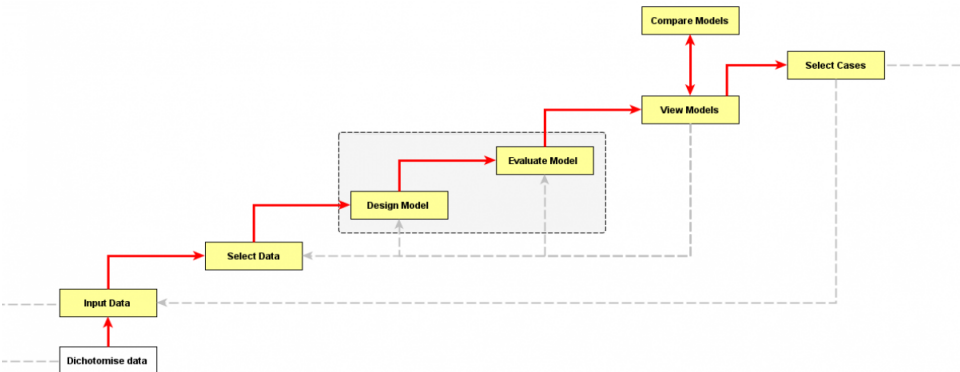


EvalC3

...tools for developing, exploring and evaluating predictive models of expected outcomes



The content that follows is a download of all the pages that were available here: <https://evalc3.net/> and generated between 2015 and 2020, by Rick Davies. Originally designed as an Excel app, EvalC3 has been superseded by EvalC3 Online, which is now accessible here: <https://evalc3online.org/authenticate>.
Rick Davies, March 2025

EvalC3 Contents [See also: Bookmarks, in a pdf viewer]

0. About EvalC3

- 0.0 About EvalC3
- 0.1 Example Uses
 - 1. A Typology of Uses
 - 2. Published Papers on the Use of EvalC3
- 0.2 Types of Causes
- 0.3 Prediction vs Explanation
- 0.4 Compared to What...?
- 0.5 Does EvalC3 Use Machine Learning?
- 0.6 Internal and External Validity
- 0.7 Contra Regression Analysis
- 0.8 Pro and Contra QCA
 - What I Like About QCA
 - Where I Am in Disagreement
 - 1. Defining Necessity and Sufficiency
 - 2. Measuring Consistency and Coverage
 - 3. Using the Quine McCluskey Algorithm
 - 4. The Consequences of Using a Truth Table
- 0.9 Realist Evaluation and Process Tracing
 - Realist Evaluation
 - Process Tracing
- 0.10 Background Reading
- 0.11 Origins

1. Input Data

- 1.0 Input Data
- 1.1 Usable Data
 - 1.1.1 Multiple Observations of One Case
- 1.2 Data Sets
- 1.3 Data Preparation
 - 1.3.1 Dichotomising Variable Data
 - 1.3.2 Using a Data Analysis Matrix
- 1.4 Participatory Predictive Modelling
 - 1. Participation in the Generation of Data
 - 2. Participation in the Development of Predictive Models

2. Select Data

- 2.0 Select Data
- 2.1 Selecting Cases
 - 1. At the Beginning: When the Data Set is Imported

- 2. Towards the End: When Predictive Models Have Been Identified
- 2.2 Selecting Attributes and Outcomes
 - Choices when importing data
 - Risks

3. Design Model

- 3.0 Design model
 - 3.1 Search options
 - 3.1.1 Search parameters
 - 3.2 Analysis sequence
 - 3.3 Decision Trees
 - 3.4 Solver - a genetic algorithm

4. Evaluate Model

- 4.0 Evaluate model
 - 4.1 Sensitivity and INUS Analysis
 - 4.2 The adjacent possible
 - 4.3 Context effects (aka Scope conditions)
 - 4.4 "Boring" versus "interesting" models
 - 4.5 Finding Positive Deviants
 - 4.6 Testing models with new data

5. Compare Models

- 5.0 Compare models
 - 5.1 Reviewing models
 - 5.2 EvalC3 versus QCA results
 - 5.3 Mapping a fitness landscape?

6. Select Cases

- 6.0 Select cases
 - 6.1 Within-case analysis
 - 6.2 Network analysis of cases

7. Obtain EvalC3

- 7.0 Obtain EvalC3

0.0 About EvalC3

...tools for developing, exploring and evaluating predictive models of expected outcomes

EvalC3 is an Excel app designed for use in the monitoring and evaluation of the achievements of development aid projects (and parts thereof). But it also has much wider applicability.

EvalC3 enable users:

1. To identify sub-sets of attributes that describe an intervention & its context, and which are good predictors of the achievement of an outcome of interest.
2. To compare and evaluate the performance of these predictive models,
3. To identify relevant cases for follow-up within-case investigations to uncover any causal mechanisms at work.

Examples of four different kinds of uses are described in [Example Uses](#)

These predictions are based on the screening of a data set that (ideally) describes the attributes of a set of those interventions, their context and their outcomes. EvalC3 uses binary data (i.e. 0/1 values) that can represent category membership, or two halves of a range of numeric values.

While EvalC3 enables different forms of systematic

quantitative cross-case comparison, its use should be informed by [within-case knowledge](#) at both the pre-analysis planning and post-analysis interpretation stages.

The overall approach is based on the view that “association is a necessary but insufficient basis for a strong claim about causation”, which is a more useful formulation than simply saying “correlation does not equal causation”.

Influences: The design of EvalC3 makes use of two sets concepts and methods:

- Qualitative Comparative Analysis, a body of methods developed in Political Science. Especially its view of causality (equifinality, asymmetry, conjunctural) and importance of combining cross-case and within-case analysis)
- Predictive Analytics, a body of methods used largely for commercial purposes. Especially what is known about different search algorithms and how the performance of prediction models generated by these algorithms can be evaluated.

Goertz and Mahoney’s (2012) “A Tale of Two Cultures” was also an important influence. [Look here for other relevant references](#)

Four main tools are available to develop these predictions:

1. Manual hypothesis-led inquiry, used to explore the predictive power of specific attributes of prior interest. Suitable for data sets of any size. Ideally the first step in the process of analysis using EvalC3
2. Algorithm-based searches

1. To find the single best predictive model
 1. For quick answers
 1. Cumulative single attribute searches
 2. To avoid the “local optimum problem”
 1. Exhaustive searches of multiple attribute combinations, useful for small data sets
 2. Evolutionary searches, using a genetic algorithm, usefull in larger data sets
2. To find the best set of predictive models, covering all observed outcomes
 1. Decision Tree searches.

The results are generated instantaneously in the case of manual hypothesis testing, quickly with evolutionary and Decision Tree searches and sometimes much longer with exhaustive searches for combinations of attributes.

A range of performance measures: The results of each search is a predictive model, which describes a sub-set of attributes that is consistently associated with a specific kind of outcome. The number of the cases identified (and missed) by predictive models is summarised in the form of a truth table, commonly known as a Confusion Matrix. This table is then used to generate a range of measures of the performance of a given model, which are suitable for use in different contexts.

There is also a model store, where results of any previous model can be accessed: (a) to compare against the design and performance of the current model and (b) reloaded for further exploration.

Supporting tools: The EvalC3 application also two supporting tools:

1. Post cross-case analysis: A measure of project similarity which enables identification of cases most suitable for subsequent within-case investigation in order to identify the nature of any common causal mechanism underlying the project attributes that have been found to be good predictors of outcomes
2. Pre cross-case analysis: Two measures describing the whole data set.
 1. Diversity: The percentage of all possible configurations of the current set of attributes that are present in the data set. The higher the percentage the less likely a current model will be contradicted by new data
 2. Consistency: The proportion of all the configurations that have consistent outcomes e.g. all present or all absent. Higher levels of consistency will mean models that are found are less likely to have False Positive cases that will require additional attributes to explain their existence.

Additional options

Analysis of “effects of a cause”: The default setting for EvalC3 is to analyse “causes of an effect” where multiple project attributes may be contributing to an outcome of

interest.

However, EvalC3 can also analyse “effects of a cause”, where a particular project intervention (described by a specific attribute in a data set) may be contributing to multiple outcomes.

Triangulation: Data that has been analysed using Qualitative Comparative Analysis(QCA) or Decision Tree algorithms can also be imported and analysed using EvalC3 tools. See [the Data Sets page](#) for examples that can be experimented with.

Predictive models first developed by EvalC3 can also be triangulated by later re-analysis using Qualitative Comparative Analysis(QCA) or Decision Tree algorithm

Origins: The original Excel application was designed in 2015 by Rick Davies, who is now working with [Aptivate](#) to develop the current more user-friendly and robust Excel version. This is being done with two purposes in mind: (a) To widen the range of tools available to identify and analyse complex causal configurations, (b) To widen the use of such tools, among the global community of evaluators.

A pdf version of this page is available here:[EvalC3 tools for exploring and evaluating complex causal configurations](#)



EvalC3 by [Rick Davies](#) is licensed under a [Creative Commons Attribution-NonCommercial-ShareAlike 4.0 International License](#).

Based on a work at <https://evalc3.net/>.

0.1 Example uses

1. A typology of uses

EvalC3 can be of use at all stages of a project cycle:

1. During project selection:
 - To identify what attributes of project proposals are the best predictors of whether a project will be chosen for funding, or not
 - To identify how well a project proposal appraisal and screening process is as a predictor of the subsequent success of projects in achieving their objectives

2. During project implementation
 - When the effectiveness of specific activities are being measured using survey instruments which include both specific (facet) and general (global) measures of satisfaction with service or product delivered. EvalC3 can identify what combination(s) of facets best predict global satisfaction. For example:
 - Participants experiences with workshops and training events
 - Donors and grantees experiences of their working relationships with each other

3. During a project evaluation
 - “Causes of effects” analysis: To identify what combination(s) of project activities (and their contexts) were associated with a significant

improvement in beneficiaries lives.

- “Effects of causes” analysis: To identify what combinations of improvements in beneficiaries lives were associated with a specific project activity (or combination of)
 - To identify “positive deviants” – cases where success is being achieved despite the fact that failure is the most common outcome. See Postscript note below for details.
4. During a review of existing evaluations
- Re-analysing data that was collected, to verify the results. This is often possible with QCA based evaluations because QCA data sets are usually published as annexes to evaluations
 - Synthesising the results of multiple evaluations, into prediction rules concerning different types of outcomes

“Loose” Theories of Change

More generally, EvalC3 is suitable for use where a project, or part thereof, has a “loose” Theory of Change. Loose in the sense that while the outcomes have been identified, the activities needed to achieve these may not yet be clear and even less so the specific causal pathways that will be involved.

Loose Theories of Change are more likely to be found in participatory development projects, or projects involving a substantial degree of decentralization, as is often the case with projects covering large geographic area and/or many sectors.

For an extended discussion of loose ToC, see my 2016 paper "[Evaluating the impact of flexible development interventions using a 'loose' theory of change Reflections on the Australia-Mekong NGO Engagement Platform](#)". ODI Methods Lab Working Paper, March 2016

2. Published papers on the use of EvalC3

So far, these are few in number

[RESILIENCE IN IRAQ Impact evaluation of the "Safe access to resilient livelihoods opportunities for vulnerable conflict-affected women in Kirkuk" project](#). Alexia Pretari and Filippo Artuso, Farah Abdulrazzaq Salih, Kayghan Muhamed Saeed Taher, Mahran Alhaeyk, Sarah Nijholt for Optimum Analysis: data gathering on behalf of Oxfam. Effectiveness Review Series 2019/20

[SUSTAINABLE WATER AND SANITATION IN SIERRA LEONE Impact evaluation of the 'Improved WASH Services in WAU and WAR Districts' project](#). Jaynie Vonk, Oxfam GB. Effectiveness Review Series 2019/20

[Prediction Modelling with Qualitative Comparative Analysis](#). Hur Hassnain , 2019 YouTube video

[WOMEN'S EMPOWERMENT IN LEBANON Impact evaluation of the project 'Women's Access to Justice' in Lebanon](#). Lombardini, Simone, Hassnain, Hur, Garwood, Rosa, 2019. Oxfam Effectiveness Review Series 2017/18

[Learning From The Civil Society Challenge Fund: Predictive Modelling](#). TripleLine Briefing Paper, Rick Davies September 2015

A pdf version of this page is available here: [Example uses EvalC3](#)



EvalC3 by [Rick Davies](#) is licensed under a [Creative Commons Attribution-NonCommercial-ShareAlike 4.0 International License](#).

Based on a work at <https://evalc3.net/>.

0.2 Types of causes

EvalC3 can be used to explore a number of different types of causal relationships. At the very macro level these fall into these two categories:

- Causes of an Effect: An effect can have multiple causes.
 - E.g. There may be a number of factors, which led people to attend a conference
- Effects of Cause: A cause can have multiple effects.
 - E.g. Attending a conference may have many different effects on what I do afterward

More often people are trying to identify causes of an effect. These can take various forms:

- Conjunctural causes: Many events are caused by combinations of factors, rather than single factors.
 - E.g. I went to a conference in June because I was interested in the subject of the conference, I had friends going there who I would like to see and I had the time available to go.
- Equifinal causes: Events can arise as a result of many different conjunctions of factors.
 - E.g. Some other people went to the same conference in June because their boss told them to go and they had the relevant subject knowledge within their organisation.

- Multifinal causes: Particular factors (or combinations of these) can lead to many different effects.
 - E.g. People attending the same evaluation conference session end up making use of the session contents in many different ways

- Asymmetric causes: The causes of absent events may not be simply the absence of factors that cause them, but the occurrence of other additional factors.
 - E.g. One friend of mine did not go to the conference even though he had the time and was motivated to go. Unfortunately, his child was sick and needed to be taken for a medical checkup.

- Necessary but insufficient causes.
 - E.g. Having the relevant expertise to attend a conference may be necessary but insufficient. Permission from one's boss is also needed

- Sufficient but unnecessary causes:
 - E.g. Some people went to the conference because they were invited as speakers

- Necessary and sufficient causes
 - E.g. For some people, the combination of being told to go to the conference by their boss, and having the relevant expertise was both necessary and sufficient.

- Neither necessary or sufficient causes:
 - E.g. Being bored with what I was doing was not necessary or sufficient to lead me to go to the conference in June. but it may have been influential.

- PS: These can be as important and useful as necessary or sufficient conditions. See [this blog posting](#), specifically the section about satisficing versus optimising
- INUS causes: Insufficient but necessary parts of a configuration that is sufficient but not necessary.
 - For example, having the most relevant subject knowledge, among all the others in an organisation may be an insufficient but necessary factor that led to someone being sent to attend a conference. $(A+B)$ or $(C+D)$ leads to E
- SUIN causes: Sufficient but Unnecessary part of a configuration that is Insufficient but Necessary. $(A$ or $B) + (C$ or $D)$ leads to E. I can't think of an example here 😊
- Exclusive Or causes: Using a different example from the above, both credit and grant assistance may be sufficient to improve people's livelihoods. But providing them with both together may be counterproductive. $(A + \text{not}B)$ or $(\text{not}A + B)$ leads to E

What EvalC3 can do

1. Find attributes which are Sufficient and/or Necessary for an outcome
2. Find combinations of attributes (aka configurations) that are Sufficient and/or Necessary
3. Enable manual tweaking of predictive models to identify the extent to which they are INUS conditions, i.e. if the model fails to perform if they are removed.

4. Develop separate predictive models for either the presence and absence of an outcome
5. Developed predictive models for the causes of an effect.
6. Develop multiple predictive models, each which predicts some but not all of the outcomes in a data set.

0.3 Prediction vs explanation

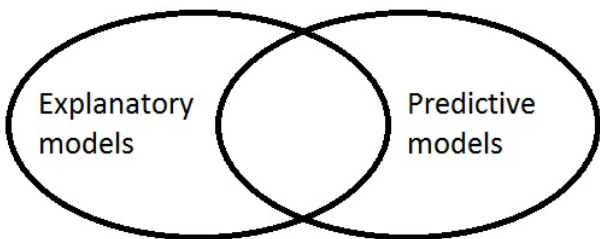
The search for causal explanations can be likened to a search for a needle in a haystack.

The development of predictive models is a way of identifying what part of the haystack we should be looking in. But the best performing model (i.e which identifies the part of the haystack which should be looking into) does not by itself provide a *causal* explanation that we may be looking for. Associations are a necessary but insufficient basis for a good causal claim.

Additional steps need to be taken once a good performing predictive models is found. There needs to be a detailed [within-case analysis](#) to investigate how, if at all, the attributes in the model are causally connected in real life. To extend the haystack metaphor, this is like deciding to open up the hay bales in the area where the predictive model said we should be looking

Even if it is found that there is no underlying causal connection for a given predictive model this is not necessarily a bad finding. Predictive models can still be useful on their own. A predictive model that would enable donors to fund projects that were successful in achieving their objectives 75% of the time, versus a chance based choice of 50% of the time, would still be a very useful product. Many grant making bodies are not even able to quantify their performance in these terms.

And different kinds of models can be useful for different kinds of people...



Historians

Surgeons

Stock market traders

0.4 Compared to what...?

What is special about EvalC3

- EvalC3 combines both manual hypothesis testing and algorithmic search
- EvalC3 provides a range of search algorithms
- EvalC3 provides a range of model performance measures
- EvalC3 provides case selection tools not available elsewhere
- EvalC3 is tolerant of missing data
- EvalC3 is available in user-friendly Excel

EvalC3 has similarities and differences with two other methods of analysis:

- [Qualitative Comparative Analysis \(QCA\)](#)
- [Decision Tree learning](#), as available in software packages like [Rapid Miner](#)

The similarities

- All three can work with binary data, which is more widely available than numerical data [...because all numeric data can be converted into binary data]
- All three analyse relationships between cases in terms of how their sets of attributes overlap, or not

- The results of all three analyses can be compared using some common performance measures
- Their findings can all be interpreted within the same view of causality, known as multiple conjunctural causation

The differences

- **Missing data:**
 - EvalC3 is tolerant of missing data points, more so than QCA.
 - Decision Tree software in packages like Rapid Miner also seem to be more tolerant than QCA, and it is currently more transparent than EvalC3 in the way it deals with missing data.
- **Minimisation / Search:** (i.e. finding a configuration that fits many cases)
 - Exhaustive search is used by EvalC3, and I think to some degree by QCA software. When used to find important single attributes this is both quick and effective. When used to find combinations of multiple attributes this can be very slow, though still very effective.
 - The Quine-McCluskey algorithm is used by QCA to reduce many configurations to the smallest possible number that have the same core elements. It is not helpful in data sets with limited diversity of configurations e.g. where many of the configurations are significantly different from each other (i.e. more than one attribute different).
 - Decision Tree models can be generated using

EvalC3 and Rapid Miner software to produce very readable results. However, Decision Tree algorithms can be prone to over-fitting i.e. prioritizing accuracy over generalisability.

- A genetic algorithm is also built into EvalC3. This is efficient when dealing with large data sets and many attributes, but it may require more than one application to find the best solution.
- **In summary**, all algorithms have their strengths and weaknesses. Ideally, we should test the results found by using one algorithm by also using an alternative algorithm.
- **Performance measures:**
 - EvalC3 uses provides multiple measures of model performance, which will be suitable in different contexts.
 - QCA focuses on two (consistency and coverage) and of these, it then seems to privilege consistency.
 - Rapid Miner has a similar range to EvalC3 but fails to use QCA measures like necessity and sufficiency, which EvalC3 does use
- **Visualisation of results:**
 - Decision Trees are the most user-friendly visualisation.
 - Venn diagrams as sometimes used by QCA require more familiarisation/explanation.
 - EvalC3 uses three methods: Decision Tree diagrams, plus a combination of a Design menu and a Confusion Matrix (truth table) to display

the results.

- **Manipulation of results:**

- EvalC3 includes the capacity to manually configure prediction models, and to tweak models developed by its two algorithms.
- These options are not available in Rapid Miner and QCA software

- **Sensitivity / contribution analysis:**

- Prediction models developed by EvalC3 can be manually adjusted to identify which attribute in the model contributes most to its overall performance.
- QCA can do a similar form of analysis (known as INUS analysis) but not with the same degree of precision.

- **Case selection:**

- EvalC3 has a systematic process for identifying individual cases most suitable to follow up within-case analysis.
- This option is not available in Rapid Miner and QCA software

0.5 Does EvalC3 use Machine Learning?

Answer: Yes – in a very simple form, and thus it is more transparent and less like a [black box](#)

Machine learning of the kind used in EvalC3 has these elements:

1. A systematic search process that finds a set of attributes that MAY be a good predictor of the outcome of interest
2. An evaluation function, that tests the performance of this possible predictor
3. A memory function, that stores this result or any previous result, whichever was the better performer.
4. Reiteration of the above process (1-3)
5. A stopping rule, which says when to stop the search and evaluate functions and to publish the best performing predictors identified so far.

Machine learning is all about incremental search and progressive improvement of candidate solutions (predictive models)

Some forms of machine learning are more sophisticated than others in how they do this. For example, [genetic algorithms](#) (built into the Solver add-in used in EvalC3) don't systematically search all possible combinations (which can take a lot of time). Instead, they mimic an evolutionary

process using re-iteration of variation, selection, and reproduction.

0.6 Internal and external validity

Internal validity

The design of the EvalC3 workflow, which progresses from cross-case analysis to within-case analysis (albeit with some recycling between the states sometimes), is orientated towards establishing a form of internal validity. In cross-case analysis an association is (hopefully) found between particular attributes of cases and an outcome of interest. Then, through within-case analyses, we might find evidence of causal mechanisms at work underlying that association. Ideally, we can then anchor the description of an abstract association with the verifiable particulars of individual known cases.

External validity

We can also give some attention to the prospect of external validity, the prospect that the association that has been found might also be found in other settings. Evidence of this possibility can be seen by looking at the diversity, or the lack of it, of the attributes of the cases within the dataset. The most extreme possibility in a set of cases would be where each had a unique set of attributes, there were no duplicates. We could also compare such set of cases in terms of the number of attributes that were documented. No duplication amongst a set of cases with many attributes per case would be indicative of greater diversity compared to no duplication amongst a set of cases with only a few attributes per case.

The presence of diversity within a set of cases is encouraging. The existence of each type of configuration tells us what is possible (outcome -wise) when that type of

configuration occurs. But where there are gaps in the range of possible configurations in a dataset this does point to potential risks for external validity. It suggests that any model that has been found work within the current set of cases may not work when applied in other settings, where there are cases with configurations not found present in a dataset that was used to develop the model. Within EvalC3, on the Select Data worksheet there is a measure of diversity present in the dataset currently being used. This tells us what proportion of all the possible configurations of attributes that could exist amongst the cases are not actually present. So it can be seen as a kind of measure of risk, the risk of a predictive model not being applicable in the wider world.

Sampling and external validity

Random sampling of a population of cases is one way of ensuring that the results of an analysis can be generalised. But they only give confidence about generalisation from the sample to the population at large where the sample came from. Not necessarily beyond that population. However, if the diversity of the attributes of the cases within the sample is high, rather than low, we might have a bit more confidence about the ability of the model to work outside the sampled population. Where we are dealing with small populations we may not have to resort to sampling, but the degree of diversity of cases within the population will still have significance in terms of potential for generalisation of findings to other populations.

0.7 Contra Regression Analysis

My main concern about using regression analysis is that it is suitable for some situations and not others. It is suitable for use where:

- Causal processes can be expected to be symmetric i.e. the causes of the absence can be expected to be the absence of the causes of the presence of the outcome
- One single model is being sought to account for all cases of outcomes present and absent
- The variables that make up a regression model can be assumed to act independently of each other.

These assumptions are different from those embedded in a QCA perspective, which assumes:

- Causal processes may not be symmetric
- There may be multiple different packages of causes generating all known cases of an outcome
- Some causes may only work when present as part of a package of causes, i.e, they are not independent

In Barbara Befani's very informed 2016 book "[Pathways to Change: Evaluating development interventions with Qualitative Comparative Analysis \(QCA\)](#) Annex B explains the differences between QCA and regression analysis. The same explanation also applies to EvalC3, because like QCA it is also a form of comparative configurational analysis. I have quoted the annex in full here:

“The QCA is often compared with regression analysis because both methods attempt to establish an association between a number of causal factors and an outcome (see for example (Vis, 2012)). In regression analysis, these factors are referred to as “variables” because they usually can take any value in an interval of real numbers; while in QCA they are referred to as “conditions” because they denote presence or absence of a certain quality or state in a given case. However, despite some apparent similarities, the differences between QCA and regression are numerous and substantial (Thiem, Baumgartner, & Bol, 2015).

First of all, in regression analysis, association is intended as “concomitant variation” between a single variable and an outcome (see Annex A): if the value of the outcome tends to increase with the value of the independent variable, we observe a correlation between the variable and the outcome. By contrast, in QCA, association is intended as a set relation: union, intersection or inclusion. If the outcome is “included” in the condition, or logically implies the condition, the association will be one of “necessity”; conversely, if the condition is “included” in the outcome and logically implies the outcome, the association will be one of “sufficiency”. While correlation is symmetrical (if x is correlated with y , then y is correlated with x), association in QCA isn’t: conditions can be necessary but not sufficient, or sufficient but not necessary. This property is also referred to as “causal asymmetry”.

The second important difference between QCA and the most common type of regression analysis (that doesn’t take interaction effects into account) is that, while in regression analyses associations are established between the outcome and one variable at a time, QCA considers cases “as wholes” or “packages”, analysing associations between combinations of conditions and the outcome; which makes the emergence

of contextual influence easier to spot. While in regression analysis the causal power of one variable, identified by the regression coefficient, is valid “on average” across the entire sample, in QCA the causal power of one condition is dependent on which other conditions it is combined with. In other words, the association is “conjunctural” (hence the word “conjunctural” in multiple conjunctural causation, see Annex A), or dependent on a specific context or setting.

Thirdly, while regression analysis aims at the identification of the one single model that fits the data best, QCA allows the identification of multiple, equally important pathways to the outcome; for example, two or more conditions that can be equally necessary for an outcome; or two or more combinations of conditions that are equally sufficient (hence the term “multiple” in multiple-conjunctural causality)”

0.8 Pro and Contra QCA

What I like about QCA

1. The perspective on causality: equifinality, asymmetry, conjectural causation, the concepts of necessary and/or sufficient causes, all described [in more detail here](#)
2. The combination of cross-case analysis and within-case analysis, the idea of moving back and forward between these levels of analysis

Where I am in disagreement

1. [Defining Necessity and Sufficiency](#)
2. [Measuring Consistency and Coverage](#)
3. [Using the Quine McCluskey algorithm](#)
4. [The consequences of using a Truth Table](#)

1. Defining Necessity and Sufficiency

EvalC3 uses a categorical definition of necessity and sufficiency. Following colloquial and [philosophical](#) use, a prediction model attribute is either necessary or not, or sufficient or not. It is a black and white status, there are no degrees of necessity or degrees of sufficiency. To me, the idea of having degrees of sufficiency or necessity is contradictory to the very kernel of the meaning of both of

those terms.

Yet QCA experts allow for this possibility when they talk about consistency of sufficient conditions and consistency of necessary conditions. For example, a configuration that has 20 True Positives and 5 False Positives would be described as having a Sufficiency consistency of 80%. Or a configuration with 20 True Positives and 10 False Negatives would be described as having a Necessity consistency of 66%. Along with this comes the more difficult notion of a threshold on these measures when a set of conditions aka a model then qualifies for a more categorical status of being sufficient, or necessary. For example, anything having more than 75% Sufficiency consistency is deemed to be Sufficient. But how this threshold is to be defined in any objective and accountable way escapes me. All [Schneider and Wagemann \(2012\)](#) say can say is “...the notion that the exact location of the consistency threshold is heavily dependent on the specific research context”

2. Measuring consistency and coverage

QCA experts have made the task of communicating their analyses to others more challenging by defining these two terms differently, according to whether they are talking about conditions that are necessary or sufficient.

- Consistency of sufficient conditions = True Positive / (True Positive and False Positive)
- Consistency of necessary conditions = True Positive / (True Positive and False Negative)
- Coverage of sufficient conditions = True Positive / (True Positive and False Negative)
- Coverage of necessary conditions = True Positive /

(True Positive and True Positive)

Again, keeping closer to the commonplace meaning of these terms, EvalC3 has only one definition each for consistency and coverage:

- Consistency of a model = True Positive / (True Positive and False Positive)
- Coverage of a model = True Positive / (True Positive and False Negative)

These two terms have others names in other fields of work:

- Consistency is also known as Positive Predictive Value (PPV), or Precision
- Coverage is also known as True Positive Rate (TPR), Recall, or Sensitivity

3. Using the Quine McCluskey algorithm

This is called a “minimisation” algorithm, because it tries to reduce a larger set of configurations down to a smaller subset, that still accounts for all cases of outcomes present and absent. As I understand it the key to the way this algorithm works is by finding cases where there is only one condition/attribute difference between the two case configurations, and where they either have the same outcome present, or same outcome absent. Because the presence or absence of this one different condition seems to make no difference to the outcome, it is treated as disposable, and removed from both configurations. A search continues for any other case that is the same as these two

reduced configurations, except for the presence of one other condition, or the absence of one other existing condition. The same reducing rule applies, if the outcome is the same when the condition is present or absent, then it can be removed from the configurations being examined. The process of comparing cases with different configurations continues until no more redundant configurations can be removed. The simplified i.e. shortened configurations that remain are the “solutions” i.e. predictive models found by the algorithm.

The problem with this algorithm, as I see it, is that because it is very incremental, only continuing to work where there is one condition difference, it seems by definition unable to find common minimal configurations in cases with two or more differences. This is not a problem when the data set contains all possible configurations of conditions. But it becomes problematic as this case diversity becomes a smaller and small sub-set of all the possible configurations. In this situation, the final set of “solutions” (models) may be more numerous than those that can be found by other algorithms, like Decision Tree searches.

In contrast, search algorithms of the kind used in EvalC3 don't depend so much on adequate diversity within a set of cases. They can find the best fitting set of attributes in two very different configurations. That said, they can still generate more than one equally good fitting model where there are relatively few cases and relatively many attributes.

“Limited diversity” in a data set also presents another problem common to both approaches. It means that any good fitting model may have limited external validity. Other new cases with new and different configurations may well contradict and thus cause the failure of these models.

4. The consequences of using the Truth Table

Here is an example of a Truth Table, see in this recent paper: Kien, Christina, Ludwig Grillich, Barbara Nussbaumer-Streit, and Rudolf Schoberberger. 2018. '*Pathways Leading to Success and Non-Success: A Process Evaluation of a Cluster Randomized Physical Activity Health Promotion Program Applying Fuzzy-Set Qualitative Comparative Analysis*'. BMC Public Health 18 (1): 1386. <https://doi.org/10.1186/s12889-018-6284-x>. PS: My use of this data set as an example is not a critique of this paper, it simply happens to be the one I am most immediately familiar with.

Table 5 Truth Table Summarizing Recipes for Achieving Pupils' Emotional and Social School Experience

Conditions					Outcome	n	Consistency
PAB	QOI	PSE	BOI	KAI	SCE		
0	1	1	0	1	1	1	0.873
1	1	1	0	0	1	1	0.821
1	1	1	1	1	1	3	0.799
1	0	1	0	1	0	1	0.750
0	1	0	1	1	0	1	0.748
0	0	1	0	1	0	2	0.662
1	0	1	1	1	0	2	0.631
1	1	0	1	1	0	1	0.628
1	1	1	1	0	0	1	0.626
1	0	0	1	1	0	2	0.573
0	1	0	0	0	0	1	0.514
1	0	0	0	0	0	1	0.497
0	0	0	0	0	0	1	0.478
0	0	0	1	0	0	1	0.386
0	0	0	1	1	0	2	0.343
1	0	1	1	0	0	3	0.225

Abbreviations: *BOI* benefits of intervention, *KAI* knowledge about intervention, *n* number of school classes showing a specific causal condition, *PAB* physical activity breaks (dosage), *PSE* perceived self-efficacy, *SCE* improvement in school experience, *QOI* quality of implementation, *0* absent, *1* present

Each row represents a type of configuration, a unique pattern of case attributes. The column “n” tells us how many cases have each of these unique patterns. It is the rows in this table that QCA works with, more specifically the Quine McCluskey minimisation algorithm that finds the simplest possible set of versions (i.e. “solutions”) of these that still accounts for all the outcomes observed and not observed. The performance of each of these solutions is

measured in terms of their coverage and consistency.

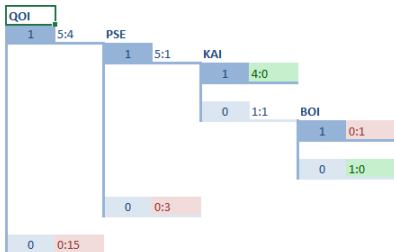
My understanding of the calculation of these measures is that they are also based on the contents of the truth table i.e. the incidence of each **type** of configuration (16 above), not the total number of cases they represent (24 above). This is an important difference, especially for someone wanting to operationalise the findings in real life. In the worst case there may be many cases with one type of configuration but only 1 of others. This could seriously skew the significance of the consistency and coverage measures of a given configuration.

In contrast, when the same data set is used for prediction modelling the Truth Table is “unpacked” into rows that represent all the cases, one by one, as shown below – for the Truth Table above.

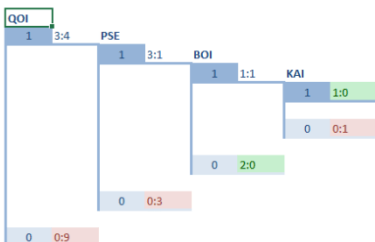
ID	PAB	QOI	PSE	BOI	KAI	SCE	
1	0	1	1	1	0	1	1
2	1	1	1	1	0	0	1
3	1	1	1	1	1	1	1
4	1	1	1	1	1	1	1
5	1	1	1	1	1	1	1
6	1	0	1	0	0	1	0
7	0	1	0	0	1	1	0
8	0	0	1	0	0	1	0
9	0	0	1	0	0	1	0
10	1	0	1	1	1	1	0
11	1	0	1	1	1	1	0
12	1	1	0	1	1	1	0
13	1	1	1	1	1	0	0
14	1	0	0	0	1	1	0
15	1	0	0	0	1	1	0
16	0	1	0	0	0	0	0
17	1	0	0	0	0	0	0
18	0	0	0	0	0	0	0
19	0	0	0	0	1	0	0
20	0	0	0	0	1	1	0
21	0	0	0	0	1	1	0
22	1	0	1	1	1	0	0
23	1	0	1	1	1	0	0
24	1	0	1	1	1	0	0

Here are two Decision Tree models generated from the original and unpacked versions of the Truth Table. The unpacked version has 8 more rows i.e. 50% more.

Balanced accuracy: 100.00% True Positive: 20.83% True Negative: 79.17% False Positive: 0.00% False Negative: 0.00%



Balanced accuracy: 100.00% True Positive: 18.75% True Negative: 81.25% False Positive: 0.00% False Negative: 0.00%



0.9 Realist Evaluation and Process Tracing

Realist Evaluation

[Updated 2018 10 07] A core concept in Realist Evaluation is the CMO configuration. CMO stands for Context, Mechanism, Outcome. [The original text worth reading is Pawson and Tilley (1997) [Realist Evaluation](#)]

Context describes what are thought to be important features of the setting in which things may happen. This may include a description of an intervention taking place in that setting.

Mechanism is a causal process happening within individual people or within organisations (depending on the scale of analysis) that is triggered by the particular set of context conditions. It leads to an...

Outcome, which is the result of the interaction between Context and Mechanism

As in QCA there can be many different CMO configurations that can be at work in a given programme, policy or project. Each of these, as initially imagined, is a theory that needs to be tested.

One of my interests is how to do this. Especially if there are a lot of potential CMOs to consider, and not much time (if you are an evaluation team)

My conjecture is that you need a multi-stage process:

1. Find out what happened.
 1. What outcomes occurred. Prior theory should help
 2. In what different kinds of contexts did they occur and not occur. Prior theory should help
2. Find out what conjectured Context and Outcome events were actually co-occurring in reality. Use QCA or EvalC3 software to find this out. This is cross-case analysis.
3. Focus in only on those CMOs where there is an association. In this order of priority:
 1. Where the context conditions are necessary and sufficient for the outcome
 2. Where the context conditions are necessary or sufficient for the outcome
4. Then invest resources in within-case analyses to find out if the conjectured mechanism is at work, or if some other mechanism is in operation, or if there is none at all.

This conjecture relates to a discussion of the sequencing of realist evaluation and related methods, on [a recent ITAD blog posting](#)

Which brings us to Process Tracing...

Process Tracing

This is a method of within-case analysis to identify causal mechanisms at work within individual cases. [Look [here for some references](#), there is a lot written on the subject] It is

one way of investigating the Mechanism element of a CMO, and how it causally connects Context with Outcome. Key items of evidence, in a plausible story of how things happened, fall into four categories:

1. Hoop tests: A person guilty of a murder will not have an alibi. If they do the theory that they committed the crime fails. Not having an alibi is a necessary part of the causal process, and the basis for prosecution. But it is not sufficient.
2. Smoking gun test: Being in possession of a gun at the scene of a murder is sufficient to enable the murder, but it is not necessary.
3. Doubly decisive test: Being in possession with a gun at the scene of the murder and the dead person's body containing a bullet of the same character (bore marks) as those in the gun, is both necessary and sufficient
4. Straw-in-the-Wind test: This is evidence that is relevant but neither necessary nor sufficient

How does this relate to EvalC3? The EvalC3 workflow ends with case selection and within-case analysis. One important form of within-case analysis is to examine True Positive cases, comparing modal and outlier cases. With each of these types of cases, it would be useful to do some form of process tracing, exploring the expected causal connection between the model attributes and the outcome. This is where the above four tests could be used.

0.10 Background reading

Background reading I have found of value

Goertz, G., Mahoney, J., 2012. [A Tale of Two Cultures: Qualitative and Quantitative Research in the Social Sciences](#). Princeton University Press. Great background context for the kind of analyses possible with EvalC3

See also Gary Goertz new book: [Multimethod Research, Causal Mechanisms, and Case Studies: An Integrated Approach](#), 2017. Princeton University Press.

Kotu, V., Deshpande, B., 2014. [Predictive Analytics and Data Mining: Concepts and Practice with RapidMiner](#). Morgan Kaufmann. Contains very useful chapters on Decision Tree method

[Befani, B. Evaluating Development Interventions With QCA: Potential And Pitfalls](#). Forthcoming 2016. The definitive book on QCA written from an evaluation perspective

Papers I have written

[Alternative approaches to exploring and testing complex causal models of development interventions](#). Presentation to UK Evaluation Society Conference, March 2016. Available as YouTube video

[“Evaluating the impact of flexible development interventions using a ‘loose’ theory of change: Reflections on the Australia-Mekong NGO Engagement Platform”](#) ODI Methods Lab. March 2016

[“Evalating loose theories of change”](#) A YouTube video of Rick Davies’ presentation to DFID EvD staff in October 2015. This presentation put the methods used by EvalC3 in a wider context

See also

[The Compass website](#): “COMPASSS (COMPARative Methods for Systematic cross-caSe analySis) is a worldwide network bringing together scholars and practitioners who share a common interest in theoretical, methodological and practical advancements in a systematic comparative case approach to research which stresses the use of a configurational logic, the existence of multiple causality and the importance of a careful construction of research populations”

0.11 Origins

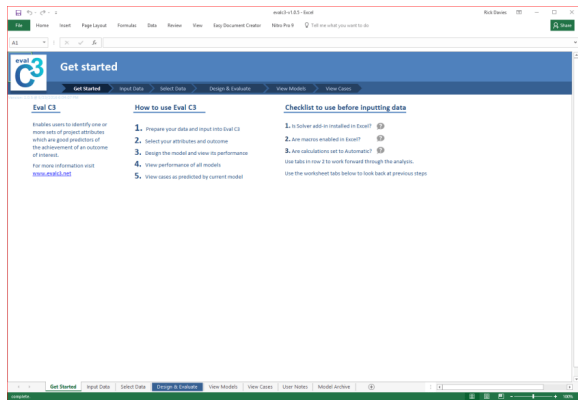
Rick Davies developed the basic idea (and its basic functioning) for this software application in 2015 while doing some prediction modelling for [TripleLine](#), using the [Rapid Miner](#) suite of data mining tools (in particular the Decision Tree module). More information on his consulting background is [available here](#)

Since early 2016 Rick has been working with Aptivate (especially Mark Skipper), a Cambridge (UK) based software development company, to make a package of data analysis tools available in Excel. Excel was chosen as the platform because it is probably the most widely used data analysis tool, amongst Monitoring Evaluation staff and consultants

1.0 Input data

1. Open EvalC3 version of Excel

1. This will open at the *Get Started* worksheet.
Read this first



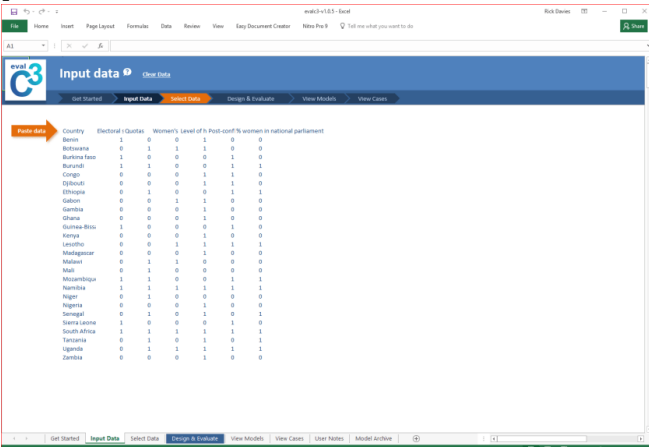
2. Pay attention to the Checklist on the right side
3. Click on *Start – Input Data*, which will take you to the *Input Data* worksheet. Clue: Don't use the Worksheet tabs at the bottom to *progress forward with the analysis*. Use the blue sequence of tabs at the top. But you can use the bottom tabs *to go back, without altering the current analysis process*

2. Import data into Excel

1. Cut and paste the relevant rows and columns of your data into the *Input Data* worksheet, which should be the first worksheet to open
 1. Make sure you have included the column header names
 2. Clue: If you want to edit your data by

adding extra columns or rows, either do it now, or even before you cut and paste the data here

3. Here is an example of a data set that has been cut and pasted



The screenshot shows a software interface with a data table. The table has three columns: 'Country', 'Electoral / Quotas', and 'Women's level of 1's Rank-coef. The data is as follows:

Country	Electoral / Quotas	Women's level of 1's Rank-coef.
Benin	1 0 0	1 0 0
Bhutan	0 1 1	1 0 0
Burkina Faso	1 0 0	0 1 0
Burundi	1 1 0	0 1 1
Congo	0 0 0	1 1 0
Cyprus	0 0 0	1 1 0
Ethiopia	0 1 0	0 1 1
Gabon	0 0 1	1 0 0
Gambia	0 0 0	1 0 0
Ghana	0 0 0	1 0 0
Guinea-Bissau	1 0 0	0 1 0
Japan	0 0 0	1 0 0
Lesotho	0 0 1	1 1 1
Madagascar	0 0 0	1 0 0
Malawi	0 1 1	0 0 0
Mali	0 1 0	0 0 0
Mozambique	1 1 0	0 1 1
Namibia	1 1 1	1 1 1
Niger	0 1 0	0 0 0
Nigeria	0 0 0	1 0 0
Senegal	0 1 0	1 0 1
Sierra Leone	1 0 0	0 1 0
South Africa	1 1 1	1 1 1
Tanzania	0 1 0	1 0 1
Uganda	0 1 1	1 1 1
Zambia	0 0 0	1 0 0

4. Click on *Select Data* button, which will take you to [the Select Data worksheet](#)

1.1 Usable data

The current version

A data set has to have the following structure:

- Rows = cases, such as individual projects, households, people,
- Columns = aspects of those cases, which include
 - At least one ID column, uniquely identifying each row of data
 - At least one Outcome measure
 - Multiple Attributes of the cases, that may or may not be good predictors of the outcome, by themselves or in combinations with others

The cells contain binary data. Here the values of 0 and 1 are used to code the absence or presence of an attribute or outcome.

Nuance: If you are concerned that 0 or 1 is too crude a description of a case attribute then the alternative is to break that attribute down into a number of subsidiary attributes, and then code for the presence or absence of each of these. If there are five subsidiary attributes this means there can be 2 to the power of 5 (i.e. 32) different forms of the original attribute, which should be more than sufficient in many situations.

Missing data: EvalC3 manages missing data values in

predictable ways. If the attribute in a predictive model is a “1” i.e. is expected to be present in a case, then a missing value is interpreted as a “0”. On the other hand, if the attribute in a predictive model is a “0” i.e. is expected to be absent in a case, then a missing value is also interpreted as a “0”. In the first of these two instances, the model is “pessimistic”, i.e. assumes cases with missing values do not have the model attributes. In the second instance, the model is “optimistic” i.e. assumes the cases with the missing values do have the model attributes. But if the predictive model combines multiple attributes, some of which are expected to be present and some absent, then it will be more challenging to identify in which net direction the model is biased

Please also pay attention to point 6 here on data preparation <https://evalc3.net/data-sets/data-preparation/>

Size of dataset: The largest dataset I have used had 597 cases and 35 attributes. On this scale the Decision Tree algorithm worked quite slowly, taking about 5 minutes to be generated. In the transition from Select Data to Design and Evaluate, it would sometimes size up and display an error message.

1.1.1 Multiple observations of one case

In most circumstances an EvalC3 data set will describe multiple attributes of multiple cases, where each case is represented by one row.

However in some circumstances there may only be one case that is of interest, but it may be that multiple observations can be made of the attributes of this case over a period of time. For example, a single project that varies its approach over a period of time. Or a single family, whose welfare and wider circumstances vary over time. In both of these situations each row in an EvalC3 data set can represent a set of observations made at a given period of time.

Where there are many observations made over an extended period of time sampling issues may need to be considered. For example, whether to include all observations, or only the most recent, or only a moving fraction. The reason for the latter is that a given predictive model may only apply for a particular period of time. This may be for good or bad reasons. See the wikipedia entry on [Goodhart's Rule](#) and [Campbell's Law](#)

Postscript 1: There is a body of literature on single case research designs, which the above is an example of. Here is a quote re these designs:

“Single-case research designs (also referred to as “single subject designs”, “single-case experimental designs”, and “n-of-1 trials”; henceforth, SCRDs) have been used to assess intervention effects for many decades (Barlow & Hayes, 1979; Herson & Barlow, 1976). In contrast to experimental designs

that involve comparing average outcomes across groups of individuals in different treatment conditions, SCRDS involve introducing an intervention to an individual case or cases and measuring changes in outcomes over time. Some types of SCRDS also involve removing and then re-introducing the intervention, providing further tests of the functional relationship between the intervention and the outcome. SCRDS are critically important for understanding the effectiveness of interventions for individuals with low incidence disabilities (e.g., physical disabilities, autism spectrum disorders), given the inherent difficulties in obtaining sufficient samples sizes for between-group experimental designs with such populations. As a result, SCRDS comprise a large part of the evidence base in certain areas within fields such as special education and school psychology. The results of SCRDS can under some circumstances provide a strong basis for understanding the causal effects of interventions (Gast & Ledford, 2014). They have the added advantage of providing information about intervention effects at the level of individual cases, whereas between-group experimental designs are informative only about average effects. Thus, the results of SCRDS are relevant for informing clinical and public policy decisions, and should be considered for inclusion in systematic reviews and meta-analyses that aim to synthesize the existing evidence about intervention effects (Council for Exceptional Children Working Group, 2014; Kratochwill et al., 2013).”

Valentine, J. C., TannerSmith, E. E., Pustejovsky, J. E., & Lau, T. S. (2016). Between-case standardized mean difference effect sizes for single-case designs: A primer and tutorial using the scdhlms web application. *Campbell Systematic Reviews*, 12(1), 1–31. <https://doi.org/10.4073/cmdp.2016.1>

Postscript 2: I have just read a 2020 paper by [Sofia Pagliarin](#)

and and Lasse Gerrits pointing out that it is possible to have, and analyse, a set of cases that combine multiple different cases A-Z, along with multiple versions of some cases that are described at different points in time: At1, A t2, etc

Postscript 3: Configurations i.e. characteristics of a case may change over time. If we treat different aggregated time periods (A-J = 0 versus K-Z = 1) as the “outcome” to be predicted, we could develop predictive models which summarise the core features (=a model) of each aggregated time period. An aggregated time period could be 1960-1970, or a presidential term, or an agricultural season.

1.2 Data sets

Listed below are some example data sets that can be imported into EvalC3 and analysed there. These and others (along with their source documents) can be found on the [Compass website](#), a repository for QCA studies.

The challenge:

1. Can you replicate the results reported in the papers below?
2. Can you improve on those results, in terms of the accuracy of the prediction model, the breadth of cases it covers, or its simplicity? Or on any other criteria that could be argued to be appropriate?

The datasets

These are arranged roughly by size of data set.

The Krook data set is the data set describing 26 African countries is built into EvalC3 as an example data set to play with.

- Welle et al (2015) [Testing the Waters: A Qualitative Comparative Analysis of the Factors Affecting Success in Rendering Water Services Sustainable Based on ICT Reporting](#), June 2015. Itad. Water Aid, IRC.
 - [EvalC3-example-data-set-6-water-aid](#)
 - Outcome 1: 8 projects x 9 attributes and 1 outcome. Binary data
 - Outcome 2: 8 projects x 7 attributes and 1

outcome. Binary data

- Outcome 3: 8 projects x 6 attributes and 1 outcome. Binary data

- Balthasar, A. (2006) “[The Effects of Institutional Design on the Utilization of Evaluation.](#)” Evaluation 12 (3):353-71.
 - [EC3 Example data set 8 Balthasar](#)
 - 10 institutions x 8 attributes and 1 outcome

- Ansorg, N. 2014. “[Wars without Borders: Conditions for the Development of Regional Conflict Systems in Sub-Saharan Africa.](#)” International Area Studies Review 17 (3):295-312.
 - [EC3 Example data set 7 Ansorg](#)
 - 12 regions x 7 attributes and 1 outcome. Binary data

- Krook, M.L., 2010. [Women’s representation in parliament: A qualitative comparative analysis.](#) Political Studies 58, 886-908.
 - [EvalC3-example-data-set-5](#)
 - 22 developed countries x 5 attributes of those countries and 1 outcome measure (% of women in parliament). Binary data.
 - 26 African countries x 5 attributes of those countries and 1 outcome measure (% of women in parliament). Binary data.

- Basedau, Matthias, and Thomas Richter. 2014. “Why do some Oil Exporters Experience Civil War but others do not? Investigating the Conditional Effects of Oil.” *European Political Science Review* 6 (4):549-74.
 - [EC3 Example data set 16 39 x 4 Basedau](#)
 - 39 cases x 4 attributes and 1 outcome

- Blackman, Tim, Jonathan Wistow, and David Byrne. 2011. “A Qualitative Comparative Analysis of Factors Associated with Trends in Narrowing Health Inequalities in England.” *Social Science & Medicine* 72 (12):1965-74.
 - [EC3 Example data set 17 Blackman](#)
 - 27 cases x 10 attributes and 1 outcome
 - 27 cases x 6 attributes and 1 outcome

- A random data set
 - [EvalC3 example data set 9 Random](#)
 - 100 cases x 10 attributes and 1 outcome
 - This data set can be useful as a comparator for processing speeds using different types of search.
 - An exhaustive search for configurations took me 6 minutes on [HP Pavillion 550 153a desktop](#)
 - An evolutionary search for configurations took me less than one minute on the same machine

- Bara, C (2014). “[Incentives and Opportunities A Complexity-Oriented Explanation of Violent Ethnic Conflict.](#)” *Journal of Peace Research* 51, no. 6

(November 1, 2014): 696–710.

- [EvalC3-example-data-set-15-Bara](#)
 - 500 cases x 11 attributes and 1 outcome.
Binary data

- I have found that EvalC3 struggles with this data set, especially when using exhaustive search. but this may depend a lot on the age and size of your computer

- Itad&DFID (2015) Empowerment and Accountability macro-evaluation. See [here for details](#) of the evaluation and [here for details](#) of the data set
 - [EvalC3 -example-data-set-16-DFIDE&A](#)
 - 523 cases x 117 attributes

1.3 Data preparation

The following steps will be useful to undertake, prior to loading data into EvalC3:

1. Check each attribute column for **missing datavalues**.
Prioritise the use of those attributes where there is no missing data. EvalC3 can work with cases that have missing data, but the models that are developed will be conservative, i.e. they will assume all cases with missing data do not fit the best performing model.
 1. For ideas on how to deal with missing values see
 1. <http://www.missingdata.org.uk/>
 2. <http://www.measuringu.com/blog/handle-missing-data.php>
2. Check each attribute column to ensure there is some **variation in cell values**. If they are all the same then the attribute will be of no value as a potential predictor. Outcome columns must include the presence and absence of outcomes.
3. If an attribute or outcome values are originally in numerical form and needs to be dichotomised into binary form (1's and 0's) then take care to ensure that there is some degree of **balance in the number of presence and absence cases**.. Where presence (for example) is either rare or very common then be aware that there will be a greater than normal risk of False Positives or False Negatives respectively.
4. Try to **minimize use of attributes that are highly correlated** in the way they appear across cases in the

data set. having more than one such attribute will not improve the predictive power of models that can be developed.

5. Think about **timing**: when each attribute was collected or when it happened. You don't want a predictive model that shows X leads to Y, when in fact X happened after Y
6. **Be careful when coding qualitative data from participatory or found sources**
 1. Coding of events of interest as 1/0 can be problematic, because typically we may have evidence that x event happened, but evidence of it not happening may or may not be there. It may have happened or it may have happened but was not reported.
 2. In this situation instead of coding 1/0 for presence and absence of an attribute, 1/0 in one column could represent the known presence / unknown status of an attribute and a second column could represent known absence / unknown status of an attribute.

Analysis planning: You may also find it useful to do some planning about the types of analysis to be carried out, once you have uploaded the data. Especially if you have a data set with many attribute and outcomes of interest. One way of planning an analysis is to use a data analysis matrix as [described in detail here.](#)

1.3.1 Dichotomising variable data

Technical terms

Dichotomisation is the process of converting variable data into binary data. For example we might have a string of variable measurements such as numbers of participants in an event: 7, 15, 23, 45, 63, 75, 84, 93. These can then be converted into binary values representing the lower and upper values: 0, 0, 0, 0, 1, 1, 1, 1.

There are different technical terms describing this process. In the machine learning field it is described as “[binning](#)” but in the QCA literature it is partially covered by the term “calibration”

Information and noise

If you Google “dichotomising data” you will find lots of warnings, that this is basically a bad idea!. Why so? Because if you do so you will lose information. All those fine details of differences between observations will be lost.

But what if you are dealing with something like responses to an attitude survey? Typically these have five-pointed scales ranging from disagree to neutral to agree, or the like. Quite a few of the fine differences in ratings on this scale may well be nothing more than “noise”, i.e. variations unconnected with the phenomenon you are trying to measure. One source of noise could be differences in respondents’ “[response styles](#)”.

Different methods

In order to dichotomise some variable measures a choice

needs be made of a cut-off point, above which one value will be assigned (1) and equal to or below which another value will be assigned (0). The choice of a cut-off point can be made by a range of methods.

1. The analyst may have some **prior theory** in mind which suggests that values above a certain point will have different consequences to those below.
2. Prior **practical experience** with similar interventions might have already shown that a certain threshold has to be passed before an intervention can have noticeable effects.
3. There may be no prior theory or experience but on examination of the data might show a **significant gap in the distribution**, which could be used as the basis for the values.
4. There might not be such a gap in the distribution of values, in which case the choice might be made to simply use the **median value** as the cut-off point.
5. The choice of cut-off value might be driven by a **value concern**, rather than any empirical observations or theories about what the consequences are. For example that all participants should receive at least X amount of an expected benefit.

This last method seems particular appropriate for dichotomisation of an outcome variable. Whereas the theory and experience-based methods (1 & 2 above) seem more appropriate to the dichotomisation of a variable which might have some causal role i.e. have some consequences for an expected outcome.

An inductive approach

This 6th method involves looking at the *relationship* between the variable data that you need to dichotomise and the outcome of interest. Let's assume the outcome has already been dichotomised, on the basis of some level of performance that we think is necessary.

What we want to then do is construct a 2 x 2 table, like this:

		Outcome y =	
		1	0
Cases > X			
Cases <= X			
		8	10

X is a cut-off value, selected within the range of values that the variable of interest has. Lets start off with the median value. Based on that we fill in the cells that with the number of cases that meet the row and column criteria

		Outcome =	
		1	0
Cases > X	7	2	
Cases <= X	1	8	
		8	10

We then calculate the Chi-square statistic, which is a measure of how different the cell values are from what otherwise would be an equal distribution across all cases. The bigger the Chi-square value the more unequal the distribution. The example above has this value: 8.1.

Then manually vary the X value, choosing a value somewhere above or below the median. Here is another example with a different cut-off value. In this example the Chi-square value is 11.52. The distribution of cases is more unequal.

		Outcome =	
		1	0
Cases > X	8	2	
Cases <= X	0	8	
	8		10

We could continue varying the X value until we cannot find any other one that has a higher Chi-square value. That is the one we will choose to keep and use. This is because this cut-off value is in effect a good single attribute predictor of the outcome of interest. In the above example it has an 88% accuracy (Accuracy = (TP+TN)/(TP+FP+FN+TN)). This single attribute model now provides us with a good building block for building more complex configurational models, along with other attribute data, when using EvalC3.

[Go here to find ASC, a simple Excel tool](#) that you can use to do either a manual or automated search for the best cut-off point with your data

This method fits with my preferred definition of information, which is ‘a difference that makes a difference’ – an idea suggested by [Gregory Bateson](#) some decades ago. The first difference is between the upper and lower values on either side of a cut-off point. And the difference it makes is its ability to predict/classify the status of the outcome variable (already dichotomised)

2022 05 07: Postscript: Dichotomisation of fuzzy set values

Here is a data set of fuzzy set values from a recent paper: Meissner, K. L., & Mello, P. A. (2022). The unintended consequences of UN sanctions: A qualitative comparative analysis. *Contemporary Security Policy*, 0(0), 1–31. <https://doi.org/10.1080/13523260.2022.2059226>

Table 2 of 5
Table 2. Sanctions regimes and calibrated fuzzy-set values

Case/Sanctions Regime	Autocratic Regime	Economic Isolation	Large Scope	Long Duration	UNSC P5 Invol.	Unintended Consequences (Outcome)
Angola, 1993–2002	0.72	0.05	1.00	0.76	1.00	1.00
Central African Rep., 2013–2014	0.38	0.97	0.05	0.00	1.00	0.00
Cote d'Ivoire, 2004–2014	0.25	0.02	0.76	0.49	1.00	0.99
Dem. Rep. of Congo, 2003–14	0.08	0.15	0.05	0.46	0.00	0.95
Dem. Rep. of Yugoslavia, 1991–92	0.68	0.03	0.05	0.00	0.73	1.00
Dominican Rep., 2012–2014	0.27	0.93	0.00	0.01	0.00	0.00
Haiti, 1993–1994	0.90	0.77	1.00	0.01	1.00	1.00
Iran, 2006–2014	0.96	0.64	0.19	0.32	1.00	0.05
Libanon, 2009–2014	0.03	0.02	0.00	0.93	1.00	0.00
Libania, 1992–2014	0.17	0.00	0.97	1.00	0.00	1.00
Libya 1, 1992–2003	0.97	0.38	0.08	0.93	1.00	1.00
Libya 2, 2011–2014	0.38	0.01	1.00	0.02	0.93	1.00
North Korea, 2006–2014	1.00	0.27	1.00	0.35	1.00	0.81
Rwanda, 1994–2008	0.92	0.91	0.08	0.99	1.00	0.05
Sierra Leone, 1997–2010	0.22	0.93	0.76	0.97	0.00	1.00
Somalia, 1993–2014	0.24	0.00	0.76	1.00	1.00	0.93
Sudan 1, 1996–2001	0.97	1.00	0.01	0.13	0.00	0.00
Sudan 2, 2004–2014	0.87	0.98	0.08	0.74	0.00	0.00

It should be possible to find optimal cut off values for each attribute& outcome combination using the [ASC Excel tool](#). Then see what difference it makes to the subsequent analysis, compared to the theory led approach used in this and other QCA papers.

1.3.2 Using a Data Analysis Matrix

Why?

Sometimes you have way more data, especially attributes of cases, than you can sensibly analyse in one exercise. Developing a matrix of the kind shown below can help. It makes the planning of your analyses transparent: what you will analyse and what you will not. And thus more accountable.

Example

Here is an example I developed and used in 2015 when helping a UK consulting firm plan a data mining exercise using a data set that had 60+ cases and more than 70 potentially useful attributes. In this matrix...

- Each blue column represents a grouping of a specific kind of case attribute. At the analysis stage, any one of these could be used as an outcome in an EvalC3 data set
- Each blue row represents a grouping of a specific kind of case attribute. At the analysis stage, any one of these could be used as attribute which might be predictive of the outcome of interest in an EvalC3 data set
- Cells represent possible relationships between specific types of attributes (rows) and specific types of outcomes (column)...
 - Colored (grey and yellow) cells represent those relationships that were of interest and which would be analysed

- Initials in these cells represent the stakeholders with specific interest in this relationship

- The cell values in the summary column on the right represent the level of confidence in that row type of case attribute
- The cell values in the summary row at the bottom represent the level of interest in the potential outcomes of interest represented by each column

The analysis that was carried out focused on the 23 colored cells. They represent 27% of all the possible types of analyses ($7 \times 12 = 84$) that could have been undertaken

Lettering Guide ST- Study team JG- x key person LR- y key person	Project performance rating	Change in discourse	Policy dev	Policy adoption	Policy Implementation	Access to gov services	(Quality of gov services	Innovative service delivery	Effectiveness	Sustainability	Multiplier effects	Equity	Level of confidence in the data
CSCF objectives		ST	ST				ST	ST	JG	JG			1
Methodological approach		JG	JG	LR	LR	JG	LR	LR	JG			ST	1
Project management									JG				2
Sector (Logframe category)						JG						JG	3
Proposal appraisal	LR								LR	JG			4
Partner										JG			5
Capacity building									LR				-
Level of interest	-	1	1	-	-	2	2	-	-	3	4	5	

1.4 Participatory predictive modeling

People's participation in predictive modeling can happen at two stages:

1. The generation of data
2. The development of predictive models

1. Participation in the generation of data

Data sets that can be analysed by EvalC3 can come from different sources: a once-off research or evaluation exercise or from ongoing monitoring systems. And as explained below, they can also be generated by participatory means.

Two types of data can be generated by participatory means: (a) outcome data, which a good predictive model should be able to identify, (b) attribute data, which may be predictive of outcomes (identified by participatory or other sources of data).

Why?

Stakeholders in a project, such as those implementing the intervention, and those experiencing its effects, are likely to have views about what works, and what does not work, which may be much wider ranging and sometimes closer to the truth, than the contents of official monitoring systems. It can be worth tapping into those views.

How?

There are two ways in which people can participate in the generation of attribute and outcome data: (a) pile or card sorting, (b) online survey instruments.

Pile sorting is a long-established form of ethnographic inquiry that enables us to identify participants' view of the world, primarily the categories they use to describe their world. There are many different ways of doing pile sorting, well summarised in [Harloff and Coxons, 2007 "How to Sort" guide](#)). The simplest approach is called "free sorting". Participants are presented with a list of events, activities, people or objects that they are familiar with and then asked to sort them into piles, and then to label each pile with a description of what the items in that pile have in common, but which makes them different from the items in the other piles. The task may be explained in ways that make the focus on the inquiry as broad or narrow as needed. For example. "Please sort these projects in two piles capturing a difference between them *that you think might affect how successful they are in achieving their objectives*".

Pile sorting is typically done with one or more respondents in a face to face meeting. The researcher (or evaluator) notes down which cards are put in which pile and then asks the respondent an open-ended question designed to elicit the respondents view of what the difference is, and sometimes, why they think it is important. Two types of information are recorded:

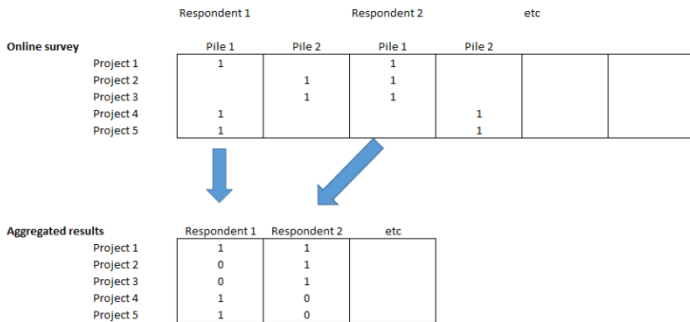
1. A text description of the nature of the difference between the two piles, and
2. The names of the items which are members of each pile.

Online survey instruments, such as those available via [Survey Monkey](#), are another means of eliciting these kinds of sorting results and judgments from participants. The use of online survey is more suitable when dealing with larger numbers of respondents and/or respondents based in many different locations.

A draft version of such a survey can be [seen here](#). In this survey cases of interest, such as projects, are listed in the rows of a matrix. Participants are then asked to sort them into two piles, by using two columns of check-boxes, representing two piles. A Comment field below the matrix is then used to capture the participants' description of how the two piles of cases differ. The differences the survey is looking for are "differences that (might) make a difference" to the outcome of interest. The same pile sorting question can then be repeated more than once in the same survey instrument, in order to capture more than one set of differences from the same respondent.

Aggregating pile sorting data on attributes

Regardless of which method is used (face to face or online survey) the results from all the participants are then aggregated into one large matrix of cases x attributes, with the attributes (and case values on these) being those contributed by the different participants via their pile-sort responses.



The matrix of data that remains can then be imported into EvalC3 for analysis. However, note that an extra column is also needed with data on the presence & absence of the outcome in each case. This may come from monitoring systems, evaluations, quantitative surveys or measures or participatory inquiry

Caveat

It is likely that some post-survey data cleansing will be needed. Some attributes will correlate highly with each other, in which case it would make sense if one or either was removed. Especially if the Comments suggest they are the same attribute or if one attribute is clearly easier to understand than the other.

Aggregating pile sorting of data on outcomes

As with the sorting of cases by attributes, cases can also be sorted into two piles according to whether the respondent's views on whether an outcome of interest is present or not. The outcome of interest can be defined in broad or narrow terms, by varying how the sorting request is expressed. Once the sorting is completed the pile labels can then be elicited to identify how the respondent was defining achievement of an

outcome

When responses from multiple survey participants are aggregated it then possible to generate an aggregate (multi-criteria) achievement scale, based on the number of respondents who placed a given case in the “successful” pile.

Sorting of cases into successful (1) and unsuccessful (0) piles

Respondents		A	B	C	D	E	F	G	H	I	J	Aggregate ranking
Cases												
1		0	1	1	1	0	0	0	1	0	1	5
2		1	1	1	0	1	1	0	1	0	1	7
3		0	0	1	1	1	1	1	1	1	0	7
4		1	1	0	0	0	0	0	1	0	0	3
5		0	0	1	0	1	1	1	1	1	0	6
6		1	1	1	1	1	1	1	0	1	1	9
7		0	0	1	1	1	0	1	1	0	0	5
8		0	0	1	1	1	1	0	1	0	1	6
9		0	1	1	0	0	1	0	1	1	1	6
10		1	1	1	1	0	1	1	1	0	1	8
Pile size		4	6	9	6	6	7	5	9	4	6	

When piles are of uneven sizes, then aggregation ranking calculation can include a weighting to take these variations in size

Associated with each of the aggregated ranking scores can be text statements (to the right of the matrix) describing the criteria used by each respondent who placed that row case in a “successful” pile. This set of statements will vary row by row in the example above, because different sub-sets of respondents will be involved in each row.

This process is simpler to use than the more traditional ranking methods, where each respondent constructs a full ranking of all cases. Especially where there are a large number of cases.

2. Participation in the development of predictive models

The data that is generated by either of the two methods described above can then be analysed in two ways. One is by using any of the four algorithms in EvalC3 to develop the

best possible predictive model, and then to follow-up with within-case inquiries to look for supporting evidence of any causal mechanisms at work.

The other is to engage the same participants in proposing and testing their own hypotheses about what combinations of attributes best predict the outcome of interest, by using the manual Design facility in EvalC3. Ideally, this would happen in a workshop setting where the Design and Evaluate view could be projected into to a large screen visible to all. A facilitator would elicit views from the participants, then enter their proposed set of attributes into the design menu. The performance of that model would then be instantly visible to the participants in the adjacent Confusion Matrix – whose contents would need to be explained by the facilitator. A discussion could then ensue on the significance of the results including the False Positives, False Negatives, the sensitivity of different attributes in the model, and changes that might improve the model.

What next?

If you would like some help in developing and using a participatory predictive modeling data set, email rick.davies@gmail.com

Postscript 1

There are a number of online pile sorting website, where people can take part in a range of types of pile sorting exercises, and the data then aggregated and exported for analysis. Although intended for use in improving the structure of websites they can be used for the same purposes as the online survey above. Most offer free non-premium services. See for example [OptimaSort](#). [See here](#) for a list of these services.

Postscript 2

The same data set (cases x attributes & outcomes) can also be analysed using social network analysis visualization software. The network structure of the matrix can be visualized in three forms:

(a) A two-mode network, showing how cases are variously connected via their shared attributes.

(b) A one-mode network, showing how cases are variously connected to each other, where the strength of these linkages is defined by the number of attributes they both share.

(c) A one-mode network, showing how attributes are variously connected to each other, where the strength of these linkages is defined by the number of cases they jointly apply to.

2.0 Select data

[A duplicate copy, in case you missed the original page]

When you click on *Select Data* button this will take you to the *Select Data* worksheet. An example is shown below, using the same example data set.

eval3 Select Data

Get Started Input Data **Select Data** Design & Evaluate View Models View Cases Compare Models

Configurations: 14 Consistency: 100% Diversity: 44% Missing data: 0% Find optimal attributes Sort by configurations

ID	Attribute	Attribute	Attribute	Attribute	Attribute	Outcome
Country	Electoral system	Quotas	Women's status	Level of human development	Post-conflict situations	% women in national parliament
Benin	1	0	0	1	0	0
Botswana	0	1	1	1	0	0
Burkina faso	1	0	0	0	1	0
Burundi	1	1	0	0	1	1
Kenya	0	0	0	1	1	0
Djibouti	0	0	0	1	1	0
Ethiopia	0	1	0	0	1	1
Gabon	0	0	1	1	0	0
Gambia	0	0	0	1	0	0
Ghana	0	0	0	1	0	0
Guinea-Bissau	1	0	0	0	1	0
Kenya	0	0	0	1	0	0
Lesotho	0	0	1	1	1	1
Madagascar	0	0	0	1	0	0
Malawi	0	1	1	0	0	0
Mali	0	1	0	0	0	0
Mozambique	1	1	0	0	1	1
Namibia	1	1	1	1	1	1
Niger	0	1	0	0	0	0
Nigeria	0	0	0	1	0	0
Senegal	0	1	0	1	0	1
Sierra Leone	1	0	0	0	1	0
South Africa	1	1	1	1	1	1
Tanzania	0	1	0	1	0	1
Uganda	0	1	1	1	1	1
Zambia	0	0	0	1	0	0

Get Started Input Data **Select Data** Design & Evaluate View Models Decision Tree View Cases Compare Models User Notes

1. Reading the characteristics of the data set. Above the data set itself are a series of measures that describe the dataset:
 1. Configurations: The number of unique configurations of attributes in the dataset. In this example dataset, there are 14, among a total of 26 cases
 1. Click on Sort by Configuration to show the cases grouped by configuration

2. Consistency: The number of configurations that have consistent outcomes i.e. all absent or all present, but not a mix of both.
 3. Diversity: The proportion of all the possible combinations (i.e. configurations) of attributes present in this data set, as a percentage of the total number that is possible given the number of attributes in this dataset. In this example Diversity of 44% = $14 / (2 \text{ to the power of } 5)$.
 4. Missing data: The percentage of all the cells in the data set that have no values (0 or 1)
2. Select Column Types and Choose Rows
1. By default, the leftmost column is automatically labeled as *ID*. To change this click on that cell and a drop-down menu will appear that gives an option to *Ignore* that column, to leave it as *ID* or to change it to *Attribute* or *Outcome*
 2. By default, the rightmost column is automatically labeled as *Outcome*. If you want to change that, click on that cell, and choose *Ignore* or *Attribute*. You will then need to click on another column heading in the same way and change that to *Outcome*.
 3. There must be one *ID* column and one *Outcome* column in any data set being prepared for use at this stage. There may be more than one outcome of interest in the data set but only one can be labeled as such at this stage, prior to going to *Design and Explore*.
 4. All the columns between *ID* on the left and *Outcome* on the right are by default labeled as

Attribute i.e. potential predictors of the outcome. But by clicking on any of these labels you can choose to change it to *Ignore*, or *Outcome*, or *ID*.

5. The status of any of the columns can be re-assigned later on. When you do this you are in effect loading a new data set. One consequence is that the findings from the analysis of the previous data selection will no longer be accessible in the View Models view – so keep a record of those findings somewhere, if they are important.

3. Click on [*Design & Evaluate*](#), which will take you to that worksheet

4. Optimizing the set of attributes being used
 1. This is an optional step to take before proceeding to Design and Evaluate. It can be useful when there are a large number of attributes in the data set, relative to the number of cases, and where there is no theory-based basis for removing some.

 2. By clicking on *Find Optimal Attributes* button a pop-up menu will provide these three options, to:
 1. Maximize the consistency of the configurations in the data set. A high percentage means most cases with a given configuration will have the same outcome. A low percentage means that often cases with the same configuration will have a mix of outcomes, i.e. both present and absent

 2. Maximize the diversity of the

configurations in the data set. A high percentage means most of the possible configurations of the attributes are represented in the data set, a low percentage means that only a few of the possible configurations are represented in the data set.

3. Maximize both the consistency and diversity of configurations. Neither measure may reach 100% but the highest possible measure on both will be found.
-
3. For more information on when these different optimization strategies will be useful, see [Selecting attributes and outcomes](#)

2.1 Selecting cases

[Update 2018 10 07] Cases are the examples or instances listed row by row in a data set. For users of this Excel application, these may be projects, or locations or groups within projects

Cases can be selected at two stages of analysis:

1. At the beginning: When the data set is imported.

Some cases may be deliberately left out with the intention that they will be used later as a “test” data set, to test the predictive power of the models developed with the portion of the data set currently being used. In the field of predictive analytics this is called Cross-Validation. The default percentage of test cases is usually 30%. See [Testing models with new data](#)

With EvalC3 cases can be left out either:

1. Prior to cutting and pasting data into EvalC3 (Input Data) or
2. When working within the Select Data view, by using the normal Excel filter function.

2. Towards the end: When predictive models have been identified that have satisfactory levels of

performance.

See the [Within-Case Analysis](#) page for advice on case selection strategies that are appropriate at this stage.

2.2 Selecting attributes and outcomes

[Updated 2018 10 07] The attributes of cases are the field names given at the top of each column in a dataset. These are sometimes called “features” in predictive analytics, or “conditions” in Qualitative Comparative Analysis (QCA).

Choices when importing data

When a data set is imported choices can be made about the status of each column of data. These choices affect the kind of models that can subsequently be developed using this data at any one time. They can be revisited and changed. Each column can be given one of four status:

1. ID: For example the name of a project. Basically any easily recognizable identifier for a case
2. Attribute: These will be the attributes of the cases that will be considered when predictive models are being developed. They are the possible “predictors” or independent variables.
3. Ignore: This are the attributes that will not seen as relevant to the current modelling exercise.
4. Outcome: One attribute must be selected as the outcome of interest, to be predicted by the models being developed. There may be more than one column of outcome data in the data set. If so, the others should be set to “Ignore”. Later on they can be re-assigned “outcome” status and used as the basis for a new model development. Or they can be assigned “attribute status” if you are looking for relationships between outcomes.

Risks

The choices made about the status of each column of data have consequences which should be born in mind

The more attributes that remain in an imported data set, the larger the number of possible combinations of these, and any one of these may be the most accurate model. The number of possible combinations rises exponentially, i.e. it doubles every time an additional attribute is included.

This has three consequences:

1. The required computation time increases. This is of greatest significance for the exhaustive search option. Exhaustive search works best with small numbers of attributes. or, when the model size is pre-specified in advance to be relatively small.
2. When there are many attributes relative to cases it is likely that there will be more than one good performing predictive model and in some cases it will not be possible to choose between them simply on the basis of their performance measures. However, subsequent within-case analyses may provide a basis for choosing between these.
3. For a given number of cases available, any increase in the number of attributes (and combinations thereof) reduces the probability that this set of cases will be a comprehensive representation of all those possible combinations. This means that a model may not perform so well when applied to new cases. These new cases may have new configurations of attributes that do not produce the outcome as previously predicted.

Attribute optimisation

In EvalC3 a sub-set of a larger set of attributes can now be identified which optimizes the *consistency* and/or *diversity* of the configurations in its associated data set. This is done via the *Find Optimal Attributes* button, which uses the Solver Add-In (more specifically, its genetic algorithm).

- Consistency is the extent to which all cases covering a given configuration have the same outcome or mixed outcomes (e.g. both present and absent)
 - Maximizing consistency is important if the aim is to identify/develop predictive models that have minimal levels of False Positives. Maximizing the consistency will improve the internal validity of the model
 - Calculation: Consistency is the percentage of all configurations having only one type of outcome i.e. $1 - ((\text{"# configurations ... including outcome"} - \text{"# configurations ... excluding outcome"}) / (\text{"# configurations ... excluding outcome"}))$
- Diversity is the extent to which all cases represent unique configurations versus duplicate one or more configurations.
 - Maximizing the diversity will reduce the number of models which best fit the same data. It also means that when the model is applied to new cases not in the current data set it is less likely to fail, because there are less surprises, i.e. configurations which don't fit the model. The external validity of the model will be improved.
 - Calculation: % Diversity = # of

configurations/ $(2^{\# \text{ of attributes}})$

- Consistency and diversity can be both maximized, though neither is likely to be perfect
 - Calculation: % Maximization = $(\text{Diversity} * \text{Consistency}) / (\text{Diversity} + \text{Consistency})$
 - This form of optimisation has similarities to what is known as [Quality-Diversity algorithms](#). In the EvalC3 implementation *consistency* of cases is the quality dimension and *diversity* of cases is the diversity dimension

A large set of attributes can also be reduced in size by removing *redundant* or *irrelevant* attributes. By either of these approaches:

1. Data centered: Using “[feature selection](#)” methods developed as an integral of data mining work. See Chapter 12 in Kotu, V., Deshpande, B., 2014. *Predictive Analytics and Data Mining: Concepts and Practice with RapidMiner*. Morgan Kaufmann. The simplest of these methods is to identify attributes whose values correlate highly with each other across all the cases available .i.e redundant measures. One of these can then be removed. This particular approach is not available specifically within EvalC3 but can be done using normal Excel functions
2. Theory centered: In its simplest form, this is using prior theory to inform choices about what attributes are likely to be more relevant than others. Another approach, called two-step analysis in QCA, is to divide the attributes into two or more groups and use one group at a time. e.g. a context attributes and

intervention attributes. A further option is to then take the attributes making up the models that fitted both groups, pool them into a new smaller set and then to analyse these as a whole.

3.0 Design model


There are two different ways of building a predictive model:

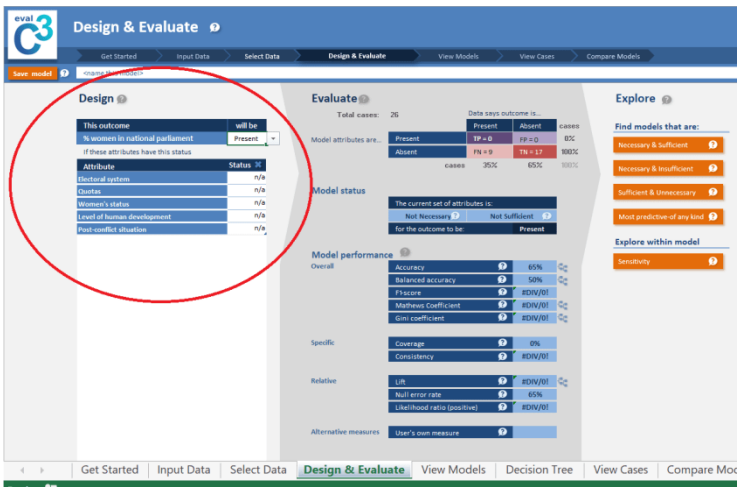
- Manually, based on a prior theory of any kind (e.g. your own or someone else's, a formal or an informal theory)
- Automatically, using one of the search algorithms built into EvalC3

1. Manual design

1. Once the *Design and Evaluate* view is open look at “Design” on the left side (circled in red in the screenshot below). Here you choose what values to place next to each of the attributes that are automatically listed here. The drop-down menu in the Status column provides three options: N/A meaning ignore this attribute; 1 = this attribute is present, 0 = this attribute will be absent.
2. The default status for each attribute when this view is first opened is N/A.
3. You also need to choose whether the Outcome is expected to be present or absent when these attributes are as described above, using the same kind of drop down menu in the Status column
4. This combination of attribute values and the selected outcome then constitute a predictive model
5. The performance of this model can then be seen

immediately in the Confusion Matrix under the heading “Evaluate”, which is explained more below

6. Click on the Save button (above left) to save details of this model and its performance. You will need to save the model with a name you will recognize later.
7. If you want to remove all the attributes of a model in one go i.e re-set them all to n/a click on the round “Stop” sign  to the right of attribute “Status”



The screenshot shows the 'Design & Evaluate' interface. The 'Design' tab is active, displaying a table with the following data:

Attribute	Status
Outcome of system	n/a
Gender	n/a
Woman's status	n/a
Level of human development	n/a
Post-conflict situation	n/a

The 'Evaluate' tab is also visible, showing model performance metrics:

- Overall: Accuracy (85%), Balanced accuracy (56%), F1 score (RDIV/0), Matthews Coefficient (RDIV/0), Gini coefficient (RDIV/0)
- Specific: Coverage (8%), Consistency (RDIV/0)
- Relative: Lift (RDIV/0), Null error rate (85%), Likelihood ratio (positive) (RDIV/0)

2. Search algorithms

The screenshot shows the Eval3 Design & Evaluate interface. The top navigation bar includes 'Get Started', 'Input Data', 'Select Data', 'Design & Evaluate', 'View Models', 'View Cases', and 'Compare Models'. The 'Design & Evaluate' section is active, showing a table of attributes and their status. The 'Evaluate' section displays a confusion matrix and model performance metrics. The 'Explore' section, highlighted with a red circle, contains buttons for finding models that are: Necessary & Sufficient, Necessary & Insufficient, Sufficient & Unnecessary, and Most predictive of any kind. Below these are buttons for exploring within a model, such as Sensitivity.

1. Chose which type of model you want to find (circled in red in the screenshot above). There are four options, each represented by a button you can click on:
 1. Necessary & Sufficient: This kind of model will consist of a single attribute, or set of attributes, which are both necessary and sufficient for the outcome. In the Confusion matrix, there will be no False Positives and no False Negatives. In reality, this kind of model is rare.
 2. Necessary but Insufficient: This kind of model will consist of a single attribute, or set of attributes, which are necessary but insufficient for the outcome. In the Confusion matrix, there will be no False Negatives but there will be some False Positives.
 3. Sufficient but Unnecessary: This kind of model will consist of a single attribute, or set of attributes, which are sufficient but unnecessary for the outcome. In the Confusion matrix, there will be no False Positives but there will be some False Negatives.

4. Most predictive – of any kind: This kind of model will consist of a single attribute, or set of attributes, which are likely to be in sufficient and unnecessary for the outcome, but still are good predictors, as measured by Accuracy, for example. In the Confusion matrix, there will be some False Positives and there will be some False Negatives. But it may also be the case that this search finds one of the above three models.

2. When any of the above buttons are clicked this will take you to a *Find New Models* pop-up menu. This presents a choice of four search algorithms. See [Search Options](#) on this website for more detailed information about these choices



Find the configuration of attributes that best predicts the presence/absence of the outcome according to a selected performance indicator.

Search type

- Exhaustively test all configurations of attributes
- Evolutionary search for best configuration of attributes (using Solver)
- Find one additional attribute that gives best performance
- Build a decision tree with maximum depth: ▾

Value to optimise

Balanced_Accuracy ▾

Number of attributes

Number of attributes in model configuration
'any' means any number of attributes

any ▾

Constraints

Add

Change

Delete

- Necessary and Sufficient
- Necessary but not Sufficient
- Sufficient but not Necessary

Cancel

OK

3. Choose the performance indicator: the measure that should be maximised by the best models that can be

found. There are three groups of these: Overall, Specific and Relative. For more information on these see [Evaluate Model](#). *Clue: Start by using the most widely used measure: Accuracy*

4. Set constraints. These can be of three types, which can be used by themselves or in combination:
 1. Particular attributes in the Design view whose values need to remain fixed. For example, as being present or absent
 2. Specific performance measures other than the one selected as the objective. For example that Lift =>100%
 3. Specific values for one or more cells in the Confusion Matrix
 1. Try setting False Positive = 0, to find Sufficient but Unnecessary attributes (or configurations of attributes)
 2. Try setting False Negative = 0, to find Necessary but Insufficient attributes (or configurations of attributes)
 3. Try setting False Negative = 0 and False Positive = 0 to find Necessary and Sufficient attributes
 4. **Postscript:** There are now three radio button options that can be used to set these constraints with one click
5. Implement the search by clicking Okay
 1. If using exhaustive search, watch the process bar in order to assess if the results will be ready within the time available. If not, cancel.

View the results of the search, given the settings above.

The attributes that have been found as the best predictors of the outcome (known as “the model”) will appear in the *Design* area, replacing any previous selection. This found model will automatically be saved and the saved name will be visible to the right of the “Save Model” button

The raw results of the prediction model will be shown in the Confusion Matrix in the *Evaluate* area. See [Evaluate Model](#) for more information on how to read the Confusion Matrix.

The performance measures derived from the Confusion Matrix can be seen listed further below the matrix. These are used to summarise the performance of the current model in predicting the outcome of interest.

Revise the results

Within the Design & Evaluate worksheet you can tweak the values of the attributes in the new model in order to:

1. Incrementally improve performance of the model
2. Identify what attributes in the model contribute most/least to its overall performance. For more on this option see [Sensitivity Analysis](#)
3. **Postscript:** There is now a Sensitivity button on the right, which if clicked will then highlight the attribute in the current model which contributes the most to its good performance. This is measured by comparing the % point reduction in model performance when each attribute is selectively removed from the model

Save the results

Save the results of each version of the model that you find to be of value. This will be done automatically, with a unique name, if exhaustive of evolutionary searches have been carried out. But if there has been any manual tweaking the resulting model will then need to be saved manually

3.1 Search options

If a dataset has information on 10 different attributes of projects this means that there could be 210 different combinations of these that might be the best predictor of an outcome of interest i.e. 1,024 possibilities. EvalC3 provides a number of ways of searching through these possibilities to find the most accurate predictor:

1. **Hypothesis-led** manual selection of attributes, based on a theory derived from past experience and/or research elsewhere. The advantage of this approach is that where the hypothesis is correct there may already be a good foundation knowledge, from prior research, on why it works. In EvalC3 a prediction model can be developed manually by inserting relevant values into the model design (under the Design), and then observing its performance. Normally this should be the first step in an analysis process using EvalC3. However it is possible that there are other solutions with an even better fit with the data, which lay out of sight outside our current understanding,
2. **Additional attribute search.** This is an incremental form of exhaustive search. There are two main ways of using it
 1. Where there is already an existing model the attributes of this model are treated as search constraints. An exhaustive search is then be made of $x+1$ attributes, where x is the number of attributes in the current model.
 2. Where there is no existing model using the “additional attribute search” will search for the best performing single attribute model. This is useful when searching for single attributes that

are necessary or sufficient for the outcome. If this search is re-iterated it will treat the result of the first search as a constraint that has to be met. The new model will have $x + 1$ attributes

3. There is a risk, that I have not substantiated, that this form of incremental search will get stuck in “local optimum“. There are two ways of checking if this is the case, which are valid search strategies in their own right:
 1. **Exhaustive search**, where every possible combination of multiple attributes is examined. Because it is exhaustive the results will be conclusive. However, an exhaustive search can be very time consuming if there are many attributes (processing time doubles with each additional attribute in a data set). This problem can now be mitigated by specifying the maximum number of attributes in any model found by exhaustive search. I often try this search with a 3 or 4 attributes maximum
 2. **Evolutionary search**. When data sets are large (deep and/or wide) an exhaustive search described above can be too slow to implement. Evolutionary searches are a very efficient means of searching for complex (i.e. multi-attribute) models within much larger combinatorial spaces. EvalC3 makes use of an existing Excel add-in known as Solver, to carry out evolutionary searches. However evolutionary searches are not necessarily as conclusive in their findings as exhaustive searches, because they sample different combinations of attributes, rather than test all of them. For this reason, the value of the results generated by an evolutionary search should be tested by repeating the search a

number of times

4. **Decision Tree** searches provide another option: a way of generating a whole set of models, which best predict all outcome, both present and absent. As with the exhaustive search, it is possible to specify the depth of the tree i.e. the maximum number of attributes in the models generated by this search.

3.1.1 Search parameters

[Last updated 2018 10 06] When any of the four algorithm based searches are to be used then the first step is to set some parameters about how the search will proceed. These are:

1. **Value to optimise:** This is where you insert the model performance measure that needs to be maximised, by clicking on the value cell next to the chosen performance measure.
2. **Model size.** This can be set in two ways:
 1. On the right of the Decision Tree search option is a drop-down menu allowing you to set the “depth” of the tree, meaning the maximum number of attributes in any of the models generated by the Decision Tree search
 2. Further below is the “Number of attributes in model configuration” setting, where you can specify the maximum number of attributes in any model found by exhaustive search
3. **Constraints:** the project attributes and/or performance measures whose values must remain within specified limits. There are two ways of setting constraints:
 1. Click on one of the three options at the bottom of the Find New Models pop-up:
 1. Necessary and Sufficient. This enters $FP=0$ and $FN=0$ in the Constraints box.
 2. Necessary but not Sufficient. This enters $FN=0$ in the Constraints box.
 3. Sufficient but not Necessary. This enters

FP=0 in the Constraints box.

2. Enter any other preferred value in the Constraints box,
 1. These can refer to Confusion Matrix cell values e.g. $FP < 5$,
 2. They can also refer to specific performance measure values, such as $Lift > 150\%$
3. You can also go to the Design menu and in the list of Attributes Status select one or more attributes to have specific values (either 1 or 0)
 1. You can explore “the adjacent possible” by setting some of the attributes of a given model as a constraint and then exploring in the vicinity of that part of the model using “one additional attribute” search

Changing the Solver (evolutionary) search parameters

In the background, outside of EvalC3, there are settings which govern how Solver runs. To see and change these go up to the top of the Excel interface and click on Data, then look to the far right and click on Solver. Click on Options, then Evolutionary. For information on all the search parameters listed here, which you can change, see this web page:

<http://www.solver.com/excel-solver-change-options-evolutionary-solving-method>

3.2 Analysis sequence

Warning, this page has been frequently re-edited 😊

There are two main stages:

1. Manual testing of pre-existing hypotheses. This is done by entering attributes hypothesized as important into the Design menu and observing the model's performance in the Confusion Matrix
 - Make sure you save any models you value.
2. Algorithmic search for better models, as described in detail below.
 - All of these models are automatically saved

In both stages the *overall* aim is to find a model with attributes that maximise the number of True Positives (TPs) and True Negatives (TNs) and minimises the number of False Positives (FPs) and False Negatives (FNs). The relevant model performance measure here is Accuracy i.e. $(\text{True Positive} + \text{True Negative}) / (\text{True Positive} + \text{True Negative} + \text{False Positive} + \text{False Negative})$. But note that there are nuances here that you will often want to explore, relating to the proportions of False Negatives to False Positives – for more, see Model Performance section [here](#)

Re algorithmic search

The best strategy may depend on the objective of the analysis.

- When trying to understand, what has happened as part of a research or evaluation exercise, we may need to find a number of models, which as a set do the best in accounting for all the outcomes.
 - The best EvalC3 tool for identifying a comprehensive set of models is the [Decision Tree algorithm](#).
- When trying to work out what best to do next a much less comprehensive analysis may be all that is needed. We just need to find one or more models which seems to work well, and which we can have some confidence in.

The advice below is oriented towards finding a smaller number of models that best account for the outcome of interest.

1. Start by searching for one or more attributes which are **Necessary and Sufficient**. These are by definition *unambiguous and essential*, so need to be found if they exist. But also bear in mind that they are uncommon.
 - Click on the Necessary and Sufficient button, in the Explore section, then use the first or third algorithm. Use the first, if your data set is small, or you have plenty of time. Otherwise, use the third.
2. Then search for **Necessary but Insufficient** attributes, using the button of the same name. These attributes are necessary for the outcome, but not sufficient by themselves.
3. Then search for the **Sufficient but Unnecessary** attributes, using the button of the same name. This is an optional solution, which will work, but it is not the

only way. The search algorithms will try to find the best Sufficient model i.e one with the largest coverage (least False Negatives)

4. Then search for one or more attributes which may be **Unnecessary and Insufficient**, but which are still a good predictor of the outcome. Use the “Most predictive of any kind” button.
 - Bear in mind that a model with only 1 False Negative and 1 False Positive will have a higher Accuracy than a Sufficient model with 5 False Negatives.
 - Which of these two kinds of models are preferred will depend, in part at least, on the acceptability of having any False Positives at all. A surgeon would want zero, but a gambler would typically tolerate a proportion of False Positives.

When good performing models have been identified consider doing a simple **sensitivity analysis** of each model.

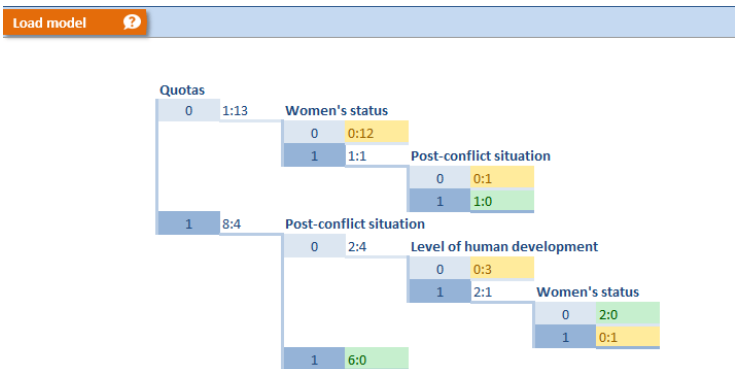
Then proceed to do within-case investigations after identifying relevant cases from the View Cases worksheet using the guidance provided on **Selecting Cases**

A pdf copy of this web page is available here: **Analysis sequence**

3.3 Decision Trees

1. What it looks like

...as generated by EvalC3



2. How read this Decision Tree

The tree should be read from left to right, as though it is a tree that has fallen over

This tree has 7 branches, each with a “leaf” at the end, shown by the color beige or green. Each of these branches is a prediction model, made up of a particular configuration of case attributes, described by the text labels.

The green and beige leaves describe the numbers and types of outcome found.

Beige= outcome absent. Green = outcome present. This is the

predicted outcome of this model, given the distribution of cases on the leaf.

First number = Number of cases with outcome present.
Second number = Number of cases with outcome absent.
Essentially the same as the top row of the Confusion Matrix.

The example here uses data from the Krook QCA study of women's participation in parliament in 26 African countries

Let's read the top branch...Where "quotas" are absent (0) and "Women's status" is absent (0) this model find there are 0 countries with high levels of women's participation in parliament, but there are 12 cases where there are low levels of women's participation in parliament.

In the next branch...Where "quotas" are absent (0) and "Women's status" is present (1) this model find there are no countries with high levels of women's participation in parliament, and a "post-conflict situation" is absent (0) there are 0 countries with high levels of women's participation in parliament, but there is 1 case where there are low levels of women's participation in parliament.

In this example, each branch represents a configuration that is **sufficient** for the outcome being either present (green) or absent (beige). But sometimes both outcomes will be present, but one will be more common than the other. In other words, the model will have some inconsistency (in QCA terms) and have limited "Positive Predictive Value" or "Precision" – to use terms used elsewhere.

3. What to do with it

To **view** a particular model in detail, click on the 0 or 1 cell to

the left of the leaf you are interested in.

Then click on Load Model. This will take you to the Design & Evaluate view, where you will see the model attributes in the Design section and its performance measures in the Evaluation section

To save this model, click on Save Model

5. Where to learn more about Decision Tree and how they work

- “[A visual introduction to machine learning](#)” – I rate this as Excellent!

3.4 Solver - a genetic algorithm

This is one of the four search algorithms built into EvalC3, and visible as an option when you click on any one of the four Explore links on the right side of the Design & Evaluate worksheet.

If you choose option 3: Evolutionary Search this will set the Solver add-in in motion, using its default settings.

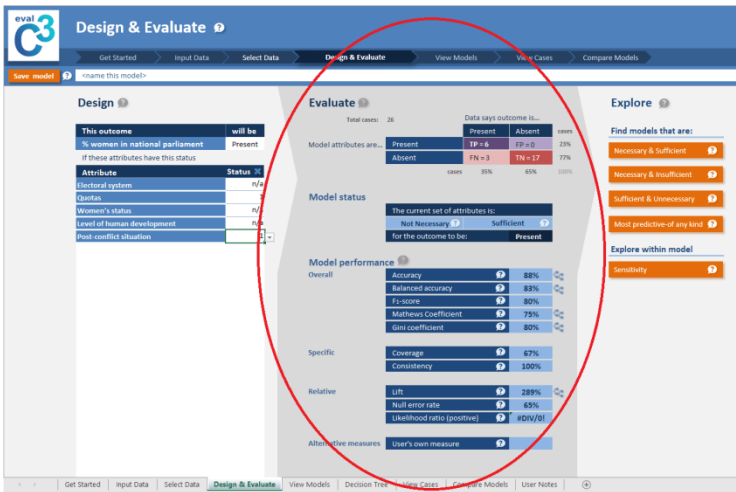
If you want to change these go to the Excel menu bar, click on Data>Analyze>Solver. When the Solver Parameters box pops up click on Options box, then in the next pop-up click on the Evolutionary tab.

To find out more about these settings look here: <https://www.solver.com/excel-solver-change-options-evolutionary-solving-method>

When you have made your choices click on the OK button. Then Solve, to use the algorithm immediately, or Close. Either way your new settings will now be the default settings in EvalC3 until you decide to come back and change them

4.0 Evaluate model

The Evaluate section of the Design and Evaluate worksheet looks like this:



The starting point: The Confusion Matrix

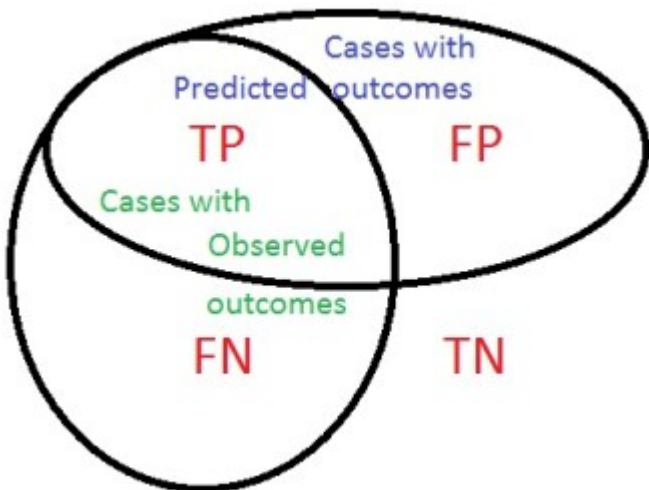
Whenever a predictive model is developed under the Design & Evaluate view, using any of the methods available, the performance of the model is automatically displayed on the right in the form of a 2 x 2 truth table, known as a Confusion Matrix, as shown above

The number displayed in each cell represents the number of cases (e.g projects) which fall into that category.

- In the TP (True Positive) cell are all the cases where the model attributes are present and the expected outcome is also present.

- In the FP (False Positive) cells are all the cases where the model are present but the expected outcome is not present.
- In the FN (False Negative) cells are all the cases where the model attributes are not present but the expected outcome is present.
- The TN (True Negatives) are all the cases where the model attributes are not present and the expected outcome is also not present.

Another way of viewing the results is in the form of two overlapping sets of cases: (a) those with the model attributes (TP&TN) and (b) those with the outcome of interest (TP&FN). Outside of these two sets is the third set of cases, which do not have the model attributes or the expected outcome (TN).



For more background information see

<https://www.dataschool.io/simple-guide-to-confusion-matrix-terminology/>
http://en.wikipedia.org/wiki/Confusion_matrix
https://en.wikipedia.org/wiki/Binary_classification

Model status

Below the Confusion Matrix are some descriptions of the model.

The first of these is a table telling us if the attributes in the model are **Sufficient and/or Necessary** for the outcome. It is easy to identify if an attribute or configuration of attributes is Necessary and or Sufficient for an outcome to be present (or absent) by examining the Confusion Matrix and identifying if any of the following patterns can be seen:

- **Where outcome is present then attributes are ...**
 - Sufficient but not Necessary (Sn) if $FP = 0$
 - Necessary but not Sufficient (Ns) if $FN = 0$
 - Necessary and Sufficient (NS) if $FP = 0$ & $FN = 0$
 - Neither Necessary or Sufficient (ns) if $TP > 0, FP > 0, TN > 0, FN > 0$

- **Where outcome is absent then attributes are ...**
 - Sufficient but not Necessary (Sn) if $FN = 0$
 - Necessary but not Sufficient (Ns) if $FP = 0$
 - Necessary and Sufficient (NS) if $FP = 0$ & $FN = 0$
 - Neither Necessary or Sufficient (ns) if $TP > 0, FP > 0, TN > 0, FN > 0$

For more background information see http://en.wikipedia.org/wiki/Necessity_and_sufficiency

Model performance

Overall measures

The first section lists a number of overall performance measures. These measure, in different ways, the extent to which the model has maximised the number of TPs and TNs and minimised the number of FPs and FNs.

- **Accuracy:** The proportion of all cases which are True Positives and True Negatives. This is the default performance measure to use. However accuracy is not a good measure to use when the Prevalence of the outcome is relatively small or relatively large. In these cases the Accuracy measure gives too much weight to the column with the more prevalent outcome.
- **Balanced accuracy:** This takes into account the prevalence of the outcome and the prevalence of the absence of the outcome – $((TP/(TP+FN))+(TN/(TN+FP)))/2$. This performance measure should be used when the presence of the outcome is either very common or very uncommon.
- **Gini Index:** This measure is used in Decision Tree algorithms as an alternative to Accuracy. It is a measure of inequality in the distribution of cases across all four categories. Perhaps not immediate relevant but data mining packages like Rapid Miner provide this measure alongside Accuracy

The next two measures try to capture good performance in the form of minimised numbers of both FPs and FNs, rather

than just one or the other. In QCA terms they measure the extent to which Coverage and Consistency have been both been improved by a model, rather than just one or the other.

- **F1 score:**
- **Mathews Correlation Coefficient:**

For more information see https://en.wikipedia.org/wiki/Evaluation_of_binary_classifiers

Specific measures

- **Coverage (QCA term) / True positive rate / Sensitivity / Recall :** The proportion of all cases with the outcome present that are correctly identified by the model. More is better!
- **Consistency (QCA term) / Positive Predictive Value / Precision / Consistency** The proportion of True Positives among all the cases where all the attributes of the model are present. More is better.

Sometimes it may be preferred to optimize one of these rather than both (e.g. via F1 score above)

Relative measures

- **Lift:** The ratio of two percentages: the percentage of correct positive classifications made by the model to the percentage of actual positive classifications in the test data
- **Null error rate:** This is how often you would be wrong

if you always predicted the majority class

- **Likelihood ratio+**: The likelihood that the outcome would be expected in a case with the model attributes compared to the likelihood that that outcome would be expected in a case without the model attributes

Interpretation of results

Two points to note:

1. It is possible that more than one model (i.e. configuration of attributes) will produce the same level of performance on one or more of the above measures, including the particular numbers of cases distributed across the four cells of the Confusion Matrix. This is more likely when the numbers of attributes is large relative to the number of cases.
 1. These alternate models can be discovered by trying both exhaustive and evolutionary searches, and by manually tweaking the models produced by both methods.
 2. See the [Reviewing Models](#) page for advice on how to make choices between these models
2. When an existing model is manually tweaked it is possible that performance may only be marginally improved or reduced. This fact highlights that it is not a black and white world out there where things either work or don't work. This is a "feature not a bug" because it suggests that experimentation with project design is not necessarily a high cost "either win or lose" proposition.

Missing data

This is how EvalC3 treats missing data:

1. If a case has no data on the status of the outcome of interest being present (1), then it is treated as an outcome which is absent (0). In this situation the case with missing outcome data will be one of the cases counted as False Positives or True Negatives. The same will be the case if it is the absence of the outcome which is of interest.
2. If a case has no data on the status of an attribute being present (1) which is part of a predictive model, then it is treated as an attribute which is absent (0). That case will be one of the cases counted as False Negative or True Negatives. The same will be the case if it is the absence of the outcome which is of interest.
3. Where a case has no data on either the outcome or attributes that form part of the predictive model then that case will be counted as a True Negative.

The net result is the performance of a prediction model constructed using a data set with missing data is likely to be a conservative one, being the lowest likely.

4.1 Sensitivity and INUS Analysis

When a good prediction model has been found it may consist of multiple project attributes (required to be present and/or absent). A question may then be asked as to how important each of these attributes is, within the model as a whole.

This question can be answered by systematically removing each attribute from the model, one at a time, and on each occasion observing how the overall performance of the model changes. Removal here means changing an attribute value of 1 or 0 to n/a. The removal of attributes which are more important will be associated with a bigger deterioration in the performance of a model. The question then is which attribute removal has been associated with the biggest deterioration in model performance.

PS for nerds: The same method has been used to make models generated by neural networks more transparent. See <https://www.oreilly.com/learning/introduction-to-local-interpretable-model-agnostic-explanations-lime>

An example using the Krook data set

As a result of using an evolutionary search it was found that the presence of “quotas” for women in parliament and the country being in a “post-conflict” situation was *sufficient but not necessary* for high levels of women’s participation in parliament. This model accounted for 6 of the 9 cases where there were high levels of women in parliament.

When the “presence of quotas” was removed from the model, the model performance fell from 83% to 74%, when

using Averaged Accuracy as the performance measure. The number of TPs increased slightly from 6 to 7, and the FPs increased from 0 to 5.

When the presence of a post-conflict situation was removed from the model, the model performance remained at 83%, but the number of FPs increased from 0 to 4

Therefore, it appears that the presence/absence of quotas was the attribute of the model that made the biggest difference to its performance

This type of analysis can be seen as a particular form of contribution analysis.

INUS analysis

The following is a quote from the TORs of an evaluation: *“Even though it is well recognized that multiple factors can affect the livelihoods of individuals or the capacities of institutions, it is important for policymakers as well as stakeholders to know what the added value of the xxxx program is”*

What they are looking for, I suggest, is an INUS condition, an attribute that is Insufficient but Necessary part of a configuration that is Sufficient but Unnecessary for an outcome to occur.

INUS attributes can be identified using EvalC3. The first step is to develop a good predictive model for an outcome. Typically this will consist of a number of attributes. Then by selectively changing the status of each attribute in a configuration and then observing its effects on the model's performance, we can identify the extent to which an attribute is an Insufficient but Necessary part of a

configuration that is Sufficient but Unnecessary.

This can be observed in two forms:

- To a degree: A model that was previously Sufficient but Unnecessary may no longer be so if the removal of an attribute from the model leads to some False Positives. The model may still be a good predictor of the outcome, but its attributes are no longer *sufficient* for the outcome. Both of the example changes to the Krook data model, discussed above, are of this kind.
- Categorically: A model that already had some False Positive cases may now have more of these than True Positives – in which case the model is now in effect a better predictor of the *absence* of outcome.

2018 09 03 Update

It is now possible to do a quick sensitivity analysis using the sensitivity button in the Explore section of the Design and Evaluate view. When you click on this button it will then highlight two attributes of your current model:

- A green highlighted attribute, whose removal makes the biggest difference to the performance of the model
- A mauve highlighted attribute, whose removal make the least difference to the performance of the model

2018 10 12

The categorical definition of an INUS attribute in a prediction model has an interesting connection to the notion of

“actionable recourse” in discussions of fairness of decision making by algorithm. Recourse is the ability to flip the decision of an algorithm by changing an attribute of a case. While people may not be able to change some of their attributes e.g. their age, they may be able to change others e.g. education level.

4.2 The adjacent possible

This phrase was first coined, I think, by Stuart Kauffman, in his book “[Investigations](#)”

The basic idea: Evolution may change speed, but it does not make big jumps. It typically progresses through numerous small moves, exploring adjacent spaces of what else might be possible. Some of those spaces lead to better fitness, some to less. This is low-cost exploration, big mutational jumps involve much more risk that the changes will be dysfunctional or even terminal in the immediate short term.[But for a counter view, read about “[hopeful monsters](#)“]

The same strategy may apply to many development programs, where big changes may require a very different set of human capital, whereas incremental changes would only require small changes in human capital requirements

Incremental searches for small improvements in a predictive model can be made in two ways:

1. **Testing the addition of new attributes, one at a time.** This can be done in two ways:
 1. Manually, by clicking on one attribute at a time in the Design View and noting how it changes the performance of the current model. This could be described as a breadth-first search.
 2. Automatically, by clicking on “Most predictive of any kind” and then choosing “Find one additional attribute that gives the best performance”. This will enlarge the current model by one attribute. Repeating this process

will expand the size of the current model by one attribute at a time. This could be described as a depth-first search

2. **Testing the effects of the removal of existing attributes of the design, one at a time.** This is covered in detailed under [Sensitivity Analysis](#)

Caveat: If you are exploring the fitness landscape around an existing model, then adding an extra attribute can have two effects on performance. Firstly, say if you add an attribute that specifies a particular context, this is likely to reduce the coverage of the model ($=TP/(TP+FN)$). That is to be expected and not necessarily a problem. What matters is that within that more circumscribed context has the consistency of the model ($=TP/(TP+FP)$) increased or decreased? This additional attribute is in effect a scope condition.

For more on the idea of “the adjacent possible” see:

[Spaces of the possible: universal Darwinism and the wall between technological and biological innovation.](#) Andreas Wagner, William Rosen. 2014

Kauffman, Stuart A. [Investigations](#). Oxford University Press,

Johnson, Steven. [Where Good Ideas Come From: The Seven Patterns of Innovation.](#) [Where Good Ideas Come From: The Seven Patterns of Innovation.](#) London: Penguin, 2011.

“The Atlas of Economic Complexity.” MIT Press. Accessed January 12, 2016. <https://mitpress.mit.edu/books/atlas-economic-complexity>.

4.3 Context effects (aka Scope conditions)

Analysing a data set, like the Krook dataset built into EvalC3, we can come up with a predictive model that performs well. In the Krook data set, the combination of “quotas for women in parliament” and “country in a post-conflict situation” is a good predictor of above-average levels of women’s representation in parliament.

But we then might want to identify how this predictive model is affected by other factors which might also present. These could be seen as contextual attributes of the cases (countries in the Krook dataset). They can also be described as “scope conditions“, in that they define the scope within which a particular model performs.

If we manually add another attribute to a model, for example, “the use of proportional voting”, it can have two effects on model performance. Firstly, it could change the coverage (/recall) of the model. This is highly likely because the more highly specified a model is the more likely it only fits a smaller proportion of cases. Secondly, it could change the consistency (/precision) of the model, for better or worse.

When “the use of proportional voting” is added to the two attribute model described above this does reduce the coverage of the model (from 67% to 44%) but it has no effect on the consistency of the model, which remains at 100%. So, in a sense, it is not a scope condition of much interest. If an additional attribute did reduce the consistency of the model it would be more important, because of potential practical

implications for efforts to influence or engineer the presence of model attributes. In the Krook dataset, none of the five attributes functioned as scope limiting attributes. But it is very conceivable that in a larger data set of African countries, some attributes in this set or otherwise would so so. They would be worth identifying.

4.4 "Boring" versus "interesting" models

How to deal with each type

In an analysis of data on 65 projects funded by the Civil Society Challenge Fund data some of the prediction rules confirmed existing expectations, they held no surprise and ran the risk of being quickly dismissed as “boring”.

But there is a useful step that can then be taken with such “boring” cases. That is to examine the False Positives, where the model predicted the outcome to be present, but where the data showed the outcome to be absent. It is these kinds of cases that are important to examine in detail, to find out why, despite the presence of the model conditions, the outcome was not present.

Understanding these cases will help us define the boundaries of our confidence in the prediction model we have taken for granted. It may help prevent us from being excessively confident in the model, if the causes of the False Positives are beyond our control. Or it may help us widen the applicability of the model, if the causes of the False Positives are within our control.

On the other hand, where a prediction rule contradicts existing expectations it is the True Positives that are most in need of investigation in detail, in order to find out if and how the attributes of the model interacted to cause the predicted and observed outcome.

So, it is worth asking clients of an analysis which of the

results they expected versus which were surprises to them. Or, better still, before sharing the results, ask them to predict the results. That may give a more direct answer.

4.5 Finding Positive Deviants

For background reading on the value of finding “positive deviants” see these resources:

- The [Positive Deviance Initiative](#) website
- Wikipedia on [Positive Deviance](#)
- “[The Power of Positive Deviance: How Unlikely Innovators Solve the World’s Toughest Problems](#)“

This brief post outlines how EvalC3 can help find cases which may be usable examples of Positive Deviance.

1. First, develop a predictive model that is good at predicting the *absence* of the outcome of interest. Usually, we are trying to predict its presence. This can be done by using EvalC3’s search algorithms. Or it can be done by testing out combinations of attributes that according to our prior knowledge and theory are conducive to the outcome not occurring – especially attributes of this kind that we think are quite prevalent.
2. Then focus on the False Positives i.e. those cases where the model attributes predicted the absence of the outcome but in practice the outcome was present. These cases qualify, on first glance, as Positive Deviants. They are the cases where it would be well worthwhile doing a within-case investigation in order to find out how they managed to succeed against the odds.
3. Try to minimise, but not totally eliminate, the number

of False Positives. If there are a lot of False Positives all this may tell us is that the current prediction model is not very good, and is lacking some important attributes. If there are very few, perhaps only one, it is more likely this is a genuine Positive Deviance case achieving the outcome despite all the odds being against it doing so

4. Try to minimise the number of False Negatives. This is not essential, but the wider the coverage of the prediction model the more likely the Positive Deviance case will be of wider interest
5. Carry out a within-case investigation of the identified Positive Deviance case, (a) to verify if it has been accurately described and thus correctly classified as False Positive, (b) to identify any causal mechanism at work that can explain its performance.

This approach can be tested out using the [Krook data set](#) , which is built into EvalC3. The absence of quotas for women in parliament is sufficient for low levels of women's participation in parliament. It predicts 13 of the 14 countries with such low levels. The one exception is Lesotho, where there are no quotas but there are high levels of participation of women in parliament. This is an example of a “positive deviance” case that would be worth doing within-case investigations to identify and understand the causal processes at work.

Outliers of different kinds

Positive deviance cases are one kind of outlier. But there different kinds of outliers can be found in the contents of a Confusion Matrix. At one level there are the False Positive and False Negative cases, if they are in a minority compared

to numbers of True Positives and True Negatives respectively. At another more detailed level, within each of the four Confusion Matrix categories, we can find both modal and outlier cases. To find examples of this latter category of outliers go to the View Cases view and click on Calculate Similarity and then look for cases that have the lowest Similarity measure within their Confusion Matrix category. These are worth investigating as part of a case comparison strategy [discussed here](#), as the end point of an EvalC3 analysis.

Postscript 2018 04 18: Here is a paper that has been waiting to be published, and is well worth reading...”[Searching for Success: A Mixed Methods Approach to Identifying and Examining Positive Outliers in Development Outcomes](#)” by Caryn Peiffer and Rosita Armytage, April 2018. Well worth reading, on how and why a combination of quantitative and qualitative analysis is the best way to identify positive outliers (aka positive deviants) and the reasons why some of these might not otherwise see the light of day.

Postscript 2018 09 27: See “[EXPLORING POSITIVE DEVIANCE – NEW FRONTIERS IN COLLABORATIVE CHANGE.](#)” 2010. Said Business School, as mentioned in Duncan Green’s blog posting “[Should Positive Deviance be my next Big Thing?](#)” (27/09/18)

Postscript 2019 04 05: See “Albanna, Basma, and Richard Heeks. 2019. “[Positive Deviance, Big Data, and Development: A Systematic Literature Review.](#)” The Electronic Journal of Information Systems in Developing Countries 85 (1): e12063. <https://doi.org/10.1002/isd2.12063>.”

4.6 Testing models with new data

[Updated 2018 10 07] In the field of predictive analytics (a subset of data mining methods in the more general sense) it is common good practice to test predictive models against new data. I.e. a data set that was not used as the basis for developing the model. These two datasets are typically called “training” and “test” datasets

Why doesn't EvalC3 address this issue?

Some may notice that there is no provision within EvalC3 for this sort of case separation. There is a reason. Testing a model on a test data set is important if you want to generalize and use your model in new settings. This is typically the case with many commercial applications of predictive modeling e.g. being able to find likely loan defaulters among new clients.

But EvalC3 was designed with a different set of users in mind, those engaged, one way or another, with development aid programmes. These often have small rather than big data sets, and external validity may not always be the top priority. Internal validity may be more important i.e. working out what is going on within the existing (and often small) data set

How can you do this, if you want to?

There are simple and complex ways of doing this. With large datasets, they can be split into two sections, one for training and another for testing purposes. Two-thirds of cases in a training data set and one-third of cases in a test data set are commonly used proportions. Cases need to be assigned

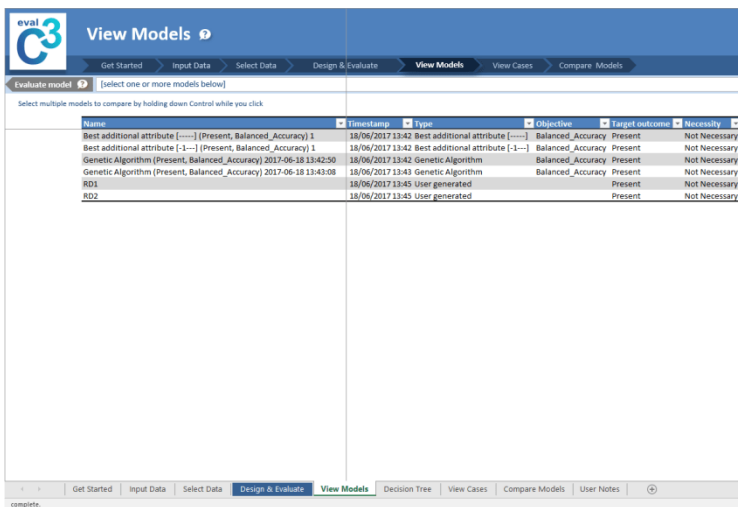
randomly, e.g. by numbering all cases randomly 1,2 or 3, then assigning 1 and 2 into the training set and 3 into the test set.

When datasets are small, other more complex methods can be used. These go by the generic name of [cross-validation](#). Basically, a small part of the data set is withheld as a test set, used, then replaced by another small part, used, then replaced by another, etc. There are many variants of this practice. You need other tools like [Rapid Miner Studio](#) to do this kind of testing. Rapid Miner Studio is free and module based, You don't need to know how to code. It can use the same kind of data set as used in EvalC3.

5.0 Compare models

1. The View Models perspective

When you click on “View Models” this View Models worksheet will appear.



Name	Timestamp	Type	Objective	Target outcome	Necessity
Best additional attribute [----] (Present, Balanced_Accuracy) 1	18/06/2017 13:42	Best additional attribute [----]	Balanced_Accuracy	Present	Not Necessary
Best additional attribute [-1-] (Present, Balanced_Accuracy) 1	18/06/2017 13:42	Best additional attribute [-1-]	Balanced_Accuracy	Present	Not Necessary
Genetic Algorithm (Present, Balanced_Accuracy) 2017-06-18 13:42:50	18/06/2017 13:42	Genetic Algorithm	Balanced_Accuracy	Present	Not Necessary
Genetic Algorithm (Present, Balanced_Accuracy) 2017-06-18 13:43:08	18/06/2017 13:43	Genetic Algorithm	Balanced_Accuracy	Present	Not Necessary
RD1	18/06/2017 13:45	User generated	Present	Present	Not Necessary
RD2	18/06/2017 13:45	User generated	Present	Present	Not Necessary

Each row represents a specific model. The columns describe three types of features for each model: (a) model identifiers – name and date, (b) the performance of the model according to different measures, (b) the attributes of each model (not yet visible in the above screenshot). For more on how to make use of this data, see [Reviewing Models](#)

Any one of these models can be re-loaded by selecting the relevant row, then clicking on Evaluate Model button at the top left of the worksheet. It will be highlighted in orange as soon as any one model is selected.

2. The Compare Models perspective

This is a new feature that is now accessible via the View Models worksheet.

While in View Models, select two or more models which are of interest to you, by holding down Control as you click, as shown below. Then click on the now highlighted orange “Compare Models” button on the right.

eval3 View Models

Get Started | Input Data | Select Data | Design & Evaluate | **View Models** | View Cases | Compare Models

Evaluate model 3 models selected

Select multiple models to compare by holding down Control while you click

Name	Timestamp	Type	Objective	Target outcome	Necessity
Best additional attribute [----] (Present, Balanced_Accuracy) 1	18/06/2017 13:42	Best additional attribute [----]	Balanced_Accuracy	Present	Not Necessary
Best additional attribute [1--] (Present, Balanced_Accuracy) 1	18/06/2017 13:42	Best additional attribute [1--]	Balanced_Accuracy	Present	Not Necessary
Genetic Algorithm (Present, Balanced_Accuracy) 2017-06-18 13:42:50	18/06/2017 13:42	Genetic Algorithm	Balanced_Accuracy	Present	Not Necessary
Genetic Algorithm (Present, Balanced_Accuracy) 2017-06-18 13:43:08	18/06/2017 13:43	Genetic Algorithm	Balanced_Accuracy	Present	Not Necessary
RD1	18/06/2017 13:45	User generated		Present	Not Necessary
RD2	18/06/2017 13:45	User generated		Present	Not Necessary

Get Started | Input Data | Select Data | Design & Evaluate | **View Models** | Decision Tree | View Cases | Compare Models | User Notes

complete

This will take you to the Compare Models worksheet. See the example below.

eval **Compare Models**

Get Started Input Data Select Data Design & Evaluate View Models View Cases **Compare Models**

Index	Cases	Genetic Algorithm (Present, Balanced Accuracy) 2017-06-18 11:41:08	RD1	RD2	# of models
1	Benin	0	0	0	0
2	Botswana	0	0	0	0
3	Burkina Faso	0	0	0	0
4	Burundi	1	0	0	1
5	Congo	0	0	0	0
6	Djibouti	0	0	0	0
7	Ethiopia	1	0	0	1
8	Gabon	0	0	0	0
9	Gambia	0	0	0	0
10	Ghana	0	0	0	0
11	Guinea-Bissau	0	0	0	0
12	Kenya	0	0	0	0
13	Lesotho	0	0	0	0
14	Madagascar	0	0	0	0
15	Malawi	0	0	0	0
16	Mali	0	0	0	0
17	Mozambique	1	0	0	1
18	Namibia	1	1	1	3
19	Niger	0	0	0	0
20	Nigeria	0	0	0	0
21	Senegal	1	0	1	2
22	Sierra Leone	0	0	0	0
23	South Africa	1	1	1	3
24	Tanzania	1	0	1	2
25	Uganda	1	1	1	3
26	Zambia	0	0	0	0
27	Zimbabwe	0	0	0	0
% True Positive coverage:		31%	0%	0%	
			Aggregate coverage:		89%

Get Started Input Data Select Data **Design & Evaluate** View Models Decision Tree View Cases **Compare Models** User Notes

complete

In this worksheet, the selected models are listed in the columns and all the cases in the dataset are listed in rows. Cell values tell which cases is predicted by the column model to have the expected Outcome and found to be a True Positive.

The right-hand column counts the number of models that predict a given case. By using the sort function in Excel the cases most frequently predicted by the different models can be sorted and made available as above

The bottom two rows tell us the number of TPs predicted by each model, and what percentage of all Outcomes are uniquely predicted by that model alone. Ideally, we would find a model that accurately and uniquely predicted many cases with the expected Outcomes.

However, where more than one model predicts a case as a TP this has practical implications. These cases could be worth selecting for within-case analysis to see where there is most

evidence for an underlying causal mechanism at work, supporting one model versus another.

Minimisation

If two prediction models predict the same set of cases, it is worth examining the two models to identify how different they are. If they differ in respect to one attribute only, a QCA type minimisation process may be appropriate. The simpler of the two models could be preferred.

5.1 Reviewing models

Saving models

All models generated by exhaustive or evolutionary searches are automatically saved, with a name that specifies the search criteria that was used. The details of each saved model are listed in the View Models screen. Manually designed models can also be seen here, if they have been saved with a manually designated name.

Sometimes a search result may generate more than one model, because more than one model performs equally well on the selected performance criteria, such as Accuracy. In this situation each saved model with the same level of performance is given a consecutive number at the end of its saved name.

Comparing models

1. Using alternate evaluation criteria

Multiple models can be evaluated using secondary and tertiary evaluation criteria, such as

- Lift – being how well the model predicts the outcome relative to chance. A higher lift value signifies a better performance relative to chance
- Simplicity – being how few attributes are used in the model, relative to the number available in the design menu. Fewer can be better for two reasons (a) Simple models can have wider applicability across cases that exhibit the range of all possible combinations of attributes – the number of cases with the intersection

of A, B, C and D will be smaller than the number of cases with the intersection of A and B, (b) Simpler models can be easier to implement in real life. [But both of these arguments assume some degree of similarity of scale and complexity across A to D]

2. Removing redundancies

There seem to be two ways of proceeding...

1. Finding redundant *models*

In Figure 1 the rows represent models. The columns represent the attributes used in each of those models, where X means the attributes have a 0 or 1 value. Looking at Figure 1, Model 2 is redundant because its combinations of three attributes are covered by Model 1. Likewise, Model 3 is redundant because its combination of three attributes are covered by Model 4. Model 1 is redundant because its combination of four attributes are covered by Model 4. By covered I mean they are a subset of the other.

Model	A	B	C	D	E	F	Outcome
1	1	1	0	1	0	1	1
2	1	0	0	1	0	1	1
3	0	1	0	1	0	1	1
4	1	1	1	1	0	1	1

Figure 1

2. Finding redundant *configurations*

Here we can use something called a “Prime Implicants

Chart”. The Prime Implicant Chart is the second part of the Quine-McCluskey procedure, a central feature of a QCA analyses. According to Schneider and Wagemann (2012:109) “Prime implicants can be defined as the end products of the logical minimisation process through pairwise comparison of conjunctions...Under certain circumstances, though, it happens that one or more of these prime implicants are logically redundant... They can be dropped from the solution term in order to obtain the most parsimonious formula....” A Prime Implicant (PI) is the equivalent of an EvalC3 model.

The process is described below in figures 2, 3 and 4. The PIs are listed by row, and the columns list the different configurations (‘minterms’) that they might apply to. The x’s in the cells indicate which mean term is covered by which PI. In Figure 2 the process starts with identification of a PI that covers one attributes than no other prime implicants do (see row 3). This is an essential rather than redundant PI. Then other attributes in other PIs which are also present in the essential PI are rule out (see vertical red lines)

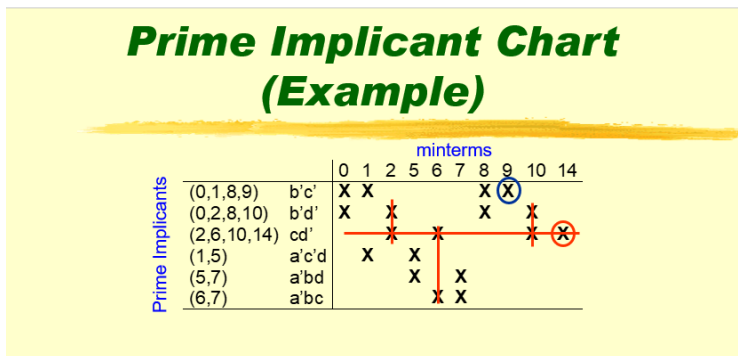


Figure 2

In the next Figures 3 & 4 the same process is repeated

Prime Implicant Chart (Example)

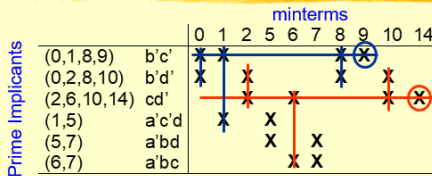


Figure 3

Prime Implicant Chart (Example)

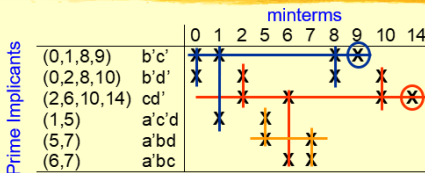


Figure 4

This leaves us with two essential PIs, in row 1 and 3.

This seems to be the approach used by S&W (2012)“.. we introduce a second formula for the minimisation of solution formula: a prime implicant is logically redundant if all the primitive expressions are covered without it being included in the solution formula” (a primitive expression is the same as the row in truth table, also known as a minterm)

S&W also helpfully suggest that removal of redundant models is an option, not a necessity, redundant models “may be of substantive interest”. Parsimony may not be the only

concern. They also suggest that when there can be some redundant models where the ExclusiveOr applies: one can be removed but not both. Meaning there can be more than one parsimonious solution.

5.2 EvalC3 versus QCA results

Sometimes an EvalC3 analysis of a given data set will generate different findings to those generated by a crispset QCA analysis of the same data set. (Example to be placed here shortly). There are , it seems to me, at least three possible reasons:

1. The QCA analysis may have made use of “logical remainders” i.e non-existent cases representing configurations that have not been observed, but where it may be reasonable to make assumptions about whether the outcome would be present or absent in those situations. There are two types of QCA analysis solutions of this type, known as “intermediate” and “most parsimonious”. The results of these analyses may differ from those where machine learning methods have been used, such as those used by EvalC3, because the later do not make use of logical remainders – so the set of cases they use will not be the same.
2. Sometimes when a Truth Table of all the existing configurations is developed as part of a QCA analysis it is found that cases within a particular configuration are inconsistent i.e. some cases of this type have the outcome present, and some do not. One of the possible solutions is to define a “sufficiency threshold”, where if say 80 % of cases with the same configuration have the outcome “present” then the outcome will be deemed to be present for the whole configuration. But in an EvalC3 analysis, the basic unit of analysis is cases, not configurations, so this initial problem of inconsistency is not an issue and each case retains its original outcome status. If a QCA analysed data set is being reanalysed using EvalC3 the original

outcome status will have to be reassigned back to each case in an “inconsistent” configuration. So the two data sets will differ, one will have more cases with the outcome present than the other.

3. Sometimes there will be insufficient diversity in a data set, so an incremental minimisation process (using the QuineMcCluskey algorithm) will not proceed very far, and may end up finding a larger number of “solutions” than will be found by simple machine learning algorithms. In the small imagined data set below there is more than one difference between any pair of the available configurations, so it is not possible to do a “minimisation” at all. But a simple visual scan, or a machine learning algorithm, could identify some simple prediction rules i.e. $A * b = \text{Outcome present}$, $a * b + A * B = \text{outcome absent}$

Cases	Conditions							Outcome
	A	B	C	D	E	F	G	x
1	1	0	1	0	1	0	0	1
2	1	0	1	1	0	1	0	1
3	1	0	0	1	0	1	1	1
4	0	0	0	0	0	1	0	0
5	1	1	0	0	1	0	0	0
6	1	1	0	0	0	1	1	0

5.3 Mapping a fitness landscape?

[Last updated 2018 10 06] Warning: This page is not about any of the core functions of EvalC3. It's simply an exploration of related ideas...

The concept of a “[fitness landscape](#)” has its origins in evolutionary biology and was first proposed by [Sewell Wright](#) in 1932. Here is a recent book on the history of the idea: “[The adaptive landscape in evolutionary biology](#)”. See also Colin Reeve’s book chapter on [Fitness Landscapes](#).

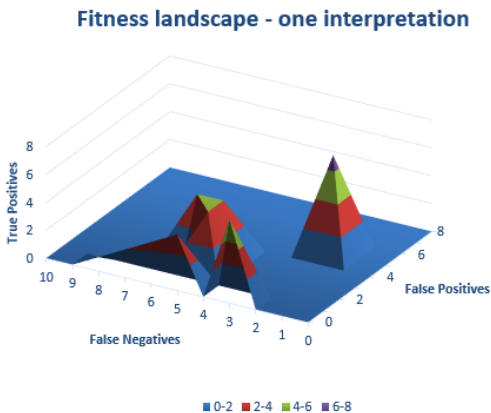
It was proposed as a metaphor, with altitude as the measure of fitness. Fitness, in the form of higher altitudes, might be spread across a landscape in different forms, as isolated peaks, ranges, and plateaus. The four compass directions represent a two-dimensional version of what in reality is a multi-dimensional space of attributes that would fully describe a species’ attributes. A species would be located somewhere on this 2D landscape and would be exploring it through genetic and behavioral variation. A species might reach the top of a hill, but this might only be a “local optimum” i.e not the highest hill in the whole landscape.

It occurred to me that *a version* of this idea could be implemented using data on prediction model fitness, as it is summarised in a Confusion Matrix. In place of species, we have prediction models, each of which has a potential location on a landscape. The N-S and E-W axes of the landscape could represent the number of False Positives and False Negatives and the vertical dimension could represent the number of True Positives. In this landscape, each prediction models will have a distinct location, defined by their number of True Positives, False Positives and False

Negatives.

The shape of the whole surface of the landscape could be identified by locating every possible model for a given set of data i.e. 2^n number of attributes in that data set.

Here is a *very* simple and incomplete example, where performance measures from 9 different models, (among the $2^5 = 32$ possible models in the Krook dataset) were plotted on to a landscape surface diagram generated by Excel. In this landscape, the bottom right corner is where Necessary and Sufficient models would be found because it is there where $FP=0$ & $FN=0$. But no models of this type were found in the Krook dataset I was working with. But there were plenty of Sufficient models, represented by the mountains on the bottom left side, where $FP = 0$ but there some FNs. Higher altitudes represent models with greater numbers of True Positives. PS: I think this visualisation could be improved by using a better 3D graphing app. Also, a contour map might be better



What would this landscape shaped meta-model then tell us?

I had hoped some measure of landscape smoothness/ruggedness would tell us how risky or safe incremental innovation in model design might be, in a given dataset. Some landscapes might be rugged in the sense that a small change in model design would lead to a big change in fitness and vice versa. However, I don't think this type of landscape example let us see this – the distance between two locations in this landscape does not represent the degree of similarity in two models – rather it represents how different the performance of two models is, in terms of numbers of FPs and FNs. To show landscape ruggedness/smoothness some other way needs to be found to represent similarity of model attributes on the horizontal axes.

To be continued....

6.0 Select cases

This is the point in the work flow when the focus changes from across-case analysis to within-case analysis. This is where case selection strategies and tools become relevant. Before doing any within-case investigations choices need to be made about which case(s) to focus on.

EvalC3 now has three sets of tools for comparing cases and to use for case selection.

1. Similarity

This is the first screen that becomes visible after clicking on “View Cases”

The screenshot shows the 'View Cases' interface in EvalC3. The interface includes a navigation bar with steps: Get Started, Input Data, Select Data, Design & Evaluate, View Models, View Cases (selected), and Compare Models. Below the navigation bar are buttons for Compare, Calculate Similarity, and Model: test. The main area contains a table with the following columns: Cases, Status, Similarity, Electoral system, Quotas, Women's status, Level of human development, Post-conflict situation, and Outcome. The table lists 28 cases, each with a country name and a type (FN, FP, TN, TP). The rows are color-coded: blue for FN and FP cases, and red for TN and TP cases. The Outcome column shows values of 0 or 1.

Cases	Status	Similarity	Electoral system	Quotas	Women's status	Level of human development	Post-conflict situation	Outcome
Lesotho	FN	---	0	1	1	1	1	1
Botswana	FP	---	0	1	1	1	0	0
Malawi	FP	---	0	1	1	0	0	0
Mali	FP	---	0	1	0	0	0	0
Niger	FP	---	0	1	0	0	0	0
Ghana	TN	---	1	0	0	1	0	0
Burkina Faso	TN	---	1	0	0	0	1	0
Congo	TN	---	0	0	0	1	1	0
Djibouti	TN	---	0	0	0	1	1	0
Gabon	TN	---	0	0	1	1	0	0
Gambia	TN	---	0	0	0	1	0	0
Ghana	TN	---	0	0	0	1	0	0
Sierra Leone	TN	---	1	0	0	0	1	0
Senegal	TN	---	0	0	0	1	0	0
Madagascar	TN	---	0	0	0	1	0	0
Nigeria	TN	---	0	0	0	1	0	0
Sierra Leone	TN	---	1	0	0	0	1	0
Sierra Leone	TN	---	0	0	0	1	0	0
Burundi	TP	---	1	1	0	0	1	1
Ethiopia	TP	---	0	1	0	0	1	1
Mozambique	TP	---	1	1	0	1	1	1
Namibia	TP	---	1	1	1	1	1	1
Senegal	TP	---	0	1	0	1	0	1
South Africa	TP	---	1	1	1	1	1	1
Tanzania	TP	---	0	1	0	1	0	1
Uganda	TP	---	0	1	1	1	1	1

Here you can see the cases listed row by row. Their attributes are listed column by column, with the outcome column being on the far right (often initially out of sight). In the Status column on the left, all the cases are sorted into

four groups, representing the four categories of cases seen in the Confusion Matrix (True Positive, False Positive, False Negative, True Negative). The values of the attributes which are part of the model that is currently loaded in the Design and Evaluate view can now be seen in red font (see Quotas = 1, in red above)

Now click on “Calculate Similarity”. This will generate the next view.

Cases	Status	Similarity	Electoral system	Quotas	Women's status	Level of human development	Post-conflict situation	Outcome
Lesotho	FN	52%	0	0	1	1	1	1
Botswana	FP	52%	0	1	1	1	0	0
Malawi	FP	46%	0	1	1	0	0	0
Mali	FP	55%	0	1	0	0	0	0
Niger	FP	55%	0	1	0	0	0	0
Benin	TN	55%	1	0	0	1	0	0
Burkina Faso	TN	48%	1	0	0	0	1	0
Congo	TN	62%	0	0	0	1	1	0
Djibouti	TN	62%	0	0	0	1	1	0
Gabon	TN	54%	0	0	1	1	0	0
Gambia	TN	63%	0	0	0	1	0	0
Ghana	TN	65%	0	0	0	1	0	0
Guinea-Bissau	TN	45%	1	0	0	0	1	0
Kenya	TN	65%	0	0	0	1	0	0
Madagascar	TN	63%	0	0	0	1	0	0
Nigeria	TN	63%	0	0	0	1	0	0
Sierra Leone	TN	45%	1	0	0	0	1	0
Zambia	TN	63%	0	0	0	1	0	0
Burundi	TP	46%	1	1	0	0	1	1
Ethiopia	TP	54%	0	1	0	0	1	1
Mozambique	TP	48%	1	1	0	0	1	1
Namibia	TP	43%	1	1	1	1	1	1
Senegal	TP	62%	0	1	0	1	0	1
South Africa	TP	43%	1	1	1	1	1	1
Tanzania	TP	62%	0	1	0	1	0	1
Uganda	TP	51%	0	1	1	1	1	1

In the Similarity column, there are now some percentage figures. Similarity is measured as 1-Hamming Distance. Hamming Distance is the proportion of all values in one row which are different from the values in a row representing another case. In the worksheet shown above, the Similarity measure is the *average* for a case, when compared to all other cases in the dataset.

It is best to focus on one Confusion Matrix category at a time, by using the Excel filter option at the made of the Status column. Start by filtering out all but the True Positive cases.

The similarity measure will then show you how similar each True Positive case is to all other True Positives. The row highlighted in color, across the whole table, is the case with the highest similarity to the others in view. We can call this a Modal case because it is a type of average, it has many attributes in common with other cases in that group. Cases with the lowest similarity measure can be called Outlier cases because they have few attributes in common with the other cases in that group.

2. Compare

Now select a case of interest with a cursor click, then click on the Compare button. For example, the Benin case. The following screen will appear.

The screenshot shows the 'View Cases' interface with a table of cases. The Benin case is highlighted in blue, indicating it is the selected case. Other cases are highlighted in light green or beige, indicating their similarity to the selected case. The table includes columns for MS & MD, Status, Similarity, and various attributes.

Cases	MS & MD	Status	Similarity	Electoral system	Qeios	Women's status	Level of human development	Post-conflict situation	Outcome
Lesotho	40%	FP	52%	0	0	1	1	1	1
Botswana	40%	FP	52%	0	1	1	1	0	0
Malawi	20%	FP	46%	0	1	1	0	0	0
Mali	40%	FP	55%	0	1	0	0	0	0
Niger	40%	FP	55%	0	1	0	0	0	0
Benin	100%	TN	55%	1	0	0	1	0	0
Burkina Faso	60%	TN	46%	1	0	0	0	1	0
Congo	60%	TN	62%	0	0	0	1	1	0
Dominican Republic	60%	TN	62%	0	0	0	1	1	0
Gabon	60%	TN	54%	0	0	1	1	0	0
Gambia	80%	TN	61%	0	0	0	1	0	0
Ghana	80%	TN	61%	0	0	0	1	0	0
Guinea-Bissau	60%	TN	49%	1	0	0	0	1	0
Kenya	80%	TN	61%	1	0	0	1	0	0
Madagascar	80%	TN	61%	0	0	0	1	0	0
Nigeria	80%	TN	61%	0	0	0	1	0	0
Sierra Leone	60%	TN	48%	1	0	0	0	1	0
Zambia	80%	TN	61%	0	0	0	1	0	0
Burundi	40%	TP	46%	1	1	0	0	1	1
Ethiopia	20%	TP	54%	0	1	0	0	1	1
Mozambique	40%	TP	46%	1	1	0	0	1	1
Namibia	40%	TP	41%	1	1	1	1	1	1
Senegal	60%	TP	62%	0	1	0	1	0	1
South Africa	40%	TP	41%	1	1	1	1	1	1
Tanzania	60%	TP	62%	0	1	0	1	0	1
Uganda	20%	TP	51%	0	1	1	1	1	1

To the left, there are now two new columns. The selected case is any case that is of particular interest (e.g. Benin, highlighted in blue). Clicking on Compare generates the percentage values seen in the MS&MD column. The light green highlighted cases are those most similar (MS) to the selected case, the beige highlighted cases are those most

different (MD) from the selected case. Whenever we choose another row as the selected case, the percentages will be recalculated and the highlighted colors re-located to the highest and lowest valued cells. The Compare function gives us a view of how specific cases compare to each other.

3. Case filtering by attribute

We can also carry out more focused comparisons, according to our interest. By opening the drop-down menu on any field we can choose to remove some types of cases from the current view. For example, we may only want to find MS & MD among the cases that do have the outcome present. If we do this, the MS & MD values will automatically be recalculated.

Case selection

The next step is to select cases for subsequent within-case investigations, to identify causal mechanisms that may be at work underlying the associations represented in the predictive model. See the [within-case analysis](#) page for more information on the options here.

Here is a [PDF copy of this page](#)

6.1 Within-case analysis

A very useful book by Mahoney and Goertz ([A Tale of Two Cultures: Qualitative and Quantitative Research in the Social Sciences, 2012](#)) makes a distinction between within-case analysis and cross-case analysis. EvalC3 is designed primarily to facilitate cross-case analysis. But to get the maximum value from this kind of analysis it is important that it is well informed at two different stages by within-case analysis.

When and why

1. **Before a cross-case analysis:** When selecting what attributes to include in a data set and to make use of when analysing that data, either through the use of EvalC3 or other methods such as QCA or using Decision Tree algorithms. Ideally the selection of which attributes to investigate in terms of their possible relationship to which outcomes, would be informed by some prior notion or theory of what might be happening, rather than random choice. The development of those views is likely to be enhanced by familiarity with the details of the cases that are making up the data set.
2. **After a cross-case analysis:** When good prediction rules have been found and modal (i.e. representative) cases have been identified (see [Selecting Cases](#)). Once modal cases have been selected they can be put to use in various ways:
 1. **As illustrative examples** of the results predicted by the model (True Positives), and or incorrect results (False Positives). At the same time, within-case inspection can be used to

verify if the attributes of the case in the data set are a correct description of the actual modal case i.e. do a measurement validity check

2. **As sources of causal explanations.** The examination of individual cases should provide much more detailed information which could shed light on what (if any) causal mechanisms are at work that makes the prediction work.
3. **As sources of contradictory information,** not available within the data set, which could disprove causal explanations that are developed. These could include **confounders**, i.e. a background factor that is a cause of both the attributes in a model and the associated outcome

Steps to take to identify and test likely causal mechanisms

There are four types of cases that can be selected for more in-depth inquiries about any underlying causal mechanisms that may be at work.

1. **Cases which exemplify the True Positive results,** where the model correctly predicted the presence of the outcome. Look within these cases to find any likely causal mechanisms connecting the conditions that make up the configuration. Two sub-types would be useful to compare:
 1. Modal cases, which represented the average characteristics of cases in this group, taking all attributes into account, not just those within the prediction model. Click the Calculate Similarity button in View Cases to find these

cases.

2. Outlier cases, which represent those which were most dissimilar to all other cases in this group, apart from having the same prediction model characteristics. Click the Calculate Similarity button in View Cases to find these cases.

1. I think this is like what others have called a MDSO (most different, same outcome) analysis – *“one has to look for similarities in the characteristics of initiatives that differ the most from each other; firstly the identification of the most differing pair of cases and secondly the identification of similarities between those two cases”* ([De Meur et al, 2006:71](#)).

2. **Cases which exemplify the False Positives**, where the model incorrectly predicted the presence of the outcome. There are at least two possible explanations that could be explored:

1. In the False Positive cases, there are one or more other factors that all the cases have in common, which are *blocking* the model configuration from working i.e. delivering the outcome
2. In the True Positive cases, there are one or more other factors that all the cases have in common, which are *enabling* the model configuration from working i.e. delivering the outcome, but which are absent in the False Positive cases.
3. There is another kind of analysis possible here called MSDO (most similar, different outcome)

– “ to explain why within a set of legislative initiatives, some initiatives result in other decision-making patterns than other initiatives, one has to look for dissimilarities in the characteristics of initiatives that are similar to each other; firstly the identification of the most similar pair of cases and secondly the identification of dissimilarities between those two case” ([De Meur et al, 2006:71](#)).

3. **Cases which exemplify the False Negatives**, where the outcome occurred despite the absence the attributes of the model. There are two types of interest here:
 1. There may be some False Negative cases that have all but one of the attributes found in the prediction model. These cases would be worth examining, in order to understand why the absence of a particular attribute that is part of the predictive model does not prevent the outcome from occurring. There may be some counter-balancing enabling factor at work, enabling the outcome. Such almost-the-same cases can be found using the Compare function in View Cases.
 2. Where a data set has some missing data points (i.e. blank cells) it is possible that some cases have been classed as FNs because they missed specific data on crucial attributes that would have otherwise classed them as TPs. In these circumstances it would be worth investigating the incidence of missing data on each of the attributes of a good performing model, and then scanning FN cases for those which have many of the necessary attributes but where the data on the others are missing.

3. Where multiple models have been developed by using EvalC3 or QCA, it is possible that some cases with the expected outcome are still not covered by any of the models. By default, these will fall into the False Negative category. These case should be subject to particular attention because it is likely that the attributes that predict this outcome are outside the data set. They can only be discovered by doing a within-case investigation of these uncovered cases.
4. **Cases which exemplify the True Negatives**, where the absence the attributes of the model is associated with the absence of the outcome
 1. There may cases here with all but one of the model attributes. These can be found using the Compare function in View Cases, after selecting a modal case in the True Positives group as the comparator. If found then the missing attribute may be viewed as an INUS attribute i.e. an attribute that is Insufficient but Necessary in a configuration that is Unnecessary but Sufficient for the outcome (See [Befani, 2016](#)). It would then be worth investigating how these critical attributes have their effects by doing a detailed within-case analysis of the cases with the critical missing attribute.
 1. Caveat: INUS status cannot be claimed for an attribute if the same configuration with all but one essential model attributes can also be found in the False Negatives group of cases (i.e. where the outcome is present).
 2. (Updated 2020 10 20) Cases may become true negatives for two reasons. The first, and most expected, is that the causes of positive

outcomes are absent. The second, which is worth investigating, is that there are additional and different causes at work which are causing the outcome to be absent. The first of these is described as causal symmetry, the second of these is described as causal asymmetry. Because of the second possibility is worthwhile paying close attention to true negative cases to identify the extent to which symmetrical causes and asymmetrical causes are work. The findings could have significant implications for any intervention that is being designed.

The cases that fit each of the four types can be seen in the “View Cases ” worksheet, and found by using the Calculate Similarity and Compare functions.

Sensitivity

When looking at individual True Positive cases in order to find causal mechanisms at work it may be of value to look at particular attributes in the model. Tweaking of a model, by selectively removing and replacing one attribute at a time, will show which attributes make the biggest difference to the model’s overall performance. It is these attributes which should be of particular interest when looking for the causal mechanism at work within a TP case.

There is now a Sensitivity button on the Design and Evaluate view, under the Explore section. Clicking on this will highlight the attribute in the currently loaded model whose removal makes the biggest difference to the model performance.

Worth reading

Elizabeth A. Stuart (2010) [Matching methods for causal inference: A review and a look forward](#)

Gary Goertz (2017) [Multimethod Research, Causal Mechanisms, and Case Studies: An Integrated Approach](#), Princeton University Press

See a [PDF copy of this page here](#)

6.2 Network analysis of cases

At present one of the hidden worksheets in EvalC3 is a cases x attributes matrix. It is used to calculate the similarity measures in the View Cases worksheet. The same matrix, if imported into a Social Network Analysis (SNA) software package (like Ucinet&Netdraw) can be treated as a “two-mode” network. It is possible to construct two types of network visualisations with this data, showing:

- How cases are connected to cases, where link strength between any two cases reflects the number of the same attributes they share
- How attributes are connected to attributes, where link strength between any two attributes reflects the number of the same cases they connect.

After calculating the average link strength, for either type of network, it is then possible to filter out the below average links, thus highlighting the clusters of cases (or attributes) that are co-occurring at above the average levels.

In the cases x cases network example below there are two distinct clusters. One is the densely connected group on the left, the other is the much less densely connected group on the right. All the members on the right were evaluated as “less successful” than those in the left, on a measure that included self-assessments by the NGOs themselves. It appears that the “less successful” group was more diverse in its attributes than the more successful group.

[Network diagram to come]

7.0 Obtain EvalC3

EvalC3 is free, but before you request a copy using the form below please read these two notes, re copyright and limitation of liability.



1. Copyright:

EvalC3 by [Rick Davies](#) is licensed under a [Creative Commons Attribution-NonCommercial-ShareAlike 4.0 International License](#).

Based on a work at <http://evalc3.net/>.

2. Limitation of Liability. In no event shall Rick Davies be liable to you or any party related to you for any indirect, incidental, consequential, special, exemplary, or punitive damages or lost profits, even if Rick Davies has been advised of the possibility of such damages. In any event, Rick Davies total aggregate liability to you for all damages of every kind and type (regardless of whether based in contract or tort) shall not exceed the purchase price of the product.

3. Privacy: The information you provide below will not be shared with any other party for any purpose. I may use it to make contact with you in the future to inquire about your use of EvalC3 and to share information about new developments of EvalC3. I will not use this information for cross-marketing purposes i.e. to try to interest you in other unrelated products. If at any stage in the future you wish for your data to be removed from my records email me: rick.davies@gmail.com

What happens next: When the completed form is received by me (Rick Davies) an email will be sent to you with a link to

a DropBox folder where the most current version of EvalC3 can always be accessed. EvalC3 is a custom-designed Excel file.

Name(required)

Email(required)

Name of your organisation

What sort of data do you hope to analyse?(required)

Which type of computer are you using?(required)

PC Mac

Which version of Windows are you using?

Which version of Excel are you using?(required)

Email rick.davies@gmail.com for a copy of the Excel version of EvalC3