

# Tackling question order effects to improve the accuracy of your survey

Technical Brief / August 2022

laterite  
DATA | RESEARCH | ADVISORY

**Do you agree with the United Nation’s endorsement to include nuclear power in the list of renewable energies? Do you think that the United Nations should act more fiercely to tackle climate change?**

**Do you think that you would have answered differently if those questions were asked in reverse order?**

We might think that our values, choices and judgments would not be affected by something as trivial as the order of questions. But behavioral science has shown that the way we answer and interpret questions is affected by context and biases. The tendency to answer the same question differently merely due to the order in which the questions are asked is known as **order effect** or **order bias** (Schwarz & Sudman (eds.) 1992).

## What triggers order effects?

Order effects might occur for various reasons (Bradburn & Mason, 1964):

- Asking more detailed questions first can influence a person’s responses to a similar, but less detailed, question that follows.
- A respondent might feel a need to be consistent to previous answers or start doubting their consistency with answers to previous questions.
- A respondent might perceive questions on overlapping subjects to be repetitive, influencing their answers to questions that come later in the survey.

## Using permutations to randomize survey elements

One technique that does help minimize this bias is to randomize the order of questions (as long as the questions are independent and their order does not matter for the logic of the survey).

As an example, let’s look at survey questions taken from the New General Efficacy Scale (Chen et al. 2001), designed to assess a respondent’s belief whether he or she can reach set goals, despite whatever difficulties one may encounter:

*“Please listen to each of the following statements. Think about how each statement relates to your life, and then tell me how much you agree or disagree with the statement.”*

The respondent is then asked about a series of eight statements, for example:

*“I will be able to achieve most of the goals that I have set for myself.”*

*“Even when things are tough, I can perform quite well.”*

*“I believe I can succeed at most any endeavor to which I set my mind.”*

Randomizing the order of statements gives the researcher some certainty that aggregate responses are not influenced by the order in which questions were asked. Note that we consider here bias relating to the position of questions, rather than the other order effects relating to the positioning of specific words within a question, or the order of answers in a choice list.

In practice, this can quickly snowball into a massive task. Let’s say you have to randomize the order of eight questions. Even though the task seems easy, you quickly realize that coding all possible arrangements (or *permutations*) of these eight questions is challenging, since there are 40,320 possible permutations!

Challenging but not impossible. Below we have set out several steps outlining how to do this using SurveyCTO and Stata.

### Step 1: Generate all possible permutations

SurveyCTO users will know that the program has an in-built function to randomize answer choices, but it doesn’t have a system to randomize the order of questions.

Using a combination of Stata and SurveyCTO avoids the need to hard code all possible permutations to randomize the question order. The idea is to load into SurveyCTO a CSV file containing all (40,320) possible arrangements. In our case:

1 2 3 4 5 6 7 8

1 2 3 4 5 6 8 7

1 2 3 4 5 7 6 8

Etc. (see Figure 1)

Figure 1. CSV of possible permutations

	A	B	C	D	E	F	G	H	I
1	permutation	v1	v2	v3	v4	v5	v6	v7	v8
15081	15080	3	8	7	5	2	1	6	4
15082	15081	3	8	7	5	2	4	1	6
15083	15082	3	8	7	5	2	4	6	1

The CSV file with all permutations can be generated using the following Stata script. The output CSV file can then be loaded as an attachment in SurveyCTO.

Table 1. Stata code to produce list of permutations

```
* Specify the number of elements that you would like to order below
local N 8

* Creates 'N' temporary datasets
forv i = 1/^N' {
    clear
    set obs `N'
    gen v`i' = _n
    tempfile temp`i'
    save `temp`i"', replace
}

* Crosses the 'N' temporary datasets
clear
use `temp1'
forv i = 2/^N' {
    cross using `temp`i'
}

* Drops observations with repeated numbers
forv i = 1/^N' {
    egen how_many_`i' = anycount(v1-v`N'), values(`i')
    drop if how_many_`i' > 1
    drop how_many_`i'
}

* Creates the permutation identifier and carries out last cosmetic changes
sort v1-v`N'
gen permutation = _n
order permutation, first
```

## Step 2: Upload the possible arrangements to SurveyCTO

In SurveyCTO, the eight questions are included in the choice lists and are numbered from 1 to 8 (Figure 2).

Figure 2. Addition of choices related to the select\_one field

	A	B	C	D
1	list_name	value	label:english	label:amharic
2	statements		1   I will be able to achieve most of the goals that I have set for myself:	እኔ ለራሴ ያወጣኋቸውን አብዛኞቹን ግቦች ማሳካት እችላለሁ
3	statements		2   When facing difficult tasks, I am certain that I will accomplish them:	አስቸጋሪ ሥራዎችን በምጋፈጠብት ጊዜ እነርሱን አንደምፈጽም እርግጠኛ ነኝ
4	statements		3   In general, I think that I can obtain outcomes that are important to me:	በአጠቃላይ ፣ ለእኔ አስፈላጊ የሆኑ ውጤቶችን ማግኘት የምችል ይመስለኛል
5	statements		4   I believe I can succeed at most any endeavor to which I set my mind:	አንድ ነገር አልጫ ከተነሱ በደንበ እንደሚሳካልኝ አምናለሁ
6	statements		5   I will be able to successfully overcome many challenges:	ብዙ ተግዳሮቶችን በተሳካ ሁኔታ ማሸነፍ እችላለሁ
7	statements		6   I am confident that I can perform effectively on many different tasks:	በተለያዩ የስራ ዘርፎች ላይ ውጤታማ እንደምሆን አተማመናለሁ
8	statements		7   Compared to other people, I can do most tasks very well:	ከሌሎች ሰዎች ጋር ሲነጻጸር ፣ አብዛኞቹን ተግባራት በደንብ መሥራት እችላለሁ
9	statements		8   Even when things are tough, I can perform quite well:	ነገሮች ከባድ በሚሆኑበት ጊዜ አንኳን እኔ ነገሮችን በጥሩ ሁኔታ ማከናወን እችላለሁ

At the same time, the eight elements are loaded in the survey tool using the select\_one field that has 'no' specified in its relevance (Figure 3). A select\_one field can be used not only to answer a specific question, but also to fetch the labels of specific elements of a choice list. We can see this in the form of the 8 statements in Figure 2.

Figure 3. Addition of a select\_one field to load the statements

1	type	name	label:english	relevance
14	select_one statements	statements	Field used to load the reference to the various statements	no

## Step 3: Have SurveyCTO randomly choose a permutation

Next, a random number generator that is calculated in SurveyCTO chooses one of the permutations from the CSV that was created in Stata (step 1, see Figure 4).

Figure 4. Generation of random number to select permutation

1	type	name	label:english	calculation
15	calculate	permutation_max	Maximum number of permutations. This has to be manually changed.	40320
16	calculate	permutation_number	Random number generator	once(random())
17	calculate	permutation_selection	Selection of permutation based on random number generator	if(\${permutation_number}= 1, \${permutation_max}, int(\${permutation_number}*\${permutation_max}) + 1)

This is then exported and loaded as an attachment to the SurveyCTO survey instrument.

SurveyCTO might then select, for example, the 15,080th permutation, which is equal to:

3 8 7 5 2 1 6 4

This implies that the first element to be displayed will be question 3, the second element will be question 8, and so on. These can be selected through a 'calculate' field as shown in Figure 5.

Figure 5. Selection of relevant elements given the order determined by the permutation

1	type	name	label::english	calculation
18	calculate	statement_1_code	Statement 1: Code	pulldata('question_randomization', 'v1', 'permutation', \${permutation_selection})
19	calculate_he	statement_1_label	Statement 1: Label	jr:choice-name(\${statement_1_code}, '\${statements}')
20	calculate	statement_2_code	Statement 2: Code	pulldata('question_randomization', 'v2', 'permutation', \${permutation_selection})
21	calculate_he	statement_2_label	Statement 2: Label	jr:choice-name(\${statement_2_code}, '\${statements}')
22	calculate	statement_3_code	Statement 3: Code	pulldata('question_randomization', 'v3', 'permutation', \${permutation_selection})
23	calculate_he	statement_3_label	Statement 3: Label	jr:choice-name(\${statement_3_code}, '\${statements}')
24	calculate	statement_4_code	Statement 4: Code	pulldata('question_randomization', 'v4', 'permutation', \${permutation_selection})
25	calculate_he	statement_4_label	Statement 4: Label	jr:choice-name(\${statement_4_code}, '\${statements}')
26	calculate	statement_5_code	Statement 5: Code	pulldata('question_randomization', 'v5', 'permutation', \${permutation_selection})
27	calculate_he	statement_5_label	Statement 5: Label	jr:choice-name(\${statement_5_code}, '\${statements}')
28	calculate	statement_6_code	Statement 6: Code	pulldata('question_randomization', 'v6', 'permutation', \${permutation_selection})
29	calculate_he	statement_6_label	Statement 6: Label	jr:choice-name(\${statement_6_code}, '\${statements}')
30	calculate	statement_7_code	Statement 7: Code	pulldata('question_randomization', 'v7', 'permutation', \${permutation_selection})
31	calculate_he	statement_7_label	Statement 7: Label	jr:choice-name(\${statement_7_code}, '\${statements}')
32	calculate	statement_8_code	Statement 8: Code	pulldata('question_randomization', 'v8', 'permutation', \${permutation_selection})
33	calculate_he	statement_8_label	Statement 8: Label	jr:choice-name(\${statement_8_code}, '\${statements}')

### Step 4: Your randomization is ready to go

Finally, SurveyCTO generates the randomized list of eight statements according to each randomly generated permutation, ready for use in the survey instrument (Figure 6).

Figure 6. Calling and displaying the randomized elements

1	type	name	label::english	label::amharic
34	begin group	g_g8b_1_pf		
35	note	G8_intro_pf	Enumerator, say "Now I'm going to ask you some questions about different feelings you might have. Please listen to each of the following statements. Think about how each statement relates to your life, and then tell me how much you agree or disagree with the statement."	መረጃ ለሰጠህ፡ ይህንን በል/ይ "አሁን አርቢያ ሊኖርት ስለሚችሉ የተለያዩ ስሜቶች ልጠይቅም ነው። እነሱትን ተከታታይነት አባባሉት ያደምጡ። እንዲሁም አባባል ከህይወቶች ጋር እንዲት እንደሚመድ ያስቡት እና በአባባሉ ምን ያህል ይስማሙ ወይንም አይስማሙ እንደሆነ ከ 1 እስከ 5 ባለው ልክት መስረት ይገለጹ። 1 ማለት 'በጣም አልስማማም' ማለት ሲሆን 5 ደግሞ 'በጣም እስማማለሁ' ማለት ነው። "
36	select_one agreement	statement_1	A. \${statement_1_label}	A. \${statement_1_label}
37	select_one agreement	statement_2	B. \${statement_2_label}	B. \${statement_2_label}
38	select_one agreement	statement_3	C. \${statement_3_label}	C. \${statement_3_label}
39	select_one agreement	statement_4	D. \${statement_4_label}	D. \${statement_4_label}
40	select_one agreement	statement_5	E. \${statement_5_label}	E. \${statement_5_label}
41	select_one agreement	statement_6	F. \${statement_6_label}	F. \${statement_6_label}
42	select_one agreement	statement_7	G. \${statement_7_label}	G. \${statement_7_label}
43	select_one agreement	statement_8	H. \${statement_8_label}	H. \${statement_8_label}
44	end group	e_e8b_1_pf		

The outcome of this process is that, once it has been set up, every respondent is asked questions in a different, random order.

## When to use this approach

This approach is straightforward to use, produces relatively efficient code, and works well with different languages. It is also easy to reproduce, and the example given can be quickly modified to accommodate various question types and formats.

However, this approach may be limited in terms of the number of elements you are trying to randomize. If you are trying to randomize a large number of survey elements - in our experience, any more than eight elements, or 50,000 permutations – would exhaust the computing power of SurveyCTO and tablets and lead to software crashes. In this instance, you might want to consider a pseudo-randomization, and reduce the number of permutations considered.

## References

Bradburn, N. & Mason, W. M. (1964). "The effect of question order on responses." *Journal of Marketing Research*, 1(4), 57-61. <https://doi.org/10.2307/3150380>

Chen, G., Gully, S. M. & Eden, D. (2001). "Validation of a new general self-efficacy scale." *Organizational Research Methods*, 4(1), 62-83. <https://doi.org/10.1177/109442810141004>

Schwarz, N. & Sudman, S. (eds) (1992). *Context effects in social and psychological research*. New York: Springer. <https://doi.org/10.1007/978-1-4612-2848-6>

This Technical Brief was written by Matteo Ramina, with contributions by Mercy Chepchirchir Muttai and Rik Linssen.

### Suggested citation:

**Ramina, M. (2022).** Tackling question order effects to improve the accuracy of your survey. Technical Brief. Laterite.

# laterite

DATA | RESEARCH | ADVISORY

From data to policy

Rwanda | Ethiopia | Kenya | Uganda | Tanzania | The Netherlands

[www.laterite.com](http://www.laterite.com)