Necessary Condition Analysis (NCA) with R (Version 5.0.0) A Quick Start Guide

20 March 2026

Jan Dul

Erasmus University Rotterdam (EUR) – Rotterdam School of Management

What are the changes in version 5.0.0?

Version 5.0.0 of the NCA software in R is a major upgrade compared to earlier versions. It has new functions (including `nca_difference` for statistical testing of differences between two effect sizes) and new output with extra model fit parameters. The new version draws linear ceiling lines with extensions on the bounding box, based on the formal mathematical descriptions of ceiling lines and feasible area.

The software corresponds to the principles and application of NCA as described in the book:

Dul, J. (2026). *Necessary Condition Analysis - NCA: Principles and Application*. Chapman & Hall/CRC Press

An online version of the book can be accessed via: https://jandul.github.io/NCA/. There you can also access additional supportive materials discussed in the book.

If you install the package for the first time: see below from section 'Abstract'. If you have installed an older version of the NCA package you can update the package as follows:

```
update.packages("NCA")
```

The NCA package uses other packages. It is possible that these packages that are not installed on your computer. In that case the following error message may appear: `There is no package called …` , where … corresponds to the name of the missing package. You then need to install the specific packages first.

```
install.packages("…")
```

Abstract

Necessary Condition Analysis (NCA) is an approach and data analysis technique for identifying necessary conditions in datasets. It can complement traditional regression-based data analysis as well as methods like QCA (see then NCA website www.erim.nl/nca for more information on NCA). This guide helps a novice user without knowledge of R or NCA to

install the free R and NCA software on the user's computer and to perform an NCA analysis within 15 minutes. The main instructions are:

I. Install R
II. Install NCA
III. Load data
IV. Run NCA.

I. Install R (for new users)

1. What is R?

R is an open source programming language that is increasingly used for data analysis in different scientific fields, including the social sciences. It contains many statistical, mathematical and graphical functions that are also part of commercial statistical software such as SPSS and Stata. Additionally, R can run specific user-defined functions ("packages"). One such package is NCA. Only some basic knowledge about R (presented in this guide) is needed to run NCA with R.

2. How can I install R (for new users)?

R can be installed (downloaded) on your computer from the central R-website (see below). You need to have administration rights on your computer to install this software. The version of R that you must download depends on the platform of your computer: Windows or OS X (Mac). There is also a version for Linux.
For Windows users:

- Go to http://cran.r-project.org/bin/windows/base/
- Download "R x.y.z for Windows", where x,y,z, is the latest version number.
- Open the downloaded file and follow the instructions (accept all defaults).

For OS X (Mac) users:

- Go to http://cran.r-project.org/bin/macosx/
- Download the correct version for your OS X.
- Open the downloaded file and follow the instructions (accept all defaults).

Additionally we recommend to install RStudio, which is a user-friendly environment from which you can work with R. There are RStudio versions for Windows, OS X and Linux.

- Go to https://www.rstudio.com/products/rstudio/download/
- Select RStudio Desktop (open source license)
- Download the appropriate installer of RStudio x.y.z. for your platform (Windows, OS X, etc.).
- Follow the instructions (accept all defaults).

3. How can I start R?

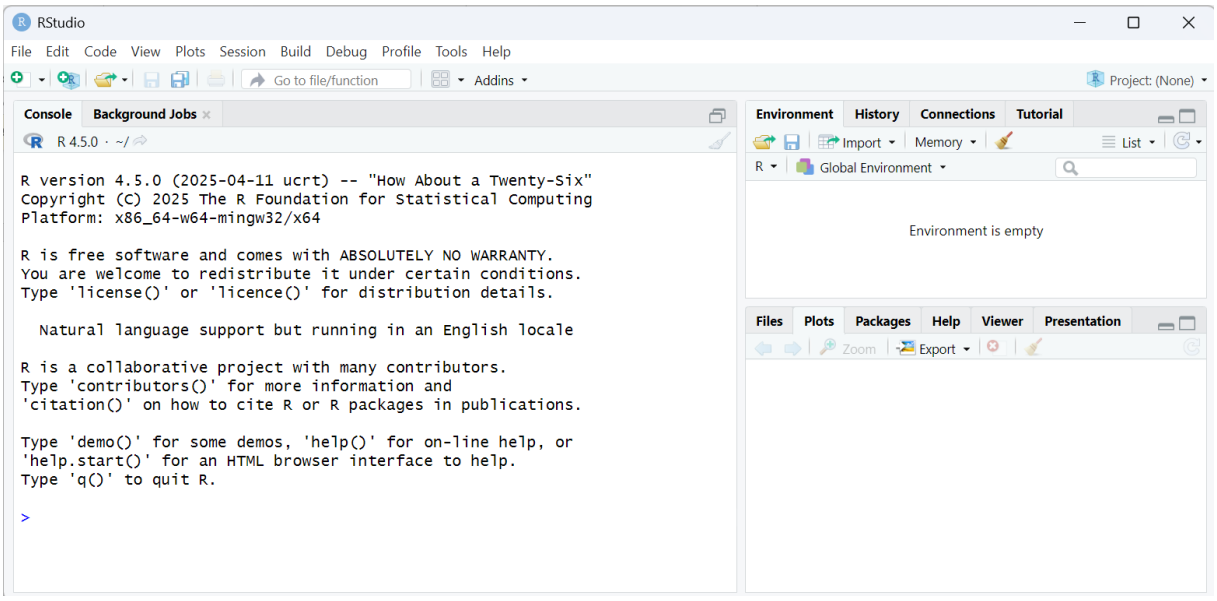
R can be started in two ways: by opening R's interface (RGui), or by using Rstudio.

RGui:
Clicking on the R shortcut on the desktop will give you R's Graphical user interface (RGui).

Opening R with RGui screen (console)

RGui opens with the window called “R console”. Here you can find some basic information about R. In the R console you can type instructions after the ">" prompt[1]. Each instruction must be followed by <enter>. In this guide the instructions for R are shown in `courier` font (“typewriter characters”). These instructions can be typed (or copy-paste) after the prompt in the console. Numerical output of NCA will be displayed in the console. Graphical output of NCA will be displayed in new windows that open in the RGui screen.

RStudio:
Go to the folder RStudio in the program files, open this folder, and click on the RStudio executable file. This will open the RStudio screen.



Opening of R with RStudio (console on the left and two other windows on the right) opens two other windows. The upper right window contains tabs with Environment, History, Connection and Tutorial, and the lower right window contains tabs with Files, Plots, Packages, Help, Viewer, and Presentation. The Help tab displays the manuals for the packages that are installed on your computer, including the manual for NCA. This manual provides details of all instructions and options that can be used in the NCA package. The Files tab shows the folder structure on your computer. For the purpose of this quick start guide we primarily use the Plots and Viewer tabs in the lower-right window that displays plots produced by NCA. Further information about RStudio and free introduction course can be found on the Internet.

Click on the Plot tab in the lower right window. In the remainder of this guide this window will be called “plots window”. Numerical output of NCA will be printed in the console. Just like for the console in RGui, in RStudio’s console you can type instructions after the ">" prompt. However, we recommend using the “script window” that can be started with the pull

[1] For certain instructions you can also use the pull-down menu at the top of the RGui or RStudio page. In this guide menu instructions are printed in calibri font and successive steps are connected by an arrow “→”.

down menu in RStudio as follows: File → New File → R Script. In the script window the instructions can be typed and edited. The script can be saved by using File → Save (or using the save button), or File → Save As. The stored script is a file with the extention ".R", for example "Myscript.R". You can load ("source") an existing script by using File → Open file.

You can run the script in the script window by successively executing each instruction line. You can also select several instructions and run this set of instructions at once.
We advise to type instructions in the script window, so you can keep track of the instructions and store them in an R file for later replication.

4. How can I set my "working directory"?

The "working directory" is a folder on your computer where (by default) R searches your data and stores output file(s). You can check your current working directory by typing `getwd()` in the lower left window (console) after the ">" followed by a return (enter), or (preferably) in the upper left window (script window) after the line number, followed by pushing on the "Run button":

```
getwd()
```

You can change your working directory by typing:

```
setwd("...\\MyWorkingDirectory")
```

In this example the working directory is named "MyWorkingDirectory" but you can use any name for the working directory folder. Note that R uses "\\" in the directory tree. Alternatively, you can use one forward slash "/", but not one backward slash "\".

You can also use the Files tab of the lower-right window of RStudio to select the working directory. Tick the square next to the folder that you want to select as working directory, select "More", and select "Set as Working Directory".

The working directory needs to be specified in R each time that you start R.

II. Install NCA

5. What is NCA software?

NCA is a free package for R. The reference to the NCA software is:
Dul, J. and Buijs G. (20xx). Necessary Condition Analysis. R Package Version x.x,x. URL: http://cran.r-project.org/package=NCA.

6. How can I install NCA for R?

Installation of the NCA package is possible from R-version 3.0.1. In the console or script window type the following instruction**:**

```
install.packages("NCA", dependencies = TRUE)
```

Select the location nearest to you.[2]

This will install package NCA and all other R packages ("dependencies") that are used by NCA on your computer.

7. How can I install new versions of NCA?

A new version of NCA and of other installed packages can be obtained by:

```
update.packages()
```

8. How can I load NCA?

After the NCA package is installed (downloaded) on your computer, it must be loaded (activated) in R (NCA must be loaded each time you start R):

```
library(NCA)
```

Some basic information about NCA is displayed in the console:

If you get a warning message that the NCA package was built under version x.x.x. (for Windows) you have an older R version. Then it is advised to update your R package, otherwise some NCA functions may not work properly. A simple way to update your R package is by installing the package "installr". It is advised to leave RStudio and to update R from Rgui as follows:

```
install.packages("installr")
library(installr)
updateR()
```

During the installation process you can press "next", "OK", and "Yes" on everything. Note that this process (in particular copying of files and updating of packages), may take several minutes.

[2] Alternatively NCA can be installed by using the file "NCA_x.y.z.tar.gz", where x,y,z, is a version number . This file contains the NCA software to be copied to your computer.
Then you can install the package from this file by typing (or copy-paste) in the console the following instructions (note that in the final instruction (…\\ ) is the path to the NCA_x.y.z.tar.gz file on your computer, which must be specified by you; also note that R uses "\\"or "/" in the directory tree):
install.packages("…\\NCA_x.y.z.tar.gz", repos=NULL, type="source")
Note that R cannot handle too long path names when installing the package this way. Then R needs to be re-installed on your computer closer to the root.
You can also install the package from the pull down menu of RStudio: Tools → Install Packages → Install from: Package Archive File (.zip; tar.gz) → Browse to the location of the NCA package on your computer → Install
It may be necessary to install also the packages that are used by the NCA package (dependencies). Missing packages are mentioned in error messages while installing NCA.

III. Load data

9. How can I load the example dataset?

NCA comes with example datasets. The dataset `nca.example` consist of N= 28 countries with two independent variables or conditions (X_1= Individualism, X_2=Risk taking) and one dependent variable or outcome. (Y=Innovation performance). You can load (activate) the example data in your R session as follows:

```
data(nca.example)
```

You can rename the data as "data".

```
data <- nca.example
```

The combination of symbols "<" and "−" is the "assignment operator" of R, which connects two objects (in this case "data" and "nca.example"). Usually the "=" symbol could be used as assignment operator, but there are exceptions. Therefore "<−" is used in this guide.

After this instruction the example data is a data object known as "data".

The data are shown on the screen in the console by typing the data name:

```
print(data)
```

or just:

```
data
```

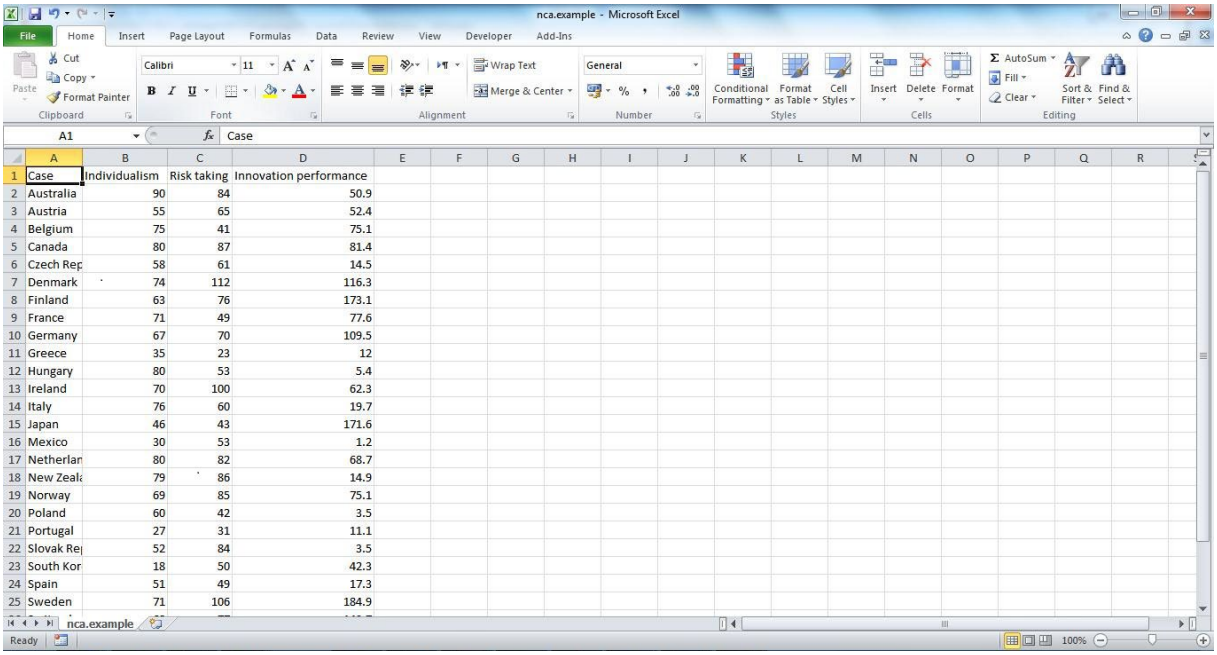
The first column on the screen contains the row names of cases: in this example "countries". The first row on the screen is the header, which contains the names of the variables. There are three data columns. In this example the first two data columns are the two independent variables, and the last column is the dependent variable.

By using the upper arrow on the keyboard you can get back previous instructions that you typed.

10. How should I prepare my own data file?

NCA presumes that the data in your data file (input file) are organized in a similar way as is commonly used in data files, for example SPSS data files. Rows correspond to cases (except for the first row, which can be a header with variable names; these names will appear in the plots and other NCA output). Columns correspond to variables (except for the first column, which can be row names). All variable values must be numbers (no letters).
A common data file type for R is .csv (e.g., an Excel file saved as .csv[3]). Missing data in a .csv file must be an empty cell (do not use NA, 999 or other symbols). Other data file types than .csv are possible as well. Examples include SPSS (.sav), Stata (.dta), and SAS (.xpt). See a general R-manual for instructions about how to import these other types of data files in R (many R manuals can be found on internet).
Data file of nca.example (.csv)

[3] Depending on the region and language settings of your computer, your Excel program uses decimal points or decimal commas, and the separator in the csv file uses a comma or a semi-colon, respectively. In this guide it is presumed that you have decimal points and comma separators.



11. How can I load my data file?

In RStudio you can load your data from the upper right window by clicking on Import DataSet and subsequently select the file location and the characteristics of your dataset. The imported data are known in R by the name of the file (e.g., mydata). You can change the name of the data as follows:

```
data <- mydata
```

Alternatively, you can load your data by giving instructions in the script window or console.

Load your .csv data file (if it contains a header and row names) as follows:

```
data <- read.csv("mydata.csv", row.names = 1)
```

(If your file uses decimal commas instead of decimal points, you can try using `read.csv2` , instead of `read.csv`, see also footnote 3).

After this instruction your data is a data object known as “data”.

If your dataset has no row names you type:

```
data <- read.csv("mydata.csv")
```

and the rows will be identified on screen with a number from 1 to N.

If your dataset has no header (and no row names) you can type:

```
data <- read.csv("mydata.csv", header = FALSE)
```

and the columns will be identified on screen with a number from 1 to the total number of variables.

For loading your dataset you can also use the Import Dataset tab in the upper-right window of RStudio.

For loading an SPSS data file (.sav) (no header) you can type in the R console :

```
library("foreign")
data <- read.spss("mydata.sav", to.data.frame = TRUE)
```

For more information on loading SPSS data:

```
?read.spss
```

IV. Run a basic NCA

12. How can I run an NCA analysis?

After the data are loaded you can run NCA by specifying the name of the data object (e.g., "data" if you have named your data as such) followed by the specification of one or more X-variables (the condition(s) or independent variable(s)) and one Y variable (the outcome or dependent variable). A variable can be specified by its column numbers (index) or variable name (column name). For the nca.example data file the first and second columns are the independent variables "Individualism" and "Risk taking", respectively. The third column is the dependent variable "Innovation performance". You can run NCA for Individualism (x_1) and Innovation performance (Y) by specifying the dataset (here the dataset nca.example is renamed as "data"), and the column numbers of the variables. Usually the analysis is given a name, for example "model":

```
model <- nca_analysis(data,1,3)
```

`nca_analysis` is the core function of NCA.

Instead of using the column numbers to identify the variables you can also use the variable names, enclosed by quotation marks (`"`) :

```
nca_analysis(data, "Individualism", "Innovation
performance")
```

The analysis does not show yet the results. The print the basic results you give the name of the analysis:

```
print(model)
```

or:

```
model
```

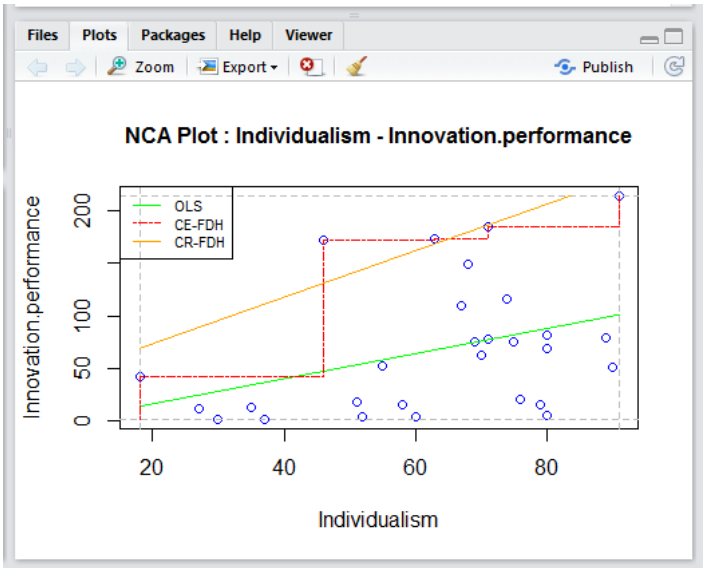
As a result the following NCA output is printed on the console:

```
---------------------------------------------------------------
Effect size(s):
             ce_fdh cr_fdh
Individualism  0.416  0.307
---------------------------------------------------------------
```

The printed output shows per independent variable (here only the output for Individualism is shown) the necessary condition effect size for two different ceiling line techniques. These are the default ceiling lines: the step function CE-FDH (Ceiling Envelopment – Free Disposal Hull) and the straight line CR-FDH (Ceiling Regression – Free Disposal Hull). The step ceiling line can be used when the data and underlying phenomenon are discrete with limited number of levels, and a straight ceiling line can be selected when the data and underlying phenomenon are discrete with a large number of levels, or continuous.

More output can be obtained with the `nca_output` function. For example, the following instruction displays the scatter plot (for Individualism and Innovation performance) in the plots window (and avoids summary output being printed in the console) .

```
nca_output (model, plots=TRUE, summaries = FALSE)
```



Note that the name Innovation performance is changed into Innovation.performance. After loading a dataset, spaces in names of columns or rows are changed into a dot ".". For example, the third column name in the csv data file is "Innovation performance", whereas the name after this file is loaded in R is changed into "Innovation.performance". In the remainder of this guide we use the names as loaded in R (hence with a dot). In the NCA package and in this guide, when variable names or other object names created by the user have separate words, a dot "." connects the words (e.g., "Innovation.performance" is a user defined object). An underscore ("_") is used to connect words in functions that are part for the package (e.g., "nca_analysis" is a function of the package).

The scatter plot shows the selected ceiling lines: the two default ceiling lines (CE-FDH in red, and CR-FDH in orange), and the OLS regression line (green) through the middle of the data (as a reference). If the effect size is greater than zero, there is an empty area in the upper-left corner of the scatter plot, which is an indication of the presence of a necessary condition. The necessary condition effect size (d) is the proportion of the scope above the ceiling: $d = C/S$. It ranges from 0 to 1 ($0 \leq d \leq 1$). The effect size indicates to what extent the condition is necessary for the outcome. In other words: to what extent the condition constrains the outcome, and the outcome is constrained by the condition. Hence, the effect size is the size of the empty zone relative to the total XY-zone where data can be expected (bounding box).

13. What is a general benchmark for the effect size?

According to Dul (2016, p.30) "An effect size can be valued as important or not, depending on the context. A given effect size can be small in one context and large in another. General qualifications for the size of an effect as 'small,' 'medium,' or 'large' are therefore disputable. If, nevertheless, a researcher wishes to have a general benchmark for necessary condition effect size, I would offer $0 < d < 0.1$ as a 'small effect,' $0.1 \leq d < 0.3$ as a 'medium effect,' $0.3 \leq d < 0.5$ as a 'large effect,' and $d \geq 0.5$ as a 'very large effect'."

14. How can I run a basic NCA analysis with multiple conditions?

You can perform an NCA analysis with two or more conditions (x_i), but always with only one outcome (y) at the same time. This is the multiple NCA. In the NCA instruction for multiple NCA the conditions (x variables) are specified as a vector (a list of variables) by using R's symbol for a vector which is a "c" (combine). For example, when running a multiple NCA with Individualism (x_1) in the first column of the dataset, Risk.taking (x_2) in the second column, and Innovation.performance (y) in the third column, the NCA instruction is:

```
model <- nca_analysis(data,c(1,2),3)
```

Alternatively, you can use the variable names, enclosed by quotation marks (") :

```
model <- nca_analysis(data,c("Individualism","Risk
taking"),"Innovation performance")
```

Yet another alternative is to identify the range of successive columns of conditions

```
model <- nca_analysis (data,c(1:2),3)

model
```

Now the following NCA output is printed on the console:

```
---------------------------------------------------------------
Effect size(s):
             ce_fdh cr_fdh
Individualism  0.416  0.307
Risk.taking    0.309  0.282
---------------------------------------------------------------
```

The output in the plots window now consists of two scatter plots, one for Individualism, and one for Risk.taking. You can switch between the scatter plots by using the arrows in the plots window.

15. How can I run a basic NCA analysis with different ceiling lines?

The default ceiling lines are CE-FDH (step function) and CR-FDH (straight line). With the option "ceilings" other ceiling lines can be selected. For example, the ceiling line CE-VRS can be selected, together with the two default ceiling lines (using a vector with the names of the ceiling lines) as follows:

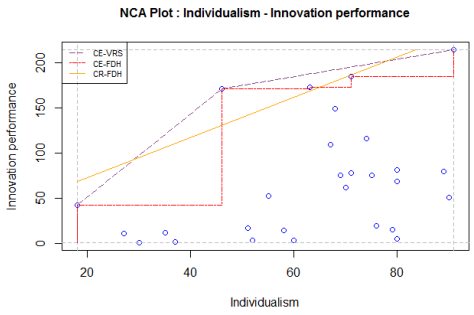
```
model <- nca_analysis(data,c(1:2),3,ceilings=c("ce_fdh",
"cr_fdh", "ce_vrs"))

model
```

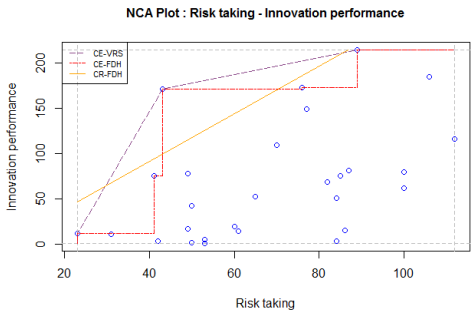
```
---------------------------------------------------------------
Effect size(s):
             ce_vrs ce_fdh cr_fdh
```

```
Individualism 0.255  0.416  0.307
Risk taking   0.181  0.309  0.282
-----------------------------------------------------------------
```

The scatterplots show the three ceiling lines

```
nca_output(model, plots=TRUE, summaries = FALSE)
```





A plot can be saved with the Export tab in the plots window. The size of the plot can be changed with by moving the border of the plots window and Re- running the nca_output instruction (re-running latter also ensures a proper positioning of the legend after changing the borders).

V. Run an advanced NCA

For further demonstration of the NCA software you can download the file "Demonstration_NCA.R" from https://jandul.github.io/NCA/ with R script for advanced NCA functions. Details of this script are described in Chapter 9 (Data Analysis) and Appendix B.3 (Demonstration of NCA with R) in Dul, J. (2026). *Necessary Condition Analysis – NCA. Principles and Application*. Chapman & Hall/CRC Press.