set implies that this particular consumer would have chosen American from any of these six other choice sets: {American, Japanese}, {American, European}, {American, Japanese, European}, {American, Korean}, {American, Japanese, Korean}, {American, European, Korean}. A conditional logistic regression model that also includes the implicit information ultimately includes seven choice sets instead of just the original choice set. This is what the expanded dataset would look like for this first consumer.

```
. cm_expand purchase, clear

. list consumerid purchase car _cmexset if consumerid==1,
> sepby(consumerid _cmexset) abbrev(10)
```

	consumerid	purchase	car	_cmexset
1.	1	1	American	1
2.	1	0	Japanese	1
3.	1	0	European	1
4.	1	0	Korean	1
3161.	1	0	Japanese	886
3162.	1	0	European	886
3163.	1	0	Korean	886
4676.	1	1	American	1771
4677.	1	0	European	1771
4678.	1	0	Korean	1771
6191.	1	1	American	2656
6192.	1	0	Japanese	2656
6193.	1	0	Korean	2656
7706.	1	1	American	3541
7707.	1	0	Japanese	3541
7708.	1	0	European	3541
10361.	1	0	European	4426
10362.	1	0	Korean	4426
11371.	1	0	Japanese	5311
11372.	1	0	Korean	5311
12381.	1	0	Japanese	6196
12382.	1	0	European	6196
14151.	1	1	American	7081
14152.	1	0	Korean	7081
15161.	1	1	American	7966
15162.	1	0	European	7966
16931.	1	1	American	8851
16932.	1	0	Japanese	8851

We can also use the extra information contained in `least` (the "worst" choice) to include additional implied information in the dataset (`_cmexset` is 4426 and 6196). Note how this inferred information is represented.

```
. cm_expand purchase least, clear

. list consumerid purchase least car _cmexset if consumerid==1,
> sepby(consumerid _cmexset) abbrev(10)

        ┌──────────────────────────────────────────────────────┐
        │ consumerid   purchase   least        car   _cmexset  │
        ├──────────────────────────────────────────────────────┤
     1. │          1          1       0   American          1  │
     2. │          1          0       0   Japanese          1  │
     3. │          1          0       1   European          1  │
     4. │          1          0       0     Korean          1  │
        ├──────────────────────────────────────────────────────┤
  3161. │          1          0       0   Japanese        886  │
  3162. │          1          0       1   European        886  │
  3163. │          1          0       0     Korean        886  │
        ├──────────────────────────────────────────────────────┤
  4676. │          1          1       0   American       1771  │
  4677. │          1          0       1   European       1771  │
  4678. │          1          0       0     Korean       1771  │
        ├──────────────────────────────────────────────────────┤
  6191. │          1          1       0   American       2656  │
  6192. │          1          0       0   Japanese       2656  │
  6193. │          1          0       0     Korean       2656  │
        ├──────────────────────────────────────────────────────┤
  7706. │          1          1       0   American       3541  │
  7707. │          1          0       0   Japanese       3541  │
  7708. │          1          0       1   European       3541  │
        ├──────────────────────────────────────────────────────┤
 10361. │          1          0       1   European       4426  │
 10362. │          1          1       0     Korean       4426  │
        ├──────────────────────────────────────────────────────┤
 11371. │          1          0       0   Japanese       5311  │
 11372. │          1          0       0     Korean       5311  │
        ├──────────────────────────────────────────────────────┤
 12381. │          1          1       0   Japanese       6196  │
 12382. │          1          0       1   European       6196  │
        ├──────────────────────────────────────────────────────┤
 14151. │          1          1       0   American       7081  │
 14152. │          1          0       1     Korean       7081  │
        ├──────────────────────────────────────────────────────┤
 15161. │          1          1       0   American       7966  │
 15162. │          1          0       1   European       7966  │
        ├──────────────────────────────────────────────────────┤
 16931. │          1          1       0   American       8851  │
 16932. │          1          0       1   Japanese       8851  │
        └──────────────────────────────────────────────────────┘
```

In the following subsection, we illustrate how the extra information is incorporated into associated models, what assumptions we are making, and what we should expect to gain. Some choice sets will affect only conditional logistic regression models of the best choice, and others will affect only conditional logistic regression models of the worst choice. Finally, there are also best–worst choice models (also called "maxdiff" models) (Cohen 2003) that simultaneously incorporate the information from the best and worst indicated choices for which we have developed additional software. We discuss that software in section 2.2.

2.1 Conditional logistic regression models using expanded data

Let's begin with a conditional logistic regression of the best choice using only the explicit information from the car choice dataset in the previous section.

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

. cmclogit purchase dealers, casevars(i.gender income) nolog cluster(consumerid)
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)    =      51.82
Log pseudolikelihood = -948.12096               Prob > chi2     =     0.0000
                           (Std. err. adjusted for 862 clusters in consumerid)
```

purchase	Coefficient	Robust std. err.	z	P>|z|	[95% conf.	interval]
car						
dealers	.0448082	.026957	1.66	0.096	-.0080266	.097643
American	(base alternative)					
Japanese						
gender						
Male	-.379326	.1715643	-2.21	0.027	-.715586	-.0430661
income	.0154978	.0064881	2.39	0.017	.0027813	.0282142
_cons	-.4787261	.3315883	-1.44	0.149	-1.128627	.171175
European						
gender						
Male	.653345	.2653239	2.46	0.014	.1333197	1.17337
income	.0343647	.0081403	4.22	0.000	.0184101	.0503193
_cons	-2.839606	.4778922	-5.94	0.000	-3.776258	-1.902955
Korean						
gender						
Male	.0679233	.465688	0.15	0.884	-.8448084	.980655
income	-.0377716	.016497	-2.29	0.022	-.0701051	-.0054381
_cons	.0511728	.8645768	0.06	0.953	-1.643367	1.745712

Let's now expand the original data and estimate the same conditional logistic regression using the implied information and the explicit information. One might think that adding the results of the implied choice sets would add power to tests associated with the model fit. However, that assumption is guaranteed only if we assume that all the added (implied) choice sets are independent. That assumption seems untenable, and we should assume that the implied choice sets of a particular respondent would be more highly correlated than two choice sets from different respondents. Thus, an analysis of the expanded dataset should be carried out using standard errors based on a modified sandwich variance estimator (or similar) as opposed to a model-based variance estimator.

```
. cm_expand purchase least, clear

. cmset consumerid _cmexset car
note: case identifier _caseid generated from consumerid and _cmexset.
note: panel by alternatives identifier _panelaltid generated from consumerid
      and car.
note: alternatives are unbalanced across choice sets; choice sets of different
      sizes found.

                   Panel data: Panels consumerid and time _cmexset
             Case ID variable: _caseid
         Alternatives variable: car
Panel by alternatives variable: _panelaltid (unbalanced)
                 Time variable: _cmexset, 1 to 9735, but with gaps
                         Delta: 1 unit

Note: Data have been xtset.
```

```
. cmclogit purchase dealers, casevars(i.gender income) nolog
note: data were cmset as panel data, and the default vcetype for panel data is
      vce(cluster consumerid); see cmclogit.
note: 2343 cases (5175 obs) dropped due to no positive outcome per case.
note: variable dealers has 197 cases that are not alternative-specific; there
      is no within-case variability.
Conditional logit choice model                 Number of obs      =     13,151
Case ID variable: _caseid                      Number of cases    =       4972

Alternatives variable: car                     Alts per case: min =          2
                                                              avg =        2.6
                                                              max =          4

                                                  Wald chi2(7)    =      54.91
Log pseudolikelihood = -3955.8711                 Prob > chi2     =     0.0000

                           (Std. err. adjusted for 885 clusters in consumerid)
------------------------------------------------------------------------------
             |               Robust
    purchase | Coefficient  std. err.      z    P>|z|     [95% conf. interval]
-------------+----------------------------------------------------------------
car          |
     dealers |   .0565795   .0270688     2.09   0.037     .0035256    .1096334
-------------+----------------------------------------------------------------
American     |  (base alternative)
-------------+----------------------------------------------------------------
Japanese     |
      gender |
       Male  |  -.4448359   .1733478    -2.57   0.010    -.7845913   -.1050805
      income |   .0156497    .006634     2.36   0.018     .0026474    .0286521
       _cons |  -.3859363   .3375641    -1.14   0.253     -1.04755    .2756772
-------------+----------------------------------------------------------------
European     |
      gender |
       Male  |   .5651856   .2690416     2.10   0.036     .0378738    1.092497
      income |   .0357682   .0081477     4.39   0.000     .0197991    .0517374
       _cons |  -2.773679   .4740539    -5.85   0.000    -3.702808   -1.844551
-------------+----------------------------------------------------------------
Korean       |
      gender |
       Male  |   .0696396   .4580696     0.15   0.879    -.8281603    .9674395
      income |  -.0374857   .0163405    -2.29   0.022    -.0695126   -.0054588
       _cons |   .1145806    .840377     0.14   0.892    -1.532528    1.761689
------------------------------------------------------------------------------
```

Doing so, we see that the number of observations and number of choice sets increase dramatically from the original explicit information but the standard errors do not change by much. Perhaps even more disconcerting is that the analysis of the original data involved 862 purchasers but the analysis of the expanded data reports 885 clusters. How is this possible? The issue is that Stata's default method for handling missing values is casewise deletion. Let's illustrate the source of these differences.

```
. list consumerid _cmexset purchase car gender income if consumerid==142,
> sepby(consumerid _cmexset) abbrev(10)
```

	consumerid	_cmexset	purchase	car	gender	income
3053.	142	142	0	Japanese	Male	46.6
3054.	142	142	0	American	.	.
3055.	142	142	0	Korean	Male	46.6
3056.	142	142	1	European	Male	46.6
3057.	142	1027	1	European	Male	46.6
3058.	142	1027	0	Japanese	Male	46.6
3059.	142	1027	0	Korean	Male	46.6
3060.	142	1912	1	European	Male	46.6
3061.	142	1912	0	American	.	.
3062.	142	1912	0	Korean	Male	46.6
3063.	142	2797	0	Japanese	Male	46.6
3064.	142	2797	0	American	.	.
3065.	142	2797	0	Korean	Male	46.6
3066.	142	3682	0	American	.	.
3067.	142	3682	0	Japanese	Male	46.6
3068.	142	3682	1	European	Male	46.6
3069.	142	4567	1	European	Male	46.6
3070.	142	4567	0	Korean	Male	46.6
3071.	142	5452	0	Korean	Male	46.6
3072.	142	5452	0	Japanese	Male	46.6
3073.	142	6337	1	European	Male	46.6
3074.	142	6337	0	Japanese	Male	46.6
3075.	142	7222	0	American	.	.
3076.	142	7222	0	Korean	Male	46.6
3077.	142	8107	0	American	.	.
3078.	142	8107	1	European	Male	46.6
3079.	142	8992	0	Japanese	Male	46.6
3080.	142	8992	0	American	.	.

Note that the original information (`_cmexset` is 142) is not included in the first analysis of the original information because of the missing data; that is, participant 142 is not included in the analysis of the original information. For that analysis, the entire choice set is deleted because casewise deletion is the default.

However, the expansion of the data includes some cases that do not have missing information (`_cmexset` is 1,027, 4,567, 5,452, or 6,337). Thus, participant 142 does end up as part of the analysis using expanded data because some of the implied choice sets do not have any missing data. The only way to prevent this is to eliminate the excluded cases prior to data expansion.

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

     Case ID variable: consumerid
Alternatives variable: car

. quietly cmclogit purchase dealers, casevars(i.gender income) nolog
> cluster(consumerid)

. keep if e(sample)  // Important to match casewise deletion in expansion
(85 observations deleted)

. cm_expand purchase least, clear

. cmset consumerid _cmexset car
note: case identifier _caseid generated from consumerid and _cmexset.
note: panel by alternatives identifier _panelaltid generated from consumerid
      and car.
note: alternatives are unbalanced across choice sets; choice sets of different
      sizes found.

                   Panel data: Panels consumerid and time _cmexset
             Case ID variable: _caseid
         Alternatives variable: car
Panel by alternatives variable: _panelaltid (unbalanced)
                Time variable: _cmexset, 1 to 9482, but with gaps
                        Delta: 1 unit

Note: Data have been xtset.
```

```
. cmclogit purchase dealers, casevars(i.gender income) nolog
note: data were cmset as panel data, and the default vcetype for panel data is
      vce(cluster consumerid); see cmclogit.
note: 2327 cases (5143 obs) dropped due to no positive outcome per case.
note: variable dealers has 196 cases that are not alternative-specific; there
      is no within-case variability.
Conditional logit choice model               Number of obs      =     13,025
Case ID variable: _caseid                    Number of cases    =       4917

Alternatives variable: car                   Alts per case: min =          2
                                                            avg =        2.6
                                                            max =          4

                                                Wald chi2(7)    =      53.58
Log pseudolikelihood = -3925.3923               Prob > chi2     =     0.0000

                           (Std. err. adjusted for 862 clusters in consumerid)
------------------------------------------------------------------------------
             |               Robust
    purchase | Coefficient  std. err.      z    P>|z|     [95% conf. interval]
-------------+----------------------------------------------------------------
car          |
     dealers |   .0541101   .0273313     1.98   0.048     .0005417    .1076786
-------------+----------------------------------------------------------------
American     |  (base alternative)
-------------+----------------------------------------------------------------
Japanese     |
      gender |
       Male  |  -.4444872   .1744817    -2.55   0.011    -.7864651   -.1025094
      income |   .0157422   .0066658     2.36   0.018     .0026776    .0288068
       _cons |  -.3944877   .3393546    -1.16   0.245    -1.059611    .2706352
-------------+----------------------------------------------------------------
European     |
      gender |
       Male  |   .5511717   .2698608     2.04   0.041     .0222542    1.080089
      income |   .0355843   .0081928     4.34   0.000     .0195268    .0516418
       _cons |  -2.764961   .4765124    -5.80   0.000    -3.698908   -1.831014
-------------+----------------------------------------------------------------
Korean       |
      gender |
       Male  |   .0727488   .4578972     0.16   0.874    -.8247132    .9702109
      income |  -.0373879   .0162645    -2.30   0.022    -.0692657   -.0055101
       _cons |   .1080895   .8384374     0.13   0.897    -1.535218    1.751397
------------------------------------------------------------------------------
```

The analysis of the expanded data now reports that the number of clusters is 862, which matches the number of cases in the analysis of the original information. There is no issue if you are running the conditional logistic regressions and specifying the `altwise` option for alternativewise deletion. Under that approach, only the specific observation with missing information is deleted instead of the entire case.

2.2 Best–worst or maxdiff models

The following discussion uses NMVR Project Team (2021) materials developed to support Aizaki, Nakatani, and Sato (2015). We specifically use the datasets from these materials to illustrate the Stata support programs that we developed and to motivate discussion of related models and data summaries.

In addition to fitting individual models on best and worst choices, we can consider a best–worst model that evaluates the maximum difference; that is, the best–worst model evaluates the difference of the best and worst choices. Three implementations of this idea are given by

$$\Pr(\text{best} = i, \text{worst} = j) = \frac{\exp(V_i - V_j)}{\sum_{p,q|p \neq q} \exp(V_p - V_q)} \tag{1}$$

$$\Pr(\text{best} = i, \text{worst} = j) = \frac{\exp(V_i)}{\sum \exp(V_p)} \frac{\exp(-V_j)}{\sum \exp(-V_q)} = \frac{\exp(V_i - V_j)}{\sum_{p,q} \exp(V_p - V_q)} \tag{2}$$

$$\Pr(\text{best} = i, \text{worst} = j) = \frac{\exp(V_i)}{\sum \exp(V_p)} \frac{\exp(-V_j)}{\sum_{q \neq i} \exp(-V_q)} = \frac{\exp(V_i - V_j)}{\sum_{p,q|q \neq i} \exp(V_p - V_q)} \tag{3}$$

Modeling the maximum difference between the best and worst choices can be seen as modeling a choice among all the pairs of "best minus worst" differences. Similarly to the application of the `cm_expand` command, we develop another command to expand each choice set into the set of all pairwise differences, `cm_bwpairs`. Note that these three models are actually the same model but applied to different subsets of the expanded collection of pairwise differences. For users who want to fit other models using the original data form, we have developed `cm_bestworst`, which will fit these best–worst models without altering the data.

3 Syntax

Software accompanying this article includes the data management command files, estimation command files, and supporting files for prediction and helps.

The basic syntax to expand best or worst choice data (the dataset has one indicator variable indicating the best or worst choice) is given by

`cm_expand` *choicevar* [`, clear`]

For best–worst data (the dataset has one indicator variable for the best choice and another variable for the worst choice), use the syntax

`cm_expand` *bestvar* *worstvar* [`, clear`]

The `clear` option is required if the data in memory have not already been saved.

Before we restructure the data using the `cm_bwpairs` dataset conversion, we can summarize the best–worst choices using

`cm_bwsumm` *bestvar worstvar* [*if*] [*in*]

This will provide a summary table where b_i records the number of times that the ith choice was chosen as the best, w_i records the number of times that the ith choice was chosen as the worst, and bw_i is the difference; that is, $bw_i = b_i - w_i$. The mean and standard deviation (across all participants) are also calculated and thus can be graphed.

The syntax for `cm_bwpairs` dataset conversion to use for maxdiff models is

`cm_bwpairs` *varlist* [`, replace best(`*bestvar*`) worst(`*worstvar*`)`]

Because this command changes the data in memory, the `replace` option is required if the data have not been saved. The resulting transformed dataset converts each variable listed in the *varlist* into the collection of pairwise differences assuming that each observation could be selected as either the best choice or the worst choice. Once converted, the individual choices of best and worst are lost and in its place is an indication of the specific best–worst difference that was chosen. Users who choose to convert their dataset may then directly fit the maxdiff models using, for example, the `cmclogit` command and specifying the subset of data using variables created and left behind by the `cm_bwpairs` command (see section 4). Data must be `cmset` before using this command with some form of

```
cmset id set_id alternative
```

and they will be newly `cmset` at the conclusion as specified by

```
cmset id _bw_caseid _bw_altbw
```

Similarly to what is done by the `cmset` command, the `cm_bwpairs` command creates and leaves behind several variables: `_bw_choice` is an indicator of the best–worst difference that was selected, `_bw_samplebw` is an indicator of the subset of data required to fit the model associated with (1), `_bw_samplema` is an indicator of the subset of data required to fit the model associated with (2), `_bw_samplesq` is an indicator of the subset of data required to fit the model associated with (3), `_bw_idb` is the item that was selected as the best, `_bw_idw` is the item that was selected as the worst, `_bw_altbw` identifies the alternative, and `_bw_caseid` identifies the choice set.

The syntax for the `cm_bestworst` estimation command is given by

`cm_bestworst` *varlist* [*if*] [*in*] [*weight*] `, best(`*varname_best*`)`
`worst(`*varname_worst*`) case(`*varname_case*`)` [`bw` <u>`marg`</u>`inal` <u>`seq`</u>`uential`
cmclogit_options]

`fweights`, `iweights`, and `pweights` are allowed; see [U] **11.1.6 weight**. All other options (including those for maximization) are passed to the `cmclogit` command that fits the maxdiff model for `cm_bestworst`.

Data must have been `cmset` to use this command and must be in long form with an indicator for the best choice and an indicator for the worst choice. This command will convert the data into the all-differences format described by the `cm_bwpairs` command and then fit the particular version of the best–worst model requested. After estimation, the dataset will be restored to the original form (unless the user specifies `replace`). This is useful if the user wants to fit other models using the original data form. That said, if the user wants to fit best–worst models, we recommend converting the data using `cm_bwpairs` and fitting models directly (or specifying `replace`) if predictions from the best worst model are needed.

Best–worst analysis can be conducted using any of the definitions above, and it is not difficult to run all three models. There is a slight distinction in the interpretation of the models depending on which denominator is at use, but the distinction rarely matters in practice. That is, the significance of covariates will not change by much. We do recommend that users convert their datasets before running the models so that postestimation is easier. Also, we recommend summarizing the data before they are expanded into the best–worst representation.

4 Examples

Here we use the synthetic dataset from chapter 3 of NMVR Project Team (2021). We point out that the cited online source provides model estimates that use a model-based estimate of variance, whereas our results use a sandwich variance estimator to adjust for the multiple choice sets per individual (Kauermann and Carroll 2000). Our utilization of the sandwich variance estimate is the default variance estimator for these data when they are properly set up using `cmset`.

The synthetic dataset includes the results of a DCE in which individuals select the best and worst attributes of rice from among taste, safety, price, variety, origin, milling, and whether the rice is washfree. In the experiment, individuals are shown sets of four of these attributes at a time and select their best and worst choices from the set. Each person is shown seven different choice sets.

The organization of the dataset provided by the cited source includes two copies of each choice set. In the first copy, the best choice is recorded, and in the second copy, the worst choice is recorded. Because this data organization is not uncommon, we begin with a listing to highlight the various structural variables. In this way, we illustrate how the structural variables need to be changed to our required presentation.

```
. insheet using data1mr.txt, clear
(20 vars, 5,040 obs)

. list obs id alt bw item resb resw res str in 1/12, sepby(str)

     +-----------------------------------------------------+
     | obs   id   alt   bw   item   resb   resw   res   str |
     |-----------------------------------------------------|
  1. |   1    1     1    1      2      7      2     0  1011 |
  2. |   2    1     2    1      3      7      2     0  1011 |
  3. |   3    1     3    1      4      7      2     0  1011 |
  4. |   4    1     4    1      7      7      2     1  1011 |
     |-----------------------------------------------------|
  5. |   5    1     1   -1      2      7      2     1  1012 |
  6. |   6    1     2   -1      3      7      2     0  1012 |
  7. |   7    1     3   -1      4      7      2     0  1012 |
  8. |   8    1     4   -1      7      7      2     0  1012 |
     |-----------------------------------------------------|
  9. |   9    1     1    1      1      1      3     1  1021 |
 10. |  10    1     2    1      2      1      3     0  1021 |
 11. |  11    1     3    1      3      1      3     0  1021 |
 12. |  12    1     4    1      6      1      3     0  1021 |
     +-----------------------------------------------------+
```

Each person identified by `id` is presented with various choice sets identified by `str`. We note that the choice set identifier is actually a combination of a three-digit choice set identifier followed by `1` for the information gathered about the best choice or followed by `2` for the information gathered for the worst choice. The `bw` variable also identifies the copies of these choices sets with `1` for best and `-1` for worst. We do not need two copies of each choice set, so the first step is to delete all the observations for `bw` $= -1$.

We further point out that the choices in the choice set can be identified in two different ways: as an "alternative number" (`alt`) across the enumeration of possible alternatives or as the "item number" (`item`) identifying the specific item from the set of all items that we use across all the choice sets. We also recognize that there are several ways to codify choices, including using an indicator variable set to 1 for the best and worst choices, using a variable set equal to the alternative number of the best and worst choices, or using a variable set equal to the item number of the best and worst choices. In this example dataset, `resb` and `resw` are set equal to the item numbers. However, our commands require indicators of best and worst, so we must generate those necessary indicator variables.

```
. drop if bw==-1
(2,520 observations deleted)

. generate byte best  = cond(resb==., ., resb==item)

. generate byte worst = cond(resw==., ., resw==item)
```

Now that we have the data in the correct layout, we can use `cmset` to communicate that structure to Stata:

```
. cmset id str item
note: case identifier _caseid generated from id and str.
note: panel by alternatives identifier _panelaltid generated from id and item.
note: alternatives are unbalanced across choice sets; at least one choice set
      does not have all possible values of item.
                   Panel data: Panels id and time str
             Case ID variable: _caseid
        Alternatives variable: item
Panel by alternatives variable: _panelaltid (weakly balanced)
                Time variable: str, 1011 to 90071, but with gaps
                        Delta: 1 unit
Note: Data have been xtset.
. sort id str obs
. list id alt item best worst str _caseid _panelaltid variety in 1/12, sepby(str)
```

	id	alt	item	best	worst	str	_caseid	_panel~d	variety
1.	1	1	2	0	1	1011	1	2	1
2.	1	2	3	0	0	1011	1	3	0
3.	1	3	4	0	0	1011	1	4	0
4.	1	4	7	1	0	1011	1	7	0
5.	1	1	1	1	0	1021	2	1	0
6.	1	2	2	0	0	1021	2	2	1
7.	1	3	3	0	1	1021	2	3	0
8.	1	4	6	0	0	1021	2	6	0
9.	1	1	1	1	0	1031	3	1	0
10.	1	2	2	0	0	1031	3	2	1
11.	1	3	5	0	1	1031	3	5	0
12.	1	4	7	0	0	1031	3	7	0

```
. label define itemlab 1 "origin" 2 "variety" 3 "price" 4 "taste"
> 5 "safety" 6 "washfree" 7 "milling"
. label values item itemlab
```

Using the `cm_bwsumm` command, we can summarize the best–worst choices:

```
. cm_bwsumm best worst
                                  Summary of best-worst data
Choice                    B         W          BW    mean(BWn)    sd(BWn)
origin                   67       103         -36          -.4   1.816281
variety                  64        97         -33    -.3666667   1.951375
price                   160        39         121     1.344444   2.239388
taste                   125        32          93     1.033333   1.856964
safety                  153        22         131     1.455556   1.824682
washfree                 24       242        -218    -2.422222   2.130452
milling                  37        95         -58    -.6444445   1.717793
Number of subjects = 90
```

The summary table illustrates that price (mean = 1.34), taste (mean = 1.03), and safety (mean = 1.46) are similarly important attributes but that there is more variability across consumers with regard to the importance of price (sd = 2.24). The variety (mean = −0.37), origin (mean = −0.40), and milling (mean = −0.64) do not play substantial roles in distinguishing between best and worst products, while washfree seems to be associated with the worst choice (mean = −2.42).

With these data, we can fit the best–worst model (1) using

```
. local vars origin variety price taste safety milling
. cm_bestworst `vars', best(best) worst(worst) case(str) bw noconstant nolog
note: data were cmset as panel data, and the default vcetype for panel data is
      vce(cluster id); see cmclogit.
note: variable origin has 270 cases that are not alternative-specific; there
      is no within-case variability.
note: variable variety has 270 cases that are not alternative-specific; there
      is no within-case variability.
note: variable price has 270 cases that are not alternative-specific; there is
      no within-case variability.
note: variable taste has 270 cases that are not alternative-specific; there is
      no within-case variability.
note: variable safety has 270 cases that are not alternative-specific; there
      is no within-case variability.
note: variable milling has 270 cases that are not alternative-specific; there
      is no within-case variability.
Conditional logit choice model              Number of obs      =      7,560
Case ID variable: _caseid                   Number of cases    =        630
Alternatives variable: _bw_altbw            Alts per case: min =         12
                                                           avg =       12.0
                                                           max =         12
                                               Wald chi2(6)    =     129.65
Log pseudolikelihood = -1318.0057              Prob > chi2     =     0.0000
                                    (Std. err. adjusted for 90 clusters in id)
```

_bw_choice	Coefficient	Robust std. err.	z	P>|z|	[95% conf.	interval]
_bw_altbw						
origin	1.130956	.2170617	5.21	0.000	.7055227	1.556389
variety	1.10765	.2141616	5.17	0.000	.6879014	1.5274
price	2.01292	.2331173	8.63	0.000	1.556019	2.469822
taste	1.846998	.2393748	7.72	0.000	1.377832	2.316163
safety	2.071936	.2188306	9.47	0.000	1.643036	2.500836
milling	.9602765	.1891358	5.08	0.000	.5895771	1.330976

or we can transform the data to a dataset of the best–worst differences and then fit the appropriate model.

```
. cm_bwpairs `vars', best(best) worst(worst) replace
. list id alt item _bw_idb _bw_idw _bw_choice _caseid variety
> if str==1011, sepby(str) nolabel abbreviate(10)
```

	id	alt	item	_bw_idb	_bw_idw	_bw_choice	_caseid	variety
1.	1	1	2	2	2	0	1	0
2.	1	1	2	2	3	0	1	1
3.	1	1	2	2	4	0	1	1
4.	1	1	2	2	7	0	1	1
5.	1	2	3	3	2	0	1	-1
6.	1	2	3	3	3	0	1	0
7.	1	2	3	3	4	0	1	0
8.	1	2	3	3	7	0	1	0
9.	1	3	4	4	2	0	1	-1
10.	1	3	4	4	3	0	1	0
11.	1	3	4	4	4	0	1	0
12.	1	3	4	4	7	0	1	0
13.	1	4	7	7	2	1	1	-1
14.	1	4	7	7	3	0	1	0
15.	1	4	7	7	4	0	1	0
16.	1	4	7	7	7	0	1	0

In the best–worst differences dataset, the expanded covariate `variety` now reflects the difference of its values for each choice. This same conversion was applied to all the covariates specified in the covariate list of the `cm_bwpairs` command. Having these differences ensures that the parameters are equal across the V_p and V_q terms describing the model. Now we can fit the model directly:

```
. cmclogit _bw_choice `vars' if _bw_samplebw, noconstant nolog
note: data were cmset as panel data, and the default vcetype for panel data is
      vce(cluster id); see cmclogit.
note: variable origin has 270 cases that are not alternative-specific; there
      is no within-case variability.
note: variable variety has 270 cases that are not alternative-specific; there
      is no within-case variability.
note: variable price has 270 cases that are not alternative-specific; there is
      no within-case variability.
note: variable taste has 270 cases that are not alternative-specific; there is
      no within-case variability.
note: variable safety has 270 cases that are not alternative-specific; there
      is no within-case variability.
note: variable milling has 270 cases that are not alternative-specific; there
      is no within-case variability.
Conditional logit choice model                 Number of obs      =      7,560
Case ID variable: _caseid                      Number of cases    =        630

Alternatives variable: _bw_altbw               Alts per case: min =         12
                                                              avg =       12.0
                                                              max =         12

                                                  Wald chi2(6)    =     129.65
Log pseudolikelihood = -1318.0057                 Prob > chi2     =     0.0000

                                    (Std. err. adjusted for 90 clusters in id)
------------------------------------------------------------------------------
             |               Robust
  _bw_choice | Coefficient  std. err.      z    P>|z|     [95% conf. interval]
-------------+----------------------------------------------------------------
_bw_altbw    |
      origin |   1.130956   .2170617     5.21   0.000     .7055227    1.556389
     variety |    1.10765   .2141616     5.17   0.000     .6879014      1.5274
       price |    2.01292   .2331173     8.63   0.000     1.556019    2.469822
       taste |   1.846998   .2393748     7.72   0.000     1.377832    2.316163
      safety |   2.071936   .2188306     9.47   0.000     1.643036    2.500836
     milling |   .9602765   .1891358     5.08   0.000     .5895771    1.330976
------------------------------------------------------------------------------
```

Similarly, by changing the `if` statement, we can run the other best–worst models as described in (2) and (3). Armed with this dataset, we are obviously not just limited to the best–worst models illustrated in this article. For example, we could fit a random-parameters model using the `bayes:` prefix along with the `clogit` command:

```
. set seed 1234
. bayes: clogit _bw_choice `covs' if _bw_samplebw, group(_bw_caseid)

Burn-in ...
Simulation ...
Model summary
------------------------------------------------------------------------------
Likelihood:
  _bw_choice ~ clogit(xb__bw_choice)
Prior:
  {_bw_choice:origin variety price taste safety milling} ~ normal(0,10000) (1)
------------------------------------------------------------------------------
(1) Parameters are elements of the linear form xb__bw_choice.
Bayesian conditional logistic regression         MCMC iterations  =     12,500
Random-walk Metropolis-Hastings sampling         Burn-in          =      2,500
                                                 MCMC sample size =     10,000
                                                 Number of obs    =      7,560
                                                 Acceptance rate  =      .1539
                                                 Efficiency:  min =      .0228
                                                              avg =      .0289
Log marginal-likelihood = -1359.9811                          max =     .03568

------------------------------------------------------------------------------
             |                                                Equal-tailed
  _bw_choice |      Mean   Std. dev.     MCSE     Median  [95% cred. interval]
-------------+----------------------------------------------------------------
      origin |   1.13011   .1208239   .006688   1.123376   .9011242   1.378372
     variety |  1.111733   .1130841   .006615   1.114149   .9046607   1.341101
       price |  2.020179   .1238479   .008202    2.02053   1.772013   2.267436
       taste |  1.860222   .1217593   .007766   1.864548   1.609564   2.081208
      safety |  2.083679   .1259612   .007466   2.085713    1.84003   2.333811
     milling |   .961468   .1161465   .006149   .9606859   .7180063   1.190508
------------------------------------------------------------------------------
Note: Default priors are used for model parameters.
```

5 Conclusions

Researchers who intend to investigate how individuals value characteristics of a product or service use advanced choice modeling techniques applied to DCEs. In those cases where respondents have indicated a best and a worst choice, researchers can fit separate conditional logistic regression models for each of those outcomes. The maxdiff model described herein allows researchers to focus on those attributes associated with the biggest differences between those qualifiers. Depending on the number of choices in a given choice set, there could be other approaches based on rank-ordered logistic regression. More importantly, researchers now allow respondents to opt out of indicating one or the other of the requested choices. How opting out should be treated is complicated, with some researchers dropping the observations and others advocating nested logistic models that start out modeling whether a choice is selected and then modeling associations of attributes with the choice. Future research is required, and we look forward to the development of ever more sophisticated models to address these important data issues.

6 Programs and supplemental material

To install the software files as they exist at the time of publication of this article, type

```
. net sj 23-4
. net install st0735      (to install program files, if available)
. net get st0735          (to install ancillary files, if available)
```

7 References

Aizaki, H., T. Nakatani, and K. Sato. 2015. *Stated Preference Methods Using R*. Boca Raton, FL: Chapman and Hall/CRC. https://doi.org/10.1201/b17292.

Cohen, S. H. 2003. Maximum difference scaling: Improved measures of importance and preference for segmentation. Technical report, Sequim, WA.

Flynn, T. N., J. J. Louviere, T. J. Peters, and J. Coast. 2007. Best–worst scaling: What it can do for health care research and how to do it. *Journal of Health Economics* 26: 171–189. https://doi.org/10.1016/j.jhealeco.2006.04.002.

Huls, S. P., E. Lancsar, B. Donkers, and J. Ride. 2022. Two for the price of one: If moving beyond traditional single-best discrete choice experiments, should we use best-worst, best-best or ranking for preference elicitation? *Health Economics* 31: 2630–2647. https://doi.org/10.1002/hec.4599.

Kauermann, G., and R. J. Carroll. 2000. The sandwich variance estimator: Efficiency properties and coverage probability of confidence intervals. Collaborative Research Center 386, Discussion Paper 189, Ludwig-Maximilians-Universität München. https://doi.org/10.5282/ubm/epub.1579.

Lancsar, E., D. G. Fiebig, and A. R. Hole. 2017. Discrete choice experiments: A guide to model specification, estimation and software. *PharmacoEconomics* 35: 697–716. https://doi.org/10.1007/s40273-017-0506-4.

Louviere, J., I. Lings, T. Islam, S. Gudergan, and T. Flynn. 2013. An introduction to the application of (case 1) best–worst scaling in marketing research. *International Journal of Research in Marketing* 30: 292–303. https://doi.org/10.1016/j.ijresmar.2012.10.002.

Louviere, J. J., T. N. Flynn, and A. A. J. Marley. 2015. *Best-Worst Scaling: Theory, Methods and Applications*. Cambridge: Cambridge University Press. https://doi.org/10.1017/CBO9781107337855.

Potoglou, D., P. Burge, T. Flynn, A. Netten, J. Malley, J. Forder, and J. E. Brazier. 2011. Best–worst scaling vs. discrete choice experiments: An empirical comparison using social care data. *Social Science and Medicine* 72: 1717–1727. https://doi.org/10.1016/j.socscimed.2011.03.027.

Scarpa, R., S. Notaro, J. Louviere, and R. Raffaelli. 2011. Exploring scale effects of best/worst rank ordered choice data to estimate benefits of tourism in alpine grazing commons. *American Journal of Agricultural Economics* 93: 813–828. https://doi.org/10.1093/ajae/aaq174.

NMVR Project Team. 2021. Non-market valuation with R. http://lab.agr.hokudai.ac.jp/nmvr/.

StataCorp. 2023. *Stata 18 Choice Models Reference Manual*. College Station, TX: Stata Press.

Teffo, M., A. Earl, and M. Zuidgeest. 2019. Understanding public transport needs in Cape Town's informal settlements: A best-worst-scaling approach. *Journal of the South African Institution of Civil Engineering* 61: 39–50. http://doi.org/10.17159/2309-8775/2019/v61n2a4.

About the authors

Farahnaz Islam is a recent PhD graduate from the University of South Carolina and is currently a senior biostatistician in the Department of Data Science–Biostatistics, Tempus Labs, Chicago, IL.

James F. Thrasher is a professor in the Department of Health Promotion, Education, and Behavior at the University of South Carolina, Columbia, SC.

Feifei Xiao is an associate professor in the Department of Biostatistics at the University of Florida, Gainesville, FL.

Robert R. Moran is an associate professor in the Department of Epidemiology and Biostatistics at the University of South Carolina, Columbia, SC.

James W. Hardin is a professor in the Department of Epidemiology and Biostatistics at the University of South Carolina, Columbia, SC.