Comparing NCA and QCA

Supplement to Dul, J. (2020) Conducting Necessary Condition Analysis. Sage publications.
Jan Dul
Version 1.0.2; April 27, 2020

Suggested reference:
Dul, J. ([year], [date]). Comparing NCA and QCA. Supplement to 'Dul, J. (2020) Conducting Necessary Condition Analysis, Sage Publications'. Retrieved from http://erim.eur.nl/nca

In this supplement I explain how NCA differs from Qualitative Comparative Analysis (QCA) and how NCA could complement QCA. In another supplement I explain how NCA is different from regression analysis and how NCA can complement regression analysis. These three methods are not competing but rather they can provide complementary insights.

1. Introduction to QCA

1.1 History of QCA

QCA is about 30 years old. The method has its roots in political science and sociology, and was developed by Charles Ragin (1987, 2000, 2008). QCA has steadily evolved and used over the years and currently many types of QCA approaches exist. A common interpretation of QCA is described by Schneider and Wagemann (2012), which I follow in this supplement.

1.2 Logic and theory of QCA

Set theory is in the core of QCA. It means that relations between sets, rather than relations between variables are studied. A case can be part of a set or not part of the set. For example, the Netherlands is a case (of all countries) that is 'in the set' of rich countries, and Ethiopia is a case that is 'out of the set' of rich countries. Set membership scores (rather than variable scores) are linked to a case. Regarding the set of rich countries, the Netherlands has a set membership score of 1 and Ethiopia of 0. In the original version of QCA the set membership scores could only be 0 or 1. This version of QCA is called crisp-set QCA. Later also fuzzy-set QCA (fsQCA) was developed. Here the membership scores can also have values between 0 and 1. For example, Croatia could be allocated a set membership score of 0.7 indicating that it is 'more in the set' than 'out of the set' of rich countries.

In QCA relations between sets are studied. Suppose that one set is the set of rich countries (X), and another set is the set of countries with happy people ('happy countries', Y). QCA uses Boolean (binary) algebra and expresses the relationship between condition X and outcome Y as the presence or absence of X is related to the presence or absence of Y. More specifically, the relations are expressed in terms of sufficiency and necessity. For example, the presence of X (being a country that is part of the set of rich countries) could be theoretically stated as *sufficient* for the presence of Y (being a country that is part of the set of happy countries). All rich countries are happy countries. The set of rich countries is a subset of the set of happy countries. No rich country is not a happy country. Set X is a *subset* of set Y. Alternatively, in another theory it could be stated that the presence of X (being country that is part of the set of rich countries) is *necessary* for the presence of Y (being a country that is part of the set of happy countries). All happy countries are rich countries. The set of happy countries is a superset of the set of rich countries. No happy country is not a rich county. Set X is a *superset* of set Y.

QCA's main interest is about sufficiency. QCA assumes that a *configuration* of single conditions produces the outcome. For example, the condition of being in the set of rich countries (X_1) AND the condition of being in the set of democratic countries (X_2) is sufficient for the outcome of being in the set of happy countries (Y).

QCA's Boolean logic statements for this sufficiency relationship is expressed as follows:

$$X_1{*}X_2 \rightarrow Y \qquad \text{Eq. 1}$$

where the symbol '*' means the logical 'AND', and the symbol '→' means 'is sufficient for'.

Furthermore, QCA assumes that several alternative configurations may exits that can produce the outcome, known as '*equifinality*'. This is expressed in the following example:

$$X_1{*}X_2 + X_2{*}X_3{*}X_4 \rightarrow Y \qquad \text{Eq. 2}$$

where the symbol '+' means the logical 'OR'. It is also possible that the absence of a condition is part of a configuration. This is shown in the following example:

$$X_1{*}X_2 + X_2{*}{\sim}X_3{*}X_4 \rightarrow Y \qquad \text{Eq. 3}$$

where the symbol '~' means 'absence of'. Single conditions in a configuration that is sufficient for the outcome are called INUS conditions (Mackie, 1965). An INUS condition is an 'Insufficient but Non-redundant (i.e., Necessary) part of an Unnecessary but Sufficient condition.' In this expression, the words 'part' and 'condition' are somewhat confusing because 'part' refers to the single condition and 'condition' refers to the configuration that consists of single conditions. **I**nsufficient refers to the fact that a part (single condition) is not itself sufficient for the outcome. **N**on-redundant refers to the necessity of the part (single condition) for the configurations being sufficient for the outcome. **U**nnecessary refers to the possibility that also other configurations can be sufficient for the outcome. **S**ufficient refers to the fact that the configuration is sufficient for the outcome. Although a single condition may be *locally necessary* for the configuration to be sufficient for the outcome, it is not *globally necessary* for the outcome because the single condition may be absent in other sufficient configurations. INUS conditions are thus usually not necessary conditions for the outcome (the latter are the conditions that NCA considers).

Hence, in above generic logical statements about relations between sets (equations 1 - 3) X and Y can only be absent of present (Boolean algebra), even though the individual members of the sets can have fuzzy scores. Both csQCA and fsQCA use logical statements where the condition and the outcome can only be absent (0) or present (1). In fsQCA absence means set membership scores < 0.5 and presence means set membership scores > 0.5.

1.3 Data and data analysis of QCA

Particularly in large N studies and in the business and management field, the starting point of the QCA data analysis is to transform variable scores into set membership scores in a 'mechanistic' way (data driven). The transformation process is called 'calibration'. Calibration can be based on the distribution of the data, the measurement scale, or expert knowledge.

The goal of calibration is to get scores of 0 or 1 (csQCA) or between 0 and 1 (fsQCA) to represent the extent to which the case belongs to the set (set membership score). In fsQCA mechanistic transformation is usually done with the logistic transformation function. The selection is somewhat arbitrary (build in popular QCA software) and moves the variable scores to the extremes (0 and 1) in comparison to just standardization of the data: low values move to 0 and high values move to 1. When no substantive reason exists for the logistic transformation, I have proposed (Dul, 2016) to use a standard transformation. This transformation keeps the distribution of the data intact. The reason for my proposal is that moving the scores to the extremes implies that cases in the XY scatter plot with low to middle values of X move to the left and cases with middle to high values of Y move upwards. As a result, the upper left corner is more filled with cases. Consequently, potential empty spaces (indicating necessity) may not be identifiable. With the standard transformation the cases stay where they are; an empty space in a corner of the XY plot with the original data stays empty. The standard transformation is an alternative to an arbitrary transformation: it just changes variable score into set membership scores, without affecting the distribution of the data. A calibration evaluation tool to check the effect of calibration on the necessity effect size is available on the NCA website.

QCA performs two separate analyses with calibrated data: a necessity analysis for identifying (single) necessary conditions, and a sufficiency analysis ('truth table' analysis) for identifying sufficient configurations. In this supplement I focus on the necessary condition analysis of single necessary conditions, which precedes the sufficiency analysis.

In csQCA the necessity analysis is similar to a dichotomous necessary condition analysis of NCA with the contingency table approach when X and Y are dichotomous set membership scores that can only be present (in the set) or absent (not in the set). By visual inspection of the contingency table a necessary condition 'in kind' can be identified when the upper left cell is empty (Figure 1).

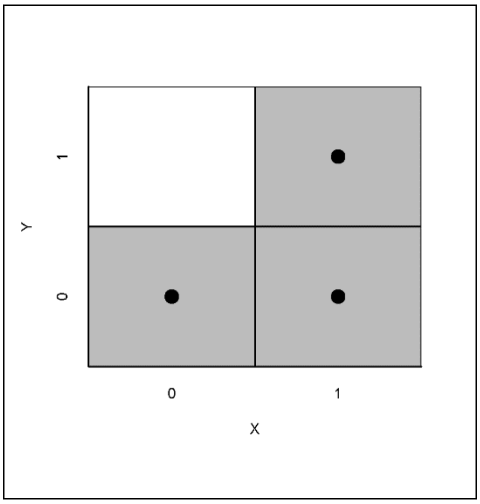

Figure 1 Necessity analysis with crisp set QCA and with NCA using set membership scores 0 and 1.

For fuzzy set membership scores the necessity analyses of fsQCA and NCA differ. In fsQCA a diagonal is drawn in the XY scatter plot (Figure 2A). For necessity, there can be no cases above the diagonal. Necessity consistency is a measure of the extent to which cases are not above the diagonal, which can range from 0 to 1. When some cases are present in the 'empty' zone above the diagonal, fsQCA considers these cases as 'deviant cases'. FsQCA accepts some deviant cases as long as the necessity consistency level, which is computed from the total vertical distances of the deviant cases to the diagonal, is not smaller than a certain threshold, usually 0.9. The necessity consistency is large enough, fsQCA makes a qualitative ('in kind') statement about the necessity of X for Y: 'X is necessary for Y', e.g., the presence of X (membership score > 0.5) is necessary for the presence of Y (membership score > 0.5).

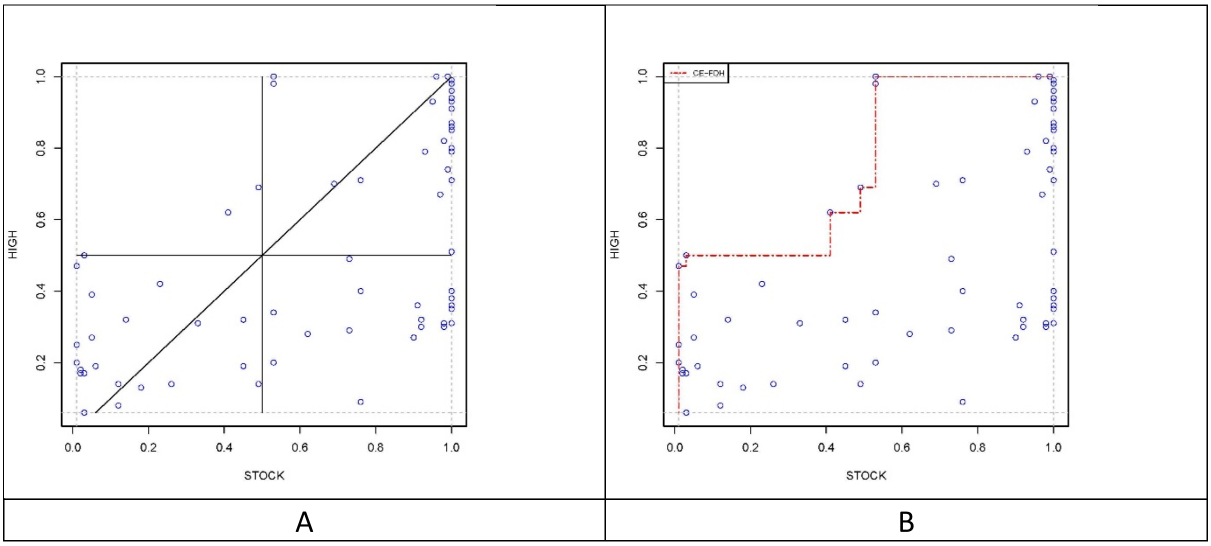

Figure 2 Comparison of necessity analysis with fsQCA's (A) and with NCA (B). Data from Rohlfing, I., & Schneider, C. Q. (2013). See also Vis & Dul (2018).

2. The difference between NCA and QCA

The differences between NCA and QCA are summarized in Table 1 and are discussed below, showing that there are major differences between NCA and QCA (see also Dul, 2016; Vis and Dul, 2018).

Table 1. Comparison of NCA and QCA.

		NCA	QCA

Logic and theory	Logic	- Necessity ('in kind'): Presence/absence of single condition X is necessary for presence/absence of outcome Y. ('in kind')
		- Necessity 'in degree'): Level of single condition X is necessary for level of outcome Y ('in degree')
			- Necessity ('in kind'):
			  - Presence/absence of single condition X is necessary for presence/absence of outcome Y. ('in kind')
			  - Presence/absence of single condition X1 OR Presence/absence of single condition X2 is necessary for presence/absence of outcome Y. ('in kind')
			- Sufficiency ('in kind'): A configuration of present/absent "INUS" conditions X is sufficient but not necessary for

			presence/absence of outcome Y. ('in kind')

	Type of data	- Variable scores with different possible levels:
		  - dichotomous (two levels)
		  - discrete (finite number of levels)
		  - continuous (infinite number of levels)
		- Set membership scores with different possible levels
		  - dichotomous (two levels): for combining with crisp set QCA
		  - discrete (finite number of levels): for combining with discrete fuzzy set QCA;
		  - continuous (infinite number of levels): for combining with fuzzy set QCA
			- Set membership scores with different possible levels
			  - dichotomous (two levels): crisp set QCA
			  - discrete (finite number of levels): discrete fuzzy set QCA;
			  - continuous (infinite number of levels): continuous fuzzy set QCA
			Fuzzy set membership scores are dichotomized for application of Boolean algebra:
			- Presence = set membership score > 0.5
			Absence = set membership score < 0.5

	Mathematics	Linear algebra	Boolean algebra
	Main interest	- Necessity of single factors that enable the outcome ('in degree')	- Sufficiency of configurations that produce the outcome ('in kind')
Data and data analysis	Research Strategy	- Experiment - Small N observational study - Large N observational study	- Small N observational study - Large N observational study
	Case selection/sampling	- Purposive case selection (small N) - Probability sampling (large N)	- Purposive case selection (small N) - Probability sampling (large N)
	Measurement	- Valid and reliable variable scores	- Meaningful set membership scores (with calibration)
	Data analysis	Necessity: - Ceiling line is reference line - Empty space is quantified (effect size), including statistical test with p-value (randomness test) - Bottleneck table	Necessity: - Diagonal is reference line - Consistency is quantified Sufficiency: - Truth table analysis

2.1 Logic and theory

NCA and QCA are only the same in a very specific situation of 'in kind' necessity: A single X is necessary for Y, and X and Y are dichotomous set membership scores (0 and 1). Then the analyses of NCA and QCA are exactly the same. However, NCA normally uses variable scores, but can also set membership scores when NCA is applied in combination with QCA (see below). In addition to the 'in kind' necessity that both methods share, NCA also formulates 'in degree' necessity. QCA also formulates 'in kind' necessity of 'OR' combinations of conditions, as well as 'in kind' sufficiency of configurations of conditions.

The main interest of NCA is the necessity 'in degree' of single factors that enable the outcome, whereas the main interest of QCA is the sufficiency 'in kind' of (alternative) configurations of conditions.

2.2 Data and data analysis

Regarding *research strategy* most NCA studies are observational studies (both small N and large N), although also experiments are possible. Most QCA studies are small N observational studies, although increasingly also large N studies are employed with QCA, in particular in the business and management area. The experimental research strategy is rare (if not absent) in QCA. Regarding *case selection/sampling* purposive sampling is the main case selection strategy in QCA. It is also possible in small N NCA studies. For large N studies sampling strategies such as those used in regression analysis (preferably probability sampling) are used both in NCA and QCA. Regarding *measurement*, NCA uses valid and reliable variable scores unless it is used in combination with QCA, in which case NCA uses calibrated set membership scores. QCA uses calibrated set membership scores and cannot use variable scores. In QCA data with variable scores may be used as input for the 'calibration' process to transform variable scores into set membership scores. Regarding *data analysis* in fsQCA a necessary condition is assumed to exist if the area above the diagonal reference line in an XY scatter plot is virtually empty (see Figure 2A). In contrast, NCA uses the ceiling line as the reference line (see Figure 2B) for evaluating the necessity of X for Y (with possibly some cases above the ceiling line; accuracy below 100%). In situations where fsQCA observes 'deviant cases', NCA includes these cases in the analysis by 'moving' the reference line from the diagonal position to the boundary between the zone with cases and the zone without cases. NCA considers cases around the ceiling line (and usually above the diagonal) as 'best practice' cases rather than 'deviant' cases. These cases are able to reach a high level of outcome (e.g., an output that is desired) for a relatively low level of condition (e.g., an input that requires effort). For deciding about necessity, NCA evaluates the size of the 'empty' zone as a fraction of the total zone (empty plus full zone), which ratio is called the necessity effect size. If the effect size is greater than zero (an empty zone is present), and if according to NCA's statistical test this is unlikely a random result of unrelated X and Y, NCA identifies a necessary condition 'in kind' that can be formulated as: 'X is necessary for Y', indicating that for at least a part of the range of X and the range of Y a certain level of X is necessary for a certain level of Y. Additionally, NCA can quantitatively formulate necessary condition 'in degree' by using the ceiling line: 'level Xc of X is necessary for level Yc of Y'. The ceiling line represents all combinations X and Y where X is necessary for Y. Although also fsQCA's diagonal reference line allows for making quantitative necessary conditions statements, e.g. $X > 0.3$ is necessary for $Y = 0.3$, fsQCA does not make such statements.

When the ceiling line coincides with the diagonal (corresponding to the situation that fsQCA considers) the statement 'X is necessary for Y' applies to all X-levels [0,1] and all Y-levels [0,1] and the results of the qualitative necessity analysis of fsQCA and NCA are the same. When the ceiling line is positioned above the diagonal 'X is necessary for Y' only applies to a specific range of X and a specific range of Y. Outside these ranges X is not necessary for Y ('necessity inefficiency'). Then the results of the qualitative necessity analysis of fsQCA and NCA can be different.

Normally, NCA identifies more necessary conditions than fsQCA, mostly because the diagonal is used as reference line. In the example of Figure 2, NCA identifies that X is necessary for Y because there is an empty zone above the ceiling line. However, fsQCA would conclude that X is not necessary for Y, because the necessity consistency level is below 0.9.

FsQCA's necessity analysis can be considered as a special case of NCA: an NCA analysis with discrete or continuous fuzzy set membership scores for X and Y, a ceiling line that is diagonal, an allowance of a specific number of cases in the empty zone given by the necessity consistency threshold, and the formulation of a qualitative ‘in kind’ necessity statement.

3. Recommendation for combining NCA and QCA

Although in most NCA applications variable scores are used for quantifying condition X and outcome Y, NCA can also employ set membership scores for the conditions and the outcome, allowing to combine NCA and QCA. The other way around is not possible: combining QCA to a regular NCA with variable scores is not possible because by definition QCA is a set theoretic approach that does not use variable scores.

How can NCA with membership scores complement QCA? For answering to this question first another question should be raised: how does QCA integrate an identified necessary condition in kind with the results of the sufficiency analysis: the identified sufficient configurations. By definition the necessary condition must be part of all sufficient configurations, otherwise this configuration cannot produce the outcome. However, within the QCA community five different views exist about how to integrate necessary condition in sufficient configurations. In the first view only sufficient configurations that include the necessary conditions are considered as a result. Hence, all selected configurations have the necessary condition. In the second view the truth table analysis to find the sufficient configurations are done without the necessary conditions and afterwards the necessary conditions are added to the configuration. This also ensures that all configurations have the necessary conditions. In the third view configurations that do not include the necessary condition are excluded from the truth table before this table is further analysed to find the sufficient configurations. This ‘ESA’ approach (Schneider and Wagemann, 2012) also ensures that all configurations have the necessary conditions. In the fourth view sufficient configurations are analysed without worrying about necessary conditions. Afterwards, the necessary conditions are discussed separately. In the fifth view a separate necessity analysis is not done, or necessity is ignored. All views have been employed in QCA; hence no consensus exists yet. And additional complexity of integrating necessity with sufficient configuration is that NCA produces necessary conditions in degree, rather than QCA’s necessary condition and sufficient configurations in kind. The conditions that are part of the sufficient configurations can only be absent or present.

Given these complexities, I suggest, a combination of the second and the fourth view:

1. Perform the NCA analysis ‘in degree’ before QCA’s sufficiency analysis;
2. Integrate a part of the results of NCA’s necessity analysis into QCA’s sufficient configurations, namely the conditions (‘in degree’) that are necessary for outcome > 0.5;
3. Discuss the full results of NCA’s necessity analysis afterwards.

In particular it could be discussed that specific levels of necessary membership scores found by NCA must be present in each sufficient configuration found by QCA. If that membership in degree is not in place, the configuration will not produce the outcome.

4. Examples

I discuss two examples of integrating NCA in QCA according to this recommendation.

The first example is a study by Emmenegger (2011) about the effects of six conditions: S = state-society relationships, C = non-market coordination, L = strength labour movement, R = denomination, P = strength religious parties and V = veto points on JSR = job security regulation in Western European countries. He performed an fsQCA analysis with a necessity analysis and a sufficiency analysis. His necessity analysis showed that no condition was necessary for job security regulation (necessity consistency of each condition was < 0.9).

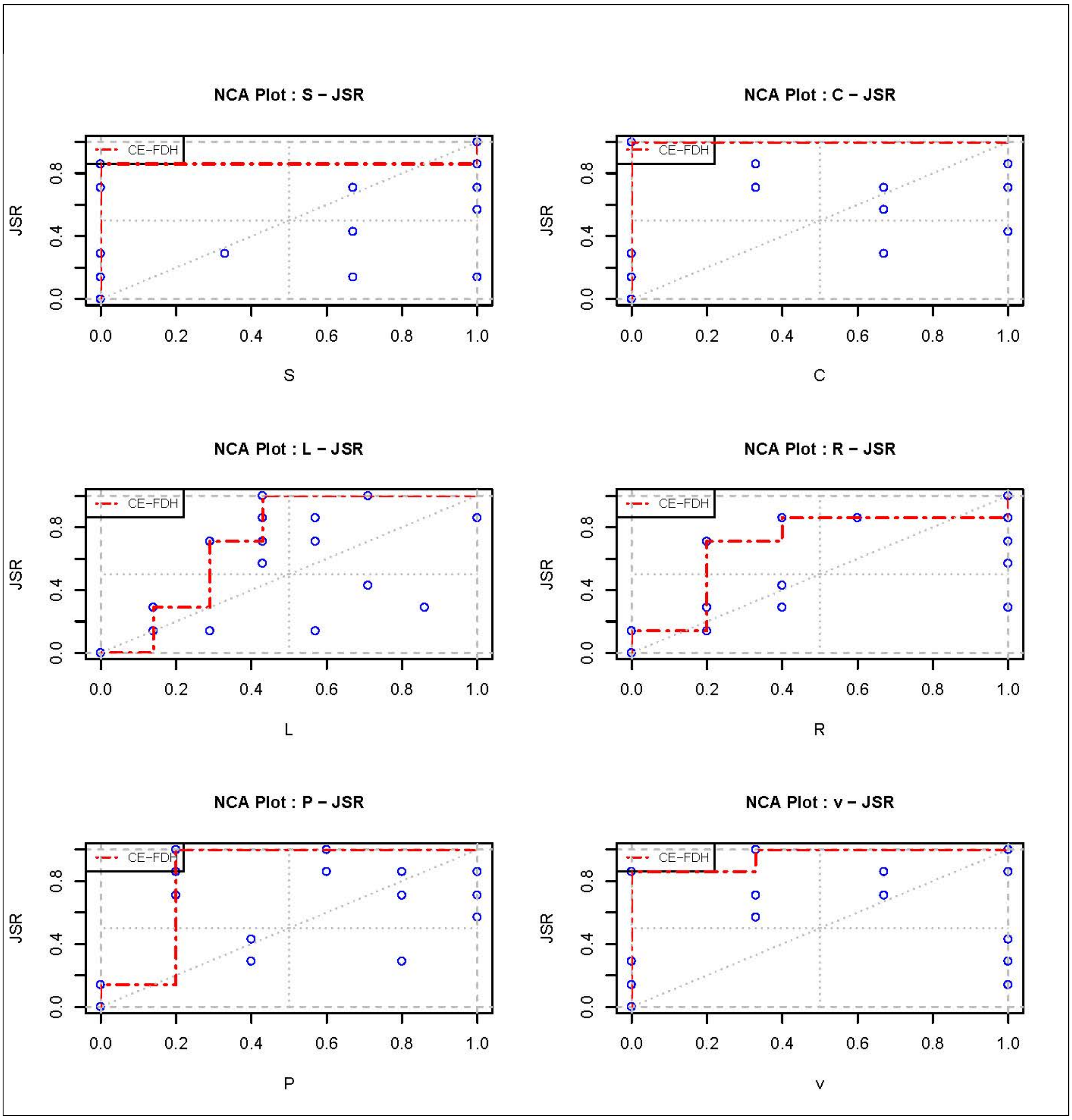

Figure 3 Example of a necessity analysis with NCA (Data from Emmenegger, 2010)

However, the NCA analysis in degree with the six conditions and using the CE-FDH ceiling line (Figure 3) showed the following effect sizes and corresponding p-values (Table 2).

Table 2 NCA necessity analysis. d = effect size; p = p-value

	d	p
S	0.14	0.136
C	0	1
L	0.287	0.002
R	0.314	0.001
P	0.172	0.009
~V (v)	0.046	0.539

A condition could be considered necessary when the effect size has a small p value (e.g. $p < 0.05$). Hence, the NCA analysis shows that certain strength of labour movement (L), a certain level of denomination (R), and a certain strength of religious parties (P) are necessary for high levels of job security regulation (JSR). From Figure 3 it can be observed that the following conditions are necessary for JSR > 0.5:

L > 0.29 is necessary for JSR > 0.5 (presence of JSR)
R > 0.20 is necessary for JSR > 0.5 (presence of JSR)
P > 0.20 is necessary for JSR > 0.5 (presence of JSR)

Although in QCA's binary logic these small necessary membership scores of L, R, P (all < 0.5) would be framed as 'absence' of the condition, in NCA these membership scores are considered small, yet must be present for having the outcome. Thus, according to NCA the low level of membership scores *must* be present, otherwise the sufficient configurations identified by QCA will not produce the outcome.

Emmenegger (2010) identified four possible sufficient configurations for the outcome JSR:

1. S*R*~V (presence of S AND presence of R AND absence of V)
2. S*L*R*P (presence of S AND presence of L AND presence of R AND presence of P)
3. S*C*R*P (presence of S AND presence of C AND presence of R AND presence of P)
4. C*L*P*~V (presence of C AND presence of L AND presence of P AND absence of V)

This combination of solutions can be expressed by the following logical expression:

S*R*~V + S*L*R*P + S*C*R*P + C*L*P*~V → JSR

The presence of a condition and the outcome means that the membership score is > 0.5. The absence of a condition means that the membership score is < 0.5. A common way to summarize the results is shown in Table 3.

Table 3. QCA sufficiency solutions (sufficient configurations).

	Sufficient configuration			
	1	2	3	4
S	●	●	●	
C			●	●
L		●		●
R	●	●	●	
P		●	●	●
V	○			○

● = presence of the condition
○ = absence of the condition

The NCA necessity results can be combined with the QCA sufficiency results as shown in Table 4. Small full square symbols (▪) are added to sufficient configurations (according to QCA) to ensure that the minimum required necessity membership score (according to NCA) is fulfilled. This symbol is inspired by Greckhamer (2016) who used large full square symbol (■) to be added to the solutions table to indicate the presence of a necessary condition (membership score > 0.5) according to QCA's necessity analysis.

Table 4. Combining QCA sufficiency results with NCA necessity results. Small full square symbols (▪) are added to sufficient configurations (according to QCA) to ensure that the minimum required necessity membership score (according to NCA) is fulfilled.

	Sufficient configuration				Necessity requirement
	1	2	3	4	
S	●	●	●		
C			●	●	
L	▪	●	▪	●	> 0.29
R	●	●	●	▪	> 0.2
P	▪	●	●	●	> 0.2
V	○			○	

The NCA results that the presence of L > 0.29 is necessary for JSR > 0.5 is already achieved in the QCA sufficient configurations 2 and 4, but not in configurations 1 and 3. In these latter configurations the requirement L > 0.29 is added to the configuration (▪). Similarly, R > 0.20 is added to configuration 4, and P > 0.20 is added to configuration 1. Without adding these requirements to the configuration, the configuration cannot produce the outcome. Only configuration 2 includes all three necessary conditions according to NCA, without a need for adding them. If the results of NCA would be ignored and the QCA of configurations 1, 3 and 4 would not produce the outcome (or only by chance if the required minimum levels of the ignored NCA necessary conditions would be present by chance). Additionally, the NCA results can show what levels of the condition would be necessary for a higher level of the outcome than a membership score > 0.5. This can be observed in Figure 3. For example, for a membership score of JSR of 0.9, it is necessary to have membership scores of S = 1, L > 0.45, R = 1, P > 0.2, and ~V > 0.35.

I therefore recommend presenting the NCA necessity results together with the QCA sufficient configurations results as in Table 4, and additionally to discuss the full results of NCA for deeper understanding of the sufficient configurations.

The second example is from a study of Skarmeas et al. (2014) that I discuss in my book (Dul, 2020, pp. 77-83). This study is about the effect of four organizational motives (Egoistic-driven motives, absence of Value-driven motives, Strategy-driven motives, Stakeholder-driven motives) on customer scepticism about an organization's Corporate Social Responsibility (CSR) initiative. The results of NCA's necessity analysis with raw scores, and with calibrated set membership scores are shown in Tables 5 and 6, respectively.

Table 5 NCA necessity analysis with raw scores.

	d	P
Egoistic-driven motives	0.14	0.073
Absence of Value-driven motives	0.15	0.000
Strategy-driven motives	0.29	0.002
Stakeholder-driven motives	0.00	1.000

d = effect size; p = p-value

Table 6 NCA necessity analysis with calibrated cores.

	d	p
Egoistic-driven motives	0.01	0.032
Absence of Value-driven motives	0.04	0.000
Strategy-driven motives	0.02	0.026
Stakeholder-driven motives	0.00	1.000

d = effect size; p = p-value

NCA with raw variable scores shows that Absence of Value-driven motives and Strategy-driven motives could be considered are necessary for Scepticism given the medium effect sizes and the small p-values ($p < 0.05$). Also, the NCA with calibrated set membership scores shows that these two conditions have low p-values; however, their effect sizes are small (0.04 and 0.02, respectively). This means that these necessary conditions may be statistically significant but may not be practically significant: nearly all cases reached the required level of necessity. Also, Egoistic-driven motives are statistically, but not practically significant.
NCA with raw variable scores (the conventional NCA approach) can be used in combination with regression analysis, as regression analysis uses raw variable scores (see the supplement 'Comparing NCA with regression'). NCA with calibrated set scores (set membership scores) can be used in combination with QCA, because QCA uses calibrated set membership scores (this supplement).

Table 7 shows the two sufficient configurations according the QCA analysis of Skarmeas et al. (2014).

Table 7. Sufficient configurations (QCA) and necessity requirements (NCA) of the study of Skarmeas et al. (2014) on four conditions and the outcome ‘customer scepticism’ about the motives of an organization’s CSR initiative.

	Sufficient configuration		Necessity Requirement
	1	2	
Egoistic-driven motives	●	●	
Value-driven motives	○	○	
Strategy-driven motives	▪	●	> 0.01
Stakeholder-driven motives	○		

●= presence of the condition according to QCA (> 0.5 membership score)

○ = absence of the condition according to QCA (< 0.5 membership score)

▪ = minimum required necessity membership score according to NCA

In each configuration the necessity of Egoistic-driven motives and the Absence of Value-driven motives are ensured in the configuration. However, the necessity of Strategy-driven motives is not ensured in Sufficient configuration 1. Therefore, the minimum required level of Stakeholder-driven motives according to NCA (0.01) is added to ensure that the configuration is able to produce the outcome. However, the practical meaning of this addition is very limited because the necessity effect size is small. It is added here for illustration of our recommendation.

1. References

Dul, J. (2016) Identifying Single Necessary Conditions with NCA and fsQCA. *Journal of Business Research* 69(4):1516–23.

Dul, J. (2020) Conducting Necessary Condition Analysis. Sage publications.

Emmenegger, P. (2011). Job security regulations in Western democracies: A fuzzy set analysis. *European Journal of Political Research*, 50(3), 336-364.

Greckhamer, T. (2016). CEO compensation in relation to worker compensation across countries: The configurational impact of country-level institutions. *Strategic Management Journal*, 37(4), 793-815.

Ragin, C.C. (1987). *The comparative method: Moving beyond qualitative and quantitative strategies*. Los Angeles: University of California Press.

Ragin, C. C. (2000). *Fuzzy-set Social Science*. Chicago: The University of Chicago Press.

Ragin, C. C. (2008). *Redesigning Social Inquiry: Fuzzy Sets and Beyond*. Chicago: University of Chicago Press.

Rohlfing, I., & Schneider, C. Q. (2013). Improving research on necessary conditions: Formalized case selection for process tracing after QCA. *Political Research Quarterly*, 220-235.

Schneider, C.Q., Wagemann, C. (2012) *Set-theoretic Methods for the Social Sciences: A Guide to Qualitative Comparative Analysis*. Cambridge, England: Cambridge University Press.

Skarmeas, D., Leonidou, C. N., & Saridakis, C. (2014). Examining the role of CSR skepticism using fuzzy-set qualitative comparative analysis. *Journal of Business Research*, 67(9), 1796-1805.

Vis, B., and Dul, J. (2018) Analyzing Relationships of Necessity Not Just in Kind but Also in Degree: Complementing fsQCA with NCA. *Sociological Methods & Research*. *47*(4), 872-899.

Appendix. R Script for producing selected figures and tables

```
#Script for Supplement NCA and QCA
#Jan Dul
#April 27, 2020
library(NCA) # R package for conducting NCA
library(QCA) # R package for conducting QCA
###Figure 1###
#pdf("Figure 1.pdf") # delete '#' for storing a pdf file of the scatter plot
X=NULL
Y=NULL
plot (X,Y, xlim = c(0,1), ylim=c(0,1), pch=19, cex=2, axes =F, xlab="", ylab="", yaxs="i", xaxs="i")
axis(1, at = c(0,0.25,0.75,1.25), pos=0, tck=0, cex.axis=1.5, labels=rep(c("","0","1", "")))
axis(2, at = c(0,0.25,0.75,1.25), pos=0, tck=0, cex.axis=1.5, labels=rep(c("","0","1", "")))
title(xlab="X", cex.lab = 2)
title(ylab="Y", cex.lab = 2)
abline(v=c(0.5,1), h=c(0.5,1))
rect(0.5,0.5,1,1,  col = "grey") # upper right
rect(0,0,0.5,0.5,  col = "grey") # lower left
rect(0.5,0,1,0.5,  col = "grey") # lower right
X<- c(0.25,0.75, 0.75)
Y<- c(0.25,0.25, 0.75)
points(X,Y, pch=19, cex=2)
#dev.off() # delete '#' for storing a pdf file of the scatter plot
###Figure 2###
# Default values that can be customized
line_colors <- list(ols="green",      lh="red3",         cols="darkgreen",
                qr="blue",       ce_vrs="orchid4",    cr_vrs="violet",
                ce_fdh="red",      cr_fdh="orange",     sfa="darkgoldenrod")
line_types  <- list(ols=1,           lh=2,              cols=3,
                qr=4,            ce_vrs=5,          cr_vrs=1,
                ce_fdh=6,        cr_fdh=1,          sfa=7)
line_width  <- 1.5
point_type  <- 1
point_color <- 'blue'
#function for adapting NCA plot
display_plotF2A <-
 function (plot) {
  flip.x <- plot$flip.x
  flip.y <- plot$flip.y
  # Determine the bounds of the plot based on the scope
  xlim <- c(plot$scope.theo[1 + flip.x], plot$scope.theo[2 - flip.x])
  ylim <- c(plot$scope.theo[3 + flip.y], plot$scope.theo[4 - flip.y])
  # Reset/append colors etc. if needed
  line.colors <- append(line.colors, line_colors)
  line.types <- append(line.types, line_types)
  if (is.numeric(line_width)) {
   line.width <- line_width
  }
  if (is.numeric(point_type)) {
   point.type <- point_type
  }
  if (point_color %in% colors()) {
   point.color <- point_color
  }
  # Only needed until release 3.0.2
```

```
    if (!exists("point.color")) {
      point.color <- "blue"
    }
    # Plot the data points
    plot (plot$x, plot$y, pch=point.type, col=point.color, bg=point.color,
          xlim=xlim, ylim=ylim, xlab=colnames(plot$x), ylab=tail(plot$names, n=1))
    # Plot the scope outline
    abline(v=plot$scope.theo[1], lty=2, col="grey")
    abline(v=plot$scope.theo[2], lty=2, col="grey")
    abline(h=plot$scope.theo[3], lty=2, col="grey")
    abline(h=plot$scope.theo[4], lty=2, col="grey")
    # Apply clipping to the lines
    clip(xlim[1], xlim[2], ylim[1], ylim[2])
    # Plot the lines
    for (method in plot$methods) {
      line <- plot$lines[[method]]
      line.color <- line.colors[[method]]
      line.type  <- line.types[[method]]
      # Add QCA reference  lines
      abline(0,1, lty=1, col="black")
      abline(v=0.5, lty=1, col="black")
      abline(h=0.5, lty=1, col="black")
    }
  }
display_plotF2B <-
  function (plot) {
    flip.x <- plot$flip.x
    flip.y <- plot$flip.y
    # Determine the bounds of the plot based on the scope
    xlim <- c(plot$scope.theo[1 + flip.x], plot$scope.theo[2 - flip.x])
    ylim <- c(plot$scope.theo[3 + flip.y], plot$scope.theo[4 - flip.y])
    # Reset/append colors etc. if needed
    line.colors <- append(line.colors, line_colors)
    line.types <- append(line.types, line_types)
    if (is.numeric(line_width)) {
      line.width <- line_width
    }
    if (is.numeric(point_type)) {
      point.type <- point_type
    }
    if (point_color %in% colors()) {
      point.color <- point_color
    }
    # Only needed until release 3.0.2
    if (!exists("point.color")) {
      point.color <- "blue"
    }
    # Plot the data points
    plot (plot$x, plot$y, pch=point.type, col=point.color, bg=point.color,
          xlim=xlim, ylim=ylim, xlab=colnames(plot$x), ylab=tail(plot$names, n=1))
    # Plot the scope outline
    abline(v=plot$scope.theo[1], lty=2, col="grey")
    abline(v=plot$scope.theo[2], lty=2, col="grey")
    abline(h=plot$scope.theo[3], lty=2, col="grey")
    abline(h=plot$scope.theo[4], lty=2, col="grey")
    # Plot the legend before adding the clipping area
    legendParams = list()
```

```
    for (method in plot$methods) {
      line.color <- line.colors[[method]]
      line.type  <- line.types[[method]]
      name <- gsub("_", "-", toupper(method))
      legendParams$names  = append(legendParams$names, name)
      legendParams$types  = append(legendParams$types, line.type)
      legendParams$colors = append(legendParams$colors, line.color)
    }
    if (length(legendParams) > 0) {
      legend("topleft", cex=0.7, legendParams$names,
           lty=legendParams$types, col=legendParams$colors, bg=NA)
    }
    # Apply clipping to the lines
    clip(xlim[1], xlim[2], ylim[1], ylim[2])
    # Plot the lines
    for (method in plot$methods) {
      line <- plot$lines[[method]]
      line.color <- line.colors[[method]]
      line.type  <- line.types[[method]]
      if (method %in% c("lh", "ce_vrs", "ce_fdh")) {
        lines(line[[1]], line[[2]], type="l",
            lty=line.type, col=line.color, lwd=line.width)
      } else {
        abline(line, lty=line.type, col=line.color, lwd=line.width)
      }
    }
  }
 #data from Rohlfing, I., & Schneider, C. Q. (2013)
 country <- c("Australia1990", "Austria1990", "Belgium1990", "Canada1990", "Denmark1990", "Finland1990", "France1990", "Germany1990", "Ireland1990", "Italy1990", "Japan1990", "Netherlands1990", "NewZealand1990", "Norway1990", "Spain1990", "Sweden1990", "Switzerland1990", "UK1990", "USA1990", "Australia1995", "Austria1995", "Belgium1995", "Canada1995", "Denmark1995", "Finland1995", "France1995", "Germany1995", "Ireland1995", "Italy1995", "Japan1995", "Netherlands1995", "NewZealand1995", "Norway1995", "Spain1995", "Sweden1995", "Switzerland1995", "UK1995", "USA1995", "Australia1999", "Austria1999", "Belgium1999", "Canada1999", "Denmark1999", "Finland1999", "France1999", "Germany1999", "Ireland1999", "Italy1999", "Japan1999", "Netherlands1999", "NewZealand1999", "Norway1999", "Spain1999", "Sweden1999", "Switzerland1999", "UK1999", "USA1999", "Australia2003", "Austria2003", "Belgium2003", "Canada2003", "Denmark2003", "Finland2003", "France2003", "Germany2003", "Ireland2003", "Italy2003", "Japan2003", "Netherlands2003", "NewZealand2003", "Norway2003", "Spain2003", "Sweden2003", "Switzerland2003", "UK2003", "USA2003")
 EMP <- c(0.07, 0.7, 0.94, 0.04, 0.59, 0.7, 0.86, 0.95, 0.07, 0.99, 0.65, 0.86, 0.07, 0.93, 0.99, 0.98, 0.09, 0.03, 0.05, 0.14, 0.7, 0.79, 0.12, 0.25, 0.59, 0.88, 0.83, 0.12, 0.97, 0.65, 0.65, 0.07, 0.83, 0.94, 0.83, 0.25, 0.07, 0.05, 0.14, 0.7, 0.79, 0.12, 0.25, 0.59, 0.88, 0.83, 0.12, 0.97, 0.65, 0.65, 0.07, 0.83, 0.94, 0.83, 0.25, 0.07, 0.05, 0.25, 0.65, 0.79, 0.12, 0.41, 0.59, 0.91, 0.79, 0.17, 0.75, 0.41, 0.7, 0.17, 0.83, 0.94, 0.83, 0.3, 0.12, 0.05)
 BARGAIN <- c(0.9, 0.98, 0.95, 0.21, 0.78, 0.97, 0.96, 0.95, 0.79, 0.92, 0.07, 0.8, 0.42, 0.85, 0.86, 0.94, 0.48, 0.36, 0.05, 0.9, 0.98, 0.95, 0.18, 0.78, 0.97, 0.97, 0.96, 0.95, 0.91, 0.06, 0.9, 0.13, 0.84, 0.88, 0.95, 0.42, 0.36, 0.05, 0.79, 0.95, 0.95, 0.2, 0.9, 0.97, 0.93, 0.84, 0.32, 0.95, 0.06, 0.85, 0.08, 0.68, 0.79, 0.95, 0.46, 0.18, 0.04, 0.9, 0.98, 0.95, 0.14, 0.92, 0.95, 0.95, 0.76, 0.32, 0.9, 0.04, 0.94, 0.08, 0.79, 0.38, 0.95, 0.24, 0.12, 0.03)
 UNI <- c(1, 0.01, 0.14, 0.99, 0.1, 0.2, 0.1, 0.05, 0.23, 0.01, 0.63, 0.01, 0.11, 0.94, 0.31, 0.04, 0.01, 1, 0.95, 1, 0.02, 0.14, 0.99, 0.95, 0.8, 0.43, 0.14, 0.86, 0.05, 0.73, 0.45, 0.98, 0.95, 0.9, 0.36, 0.01, 1, 1, 1, 0.03, 0.72, 1, 0.95, 1, 0.99, 0.14, 1, 0.38, 0.99, 1, 1, 1, 0.98, 0.95, 0.89, 1, 1, 1, 0.36, 0.73, 1, 1, 1, 1, 0.4, 1, 0.04, 1, 1, 1, 1, 1, 1, 0.2, 1, 0.99)
 OCCUP <- c(0.68, 0.91, 0.37, 0.11, 0.55, 0.95, 0.46, 0.95, 0.16, 0.3, 0.14, 0.73, 0.17, 0.65, 0.21, 0.74, 0.82, 0.08, 0.05, 0.68, 0.86, 0.94, 0.11, 0.27, 0.54, 0.56, 0.74, 0.03, 0.71, 0.15, 0.51, 0.2, 0.83, 0.16, 0.62, 0.74, 0.07, 0.03, 0.68, 0.93, 0.77, 0.11, 0.71, 0.82, 0.82, 0.71, 0.07, 0.8, 0.16, 0.66, 0.25, 0.81, 0.18, 0.24, 0.74, 0.06, 0.03, 0.48, 0.63, 0.75, 0.11, 0.66, 0.85, 0.85, 0.76, 0.19, 0.8, 0.13, 0.76, 0.25, 0.41, 0.14, 0.32, 0.71, 0.06, 0.03)
```

```
STOCK <- c(0.45, 0.01, 0.26, 0.62, 0.53, 0.02, 0.23, 0.05, 0.53, 0.01, 1, 0.05, 0.03, 0.12, 0.03, 0.91, 1, 1, 0.95,
0.98, 0.01, 0.53, 0.92, 0.76, 0.73, 0.41, 0.14, 0.53, 0.02, 1, 0.69, 0.76, 0.49, 0.06, 1, 1, 1, 1, 1, 0.01, 1, 1, 0.99, 1,
1, 0.97, 0.99, 0.73, 1, 1, 0.12, 0.45, 0.9, 1, 1, 1, 1, 1, 0.03, 0.76, 1, 0.93, 1, 0.98, 0.49, 0.96, 0.33, 1, 1, 0.18, 0.92,
0.98, 1, 1 ,1, 1)
MA <- c(0.33, 0.05, 0.14, 0.31, 0.1, 0.13, 0.27, 0.08, 0.49, 0.07, 0.06, 0.43, 0.95, 0.27, 0.13, 0.93, 0.53, 0.9, 0.15,
0.78, 0.06, 0.56, 0.4, 0.07, 0.24, 0.22, 0.13, 0.56, 0.07, 0.05, 0.67, 0.82, 0.4, 0.08, 0.92, 0.99, 0.88, 0.15, 0.91,
0.15, 1, 1, 0.98, 1, 0.99, 1, 0.99, 0.23, 0.08, 1, 0.93, 0.98, 0.85, 1, 1, 1, 0.8, 0.67, 0.2, 0.51, 0.62, 0.67, 0.91, 0.34,
0.44, 0.8, 0.12, 0.06, 0.79, 0.97, 0.69, 0.26, 0.95, 0.93, 0.96, 0.17)
HIGH <- c(0.19, 0.25, 0.14, 0.28, 0.34, 0.17, 0.42, 0.27, 0.98, 0.2, 0.85, 0.39, 0.06, 0.14, 0.17, 0.36, 0.31, 0.79,
0.93, 0.31, 0.25, 0.2, 0.3, 0.4, 0.49, 0.62, 0.32, 1, 0.18, 0.85, 0.7, 0.09, 0.14, 0.19, 0.71, 0.51, 0.87, 0.91, 0.38,
0.47, 0.4, 0.4, 0.74, 0.86, 0.86, 0.67, 1, 0.29, 0.96, 0.96, 0.08, 0.32, 0.27, 0.93, 0.98, 0.98, 0.99, 0.35, 0.5, 0.71,
0.36, 0.79, 0.86, 0.82, 0.69,1, 0.31, 0.94, 0.96, 0.13, 0.32, 0.3, 0.8, 0.99, 0.98, 0.98)
data1 <- data.frame(country, EMP, BARGAIN, UNI, OCCUP, STOCK, MA, HIGH, row.names = 1)
#NCA
modelF2<- nca_analysis(data1, "STOCK", "HIGH", ceilings = "ce_fdh")
#pdf("Figure 2.pdf") # delete '#' for storing a pdf file of the scatter plot
display_plotF2A(modelF2$plots[[1]])
display_plotF2B(modelF2$plots[[1]])
#dev.off() # delete '#' for storing a pdf file of the scatter plot

###Table 2###
data("d.Emm") # get the Emmenegger data from the QCA package
head(d.Emm) # print the head of the data file
#create data with v (absence of V) instead of V (presence of V)
data<-d.Emm # rename dataset as 'data'
v<-(1-data$V) # define absence of V
data$V<-v # change values of V
colnames(data)[6]<-"v" #change name V into v
head(data) # print the head of the new data file
#perform QCA necessity analysis for all conditions
pofind(data, outcome = "JSR")
#No single condition has consistency (inclN) > 0.9: not necessary
#perform NCA necessity analysis for each condition
modelNCA<- nca_analysis(data, c("S","C", "L", "R", "P", "v"), "JSR", ceilings = "ce_fdh", test.rep = 10000)
modelNCA # summary of NCA results
#L, R and P effect sizes (>0.1) and p-values (<0.05) support necessity

###Figure 3###
#function for adapting NCA plot
# Default values that can be customized
line_colors <- list(ols="green",       lh="red3",          cols="darkgreen",
                qr="blue",         ce_vrs="orchid4",    cr_vrs="violet",
                ce_fdh="red",      cr_fdh="orange",     sfa="darkgoldenrod")
line_types  <- list(ols=1,           lh=2,               cols=3,
                qr=4,              ce_vrs=5,           cr_vrs=1,
                ce_fdh=6,          cr_fdh=1,           sfa=7)
line_width  <- 1.5
point_type  <- 1
point_color <- 'blue'
#function for adapting NCA plot
display_plot <-
  function (plot) {
    flip.x <- plot$flip.x
    flip.y <- plot$flip.y
    # Determine the bounds of the plot based on the scope
    xlim <- c(plot$scope.theo[1 + flip.x], plot$scope.theo[2 - flip.x])
    ylim <- c(plot$scope.theo[3 + flip.y], plot$scope.theo[4 - flip.y])
```

```
# Reset/append colors etc. if needed
line.colors <- append(line.colors, line_colors)
line.types <- append(line.types, line_types)
if (is.numeric(line_width)) {
  line.width <- line_width
}
if (is.numeric(point_type)) {
  point.type <- point_type
}
if (point_color %in% colors()) {
  point.color <- point_color
}
# Only needed until release 3.0.2
if (!exists("point.color")) {
  point.color <- "blue"
}
# Plot the data points
plot (plot$x, plot$y, pch=point.type, col=point.color, bg=point.color,
      xlim=xlim, ylim=ylim, xlab=colnames(plot$x), ylab=tail(plot$names, n=1))
# Plot the scope outline
abline(v=plot$scope.theo[1], lty=2, col="grey")
abline(v=plot$scope.theo[2], lty=2, col="grey")
abline(h=plot$scope.theo[3], lty=2, col="grey")
abline(h=plot$scope.theo[4], lty=2, col="grey")
# Plot the legend before adding the clipping area
legendParams = list()
for (method in plot$methods) {
  line.color <- line.colors[[method]]
  line.type  <- line.types[[method]]
  name <- gsub("_", "-", toupper(method))
  legendParams$names  = append(legendParams$names, name)
  legendParams$types  = append(legendParams$types, line.type)
  legendParams$colors = append(legendParams$colors, line.color)
}
if (length(legendParams) > 0) {
  legend("topleft", cex=0.7, legendParams$names,
         lty=legendParams$types, col=legendParams$colors, bg=NA)
}
# Apply clipping to the lines
clip(xlim[1], xlim[2], ylim[1], ylim[2])
# Plot the lines
for (method in plot$methods) {
  line <- plot$lines[[method]]
  line.color <- line.colors[[method]]
  line.type  <- line.types[[method]]
  if (method %in% c("lh", "ce_vrs", "ce_fdh")) {
    lines(line[[1]], line[[2]], type="l",
          lty=line.type, col=line.color, lwd=line.width)
  } else {
    abline(line, lty=line.type, col=line.color, lwd=line.width)
  }
  # Add QCA reference  lines
  abline(0,1, lty=3, col="grey")
  abline(v=0.5, lty=3, col="grey")
  abline(h=0.5, lty=3, col="grey")
}
# Plot the title
```

```
    title(paste0("NCA Plot : ", plot$title), cex.main=1)
  }
#pdf("Figure 3.pdf") # delete '#' for storing a pdf file of the scatter plot
#par(mfrow=c(3,2)) # delete '#' for storing a pdf file of the scatter plot
display_plot(modelNCA$plots[[1]])
display_plot(modelNCA$plots[[2]])
display_plot(modelNCA$plots[[3]])
display_plot(modelNCA$plots[[4]])
display_plot(modelNCA$plots[[5]])
display_plot(modelNCA$plots[[6]])
#dev.off() # delete '#' for storing a pdf file of the scatter plot
```