

## Using fsQCA software to implement generalized AI

Generalized AI, like fuzzy-set Qualitative Comparative Analysis (fsQCA), is set-analytic in nature. Consequently, the two approaches share many operations and procedures. For example, both techniques utilize truth tables to simplify and model combinations of conditions, and both can work with both crisp and fuzzy sets. Thus, it is appropriate (and expedient!) to implement generalized AI as part of the fsQCA package. Generalized AI is implemented in fsQCA beginning with version 4.0, which can be freely downloaded from [www.fsqca.com](http://www.fsqca.com).

The purpose of this appendix is to provide practical instructions regarding the application of generalized AI. For this demonstration I use the data on social movement organizations (“challengers”) published by William Gamson (1990) in *The Strategy of Social Protest*. Gamson developed a sampling frame for social movement organizations in the United States from 1800 to 1945. Table C-1 presents Gamson’s raw data for the 26 SMOs that gained new advantages for their constituents within 15 years of their period of activism. The presence/absence conditions and the outcome are coded as follows:

*burorgiz*: 1 = the challenging group developed a bureaucratic organizational structure; 0 = the group lacked a bureaucratic structure

*lowstatus*: 1 = the challenging group’s constituency was low status (e.g., workers, minorities, etc.); 0 = constituency was not-low status

*displace*: 1 = the challenging group’s goal was to displace a person in a position of power; 0 = non-displacement goals

*help*: 1 = the challenging group received help from an outsider (e.g., from another challenging group); 0 = no help from outsiders

*acceptnc*: 1 = the challenging group won general acceptance as a representative of its constituents; 0 = did not win acceptance

*newadv*: 1 = new advantages accrued to the challenger’s constituency within 15 years of the challenger’s activism; 0 = no new advantages (Gamson 1990: 36)

The outcome, new advantages, is constant across the 26 cases (all are coded “1” on the outcome), consistent with generalized AI’s focus on investigating one well-specified outcome at a time.<sup>1</sup>

[table C-1]

---

<sup>1</sup> Gamson’s ‘negative’ cases were challenging groups that failed to win new advantages for their constituents, due to a wide variety of obstacles and shortcomings.

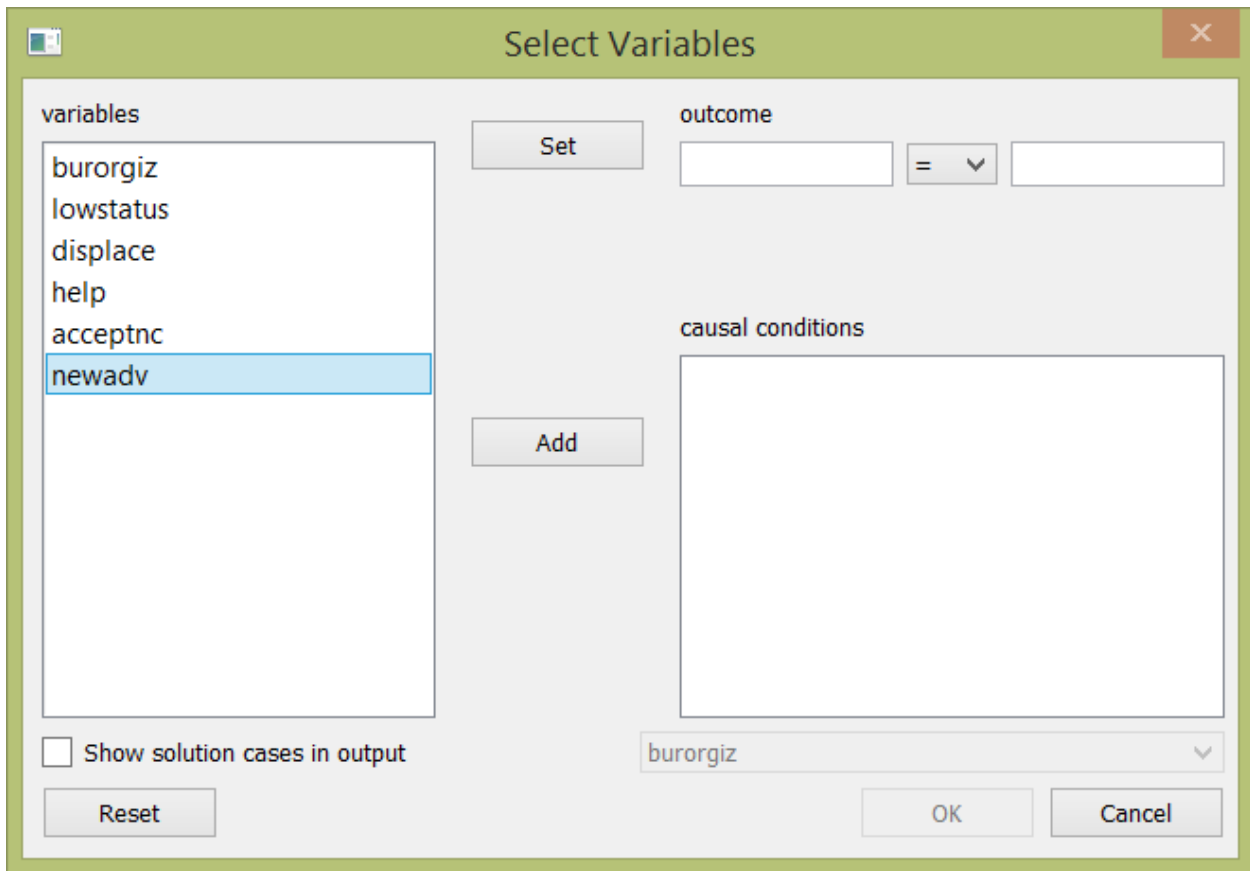
The first step is to make sure the data set is in a proper format for the software. In general, it is best to use simple variable names (three to ten characters), avoiding punctuation, dashes, underscores, and embedded spaces. Data should be numeric, with the exception of a column of case names (often the first column). While data can be entered directly into fsQCA, it is usually easier to use Excel for data entry, saving the file in \*.csv format. Sometimes Excel attaches a blank line at the bottom of the data file. This line must be deleted once the \*.csv file is opened in fsQCA. Save the data file after deleting the blank line—if Excel has inserted one. Move the cursor to the blank line; click “Cases” and then “Delete.”

fsQCA has two main windows, which are opened at start-up. The left window displays the data spreadsheet; the right window displays results. To retrieve a data file, click “File” then “Open.” (To input data directly into the program, consult the fsQCA manual—downloadable from [www.fsqca.com](http://www.fsqca.com).) In addition to \*.csv files, fsQCA also can read tab delimited files (\*.dat) and space delimited files (\*.txt). Sometimes, it is necessary to change the three-letter filename extension, to make the data file recognizable by fsQCA.

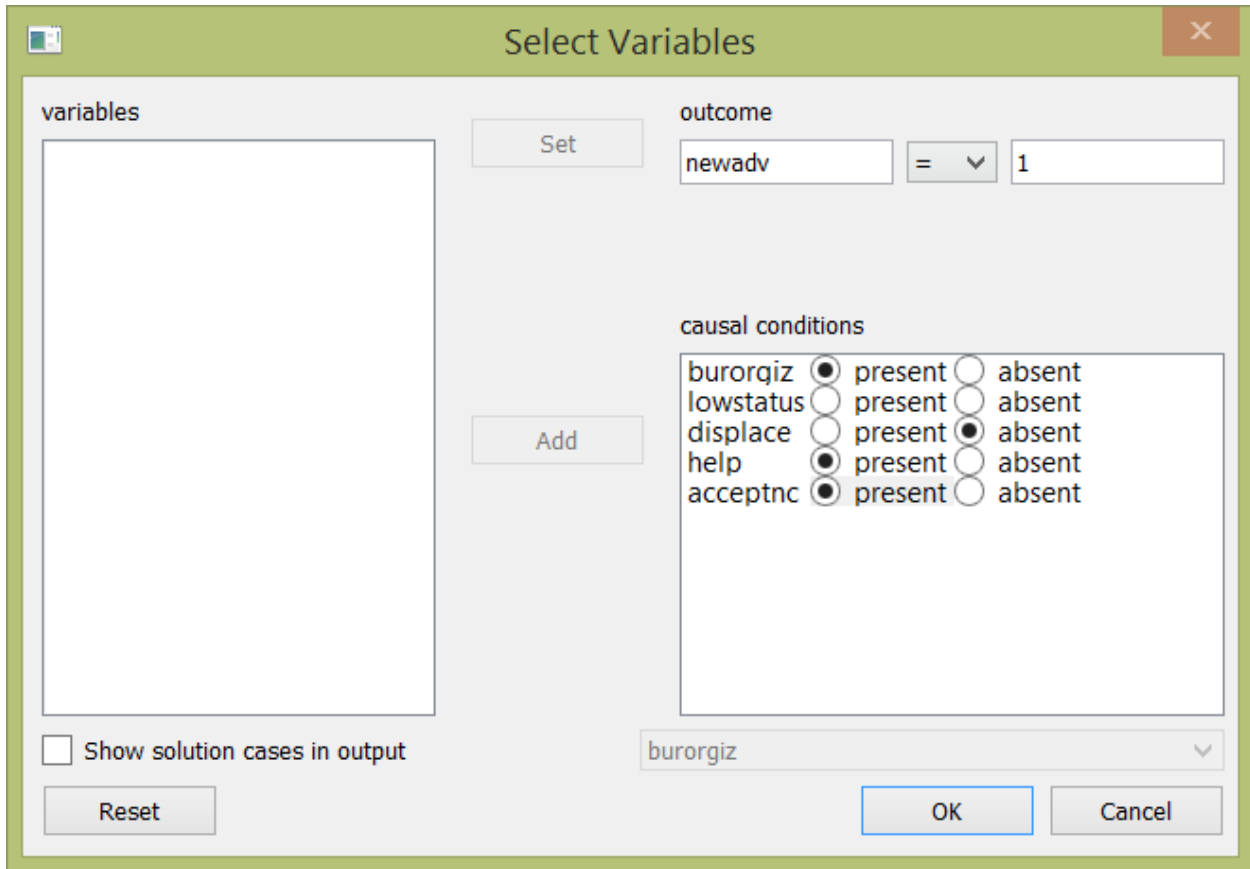
fsQCA offers a variety of data and case functions for manipulating the contents of the data file. For example, there is a calibration procedure for converting interval- and ratio-scale variables to fuzzy sets. This function is very useful when working with conventional survey or archival data (see, e.g., chapter 9). In order to utilize the truth table function, which is central to both generalized AI and QCA, it is necessary for the causally relevant conditions to be crisp sets (i.e., conventional binary variables) or fuzzy sets. The two types are often mixed in the same analysis. The demonstration presented in this appendix uses all crisp sets.

After retrieving the data file, open the generalized AI dialogue box by clicking “Analyze,” and then “Analytic Induction.” Figure C-1 shows a screenshot of the initial generalized AI dialogue box, with all the variables listed on the left. The first task is to define the outcome, which in this example is new advantages (*newadv*). Click the outcome variable and then click “Set.” Notice that there are several ways ( $=$ ,  $<$ ,  $>$ ,  $\leq$ , and  $\geq$ ) to code the outcome, which becomes a constant value of 1 across the selected cases. In this example, the outcome coding is simply  $newadv = 1$ , as shown in figure C-2. It is possible, however, to use interval- and ratio-scale variables to define qualitative outcomes. For example, in chapter 9 the first generalized AI application used an income-to-poverty ratio  $\leq 1.0$  to select the relevant cases and to code them with a constant value of 1 on the outcome.

[figure C-1]



**Figure C-1: Initial AI dialogue box**



**Figure C-2: Coded AI dialogue box**

Selecting and coding the antecedent conditions comes next. Click relevant conditions one at a time, followed by “Add.” Each condition comes with click boxes for “present” versus “absent,” the purpose of which is to implement the interpretive coding of causal conditions. The user clicks “present” if she expects the condition to contribute to the outcome when the condition is present, and “absent” if she expects the condition to contribute to the outcome when the condition is absent. If neither option is selected, the interpretation is that the condition could contribute when it is either present or absent, depending on context (i.e., other conditions).

Figure C-2 shows the interpretive coding of the five antecedent conditions. Bureaucratic organization (*burorgiz*) is coded present, based on the literature on social movement organizations. Low status constituents (*lowstatus*) remains unspecified, consistent with an expectation that its role as a contributing condition is dependent on context. This way of coding *lowstatus* allows for the possibility that the contributions of the other antecedent conditions may differ

depending on whether the challenger's constituency is low status.<sup>2</sup> Having displacement as a primary goal (*displace*) is coded as contributing when absent for the simple reason that it is very rarely successfully achieved. Receiving help from outsiders (*help*) and achieving acceptance (*acceptnc*) as a representative of its constituents are both coded as contributing when present, as indicated in the research literature on social movement organizations.

The setup for the analysis is complete. To produce the truth table based on the setup, click "OK," and the truth table window opens, as shown in figure C-3. Cases are sorted into rows based on their condition profiles, and rows are listed in order of the number of cases in each row, as shown in the "number" column. For example, there are six instances of the first row, cases that combine bureaucratic organization, low-status constituency, non-displacement goals, and acceptance. The percentages in the "number" column refer to the cumulative percentage of cases. The dashes in the table indicate that a condition does not contribute to the outcome, based on the interpretive codings input by the user. As explained in chapter 6, 'contributing versus irrelevant' codings are based on substantive and theoretical knowledge. Note that the outcome is coded 1 for every row, consistent with generalized AI's focus on cases sharing a specific outcome. [figure C-3]

---

<sup>2</sup> Another motivation for allowing *lowstatus* to remain un-coded could just as well be an interest in modelling the differences between SMOs representing low status groups versus SMOs representing moderate or high status groups.

burorgiz	lowstatus	displace	help	acceptnc	number	newadv
1	1	0	-	1	6 (23%)	1
1	1	0	1	1	5 (42%)	1
-	0	0	-	-	2 (50%)	1
-	0	0	1	-	2 (57%)	1
-	0	0	-	1	2 (65%)	1
-	0	0	1	1	2 (73%)	1
-	1	0	1	1	2 (80%)	1
1	1	0	-	-	1 (84%)	1
1	0	0	1	-	1 (88%)	1
1	0	0	-	1	1 (92%)	1
-	0	-	-	1	1 (96%)	1
1	0	0	1	1	1 (100%)	1

**Figure C-3: Initial truth table**

The next step is to select a meaningful frequency threshold, which determines which rows are included in the logical minimization of the truth table. Generally, the threshold should not be so high that many cases are excluded from the logical minimization. The threshold also should not be too low, which might give too much analytic weight to rows that are deviant in some way or perhaps that exist simply due to classification or measurement error.<sup>3</sup> In this example, I use a frequency threshold of at least two cases, which embraces 80% of the cases in the truth table. To implement a numerical threshold, click “Edit” and then

<sup>3</sup> Of course, it is always possible to use multiple frequency thresholds and assess the impact of being more versus less inclusive.

“Delete.” A small dialogue box will open, with the message “Delete rows with number less than \_\_\_\_”. In this example, the input value is 2. Then click “OK.” The resulting truth table is shown in figure C-4.

[figure C-4]

burorgiz	lowstatus	displace	help	acceptnc	number	newadv
1	1	0	-	1	6 (28%)	1
1	1	0	1	1	5 (52%)	1
-	0	0	-	-	2 (61%)	1
-	0	0	1	-	2 (71%)	1
-	0	0	-	1	2 (81%)	1
-	0	0	1	1	2 (90%)	1
-	1	0	1	1	2 (100%)	1

**Figure C-4: Edited truth table**

The truth table is now ready for logical minimization. Click “Run.” The results are displayed in the output window, which was opened at startup. Table C-2 shows the results of the application of the truth table algorithm. In this example, there are three modal configurations linked to new advantages: (1) challengers with not-low-status constituents combined with non-displacement goals, (2) bureaucratically organized challengers with non-displacement goals and acceptance, and (3) challengers that have achieved acceptance combined with non-displacement goals and help from outsiders. The first modal configuration is found in eleven of the 26 instances of new advantages (42.3%); its unique coverage (i.e., not overlapping with the coverage of the other two modal configurations) is 26.9% (seven of 26 instances). The second modal configuration is found in thirteen of 26 cases (50%); its non-overlapping coverage is six

instances (23.1%). Finally, the third modal configuration is found in ten cases, with two cases non-overlapping. Altogether the three modal configurations account for 92.3 % of the instances of new advantages (24 of 26 cases).

[table C-2]

### Clarifying the truth table results

While it is tempting to view the results of the truth table algorithm as the conclusion of the analysis, it is important to interrogate the results further. After all, the results can be expressed as a Boolean equation, which in turn can be manipulated algebraically.

First, consider the results expressed as a logical equation:

$\sim\text{lowstatus} \bullet \sim\text{displace} + \text{burorgiz} \bullet \sim\text{displace} \bullet \text{acceptnc} + \sim\text{displace} \bullet \text{help} \bullet \text{acceptnc} \rightarrow \text{newadv}$   
 Where “ $\rightarrow$ ” indicates the superset/subset relation (inclusion), “ $\bullet$ ” indicates logical *and* (combined conditions), “+” indicates logical *or* (alternate conditions or alternate combinations of conditions), and “ $\sim$ ” indicates *not* (set negation). Note that the second and third modal configurations apply to challengers representing low-status constituencies (*lowstatus*) and also to challengers representing constituencies that are not low status ( $\sim\text{lowstatus}$ ). Thus, these two recipes and the equation for new advantages can be rewritten as follows:

$\sim\text{lowstatus} \bullet \sim\text{displace} + \text{lowstatus} \bullet \text{burorgiz} \bullet \sim\text{displace} \bullet \text{acceptnc} + \sim\text{lowstatus} \bullet \text{burorgiz} \bullet \sim\text{displace} \bullet \text{acceptnc} + \text{lowstatus} \bullet \sim\text{displace} \bullet \text{help} \bullet \text{acceptnc} + \sim\text{lowstatus} \bullet \sim\text{displace} \bullet \text{help} \bullet \text{acceptnc} \rightarrow \text{newadv}$

The next step is important. Two of the terms just added are subsets of the first modal configuration. Specifically,  $\sim\text{lowstatus} \bullet \text{burorgiz} \bullet \sim\text{displace} \bullet \text{acceptnc}$  is included in  $\sim\text{lowstatus} \bullet \sim\text{displace}$ , and  $\sim\text{lowstatus} \bullet \sim\text{displace} \bullet \text{help} \bullet \text{acceptnc}$  is also included in  $\sim\text{lowstatus} \bullet \sim\text{displace}$ . Removing the redundant terms yields:

$\sim\text{lowstatus} \bullet \sim\text{displace} + \text{lowstatus} \bullet \text{burorgiz} \bullet \sim\text{displace} \bullet \text{acceptnc} + \text{lowstatus} \bullet \sim\text{displace} \bullet \text{help} \bullet \text{acceptnc} \rightarrow \text{newadv}$

And then joining the second and third modal configurations yields:

$\sim\text{lowstatus} \bullet \sim\text{displace} + \text{lowstatus} \bullet \sim\text{displace} \bullet \text{acceptnc} \bullet (\text{help} + \text{burorgiz}) \rightarrow \text{newadv}$

The clarified results reveal that there is an important difference between challengers representing low-status constituents and challengers representing constituents who are not low status. If the constituents are not low status, the only ingredient needed for success is a non-displacement goal. However, if the challenger’s constituency is low status, then not only must challengers avoid displacement goals, but they must also win acceptance and either have a bureaucratic organization or help from outsiders. In short, the path to new advantages is much narrower for challengers representing low-status constituents.



Note also that by clarifying the results in this manner, there is no longer overlapping coverage. The first modal configuration covers eleven cases; the second covers thirteen. Total coverage is the same as before: 24/26 (92.3%).