

模糊集/定性对比分析用户操作手册

原著：查尔斯 C. 拉金 (Charles C. Ragin)

加州大学尔湾分校，美国

cragin@uci.edu

译者：孙 洋

浙江理工大学，中国

sy@zstu.edu.cn

安雯雯

广东工业大学，中国

wenwenan_academic@163.com

协助：翁晨、沈倩慧

QQ 交流群：423061018

目 录

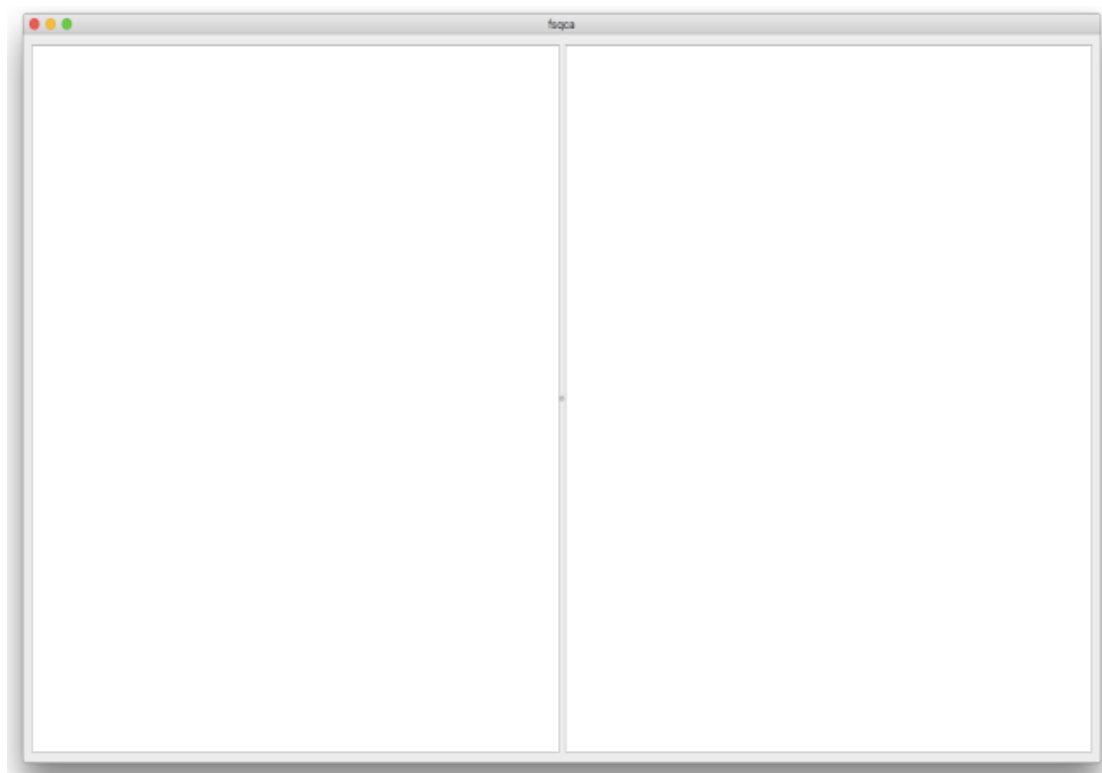
模糊集/定性对比分析用户操作手册.....	I
目录.....	I
1. 数据文件	1
1.1 打开数据文件.....	1
1.2 打开多种格式的数据文件	1
1.3 保存文件选项.....	2
1.4 用其它格式打开 fsQCA 数据	3
2. 数据编辑器	5
2.1 输入数据(在 fsQCA 中从头创建数据文件).....	5
2.2 编辑数据	6
2.3 使用输出	15
3. 基本统计和图表	15
3.1 必要条件	15
3.2 集合并存	17
3.3 子集/超集分析.....	18
3.4 描述性统计	20
3.5 XY 坐标图	21
4. 清晰集 (CRISP-SET) 分析	23
4.1 基本概念	23
4.2 数据.....	32
4.3 分析.....	32
5. 模糊集分析	40
5.1 模糊集的运算.....	40
5.2 模糊集合, 必要性和充分性 (模糊子集关系)	41
5.3 运用模糊真值表算法.....	42
5.4 “指定分析”选项的输出	49

5.5 “标准分析”选项的输出	50
5.6 一致性和覆盖率	51

1. 数据文件

1.1 打开数据文件

FsQCA 需要从以下窗口打开：



★从菜单栏中选择：

文件（File）

打开（Open）

★在打开文件对话框内选择想要打开的文件。

★点击“打开”（Open）。

1.2 打开多种格式的数据文件

数据文件有可能是由多种不同类型的文件，现有的软件能够处理如下格式的数据文件：

-由 Excel 以及其它电子表格程序生成的逗号分隔值 (*.csv) 或逗号分隔文件；

-由 WORD 文档或其它文字编辑软件生成，并且只能存为文本格式的空格分隔文件 (*.txt) 文件；

-由 SPSS (Statistical Product and Service Solutions) 及其它统计软件生成的制表符分隔 (*.dat) 文件;

-Stata 生成的原始数据文件 (*.raw) (如果以逗号分隔格式保存, 则扩展名可以更改为* .csv)。

推荐的格式为* .csv (Excel) 和* .dat (SPSS)。

请注意, fsQCA 对*csv, *.dat 以及*.txt 数据文件的结构具有如下假设: 首先, fsQCA 默认电子表格中第一行的每个单元格代表的是其各个列(数据)的变量名称。其次, fsQCA 默认从表格的第二行开始为数据行, 而每一横行表示一个单独的案例。最后, fsQCA 默认同一列所包含的所有单元格代表的是相同类型的数据(一列表示相同的变量数据)。不同列的数据类型可能不同, 但是必须与所在列保持一致。

在保存变量名称时应使用简单变量名, 仅使用没有嵌入标点或空格的字母数字字符。例如: “GNP1990” 能够使用, 而“GNP 1990” 或“GNP-1990” 等则不能使用。

-打开/保存由 **Excel** 生成的原始数据:

以*.csv 格式保存 Excel 文件。确保 Excel 表格中第一行为变量名称。在 fsQCA 软件中打开。

-打开/保存由 **SPSS** 生成的原始数据:

将 SPSS 文件保存为*.dat 或*.csv 格式。SPSS 会询问“是否将变量名称写入文件 (Write variable name to file)” **请一定选择这一选项。**

-打开/保存由 **Stata** 生成的原始数据:

以*.dta 格式保存 Stata 文件, 然后点击文件 (File), 导出 (Export), 选择文件作为逗号分隔数据。在新的窗口中插入“写入文件 (Write to the file)”的文件名, 然后为“分隔符”选择逗号分隔格式, 并点击提交。在某些 Stata 版本中或许需要将*.dta 文件重命名为*.csv 文件。

-打开/保存 **Word/Notepad** 生成的原始数据:

输入以空格分隔的数据。确保第一行包含变量的名称, 并且用空格分隔。文件以*.txt 格式保存 (纯文本), TXT (Text with Line Breaks), TXT (MS-DOS), 或者 TXT (MS-DOS with Line Breaks)。在 fsQCA 中打开。

1.3 保存文件选项

★在菜单中选择:

文件 (File)

保存 (Save...)

★保存修正的数据文件会覆盖之前版本相同路径下的同名文件。

或者★选择保存一个新文件或将文件保存为其它格式，从菜单中选择：

文件（File）

另存为（Save As...）

★文件将会以*.csv 的格式保存。

★为新数据文件输入文件名。

1.4 用其它格式打开 fsQCA 数据

当你使用 fsQCA 程序打开或对你的数据进行初步分析后，你可以选择在 fsQCA 中编辑你的数据（详见第二章），或在你更熟悉的软件中对数据进行编辑（例如：SPSS 或 Excel）。与此相似，你可以在 fsQCA 中以图表的形式展示你的数据（详见第三章）或使用 SPSS，Stata 或者 Excel 来展现更精细的图表。如果你选择使用 SPSS，Stata 或者 Excel，则需要保存 fsQCA 文件并将其传输到您选择的程序。

- SPSS

★为了能够在 SPSS 中打开 fsQCA 的数据，需要将 fsQCA 数据保存为逗号分隔值 (*.csv) 或逗号分隔文件。要确保 fsQCA 文件中的字符变量之间没有空格（不允许嵌入空格）。

★在 SPSS 中选择：

文件（File）

打开（Open）

数据（Data...）

★打开刚刚在 fsQCA 中保存的文件。

★ SPSS 会根据你的文件询问一些问题，请选择如下选项：

你的文件是否与符合预定格式（Does your test file match a predefined format）？	否（No）
如何整理你的变量（How are your variables arranged?）	分隔 （Delimited）
变量名称是否包含在你文件的顶端？	是（Yes）
变量名称在哪一行？	1
小数符号是什么？	点（Period）
数据的第一个案例从哪行开始	2
你的案例如何表现的？	每行代表一个 案例

你想输入多少案例	全部案例
变量间使用何种分隔符	逗号
合格的文本是?	无
为了以后使用你是否保存这个文件格式	Y/N
是否选用粘贴语法功能	否

然后点击“完成”（Finish）

★现在你可以在 SPSS 中编辑或以图表的形式展示你的数据了。

★为了将 SPSS 文件转换回 fsQCA 文件，请回顾 1.2 SPSS。

- Stata

★为了在 Stata 文件中打开 fsQCA 数据，需要将 fsQCA 数据保存为逗号分隔值 (*.csv) 或逗号分隔文件。要确保 fsQCA 文件中的字符变量之间没有空格（不允许嵌入空格）。

★在 Stata 中选择：

文件（File）

导入（Import）

电子表格生成的文本数据（Text data created by a spreadsheet）

★在新的窗口为了浏览 *.csv 文件和“分隔”，选择逗号分隔数据。

★现在你可以在 Stata 中编辑和使用你的数据。

★为了将 Stata 文件转换回 fsQCA 文件，请回顾 1.2 Stata。

- Excel

★为了在 Excel 文件中打开 fsQCA 数据，需要将 fsQCA 数据保存为逗号分隔值 (*.csv) 文件。要确保 fsQCA 文件中的字符变量之间没有空格（不允许嵌入空格）。

★在 Excel 中选择：

文件（File）

打开（Open...）

★打开你刚在 fsQCA 保存的文件。

★现在你可以在 SPSS 中编辑或以图表的形式展示你的数据了。

★为了将 Excel 文件传输回 fsQCA 文件，请参阅 1.2 Excel。

2. 数据编辑器

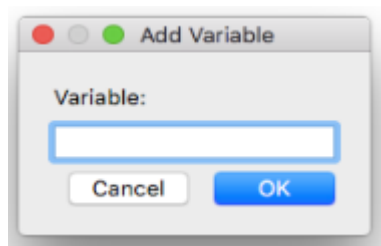
2.1 输入数据(在 fsQCA 中从头创建数据文件)

★在菜单中选择:

 变量 (Variables)

 添加 (Add) ...

★ “添加变量” 窗口将打开。



★输入变量名称。以下规则适用于变量名称:

-名称的长度不能超过 15 个字符。

-每个变量名称必须是唯一的; 不允许重复。

-变量名称不区分大小写。例如名字 NEWVAR, NewVAR 和 newvar 都被认为是相同的。

-变量名称不能包含空格或连字符或标点符号。

-只能使用字母数字字符 (0-9, a-Z)

★点击 “确定” 添加变量。

★除了添加新变量外, 还可以通过突出显示变量和单击来删除变量, 点击

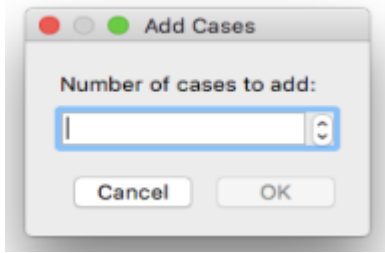
 变量 (Variables)

 删除 (Delete) ...

★现在, 在菜单中选择:

 案例 (Cases)

 添加 (Add) ...



注意：一般来说，fsQCA 能够处理大量的案例。然而，fsQCA 的一个主要特征是它处理前因（注：或译为因果）条件的组合；因此，增加更多变量会比增加更多的案例更多地影响计算时间。可能组合的数量是 2 的 k 次幂，其中 k 是前因条件的数量。根据经验法则，10 个或更少的前因条件（即 1024 个可能的组合）在计算时间方面不是问题。当处理超过 10 个条件时，需要等待程序进行分析的时间更长。大多数应用程序使用三到八个前因条件。

★输入您的数据集的案例数量，点击“确定”，将出现以下数据表窗口：

★**输入数据值。**您可以按任何顺序输入数据。对于选定区域或单个单元格，您可以按案例或变量输入数据。活动单元格以较暗的颜色突出显示。当您选择单元格并输入数据值时，该值将显示在菜单栏下的单元格编辑器中。值可以是数字或字符串。直到您按下 **Enter** 键后才会记录数据值。

★在关闭数据表之前，您需要保存它以避免丢失输入的信息。

2.2 编辑数据

添加/删除

★为了将变量添加到已存在的数据表中，请选择：

变量（Variables）

添加（Add...）

★输入变量名称并点击“确定”。

★为了删除数据表中的现有变量，请突出显示要删除的变量列中的单元格，然后选择：

变量（Variables）

删除（Delete）…

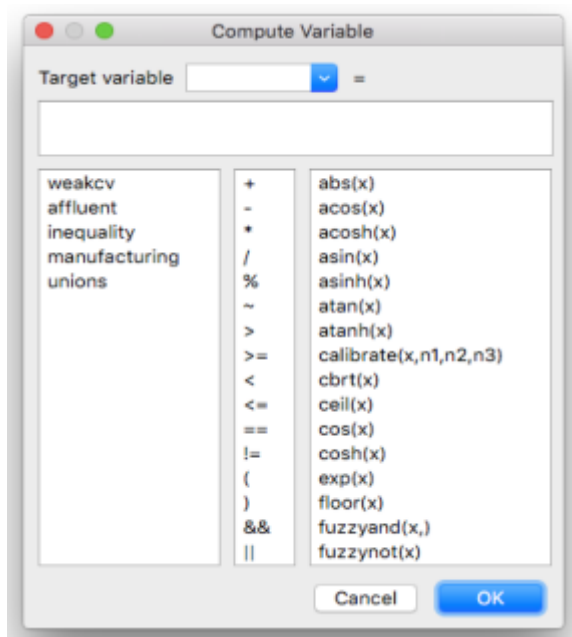
计算变量

★为了从现有的或数字或逻辑表达式中计算新变量，请选择：

变量（Variables）

计算（Compute）…

★以下窗口将打开（左边窗口中列出数据文件中变量的名称）。[本章将使用 Ragin（2005）的弱类投票国家的例子]：



★输入单个目标变量的名称。它可以是要添加到工作数据文件的现有变量或新变量。不要将单个字母用作变量名称（例如“X”）。这将导致计算功能崩溃。请遵循第 8 页的变量名称准则。

★要构建表达式，请将组件粘贴到“表达式”字段或直接在“表达式”字段（新变量字段下方的窗口）中键入。

1) 算术运算符

+	加法。前面的术语被添加到后一个术语中。两个术语都必须是数字。
-	减法。后一个术语从前一项中减去。这两个术语都必须是数字。
*	乘法。前面的和后面的项相乘。这两个术语都必须是数字。

/	除法 。前面的术语除以后一项术语。两个术语都必须是数字，后一项不能是 0。
---	--

2) 关系运算符

<	逻辑小于 。如果前一项小于后一项，则为数字项为真 (= 1)。如果前面的术语比整理序列中的下一个术语早出现（按字母顺序），则为字符串术语。该运算符通常只在逻辑条件下使用。
>	逻辑大于 。如果前一项大于下一项，则为数字项为真 (= 1)。如果前面的术语在整理顺序中显示的时间晚于下列术语（按字母顺序排列），则为字符串术语。该运算符通常只在逻辑条件下使用。
<=	逻辑小于或等于 。如果前一项小于或等于下列项，则为数字项为真 (= 1)。如果前面的术语比整理序列中的下一个术语早出现（按字母顺序排列），或两者相等，则为字符串术语。该运算符通常只在逻辑条件下使用。
>=	逻辑大于或等于 。如果前一项大于或等于下一项，则为数字项为真 (= 1)。如果前面的术语比整理序列中的下一个术语晚出现（按字母顺序排列），则为字符串术语 如果两者相等。该运算符通常只在逻辑条件下使用。
==	逻辑相等 。对于完全相同的条件，为真 (= 1)。如果字符串项的长度不等，则较短的项在比较之前在右侧填充空格。该运算符通常只在逻辑条件下使用。
!=	逻辑不等式 。对于不完全相等的条件，为真 (= 1)。如果字符串项的长度不等，则较短的项在比较之前在右侧填充空格。该运算符通常只在逻辑条件下使用。
&&	逻辑和 。如果前面的和后面的项在逻辑上都为真，则为真 (= 1)。这些术语可能是逻辑或数字；数字项大于 0 将被视为真。该运算符通常只在逻辑条件下使用。
	逻辑或 。如果前面或后面的术语在逻辑上为真，则为真。这些术语可能是逻辑或数字；数字项大于 0 将被视为真。该运算符通常只在逻辑条件下使用。该运算符只能通过将符号粘贴到表达式字段中来工作。
~	逻辑非 。如果以下术语错误，则为真。1 - （数字术语）。该运算符通常只在逻辑条件下使用。

3) 算术函数

abs (x)	返回 x 的绝对值，必须是数字。
acos (x)	返回弧度的反余弦（余弦的反函数），它必须是一个介于 0 和 1 之间的数值（以弧度为单位）。
asin (x)	返回弧度的反正弦（正弦的反函数），弧度必须是介于 0 和 1 之间的数值（以弧度为单位）。
atan (x)	返回弧度的反正切（反正切函数），它必须是一个以弧度为单位的数值。
ceil (x)	返回大于或等于 x 的的最小整数（x 必须是数字）。 例如：ceil (2.5) = 3.0

calibrate	将间隔或比例尺变量转换为模糊集; 详情见下文。
cos (x)	返回弧度的余弦, 它必须是一个以弧度为单位的数值。
cosh (x)	返回弧度的双曲余弦[$(e^x + e^{-x}) / 2$], 它必须是以弧度表示的数值。X 不能超过 230 的值。
exp (x)	返回 e 的幂 x, 其中 e 是自然对数的底数, x 是数字。x (x > 230) 的较大值会产生超出机器容量的结果。
floor (x)	返回舍入 x 向下的整数 (x 必须是数字)。 例如: floor (2.5) = 2.0
fmod (x,y)	返回 x 除以模数 (y) 的余数。两个参数都必须是数字, 模数不能为 0。
fuzzyand (x,...)	返回两个或更多模糊集的最小值。 例如: fuzzyand (1.0,0.1) = 0.1
fuzzyor (x,...)	返回两个或更多模糊集的最大值。 例如: fuzzyor (1.0, 0.1) = 1.0
fuzzynot (x)	返回模糊集的否定 (1-x) (与 Logical Not '¬' 相同)。 例如: 模糊 (0.8) = 0.2
int (x)	返回 x 的整数部分。数字四舍五入到最接近的整数。
log (x)	返回 x 的基 e 对数, 它必须是数字并且大于 0。
log10 (x)	返回 x 的基数为 10 的对数, 它必须是数字并且大于 0。
pow (x,y)	返回上一个术语, 后一个术语的权力。如果前面的项是负数, 则后一项必须是整数。该操作可能产生太大或太小的计算机无法处理的值, 特别是如果以下术语 (指数) 非常大或非常小。
round (x)	返回舍入 x 的整数, 它必须是数字。以 .5 结尾的数字完全是从 0 开始舍去的。 例如: round (2.5) = 3.0
sin (x)	返回弧度的正弦值, 该弧度必须是以弧度为单位的数值。
sinh (x)	返回弧度的双曲正弦[$(e^x - e^{-x}) / 2$], 它必须是一个以弧度为单位的数值。X 不能超过 230 的值。
square (x)	返回 x 的平方, 它必须是数字。
sqrt (x)	返回 x 的正平方根, 它必须是数字而且不是负数。
tan (x)	返回弧度的正切[正弦/余弦], 它必须是一个以弧度为单位的数值。
tanh (x)	返回弧度的双曲正切[$(e^x - e^{-x}) / (e^x + e^{-x})$], 它必须是一个以弧度为单位的数值。

4) 其他运算符

()	分组 。括号内的运算符和函数在运算符和函数之外进行评估。
"	引号 。用于指示字符串变量的值。 示例: 计算如果...: 变量 == "NA"
SYSMIS	系统缺失 。在选择案例的子集时使用。 例如: 如果...选择: 变量 == SYSMISS
Clear	删除 表达式字段中的文本。

重新编码变量

您可以通过重新编码来修改数据值。这对于折叠或组合类别特别有用。您可以重新编码现有变量中的值，也可以根据现有变量的记录值创建新变量。

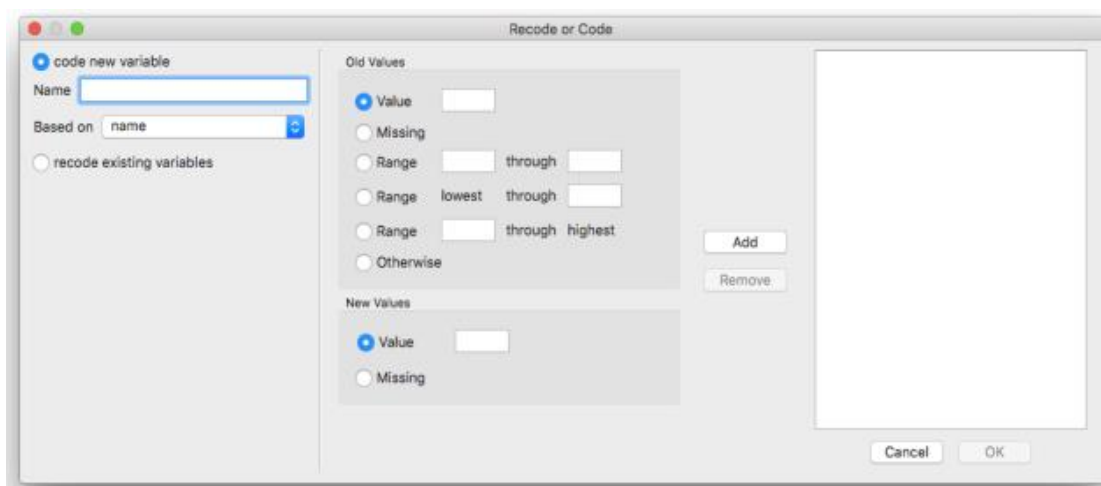
1) **重新编码为相同变量**重新分配现有变量的值或将现有值的范围折叠为新值。您可以重新编码数字和字符串变量。您可以重新编码单个或多个变量 - 它们不必是全部相同的类型。您可以一起重新编码数字和字符串变量。

★为了重新编码变量的值，请选择：

变量 (Variables)

重新编码 (Recode) ...

★以下窗口将打开：



★选择“重新编码现有变量”选项，将打开一个包含现有变量的窗口。

★选择要重新编码的变量（数字或字符串）。

★您可以随意定义要重新编码的案例的子集。

★您可以通过“旧值”和“新值”窗口定义要重新编码的值。

旧值(s)：要重新编码的值。您可以重新编码单个值，值的范围和缺失值。不能为字符串变量选择范围，因为该概念不适用于字符串变量。范围包括其端点以及属于该范围内的任何用户缺失值。

新值：重新编码每个旧值或值范围的单个值。您可以输入值或指定缺失值。

★将您的具体要求添加到右侧的列表中。

2) **重新编码为不同的变量**重新分配现有变量的值或将现有值的范围折叠为新变量的新值。

-您可以重新编码数字和字符串变量。

-您可以将数字变量重新编码为字符串变量，反之亦然。

★为了将旧变量的值重新编码为新变量，请按照上述步骤进行操作，然后选择：

变量（Variables）

重新编码（Recode）...

★将出现以下窗口：



★从下拉菜单中选择“代码新变量”以及要重新编码的现有变量。

★输入一个输出（新）变量名称。

★指定如何重新编码值。

校准模糊集

为了将常规比例和间隔尺度变量转换为模糊集合，有必要对它们进行校准，以使变量与外部标准匹配或一致。大多数社会科学家满足于使用未经校准的测量，这些测量只是显示案例相对于彼此的立场。然而，未校准的测量显然不如校准测量。例如，通过未经校准的民主衡量标准，可以知道一个国家比另一个国家更民主，或比一般国家更民主，但仍不知道它是民主还是专制。

模糊集合使用数据外部的理论和实质标准进行校准，并考虑研究者的概念化，定义和标记。最终产品是成套案例之间程度的细粒度校准，分数从 0.0 到 1.0。

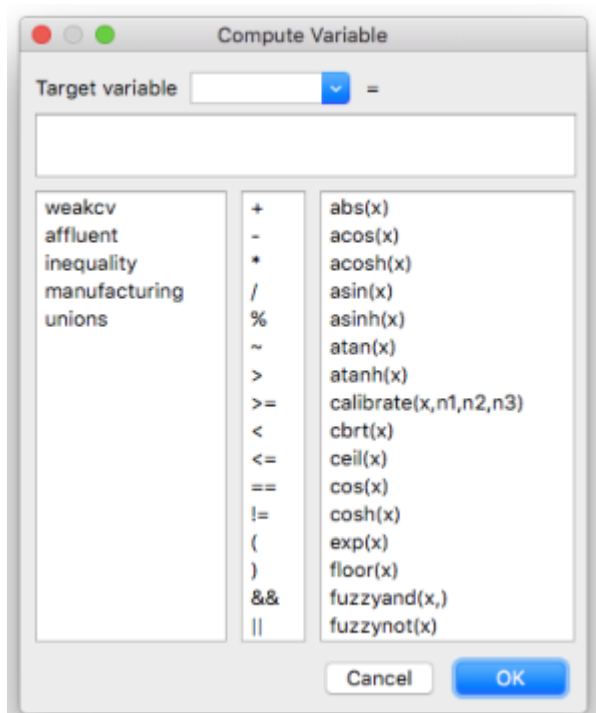
研究者必须指定一个区间尺度变量的值，这些变量对应于构成模糊集的三个定性断点：完全隶属度（模糊评分=0.95）的阈值，完全非隶属的阈值（模糊评分=0.05）和交叉点（模糊分数=0.5）。这三个基准用于使用基于完全隶属资格的对数几率的变换将原始比率或区间尺度值转换为模糊隶属度分数。

★从菜单中选择:

变量 (Variables)

计算 (Compute) ...

★下面的窗口将打开[本章将使用来自 Ragin (2005) 的弱类投票国家的例子]:



★为模糊集命名新变量 (使用 2-8 个标准字母数字字符并且不包含空格、破折号或标点符号)。

★在功能菜单中单击“校准” (x, n1, n2, n3), 然后将其转移到“表达式”窗口。

★编辑表达式校准 (,,), 例如“校准 (manf, 25,10,2)”。这里, manf 是文件中已存在的间隔或比例换算变量的名称, 您可以从左侧的“变量”菜单中进行传输。第一个数字是, 对应于目标集合中完全隶属的阈值 (0.95) 的 oldvar 的值; 第二个数字是对应于目标集合中交叉点 (0.5) 的 oldvar 的值; 第三个数字是对应于目标集合中非成员的阈值 (0.05) 的 oldvar 的值。

★点击“确定”。

★检查数据电子表格以确保模糊分数以预期的方式与原始值相对应。通过单击列标题中的变量名称, 以降序或升序对变量进行排序可能很有用。其结果是精确地确定组内案例的隶属程度, 分数范围从 0 到 1。

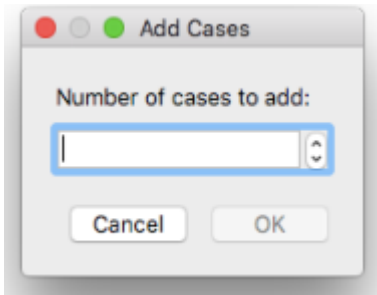
添加/插入案例

★为了将案例添加到已经存在的数据表中，请选择：

案例（Cases）

添加（Add）…

★将出现以下窗口：



★输入要添加到现有案例数量的案例数量。附加案例将出现在数据电子表格的末尾（底部）。

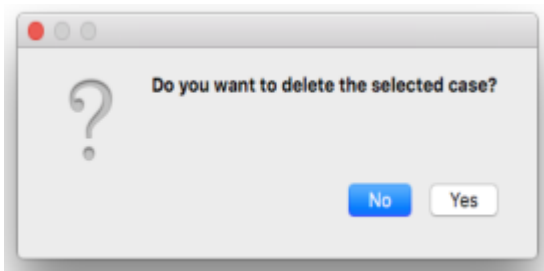
删除案例

★为了从已经存在的数据表中删除单个案例，突出显示想要删除的案例并选择：

案例（Cases）

删除（Delete）…

★将出现以下窗口：



★使用此功能，您一次只能删除一个案例。

★程序会询问您是否要删除数据表中突出显示单元格的案例。

如果……选择该案例如果……选择该案例提供了几种基于包含变量和复杂表达式的条件来选择案例分组的方法，如：

- 变量值和范围

- 算术表达式
- 逻辑表达式
- 功能

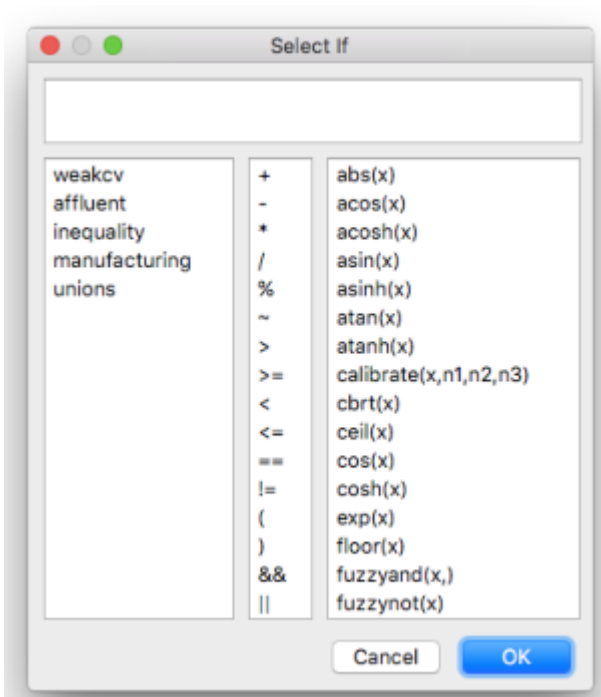
未选择的案例保留在数据文件中，但不包括在分析中。未选择的案例在数据电子表格中以浅色表示。

★为了选择案例的一个子集进行分析，请选择：

案例（Cases）

如果……则选（Select If） ...

★将出现以下窗口：



★指定选择案例的标准。

★如果条件表达式的结果为真，该案例会被选中。如果条件表达式的结果为假或不存在，该案例不会被选中。

★大多数条件表达式使用计算器面板上的六个关系运算符（<，>，<=，>=，==，!=）中的一个或多个。

★条件表达式可以包含变量名称，常量，算术运算符，数字和其他函数，逻辑变量和关系运算符。

注意：“Select If”在单变量时效果最好。例如，如果要使用结合两个逻辑语句的“Select If”函数，例如逻辑 AND 和逻辑 NOT，尝试创建反映您的选择标准的新变量（包含计算或重新编码），然后使用带有“SelectIf”的新变量。

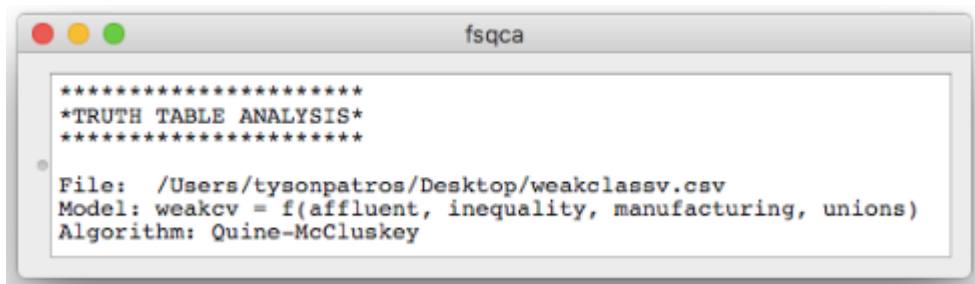
★如果您想要改变选择，请选择：

案例（Cases）

取消选择（Cancel Selection）...

2.3 使用输出

运行过程时，结果显示在 fsQCA 窗口中。您可以通过上下滚动窗口来浏览结果。



```
*****
*TRUTH TABLE ANALYSIS*
*****
File: /Users/tysonpatros/Desktop/weakclassv.csv
Model: weakcv = f(affluent, inequality, manufacturing, unions)
Algorithm: Quine-McCluskey
```

★为了打印输出，请选择：

文件（File）

打印结果（Print Results）...

★您的计算机特定打印机选项窗口将出现，您可以在其中指定打印选项。

★输出是用等宽的 New Courier（10）编写的，以便在程序之间进行简单的传输。因此，如果您在 SPSS 或某些其他程序中打开*.out 文件，表中的数字将轻微错位，除非您指定适当的字体。

★输出也可以复制并粘贴到 Word，写字板，文本或其他文件中。

★为了保存结果，请选择：

文件（File）

保存结果（Save Results）...

★fsQCA 将以*.txt（纯文本）格式保存结果。

3. 基本统计和图表

[本章将使用 Ragin（2005）的弱类投票国家的例子。]

3.1 必要条件

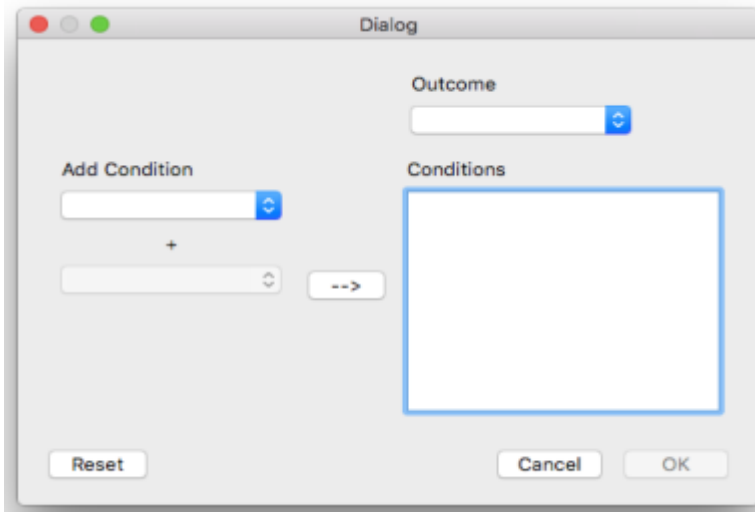
必要条件程序为各个条件和/或指定的可替代条件生成一致性和覆盖率分数。

★为了分析必要条件，请选择：

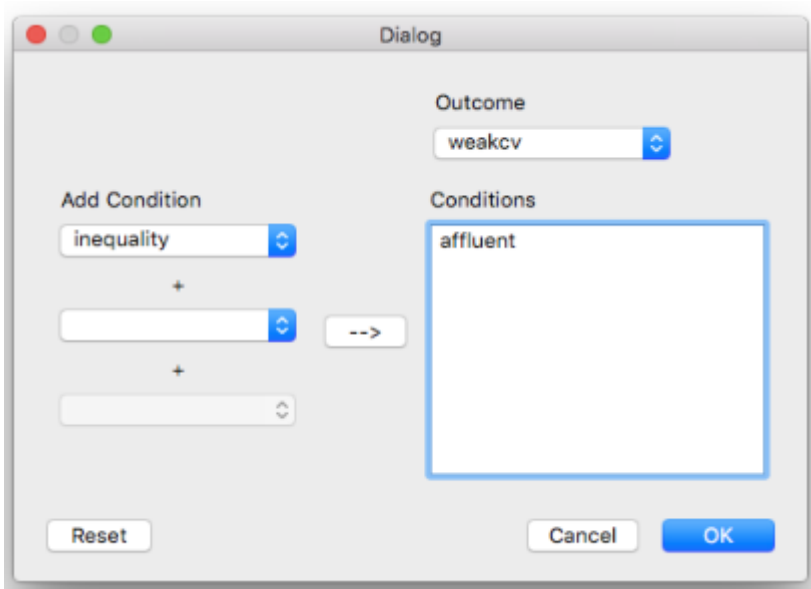
分析（Analyze）

必要条件（Necessary Conditions） ...

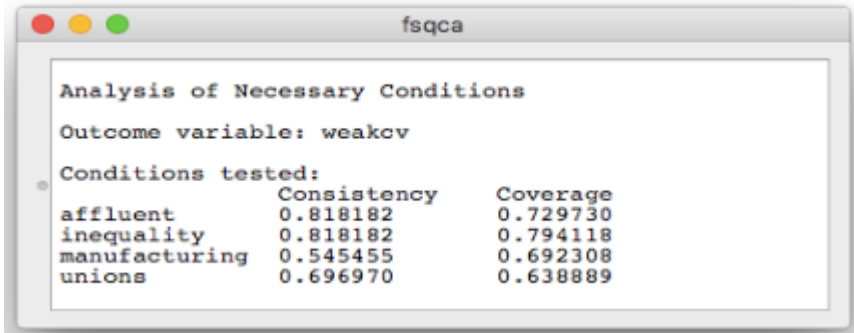
★以下窗口将打开...



★在下拉“结果”菜单中选择结果。然后从下拉“添加条件”菜单中选择一个条件，然后将其传输到对话框右侧的条件框中。您可以使用逻辑或（+）指定可替代的必要条件。



★输入具体要求后，单击“确定”，分析将显示。



在这种情况下，一致性表明前因条件是结果超集的程度；覆盖率表明一致超集的经验相关性。

3.2 集合并存

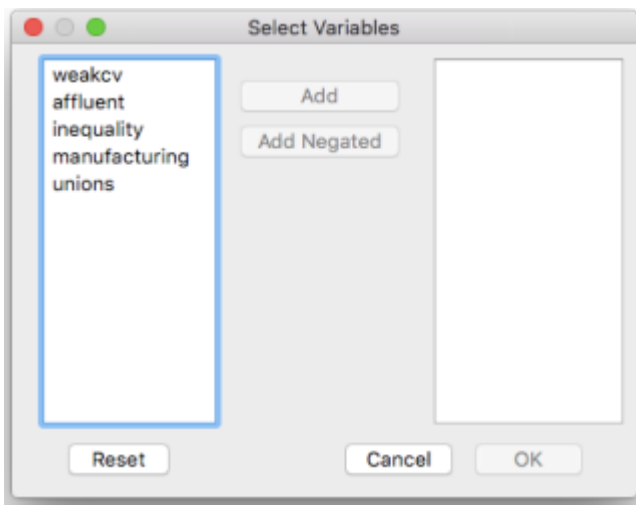
集合并存程序评估两个或多个集合的重叠程度。

★为了分析两组或更多组集合的并存，请选择：

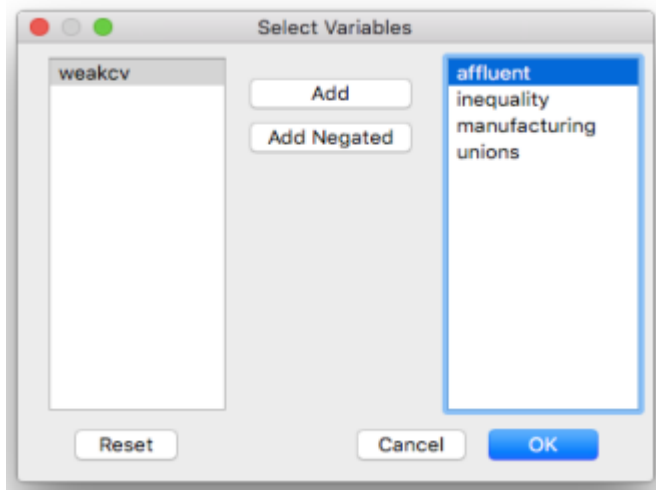
分析（Analyze）

设置并存（Set Coincidence）...

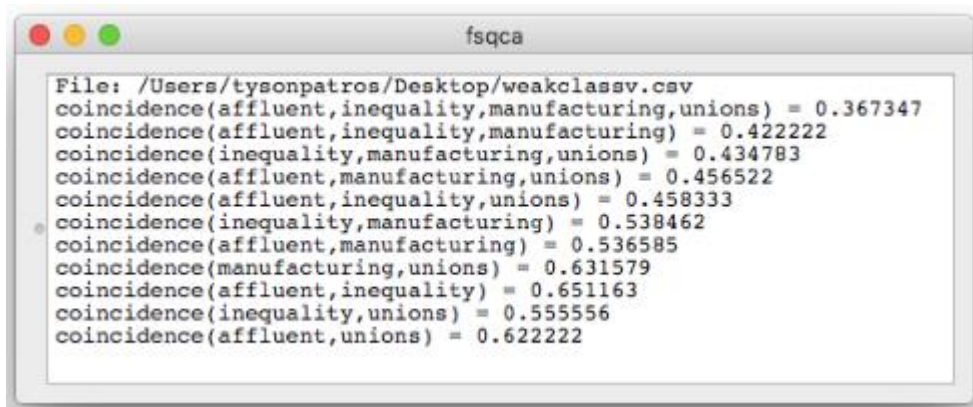
★以下窗口将打开...



★选择你想评估的条件。例如，您可以选择所有非结果条件来评估所有可能组合的重叠程度。



★输入具体要求后，单击“确定”，分析将显示。



3.3 子集/超集分析

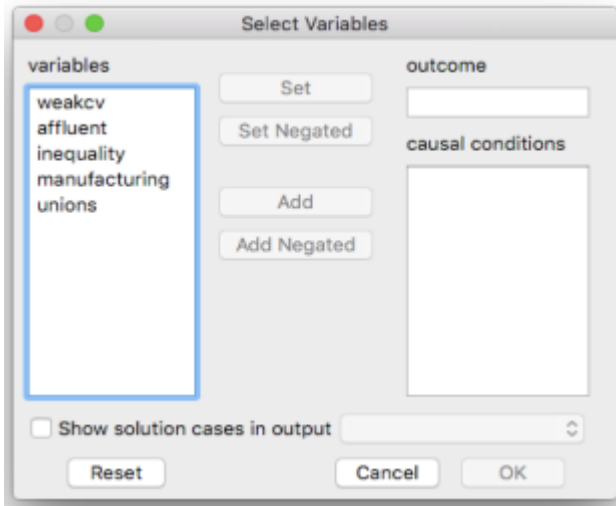
子集/超集分析程序提供了条件和条件构型的一致性和覆盖率的分数，以及组合分数（实验性）。它提供了一种检查假设的前因的充分性的方法，以及检查在给定配方中所有条件子集的方法。

★为了分析一组条件，选择：

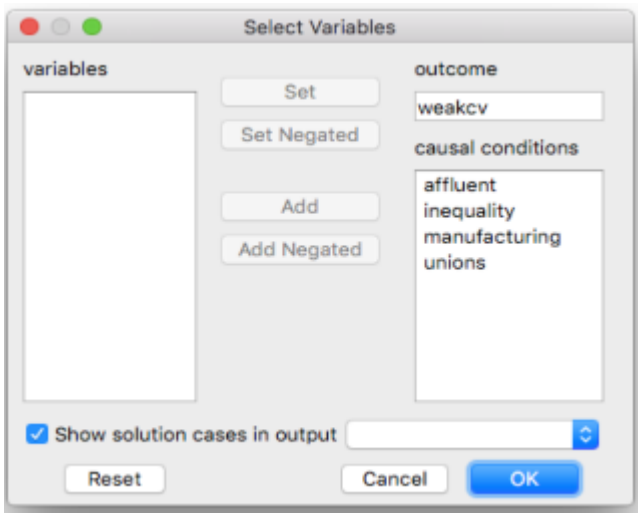
分析（Analyze）

子集/超集分析（Subset/Superset Analysis）…

★以下窗口将打开…



★选择结果变量并单击“设置”。然后选择前因条件并根据您的期望点击添加或添加否定。



★输入具体要求后，单击“确定”，将打开以下窗口：

terms	consistency	coverage	combined
affluent*inequality	0.821429	0.696970	0.765150
inequality*manufacturing	0.809524	0.515152	0.649942
inequality*manufacturing*unions	0.800000	0.484848	0.622799
inequality*unions	0.800000	0.606061	0.696311
inequality	0.794118	0.818182	0.798863
affluent*inequality*manufacturing	0.789474	0.454545	0.591608
affluent*inequality*manufacturing*unions	0.777778	0.424242	0.560303
affluent*inequality*unions	0.772727	0.515152	0.609023
affluent	0.729730	0.818182	0.694786
manufacturing*unions	0.708333	0.515152	0.522523
manufacturing	0.692308	0.545455	0.506324
affluent*manufacturing	0.681818	0.454545	0.436931
affluent*unions	0.678572	0.575758	0.485861
affluent*manufacturing*unions	0.666667	0.424242	0.385337
unions	0.638889	0.696970	0.417424

★运行分析后，您可以选择将结果保存为*.csv 格式的文件，或将结果发送到输出窗口。以下显示了输出窗口中的结果。

```

*****
SUBSET/SUPERSSET ANALYSIS
*****

Outcome: weakcv

              consistency   raw coverage   combined
-----
affluent*inequality      0.821429    0.696970    0.765150
inequality*manufacturing 0.809524    0.515152    0.649942
inequality*manufacturing*unions 0.800000    0.484848    0.622799
inequality*unions       0.800000    0.606061    0.696311
inequality               0.794118    0.818182    0.798863
affluent*inequality*manufacturing 0.789474    0.454545    0.591608
affluent*inequality*manufacturing*unions 0.777778    0.424242    0.560303
affluent*inequality*unions 0.772727    0.515152    0.609023
affluent                 0.729730    0.818182    0.694786
manufacturing*unions     0.708333    0.515152    0.522523
manufacturing            0.692308    0.545455    0.506324
affluent*manufacturing   0.681818    0.454545    0.436931
affluent*unions         0.678572    0.575758    0.485861
affluent*manufacturing*unions 0.666667    0.424242    0.385337
unions                   0.638889    0.696970    0.417424

```

3.4 描述性统计

描述性统计程序显示单个表中指定条件的单变量摘要统计信息。

★为了获得描述性统计信息，选择：

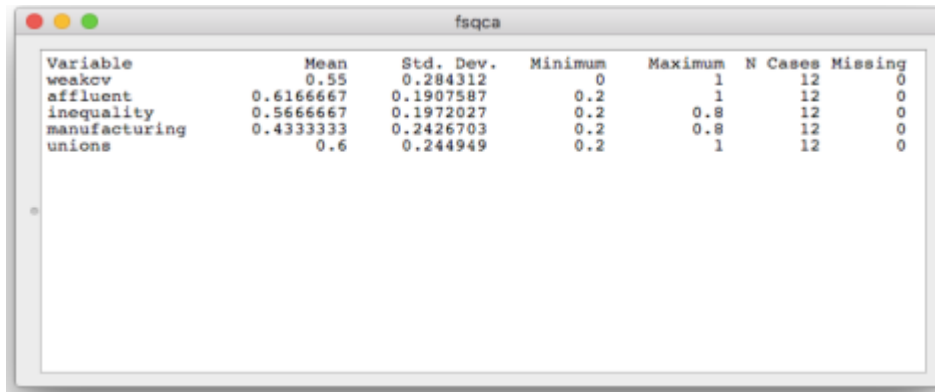
分析（Analyze）

统计（Statistics）

描述（Descriptives）…

★从“变量”列中选择一个或多个条件，并将它们传输到“描述”列。单击“确定”。

★输出窗口将显示您的描述性统计信息：



Variable	Mean	Std. Dev.	Minimum	Maximum	N Cases	Missing
weakcv	0.55	0.284312	0	1	12	0
affluent	0.6166667	0.1907587	0.2	1	12	0
inequality	0.5666667	0.1972027	0.2	0.8	12	0
manufacturing	0.4333333	0.2426703	0.2	0.8	12	0
unions	0.6	0.244949	0.2	1	12	0

★输出的第一行将指出文件名和您选择的过程（描述性统计）。描述性统计表中的列表示以下内容：

- 1.选择的变量（变量）
- 2.平均值（平均值）
- 3.标准偏差（标准偏差）
- 4.变量的最小值（最小值）
- 5.变量的最高值（最大值）
- 6.案例数量（N件）
- 7.缺失的案例数量（缺失）

图表

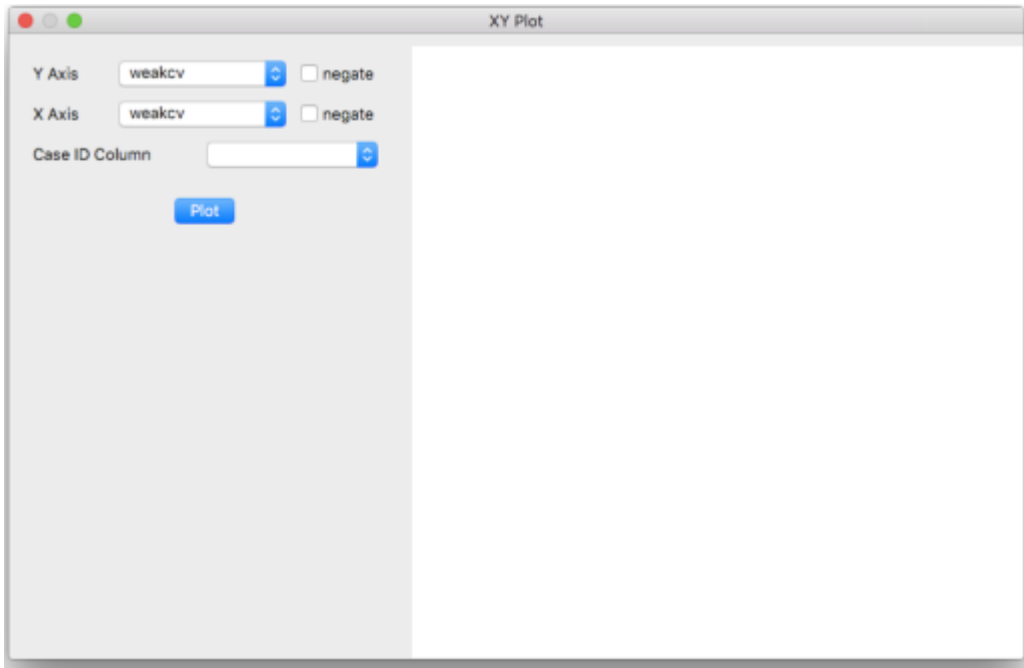
3.5 XY 坐标图

★为了生成 XY 坐标图，请选择：

图表（Graphs）

XY 坐标图（XY Plot.）...

★以下窗口将打开：

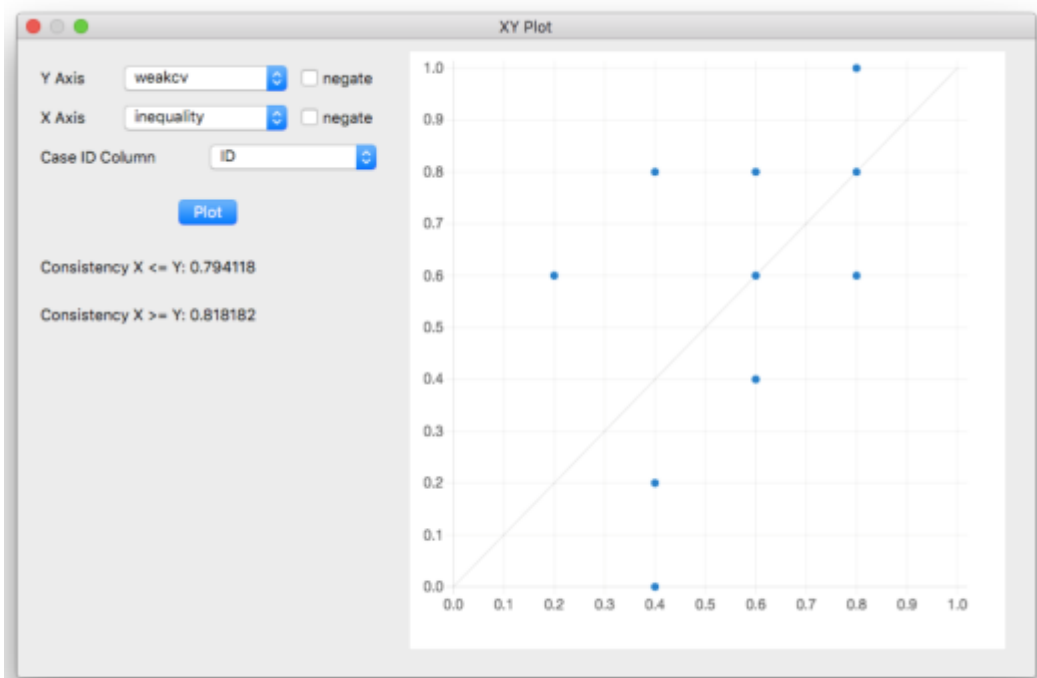


★选择一个变量来定义图表中显示的 X 轴上的值。

★选择一个变量来定义图表中显示的 Y 轴上的值。

★您还可以通过选择案例 ID 变量来添加更多信息。该变量不会在图中表示，但您可以在绘制图之后，通过将光标移动到图中的特定点来确定其值。例如，案例 ID 变量可以是一个字符串变量，其中包含数据集中国家/地区的名称。绘制完成后，可以将光标移动到绘图中的任意点，然后会出现一个窗口，其中包含案例名称以及该点的 x 和 y 值。

★输入具体要求后，点击“绘图”按钮，将显示图：



★“绘图”按钮下面的数字显示了集合论的一致性分数。上面一行显示绘制的的数据与 $X \leq Y$ (X 是 Y 的一个子集) 一致的程度。下面一行显示绘制的的数据与 $X \geq Y$ (Y 是 X 的一个子集) 的一致程度。如果这两个数字中的一个表示高一致性, 另一个可以被解释为覆盖率分数。例如, 如果上面一行中的数字是.91, 下面一行中的数字是.63, 则这些计算表明数据与 X 是 Y 的子集并且其 Y 的覆盖率是 63% 的论点基本一致。也就是说, X 对 Y 有 63% 的隶属程度。

★您可以通过单击变量名称旁边的反取选项来取消图表中的变量。该功能将从 1 中减去该变量的模糊集值。

示例: 不等式= .4; 否定不等式= .6。 [与' \neg '和'fuzzynot (x) '相同]]

★您可以将图形复制为图像并将其粘贴到 Word, 文本或其他文件中。

4. 清晰集 (CRISP-SET) 分析

本手册的这一部分涉及分析反映传统清晰的案例隶属关系的二分法社会数据。关于这种方法的深入讨论可以参见比较方法 (Ragin, 1987), 模糊集社会科学 (Ragin 2000) 第 5 章。这里使用的数据分析策略被称为定性比较分析或 QCA。QCA 基于布尔代数, 其中一个案例在一个集合之内或者在一个集合之外, 并且 QCA 使用二进制编码的数据, 其中 1 表示完全隶属, 0 表示完全不隶属。使用传统的清晰集 (crisp sets) 的 QCA 也被称为 csQCA。

4.1 基本概念

布尔代数存在定性比较的显式代数基础。布尔代数也被称为逻辑的代数和集合的代数, 由乔治布尔在十九世纪中期发展起来。定性比较分析中使用的布尔原则非常简单。布尔代数的七个方面对于算法是必不可少的, 并且在这里以粗略的顺序给出, 更简单的概念遵循更困难的概念。

1) 使用二进制数据

布尔代数有两个条件或状态: 真 (或现在) 和假 (或不存在)。这两种状态以基数 2 表示: 1 表示存在; 0 表示不存在。典型的基于布尔的比较分析解决了获得某个结果的条件的存在/不存在 (即, 是正确的)。因此, 在社会数据的布尔分析中, 所有变量, 前因条件和结果必须是名义规模的量度, 最好是二元的。区间尺度测量转化为多类别名义尺度测量。具有两个以上类别的名义尺度度量用几个二进制变量表示。

2) 布尔否定

在布尔逻辑中, 否定会将隶属程度从 1 变为 0, 从 0 变为 1。例如, 对一组清晰的男性的否定就是非男性的清晰集合。如果案例在男性组中的布尔分数为 1, 那么它在非男性组中的布尔分数为 0。

3) 使用真值表来表示数据

为了使用布尔代数作为定性比较的技术，有必要将原始数据矩阵重构为真值表。真值表背后的想法很简单。一旦数据被重新编码为名义规模变量并以二进制形式表示（如 1 和 0），就必须在偶然条件下将数据分类为不同数值组合。在前因条件上的每个逻辑组合值表示为真值表的一行。构建真值表的这一部分后，根据共享该输入值组合的案例得分（对前因分数的组合条件），为每一行分配一个输出值（结果的得数为 1 或 0）。因此，输入值（前因条件）及其相关输出值（结果）的不同组合被汇总在真值表中。

真值表具有与前因条件上的逻辑上可能的值组合相同的行数。例如，如果存在三个二元前因条件，则真值表将包含 $2^3 = 8$ 行，每个逻辑上可能的三个存在/不存在条件的组合。表 1 中显示了具有三个二元条件和一个二元结果（1 = 存在且 0 = 不存在）的中等大小数据集的真值表。从技术上讲，没有理由将每个组合的频率作为真相表的一部分。这些值包含在示例中以提醒读者，每行不是单个案例，而是具有特定输入值组合的所有案例的摘要。在这方面，真值表的一行就像来自几个分类自变量的多路交叉分类的单元格。

表 1：显示制度失败的三个原因的假设真值表

Condition			Regime Failure	Number of Instances
<i>conflict</i>	<i>death</i>	<i>cia</i>	<i>failure</i>	
0	0	0	0	9
1	0	0	1	2
0	1	0	1	3
0	0	1	1	1
1	1	0	1	2
1	0	1	1	1
0	1	1	1	1
1	1	1	1	3

conflict = Conflict between older and younger military officers
death = Death of a powerful dictator
cia = CIA dissatisfaction with the regime

4) 分组

正如可以计算逻辑上可能的组合数量（ 2^k ）一样，也可以计算逻辑上可能的分组数量。公式为 $3^k - 1$ ，其中 k 又是属性的数量（ $3^3 - 1 = 26$ ）。表 2 显示了表 1 中列出的三种二分法的 26 个逻辑上可能的分组。使用刚刚描述的公式，26 个可能的分组形成如下：8 个涉及三个属性的组合，12 个涉及两个属性的组合，并且六个涉及单个属性。

表 2：使用三种二分法的分组（来自表 1）

Initial Configuration (8 combinations of three aspects)	Groupings involving combinations of two aspects (12)	Groupings evolving a single aspect (6)
<i>conflict • death • cia</i> <i>conflict • death • ~cia</i> <i>conflict • ~death • cia</i> <i>conflict • ~death • ~cia</i> <i>~conflict • death • cia</i> <i>~conflict • ~death • cia</i> <i>conflict • death • ~cia</i> <i>~conflict • ~death • ~cia</i>	<i>conflict • death</i> <i>conflict • ~death</i> <i>~conflict • ~death</i> <i>~conflict • death</i> <i>conflict • cia</i> <i>conflict • ~cia</i> <i>~conflict • cia</i> <i>~conflict • ~cia</i> <i>death • cia</i> <i>death • ~cia</i> <i>~death • cia</i> <i>~death • ~cia</i>	<i>conflict</i> <i>~conflict</i> <i>death</i> <i>~death</i> <i>cia</i> <i>~cia</i>

5) 布尔加法

在布尔代数中，如果 $A + B = Z$ ，且 $A = 1$ 且 $B = 1$ ，则 $Z = 1$ 。换言之， $1 + 1 = 1$ 。布尔加法的基本思想是，如果满足（存在）任何附加项，那么结果为真（发生）。布尔代数中的加法相当于逻辑运算符 OR。（在本讨论中，大写的 OR 用来表示逻辑或）。因此，上述等式 $A + B = Z$ 变成：如果 A 等于 1 或 B 等于 1，那么 Z 等于 1。

考虑这个原理的最好方法是逻辑上的，而不是算术上的。例如，一个人可能会做一些不当的事情而失去他或她的工作。这个人做了多少事并不重要。如果员工做了它们中的任何一个（或全部），他或她将被解雇。做他们两个不会导致一个员工比另一个只做其中一个的员工更容易被解雇。被解雇是一种真正的定性状态。这个例子简洁地说明了布尔加法的本质：满足任何一个加法条件，并遵循预期的结果。

考虑军事政权的崩溃。假设有三种一般情况导致军事政权下降：老年军官和年轻军官之间的尖锐冲突（冲突），强大独裁者的死亡（死亡），或中央情报局不满政权（中情局）。这三个条件中的任何一个都足以促使崩溃。表 1 列出了不同国家的一些此类制度的真值表（其中 1=存在，0=不存在）。原因的每种组合都会产生政权失败或政权失灵的情况 - 没有矛盾的情况。

“简化”布尔方程

失败=冲突+死亡+中央情报局

在消极和积极的情况下，简单和优雅地表达了三个条件和制度失败之间的关系。简单地说：如果这些条件中的任何一个（或两个或全部三个）获得，那么该政权就会倒下。

6) 布尔乘法

布尔乘法与正常乘法大不相同。布尔乘法是相关的，因为布尔代数的典型社会科学应用涉及简化并称为“产品总和”的表达式过程。产品是前因条件的特

定组合。表 1 中关于军事政权崩溃的数据可以用“原始”（即未减少）的产品总和形式表示如下：

$failure = conflict \cdot \sim death \cdot \sim cia +$
 $\sim conflict \cdot death \cdot \sim cia +$
 $\sim conflict \cdot \sim death \cdot cia +$
 $conflict \cdot death \cdot \sim cia +$
 $conflict \cdot \sim death \cdot cia +$
 $\sim conflict \cdot death \cdot cia +$
 $conflict \cdot death \cdot cia$

这七个术语中的每一个都代表至少一个政权失败案例中的前因条件组合。不同的术语是产品，因为它们表示条件的交集（原因与缺失原因的联合）。该等式显示了与军事政权崩溃有关的不同原始条件组合。

布尔乘法，和布尔加法一样，不是算术。表达式冲突 \cdot ~死亡 \cdot ~cia 并不意味着冲突（1）的值乘以死亡（0）的值和 cia（0）的值来产生结果值 0。它只意味着冲突的存在与没有死亡和没有中央情报局相结合。总的来说， $failure = conflict \cdot \sim death \cdot \sim cia$ 在数据中发生两次。布尔乘法的这种结合特征塑造了上面给出的原始乘积和方程的解释：如果获得三种原因的七种组合中的任何一种，则失败（制度失败）。在布尔代数中，加法表示逻辑“或”，乘法表示逻辑“与”。这三个原因以不同的方式进行 AND 运算，以指示不同的经验构型。将这些交叉点进行“或”运算，形成一个未减少的乘积和方程，描述与政权崩溃相关的三个原因的不同组合。

7) 组合逻辑

布尔分析是按设计组合的。在对上文所述的制度失灵的分析中，从真值表的前四行（表 1）中可以看出，如果存在三种原因中的任何一种，那么制度就会崩溃。尽管采取这种捷径很有吸引力，布尔分析所采用的路径更加准确。这是因为缺少原因与布尔分析中存在原因的逻辑状态相同。如上所述，布尔乘法表示存在和不存在条件相交后组合。

考虑真值表的第二行（表 1），其中描述了与前因构型冲突有关的军事政权失败的两个例子：冲突 \cdot ~死亡 \cdot ~cia。简单的检查表明，在这种情况下，失败（政权失败）是由第一个原因冲突造成的。但请注意，如果调查人员只掌握真值表的这一行信息，而不掌握任何其他政权失灵事件的信息，则他或她可能会得出结论，即只有在死亡和中情局不存在的情况下，冲突才会导致失败。这就是冲突 \cdot ~死亡 \cdot ~cia 组合所表明的。这一行本身并不表示冲突是否会导致死亡或中情局的不存在或两者都失败。所有研究人员都知道这两个冲突的实例：冲突 \cdot ~死亡 \cdot ~cia 是因为冲突导致失败，所以可能需要其他条件（死亡和中情局）不存在。从布尔的角度来看，完全合理的是，在存在这些其他条件中的一个或两个（例如，构型冲突 \cdot ~死亡 \cdot cia）的情况下，可能不会导致失败。要回到原

来的名称，可能是因为中情局干预（cia）的存在，初级和高级官员之间的冲突（冲突）将消失，因为两个派别团结一致反对外部人企图指挥事件。

为了进一步推动这个论点，假设研究者只知道真值表的前四行。这些数据将支持这样一种观点，即三种情况中任何一种的存在都会导致失败，但数据也可能表明只有当死亡和中情局不存在时，冲突才会导致失败（冲突 $\cdot\sim$ 死亡 $\cdot\sim$ cia）；只有当冲突和cia不存在时（ \sim 冲突 \cdot 死亡 $\cdot\sim$ cia），死亡才会导致失败，等等。组合逻辑的严格应用要求将这些限制放在从有限的各种情况得出的结论上。

组合逻辑的这一特征与应该从整体上看待案例，尤其是它们的前因相关特征的观点是一致的。布尔方法的整体特征与比较社会科学中定性学者的方向一致，他们从背景中考察不同的原因。当检查真值表的第二行时（表1），它不会被解释为由冲突引起的失败的实例，而是由冲突 $\cdot\sim$ 死亡 $\cdot\sim$ cia导致失败的实例。因此，在基于布尔的定性比较中，原因不是孤立地看待，而是始终在存在和不存在其他前因相关条件的情况下。

最小化

组合逻辑的限制性特征似乎表明布尔方法简单地在复杂性之上复杂化了复杂性。事实并非如此。简化复杂性有简单明了的规则 - 用于减少原始表达式和制定更简洁的布尔语句。这些规则中最基本的是：

如果两个布尔表达式在只有一个前因条件下有所不同，但却产生相同的结果，那么区分这两个表达式的前因条件可以被认为是不相关的，并且可以被删除以创建更简单的组合表达式。

实质上，这个最小化规则允许研究人员取两个布尔表达式，这两个布尔表达式只有一个术语不同，并产生一个组合表达式。例如，冲突 $\cdot\sim$ 死亡 $\cdot\sim$ cia和冲突 \cdot 死亡 $\cdot\sim$ cia，这两者都会导致结果失败，仅在死亡方面有所不同；所有其他元素是相同的。上述最小化规则允许用一个简单的表达式来替换这两个术语：冲突 $\cdot\sim$ cia。换句话说，这两行的比较，冲突 $\cdot\sim$ 死亡 $\cdot\sim$ cia和冲突 \cdot 死亡 $\cdot\sim$ cia，因为整体表明在冲突 $\cdot\sim$ cia的情况下，死亡的价值无关紧要。死亡可能存在或不存在；失败仍然会发生。

这种简单数据缩减的逻辑与实验设计的逻辑相似。只有一种前因条件，即死亡发生变化，并且没有发现结果差异（因为冲突 $\cdot\sim$ 死亡 $\cdot\sim$ cia和冲突 \cdot 死亡 $\cdot\sim$ cia都是失败的例子）。根据实验设计的逻辑，在存在冲突的情况下，死亡与失败无关（即保持这两个条件不变）。因此，布尔最小化过程模仿了实验设计的逻辑。这是理想的社会科学比较逻辑的直接实施。

逻辑最小化的这个过程以自下而上的方式进行，直到布尔表达式不能进一步逐步减少为止。再次考虑上面提出的关于军事政权失败的数据。每个有一个原因存在和两个不存在的行可以与具有两个原因存在的行和一个不存在的行组合，因为所有这些行具有相同的结果（失败）并且每对仅在一个前因条件下不同：

冲突·¬死亡·¬cia 与冲突·死亡·¬cia 相结合产生冲突·¬cia。

冲突·¬死亡·¬cia 与冲突·¬死亡·cia 相结合产生冲突·¬死亡。

¬冲突·死亡·¬cia 与冲突·死亡·¬cia 相结合导致死亡·¬cia。

¬冲突·死亡·¬cia 与¬冲突·死亡·cia 相结合产生冲突·死亡。

¬冲突·¬死亡·cia 与冲突·¬死亡·cia 相结合产生¬死亡·cia。

¬冲突·¬死亡·cia 与¬冲突·死亡·cia 相结合产生冲突·cia。

同样，有两个原因存在和一个不存在的行可以与所有三个存在的行组合：

冲突·死亡·¬cia 与冲突·死亡·cia 相结合产生冲突·死亡。

冲突·¬死亡·cia 与冲突·死亡·cia 相结合产生冲突·cia。

¬冲突·死亡·cia 与冲突·死亡·cia 相结合导致死亡·cia。

进一步减少是可能的。请注意，第一轮产生的减少的条件可以与第二轮产生的减少的条件相结合，以产生更简单的表达式：

冲突·¬死亡与冲突·死亡相结合产生冲突。

冲突·¬cia 与冲突·cia 相结合产生冲突。

冲突·死亡与冲突·死亡相结合导致死亡。

死亡·¬cia 与死亡·cia 相结合产生死亡。

¬冲突·cia 与结合·cia 相冲突产生 cia。

¬死亡·cia 与死亡·cia 相结合产生 cia。

虽然繁琐，但这个简单的最小化过程产生了最终的简化的布尔方程：

$$\text{失败} = \text{冲突} + \text{死亡} + \text{cia}$$

对于整个真值表的简单检查来说，这是显而易见的，但所提出的问题是它的简单性而被选中的。该示例直接说明布尔最小化的关键特征。它是自下而上的。它试图找出更广泛的条件（即，前因条件的更简单的组合），其结果是真实的。而且它的实验就像是关注只有一个原因不同的构型。

1) 使用“质蕴涵”

需要引入的另一个布尔概念是蕴含的概念。如果第二项的隶属程度是第一项隶属程度的子集，则布尔表达式被认为蕴含另一项。例如， a 意味着 $a \cdot \sim b \cdot \sim c$ ，因为 a 蕴含 $a \cdot \sim b \cdot \sim c$ 的所有成分（也就是 $a \cdot \sim b \cdot \sim c$ 是 a 的一个子集）。通过示例可以最好地理解这个概念。如果 a 表示经济上依赖的国家， b 表示重工业的存在， c 表示中央协调经济体， a 包括所有依赖国家，而 $a \cdot \sim b \cdot \sim c$ 包括缺乏中央协调经济体和重工业的所有依赖国家。显然， $a \cdot \sim b \cdot \sim c$ 的隶属程度包含在 a 的隶属程度中。因此， a 意味着 $a \cdot \sim b \cdot \sim c$ 。

蕴含的概念虽然很明显，但却为最小化原始产品总和表达式提供了一个重要工具。考虑表 3 所示的假设真值表，其中总结了三个因素条件的数据，这些条件被认为会影响已经进行的罢工的成功（成功）：罢工者（市场）产生的产品市场蓬勃，同情罢工的威胁的相关行业的工人（威胁），以及大型罢工基金（基金）的存在。

显示未减少（原始）布尔表达式的成功（成功罢工）布尔方程式为

成功 = 市场 \cdot \sim 威胁 \cdot 基金 + \sim 市场 \cdot 威胁 \cdot \sim 基金 + 市场 \cdot 威胁 \cdot \sim 基金 + 市场 \cdot 威胁 \cdot 基金

表 3：假设成功罢工的三个原因的假设真值表

<i>Market</i>	<i>Condition Threat</i>	<i>fund</i>	<i>Success success</i>	<i>Frequency</i>
1	0	1	1	6
0	1	0	1	5
1	1	0	1	2
1	1	1	1	3
1	0	0	0	9
0	0	1	0	6
0	1	1	0	3
0	0	0	0	4

布尔分析这些数据的第一步是尝试尽可能多地组合真值表的兼容行。（请注意，这部分最小化过程使用输出值为 1 的行，罢工成功。）真值表最小化的第一阶段产生下面的部分最小化的布尔方程，它实际上将具有四个三变量项的原始布尔方程变成具有三个双变量项的方程：

市场 \cdot 威胁 \cdot 基金与 市场 \cdot \sim 威胁 \cdot 基金相结合产生 市场 \cdot 基金。

市场 \cdot 威胁 \cdot 基金与 市场 \cdot 威胁 \cdot \sim 基金相结合产生 市场 \cdot 威胁。

市场 \cdot 威胁 \cdot \sim 基金与 \sim 市场 \cdot 威胁 \cdot \sim 基金相结合产生 威胁 \cdot \sim 基金。

成功 = 市场 \cdot 基金 + 市场 \cdot 威胁 + 威胁 \cdot \sim 基金

使用这个简单的最小化规则产生的上述方程中的产品术语 - 如果它们具有相同的输出值，则将仅在一个原因上不同的行组合 - 称为质蕴涵。通常，每个质蕴涵涵盖（即暗含）真值表中的几个原始表达式（行）。例如，在上面给出的部分最小化方程中，质蕴涵市场•基金涵盖真值表中列出的两个原始布尔表达式：市场•威胁•基金和市场•¬威胁•基金。

这种部分减少的布尔表达式说明了布尔分析中的一个常见发现：通常存在比覆盖所有原始基本表达式所需的更多简化表达式（质蕴涵）。最重要的市场•威胁意味着原始市场•威胁•资金和市场•威胁•¬资金 - 例如，这两个原始条件也分别由市场•资金和威胁•¬资金覆盖。因此，从纯逻辑的角度来看，市场•威胁可能是多余的；它可能不是一个重要的质蕴涵。为了确定哪些素数含义在逻辑上必不可少，使用了一个称为质蕴涵图的最小化装置。最小含义图是布尔最小化的第二阶段。

简而言之，最小化过程的第二阶段的目标是尽可能多地使用逻辑上最小数量的质蕴涵来“覆盖”尽可能多的基本布尔表达式。这个目标来源于对非冗余的直接需求。质蕴涵图表映射了质蕴涵和原始表达之间的联系。表 4 列出了描述罢工结果数据中这些关联的质蕴涵图。简单的检查表明，涵盖所有原始表达式所需的最小含义量为 2。（对于非常复杂的质蕴涵图，需要复杂的计算机算法；参见 Mendelson 1970, Roth 1975 和 McDermott 1985。）质蕴涵市场•基金和威胁•¬基金涵盖了所有四种基本布尔表达式。因此，质蕴涵图的分析导致最终减少的布尔表达式仅包含逻辑上必要的质蕴涵：

$$\text{成功} = \text{市场} \cdot \text{资金} + \text{威胁} \cdot \neg \text{资金}$$

这一等式简单地说明，当工人和一个大型罢工基金（市场•基金）生产的产品蓬勃发展，或当相关产业的工人受到同情罢工威胁时，成功罢工就会发生，而罢工基金则与低罢工基金（威胁•¬基金）相结合。（也许只有当罢工工人急需其他工人的支持时，同情罢工的威胁才会受到重视。）

表 4：通过质蕴涵显示原始术语覆盖率的质蕴涵图表（假设罢工数据）

		Primitive Expressions			
		<i>market • threat • fund</i>	<i>market • ¬threat • fund</i>	<i>market • threat • ¬fund</i>	<i>¬market • threat • ¬fund</i>
Prime Implicants	<i>market • fund</i>	X	X		
	<i>market • threat</i>	X		X	
	<i>threat • ¬fund</i>			X	X

这些简单的程序允许研究者推导出描述与结果相关联的不同条件组合的逻辑最小方程。最后的简化公式显示了导致成功罢工的两个（逻辑上最小的）条件组合，因此提供了多个联合因果关系的明确陈述。

2) 使用德摩根定律

德摩根定律的应用很简单。考虑以上提出的对成功罢工的假设性分析的解决方案：成功=市场·资金+威胁·¬资金。编码存在于简化公式中的元素（比如市场中的市场·基金）被重新编码为不存在，编码不存在的元素（例如，在资金·¬威胁期限内的¬资金）被重新编码存在。接下来，将逻辑 AND 重新编码为逻辑 OR，并将逻辑 OR 重新编码为逻辑 AND。应用这两个规则，

$$\text{成功} = \text{市场} \cdot \text{资金} + \text{威胁} \cdot \neg \text{资金}$$

变为：

$$\begin{aligned} \neg \text{成功} &= (\neg \text{市场} + \neg \text{基金}) \cdot (\neg \text{威胁} + \text{基金}) \\ &= \neg \text{市场} \cdot \neg \text{威胁} + \neg \text{市场} \cdot \text{基金} + \neg \text{威胁} \cdot \neg \text{基金} \end{aligned}$$

根据这个等式，当（1）相关产品的市场没有蓬勃发展并且没有严重的同情打击威胁时，罢工失败；（2）产品的市场没有蓬勃发展，并且有大量的罢工基金，或者（3）没有同情罢工的威胁，只有一个小的罢工基金。（结合¬市场·非基金市场和大型罢工基金似乎相互矛盾 - 可能意味着经济在经过一段时间的稳定后会出现经济衰退，在这种情况下，管理层可能会倾向于退出市场。）

德摩根定律产生了给定逻辑方程的确切否定。如果真值表中存在“余数”组合，并且它们被用作“不关心”，那么应用德摩根法的结果将产生一个合乎逻辑的陈述，这与分析不存在结果不同。同样，如果在最初的分析中剩余部分被定义为“假”，那么将德摩根定律应用于（正面案例的）解决方案，将产生一个不仅包含否定案例而且包含剩余案例的逻辑陈述。

3) 必要和充分的原因

如果结果一定是由某一个原因导致的，则将原因定义为必要的。如果一个原因本身能够产生某种结果，则该原因被定义为充分的。这种区别只有在理论视角下才有意义。例如，不需要任何原因，它独立于将其指定为相关原因的理论。无论是必要性还是充分性都不存在于提出原因的理论之外。

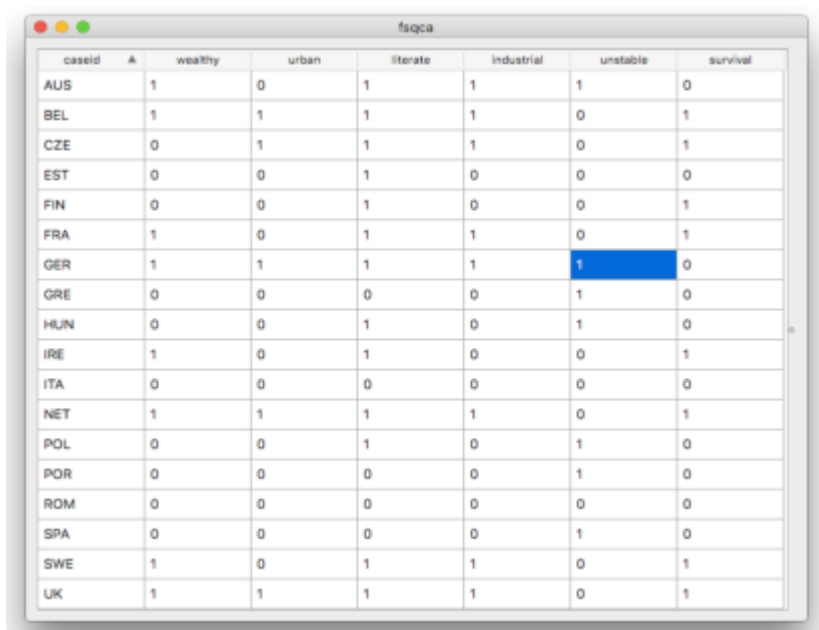
通常一起考虑必要性和充分性，因为这两者的组合都是有意义的。如果原因是产生结果的唯一原因并且是单一的（即，不是原因的组合作），则原因既是必要的也是充分的。如果一个原因能够产生结果但不是其唯一原因，那么原因就是充分但不必要的。如果原因能够与其他原因一起产生结果并且出现在所有这样的组合中，则原因是必要但不充分的。最后，如果一个原因只出现在产生结果的条件组合的一个子集中，那么原因既不必要也不充分。总而言之，有四类原因（由存在/不存在充分性与存在/不存在必要性的交叉表格构成）。

QCA（清晰或模糊）的典型应用会产生一个逻辑陈述，描述了足以产生结果的条件组合。列出的组合可能是详尽的也可能不是详尽的，也就是说，它们可能无法解释结果的所有实例。在分析足够的条件组合之前，最好检查个别条件的必要性和充分性。这可以通过查看每个条件的结果的散点图并且记下哪些

是准超集（即，必要的）以及哪些是准子集（即，充分的）（也参见子集/超集分析）来完成。

4.2 数据

以下窗口显示了一个清晰集的数据表：



caseid	wealthy	urban	literate	industrial	unstable	survival
AUS	1	0	1	1	1	0
BEL	1	1	1	1	0	1
CZE	0	1	1	1	0	1
EST	0	0	1	0	0	0
FIN	0	0	1	0	0	1
FRA	1	0	1	1	0	1
GER	1	1	1	1	1	0
GRE	0	0	0	0	1	0
HUN	0	0	1	0	1	0
IRE	1	0	1	0	0	1
ITA	0	0	0	0	0	0
NET	1	1	1	1	0	1
POL	0	0	1	0	1	0
POR	0	0	0	0	1	0
ROM	0	0	0	0	0	0
SPA	0	0	0	0	1	0
SWE	1	0	1	1	0	1
UK	1	1	1	1	0	1

所有：缩写的国家名称

富裕：高 GDP/上限与低 GDP/上限

城市：发达城市与不发达城市

识字：高水平的识字与非识字

工业：高工业工人比例与低工业工人比例

不稳定：政府不稳定与不稳定

幸存下来：民主在两次世界大战期间幸免于难

[本节中的例子来自 Rihoux 和 Ragin（2008），组态对比分析。]

4.3 分析

当前版本的 fsQCA 软件（截至撰写本文时，版本 3.0，2017 年 7 月）包含一种进行清晰分析的方法：“真值表算法”。该方法使用 Quine-McCluskey 算法。真值表算法如下所述。

真值表算法

两个重要的任务构成了清晰真值表算法的应用：（1）在不同逻辑上可能的前因条件组合下评估案例分布。（2）评估每个前因组合的证据的一致性，以及具有这种组合条件的案例构成具有结果的案例的子集。也就是说，他们分享了有关的结果。

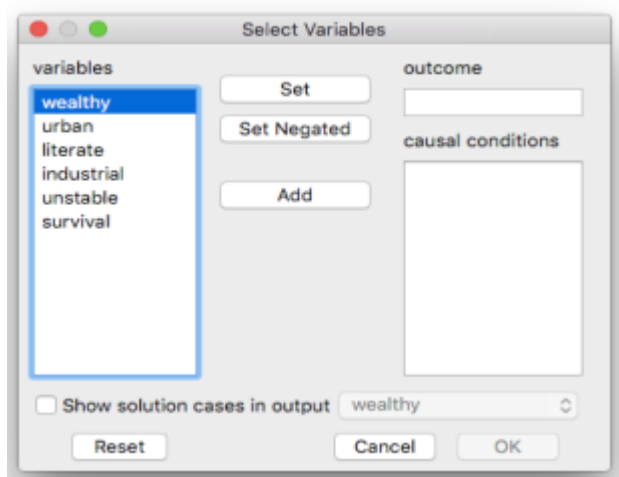
真值表算法涉及两步分析过程。第一步包括从原始数据创建真值表电子表格，主要包括指定分析中包含的结果和前因条件。第二步包括通过选择频率阈值和一致性阈值来准备用于分析的真值表电子表格。

★为了创建真值表电子表格，请选择：

分析（Analyze）

真值表算法（Truth Table Algorithm）...

以下窗口将打开，列出文件中的变量：



★确定并突出显示要解释的案例方面，并通过单击“设置”将其转移到“结果”字段中。

★选择一个前因条件的初步列表，方法是一次突出显示一个前因条件，然后单击“添加”将它们逐个移至前因条件字段。

★选中“在输出中显示解决方案案例”旁边的复选框，然后选择您的 caseID 变量。

★点击“确定”按钮，将出现包含完整真值表的以下窗口：

wealthy	urban	iterate	industrial	unstable	number	survival	cases	raw consist	PRI consist	SYM consist
1	1	1	1	0	3 (16%)		cases	1	1	1
0	0	0	0	1	3 (33%)		cases	0	0	0
0	0	0	0	0	2 (44%)		cases	0	0	0
0	0	1	0	0	2 (55%)		cases	0.5	0.5	0.5
1	0	1	1	0	2 (66%)		cases	1	1	1
0	0	1	0	1	2 (77%)		cases	0	0	0
1	0	1	0	0	1 (83%)		cases	1	1	1
0	1	1	1	0	1 (88%)		cases	1	1	1
1	0	1	1	1	1 (94%)		cases	0	0	0
1	1	1	1	1	1 (100%)		cases	0	0	0
1	0	0	0	0	0 (100%)		cases			
0	1	0	0	0	0 (100%)		cases			
1	1	0	0	0	0 (100%)		cases			
0	1	1	0	0	0 (100%)		cases			
1	1	1	0	0	0 (100%)		cases			
0	0	0	1	0	0 (100%)		cases			
1	0	0	1	0	0 (100%)		cases			

★真值表将有 2^k 行（其中k表示前因条件的数量），反映前因条件的所有可能组合（向下滚动以查看所有可能的组合）。1和0分别代表完全隶属程度和每个条件的完全非隶属程度。对于每一行，将创建以下每个变量的值：

数量：显示条件组合的案例数量

原始组成：每个真值表行中显示结果的案例比例。

PRI组成：基于准确比例减少误差计算的一致性替代度量（针对模糊集开发）。在清晰集合分析中，这将等于原始构成。

SYM组成：基于PRI一致性对称版本的模糊集合的一致性替代度量。

★请注意，标记为结果的列（在此示例中存活）为空白。由研究者根据以下程序确定每种构型的结果。

★研究人员必须首先制定一个规则，根据它们的频率将某些组合（行）分类为相关组合，而其他组合则不相关。这是通过根据数字列中显示的每行中的案例数量来选择频率阈值来完成的。当分析中的案例总数相对较少时，频率阈值应该为1或2。但是，当总N很大时，应该使用更实际的阈值。通过前因组合检查案例分布非常重要。。

★可以通过单击数字列的标题来按构型（行）的频率对其进行排序（降序或升序）。

★对行进行排序并选择频率阈值后，删除所有不符合阈值的行。如果案例按照编号按降序排列，请单击低于阈值的第一个案例，然后选择

编辑（Edit）

删除当前行到最后一行（Delete current row to last row）...

如果案例尚未排序，那么可以通过选择所选的行单独删除那些不符合阈值的案例（If cases have not been sorted then those cases that do not meet the threshold can be deleted individually by selecting the row the choosing）

编辑（Edit）

删除当前行（Delete current row）...

★下一步是区分结果子集和非结果子集。对于清晰集，使用原始编组列中报告的集合理论一致性度量来进行此确定。低于 0.75 的值表明实质上不一致。按降序对一致性评分进行排序以评估其分布（这应在**删除不符合频率阈值的行后执行**），这一点很有用。通过单击原始编组来完成排序。列标签。

★找出可能对建立一致性阈值有用的较高一致性范围中的任何差距。请记住，始终可以检查几个不同的阈值并评估降低和提高一致性截止点的后果。

★现在有必要指出哪些构型可以被视为结果的子集，哪些不能（另请参阅下面的替代方法）。对于一致性级别达到和/或超过阈值的每个构型，在结果列中输入 1（在此示例中保留）。在一致性级别不满足一致性阈值的每个构型的结果列中输入 0。

★或者，可以使用“删除和编码”功能来自动执行此过程。选择：

编辑（Edit）

删除并编码（Delete and code）...

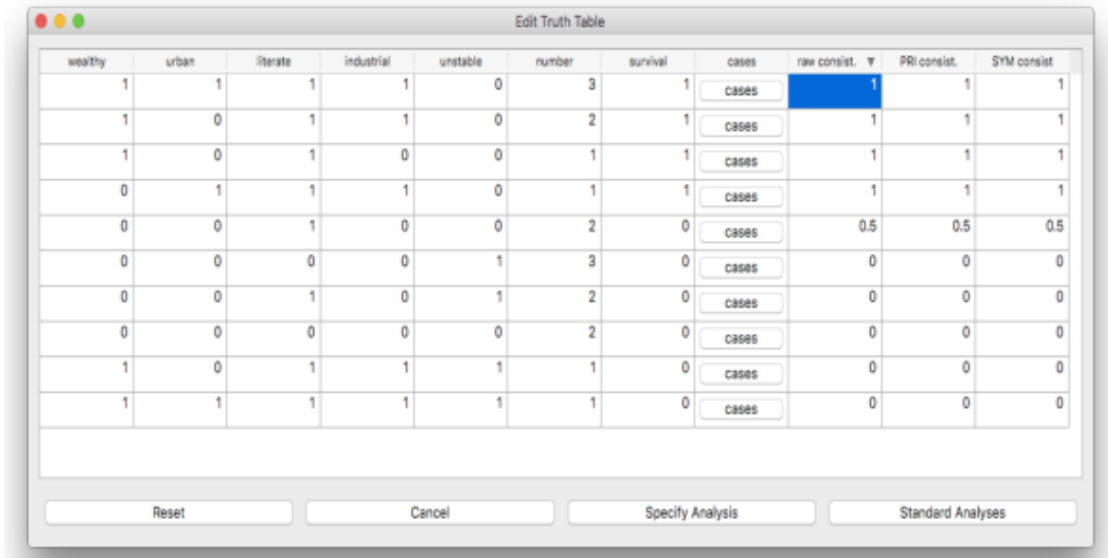
在第一个字段中，选择频率阈值。默认的个案数量是 1，但可以通过在该字段中输入所选频率阈值来更改。在第二个字段中，选择一致性阈值。默认一致性为 0.8，但可以通过在字段中键入选定的一致性阈值来更改默认一致性阈值。

单击“确定”。程序将删除不符合频率阈值的行，并根据所选的一致性阈值将结果编码为 0 或 1。

★以下窗口将显示真值表

1.对数据应用频率阈值 1 并消除没有任何观测值的构型（6 种构型）

2.选择 0.9 的一致性阈值，并为存在 0.90 一致性或更高构型（4 构型）的构型在存活列中放置 1，对于一致性较低（6 构型）的情况为 0，



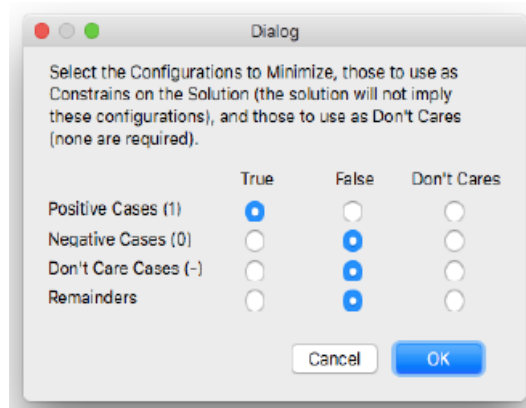
从这里，分析有两种可能性：指定单一分析与推导三种“标准”分析（复杂，简约和中级）。

建议步骤是点击“标准分析”按钮（提供三种解决方案）。

1) 指定分析选项

★建好真值表后（选择标准分析(Standard Analyses)之前），选择指定分析打开真值表分析窗口。

★在指定面板将正面案例（Positive cases）设定为“真”（True），其他设定为“假”（False），得到“最为复杂”的解。出现以下窗口：



★为了得到最为简洁的解，将正面案例（Positive cases）设定为“真”（True），负面案例（Negative cases）设置为“假”（False），逻辑余项设置为“不关心”（Don't Cares）。

★请注意：当选择质蕴涵算法无法减少真值表时，出现质蕴涵窗口（Prime Applicant Window），用户必须基于自身理论和实践知识选择应用的质蕴涵。当程序在推导简约解时，该窗口通常弹开，但也有可能出现在所有三个解中。（后面模糊集分析中会描述该窗口如何运行）。

★点击“确认”按钮运行分析，结果将会出现在输出窗口。

2) 标准分析选项

★当真值表全部建好后，选择标准分析（Standard Analyses）。标准分析会自动为用户提供复杂解、简洁解和中间解。**由于只有标准分析会提供中间解，它是建议选用步骤。**为了得到中间解，软件会基于用户提供的前因条件信息运行反事实分析。

3) 有限多样性与反事实分析

对比研究中最具挑战性的方面之一是，研究者应用的是相对较小样本的简单事实。研究者通常面临变量比样本还多的情况，这种情况非常复杂，因为对比着通常关注案例多方面的组合——案例的多个方面如何与构型匹配。例如，对于一位关注于四个前因条件交互的前因论证的研究者，理论上应该考虑这四个条件所有逻辑上可能的 16 种组合，以便能够为该论证提供全面评估。然而，自然发生的社会现象在其多样性方面受到极大限制。经验世界几乎从未向社会科学家提供与其论证相关的所有逻辑上可能的前因条件组合（如表 1 中的假设数据所示）。虽然有限多样性对于社会和政治现象的构成至关重要，但也严重复杂化了对他们的分析。

Table 1: Truth table with four causal conditions (A, B, C, and D) and one outcome (Y)

A	B	C	D	Y*
no	no	no	no	no
no	no	no	yes	?
no	no	yes	no	?
no	no	yes	yes	?
no	yes	no	no	no
no	yes	no	yes	no
no	yes	yes	no	?
no	yes	yes	yes	no
yes	no	no	no	?
yes	no	no	yes	?
yes	no	yes	no	?
yes	no	yes	yes	?
yes	yes	no	no	yes
yes	yes	no	yes	yes
yes	yes	yes	no	?
yes	yes	yes	yes	?

* Rows with "?" in the Y column lack cases – the outcome cannot be determined.

作为缺乏前因条件组合的替代，比较研究者通常进行“思想实验”（Weber [1905] 1949）。也就是说，他们通过利用理论和实践知识引导自己的评估，想象反事实案例并假设他们的结果。由于 QCA 运用真值表来评估跨案例模型，考虑反事实案例（即前因条件组合的缺失）的过程是明确和系统性的。事实上，QCA 的这一特征是它的主要优势之一。然而，这种对反事实案例的明确考虑以及将此评估结果系统纳入到对跨案例模式的陈述，对于社会科学来说是相对较新的。因此，对于 QCA 和反事实分析的最佳实践的规范是必不可少的。

思考一个案例（不是基于上述表 1）。一个研究者基于现有理论假设前因条件 A、B、C 和 D 都以某种方式与 Y 相关联。也就是说，这些条件的出现，而不是他们的缺失，应该与结果的出现相联系。实践证据表明，许多 Y 的实例伴随着前因条件 A、B 和 C 的出现相关，以及条件 D 的缺失（即 $A \cdot B \cdot C \cdot d \rightarrow Y$ ）。然而，研究者猜想，真正重要的是前三个条件 A、B 和 C 的出现。为了使 $A \cdot B \cdot C$ 产生 Y，D 并不需要，因而必须缺失。然而，实践中并没有观察到 A、B、C 出现和 D 出现相结合的实例（即 $A \cdot B \cdot C \cdot D$ 的观察实例）。因此，决定 D 缺失是否是前因组合（与 $A \cdot B \cdot C$ ）中必要的一部分的决定性实证案例根本不存在。

通过反事实分析（即，思想实验），研究者可以表明这个假设组合（ $A \cdot B \cdot C \cdot D$ ）是结果（Y）的一个可能的实例。也就是说，研究者或许可以断言 $A \cdot B \cdot C \cdot D$ （如果存在的话）会导致 Y。这个反事实分析可以通过以下逻辑简化：

$$\begin{aligned} A \cdot B \cdot C \cdot d + A \cdot B \cdot C \cdot D &\rightarrow Y \\ A \cdot B \cdot C \cdot (d + D) &\rightarrow Y \\ A \cdot B \cdot C &\rightarrow Y \end{aligned}$$

这个简化是否合理？这个问题的答案取决于当其他三个前因条件（ $A \cdot B \cdot C$ ）出现时，所陈述的 D 与 Y 之间关系的相关理论和实践知识。如果研究者基于现有知识确定有充分的理由预期在这些条件下 D 的出现有利于结果 Y（或者相反，D 的缺失不是一个促进因素），那么刚提出的反事实分析是合理的。换句话说，现有知识使得 $A \cdot B \cdot C \cdot D \rightarrow Y$ 的断言成为一个“简单”反事实，因为它涉及在已经与结果变量相连的构型中（ $A \cdot B \cdot C$ ）加入一个多余前因（D）。

QCA 的一个优势是它不仅提供了可以得到两端的复杂解和简洁解的工具，还提供了可以得到中间解的工具。考虑表 1 中的真值表，其使用 A，B，C 和 D 作为前因条件，Y 作为结果。与以前一样，假设现有的理论和实践知识认为这些前因条件的存在（而不是他们的缺失）与结果相关。反事实分析之外的分析结果（即复杂解）表明 $A \cdot B \cdot c$ 解释了 Y。对同样证据进行分析，允许产生更简约结果的任何反事实的结果（即简约解决方案），即 A 本身解释了 Y 的存在。可以将这两个结果视为复杂/简约连续体的两个端点，如下：

$$\underline{A \cdot B \cdot c} \qquad \underline{A}$$

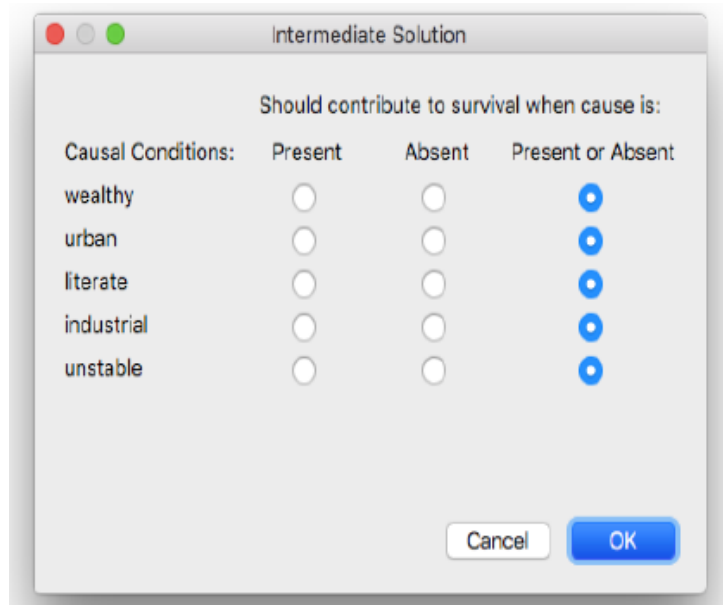
我们可以观察到复杂解（ $A \cdot B \cdot c$ ）是简约解（A）的一个子集。这也符合逻辑，两个方案都必须覆盖 Y 存在的真值表的行，简约解还包含了部分作为反事实案例的逻辑余项，因此包含了多余的行数。复杂/简约连续体中间是同一个真值表能产生的其他可能解，比如组合 $A \cdot B$ 。当用于产生简约解的逻辑余项的不同子集被包含在内时，便会产生中间解。这些中间解组成了最简约的解（这个例子中的 A）的子集以及最复杂的解（ $A \cdot B \cdot c$ ）的超集。这意味着，只要包含简约解（A）中的所有前因条件组合，任何使用复杂解（ $A \cdot B \cdot c$ ）中部分前因条件的前因组合都是真值表的有效解。因此，这个真值表有两个有效的中间解：

$$\frac{A \cdot B \cdot c \quad A \cdot B \quad A \cdot c}{A}$$

中间解 (A·B) 和中间解 (A·c) 都是简洁解的子集以及复杂解的超集。前者 (A·B) 允许反事实因子 A·B·C·D 和 A·B·C·d 作为与结果 Y 相关的组合。后者允许反事实因子 A·b·c·D 和 A·b·c·d。

这两种中间解方案的相对可行性取决于它们所包含的反事实案例的合理性。第一个中间解包含的反事实案例是“简单的”，因为它们用于从组合 A·B·c 中消除 c，并且在这个例子中，现有知识支持与结果 Y 的出现有关的是 C 的出现（而不是 C 的缺失）。然而，第二个中间解包含的反事实案例是“复杂的”，因为它们用来从 A·B·c 中消除 B。根据现有知识，B 的存在应该与结果 Y 的存在相关联。仅包含简单反事实案例的原则支持选择 A·B 作为最优中间解。该解决方案与传统案例导向研究者从这一证据中得出的方案相同。他们对以下三方面的前因条件组合具有直接兴趣：（1）正向案例（或者至少正向案例的子集）的共有部分，（2）被认为与结果相关，（3）没有由负向案例中展现。

★选择标准分析后，将出现一个指导得到中间解的窗口。在这里，如上所述，研究者必须选择理论上每种前因条件对结果的贡献。如果一个条件的出现对结果有影响，选择“存在”（Present）。如果一个条件的缺失对结果有影响，选择“缺失”（Absent）。如果一个条件的存在或缺失对结果有影响，选择“存在或缺失”（Present or Absent）。如果所有的条件都编码为“存在或缺失”，中间解就会和复杂解相同。



★ 请注意：当选择质蕴涵算法无法减少真值表时，出现质蕴涵窗口（Prime Applicant Window），用户必须基于自身理论和实践知识选择应用的质蕴涵。当程序在推导简约解时，该窗口通常弹开，但也有可能出现在所有三个解中。（后面模糊集分析中会描述该窗口如何运行）。

★点击“确认”（OK）按钮运行分析，复杂解、中间解和简约解都将会出现在输出窗口。输出窗口会清楚地标记每个方案，从复杂解开始，其次是简约解，然后是中间解。

5. 模糊集分析

本手册的这一部分涉及解决在《*Fuzzy-Set Social Science*》（Ragin, 2000）和《*Redesigning Social Inquiry*》（Ragin, 2008）中深入讨论的模糊集合的应用。模糊集不是仅允许两个相互排斥的状态（即完全隶属和完全不隶属），而是扩展了清晰集，允许 0~1 中间间隔的隶属程度。构建模糊集的方法有许多种，常见的三种方式是：

四值模糊集（0, 0.33, 0.67, 1）

六值模糊集（0, .2, .4, .6, .8, 1）

连续模糊集（任何 ≥ 0 且 ≤ 1 的值）

有一种模糊集算法，即“真值表”算法，被证明是一种最稳健的方法。

《*Redesigning Social Inquiry*》（Ragin, 2008）和《*Configurational Comparative Methods*》（Rihoux and Ragin, 2008）都介绍了这个真值表算法。

5.1 模糊集的运算

与（AND）和或（OR）在模糊集算法的逻辑运算与在清晰集中不同。以下是常见运算的介绍：逻辑加，逻辑或和逻辑非。

逻辑与。在模糊集中，逻辑与是通过在相交集合中获取每个案例的最小隶属度。例如，如果在贫穷国家集合中一个国家的隶属度为 0.34，它在民主国家的隶属度为 0.91，那么它在贫穷且民主的国家集合中的隶属度是两者的较小值，即 0.34。

逻辑或。两个或以上的集合也可以通过逻辑或连接一集合连接。例如，一个研究者或许会对“发达”或“民主”的国家感兴趣，因为这两个条件可能会为某些结果（例如官僚主义政府）提供相同的依据。通常，清晰类别可以用于编制一个“发达或民主”国家的完整清单（即，具有一个或两个特征的国家）。在模糊集中，研究者聚焦在集合中每个案例的隶属度的最大值。也就是说，由两个或多个集合组成的并集中的隶属度是集合中案例的隶属度的最大值。因此，如果一个国家在民主国家集合的隶属度是 0.15，在发达国家集合的隶属度是 0.93，那么它在“民主或发达”国家集合的隶属度是 0.93。

逻辑非。与清晰集一样，模糊集也可以被否定。在清晰集的逻辑中，逻辑非将隶属度的分数从 1~0 转换为从 0~1。这个简单的数学原理也适用于模糊集。相关数值不在局限于布尔值 0 和 1，而是扩展到了 0 和 1 之间的值。在模糊集合 A 的非集中计算一个案例的隶属度，只需从 1 中减去集合 A 中改案例的隶属度，如下：

非 A 集合中的模糊隶属度=1-集合 A 中的模糊隶属度

这也可以用 $\sim A_i = 1 - A_i$ 来展示，其中下标“i”表示第“i”个案例，“非 A”集合用 $\sim A$ 表示，符号“ \sim ”表示否定。因此，比如，如果美国在“民主国家”集合中的隶属度是 0.79，那么它在“非民主国家”集合的隶属度是 0.21。

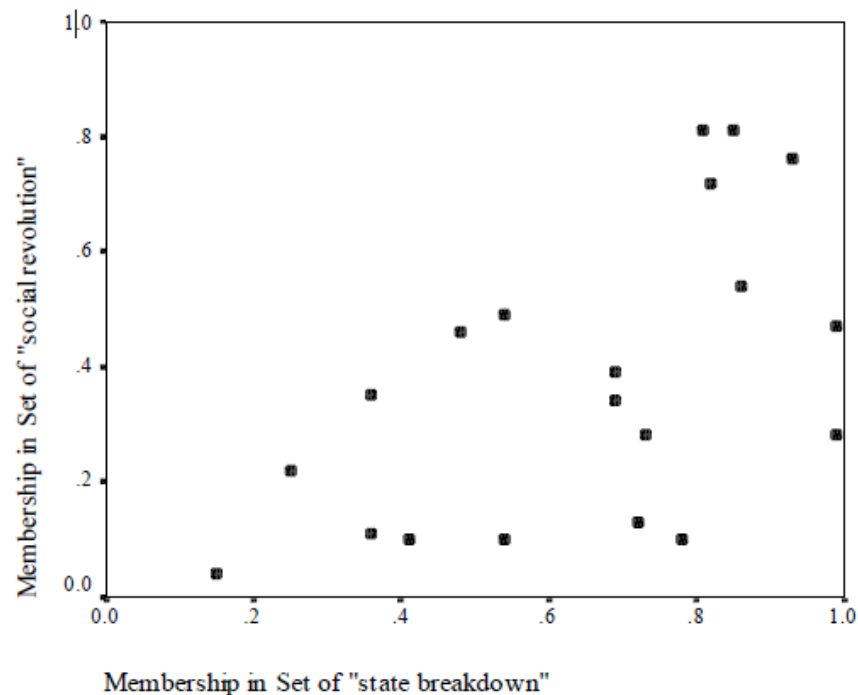
5.2 模糊集合，必要性和充分性（模糊子集关系）

清晰集合中隶属度间的子集原理和算术关系。 思考国家崩溃是社会变革的必要但非充分条件的例子（《Fuzzy Set Social Science》中第 211 页）。从逻辑上来说，如果一个条件是一个结果的必要不充分条件，那么结果的实例就组成了前因的实例的一个子集。理解子集关系的另一种方式是根据清晰集隶属度（1 和 0）的算术关系。如果结果集合中的实例是前因集中实例的子集，那么结果的布尔值（1vs0）会小于或等于前因的布尔值。

模糊集合中隶属度间的子集原理和算术关系。 在模糊集中，很难“选择”具有结果的国家（通常是清晰集中的必要条件分析的第一步），因为在展示社会变革的集合中，各国的隶属度各不相同。同样，也很难评估案例对前因条件（国家崩溃）的一致性，因为他们在集合中也具有不同的隶属度。

幸运的是，子集原理和隶属度之间的算术关系也适用于模糊集合。在模糊集中，如果集合 A 中案例的隶属度小于或等于集合 B 中该案例的隶属度，那么集合 A 就是集合 B 的子集。此外，当结果中的模糊隶属度小于或等于前因中的模糊隶属度时，则有可能认为结果中的实例是前因中的实例的子集。图 1 展示了两个维度的算术关系。当研究者发现这个这种模式时，可以引用这个证据作为对前因必要性的支持。

Figure 1: Plot of “social revolution” against “state breakdown”



对充分性的评估可以看作是对显示前因条件的案例是否构成了显示结果的案例的子集的检验。如上所示，另一种理解子集关系的方式是隶属度分数的算术关系。为了证明一个前因或者前因组合是结果的充分条件，前因中的模糊隶属度分数应该小于或等于结果中的模糊隶属度分数。

考虑来自《*Fuzzy Set Social Science*》中第 236 页中的示例。图 2 展示了充分性前因组合 (\sim cross-class \cdot \sim multiracial) 和结果 (ideological conflict) 之间的算术关系。图 2 所示的上三角图直接反映了模糊集 “race and class homogeneity” 中的隶属度分数低于或等于模糊集 “ideological conflict” 中的隶属度分数这一事实。

请注意应用子集原则到充分性评估和应用其到必要性评估的重要差异。为了证明必要性，研究者必须证明结果是前因的一个子集。为了支持充分性论断，研究者必须证明前因是结果的一个子集。

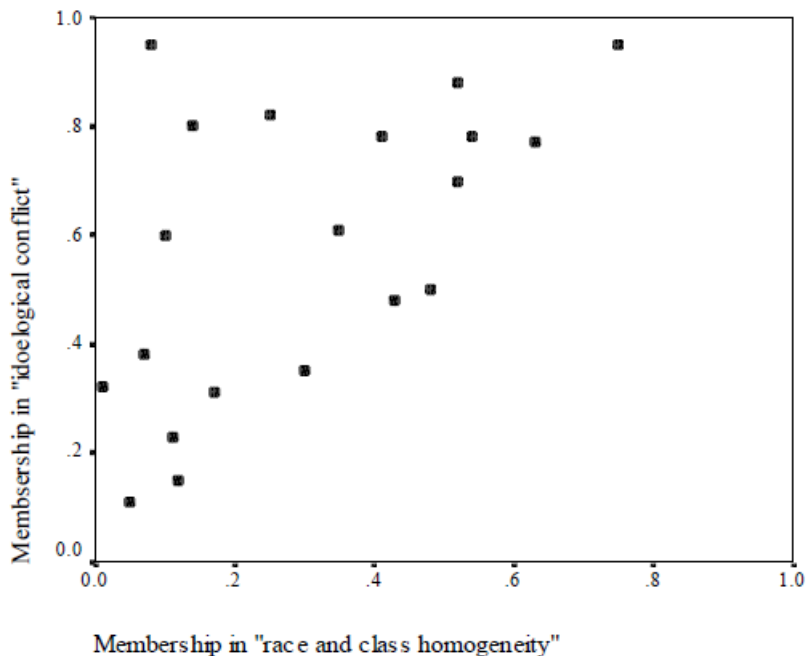


Figure 2: Plot of “ideological conflict” against “race and class homogeneity”

5.3 运用模糊真值表算法

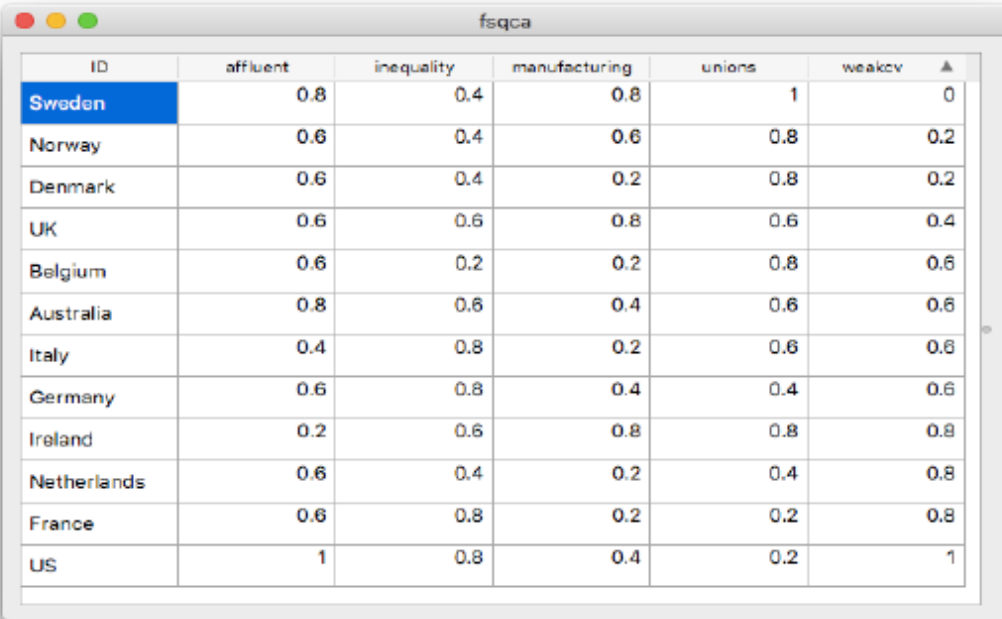
运用真值表分析模糊集合的方法在 fsQCA2.0 版本中已经介绍。Ragin (2008) 的《*Redesigning Social Inquiry: Fuzzy Sets and Beyond*》以及 Rihoux 和 Ragin (2008) 的《*Configurational Comparative Methods: Qualitative Comparative Analysis (QCA) and Related Techniques*》已详细介绍了该方法。

模糊真值表算法可以概念化为三大支柱的桥梁。第一个支柱是清晰真值表的行与模糊集前因条件的向量空间角的坐标的直接对应关系 (参见 Ragin, 2000)。第二个支柱是评估案例在不同逻辑上可能的前因条件组合 (或者是向量空间分区) 的分布情况。向量空间的某些扇区可能拥有很多具有高隶属度的案例，而其他向量空间或许拥有较弱隶属度的案例。第三个支柱是评估每个前因组合构

成结果的一个模糊子集的一致性。真值表算法涉及建立这三个支柱来构建一个清晰真值表，然后进行类似于清晰算法的分析过程。本节将解释在清晰真值表中记录多个模糊集分析结果然后分析该表所涉及的步骤。

数据

如第一章和第二章所述，模糊集数据可以通过其他程序导入或者在 fsQCA 中创建。本章将以 Ragin (2005) 的弱类投票国家为例。下表展示了数据表：



ID	affluent	inequality	manufacturing	unions	weakcv
Sweden	0.8	0.4	0.8	1	0
Norway	0.6	0.4	0.6	0.8	0.2
Denmark	0.6	0.4	0.2	0.8	0.2
UK	0.6	0.6	0.8	0.6	0.4
Belgium	0.6	0.2	0.2	0.8	0.6
Australia	0.8	0.6	0.4	0.6	0.6
Italy	0.4	0.8	0.2	0.6	0.6
Germany	0.6	0.8	0.4	0.4	0.6
Ireland	0.2	0.6	0.8	0.8	0.8
Netherlands	0.6	0.4	0.2	0.4	0.8
France	0.6	0.8	0.2	0.2	0.8
US	1	0.8	0.4	0.2	1

其中，ID 是 Country Identifier 在数据表中的简称，affluent 是 Affluent 的简称，inequality 是 Substantial Income Inequality 的简称，manufacturing 是 Strong Manufacturing Sector 的简称，unions 是 Strong Unions 的简称，以及 weakcv 是 Weak Class Voting 的简称。

等距和等比数据可以通过操作手册第二章（数据编辑）以及 Ragin (2008) 的书中介绍的“calibrate”程序转化为模糊集隶属度分数。

分析

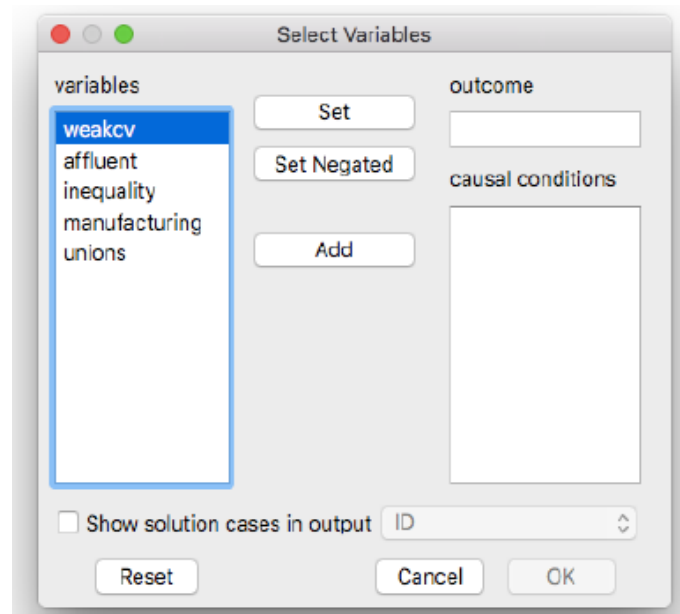
真值表算法采用两阶段分析程序。第一步包括从模糊数据中创建真值表，其中包括分析每一个构型的结果并确定要包含在分析中的构型。第二步涉及最小化前因条件和结果。这两步必须结合起来执行，并且在每一个单独分析中必须执行这两步。

★为了指定要在分析中使用的结合，请选择：

分析 (Analyze)

真值表算法 (Truth Table Algorithm) ...

以下窗口将打开：



★为了指定要在分析中使用的结合，请选择：

★确定并突出显示要解释的案例方面，并通过单击“设置”（Set）将其转移到“结果”（outcome）字段中。

★一次选择一个前因条件，然后单击“添加”（Add）将它们逐个移至前因条件（causal conditions）字段。

★选中“在输出中显示解决方案案例”旁边的复选框，然后选择您的 caseID 变量。

★单击“确定”（OK）按钮，将出现包含真值表的以下窗口：

affluent	inequality	manufacturing	unions	number	weakcv	cases	raw consist.	PRI consist.	SYM consist.
1	1	0	0	3 (25%)		cases	1	1	1
1	0	1	1	2 (41%)		cases	0.722222	0.166667	0.166667
1	0	0	1	2 (58%)		cases	0.789474	0.428571	0.428571
1	1	1	1	1 (66%)		cases	0.777778	0.2	0.2
0	1	0	1	1 (75%)		cases	0.823529	0.4	0.4
1	1	0	1	1 (83%)		cases	0.842105	0.5	0.5
0	1	1	1	1 (91%)		cases	0.875	0.5	0.5
1	0	0	0	1 (100%)		cases	1	1	1
0	0	0	0	0 (100%)		cases			
0	1	0	0	0 (100%)		cases			
0	0	1	0	0 (100%)		cases			
1	0	1	0	0 (100%)		cases			
0	1	1	0	0 (100%)		cases			
1	1	1	0	0 (100%)		cases			
0	0	0	1	0 (100%)		cases			
0	0	1	1	0 (100%)		cases			

★真值表将有 2^k 行（其中k表示前因条件的数量），反映前因条件的所有可能组合。1和0分别代表由模糊集前因条件定义的向量空间的角点。对于每一行，将创建以下每个变量的值：

数量。在改向量空间的角点超过0.5的隶属度的案例数量。括号中显示的是该向量空间中从最密集的部分开始的案例的累积百分比。

原始一致性。向量空间角中的隶属度是结果隶属度的子集的一致性程度（在清晰集中，原始一致性是每个真值表行中显示结果的案例比例）。

PRI一致性。基于误差计算中的准比例减少的模糊集合的另一种一致性计算方式（在清晰集中，它等于原始一致性）

SYM一致性。基于PRI一致性对称版本的模糊集合的一致性替代度量。

请注意，标记为结果的一列（在此示例中 *weakcv*）为空白。由研究者根据以下程序确定每种构型的结果并填写，并使用以下过程将其输入电子表格。

★研究人员必须首先基于由前因条件形成的每一向量空间中的案例数，来制定一个规则用以区分哪些是相关构型（向量空间角）哪些不是。这是通过根据数字列中显示的每行中隶属度超过0.5的案例数量来选择频率阈值来完成的。当样本总量N相对较小时，频率阈值应该为1或2。当总N很大时，应该使用更实际的阈值。通过检查样本在前因条件中的分布来确定向量空间中最多的构型变得非常重要。通常，选择的构型需要占据总样本至少75-80%的比例。

★可以通过单击数字列的标题来按照案例的频率对其进行排序（降序或升序）。

★对行进行排序并选择频率阈值后，删除所有不符合阈值的行。如果案例按照降序排列，请单击低于阈值的第一个案例，然后选择：

编辑 (Edit)

删除当前行到最后一行 (Delete current row to last row) ...

如果案例尚未排序，那么可以通过单独选择每一行来删除那些不符合阈值的案例

编辑 (Edit)

删除当前行 (Delete current row) ...

★下一步是区分结果的子集和非子集，通过原始一致性、PRI一致性和/或SYM一致性报告的集合理论一致性的度量来确定。原始一致性的值低于 0.8 表示实质不一致。按降序对一致性评分进行排序以评估其分布（这应在**删除不符合频率阈值的行后执行**）非常有用。可以通过单击原始一致性或 PRI 一致性或 SYM 一致性的列的标题来进行排序（确保单击列表提时出现的箭头指向下方）。

找出可能对建立一致性阈值有用的较高一致性范围中的任何差距。请记住，可以检验几个不同的阈值，并通过降低或提高一致性截点查看结果。

★现在有必要指出哪些构型显示出结果，哪些没有。对于一致性水平达到和/或超过阈值的构型，在结果列中输入 1。对于一致性水平不满足一致性阈值的构型，在结果列中输入 0。

★或者，可以使用“删除和编码” (Delete and code) 功能来自动执行此过程。选择：

编辑 (Edit)

删除并编码 (Delete and code) ...

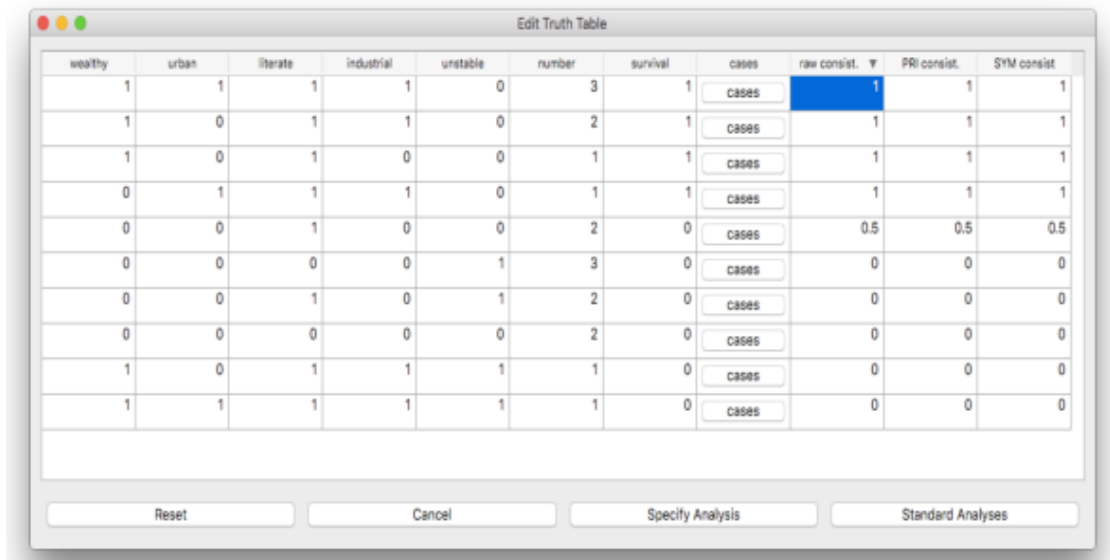
在第一个字段中，选择频率阈值。默认的个案数量是 1，但可以通过在该字段中输入所选频率阈值来更改。在第二个字段中，选择一致性阈值（原始一致性）。默认一致性为 0.8，但可以通过在字段中键入选定的一致性阈值来更改默认一致性阈值。

单击“确定” (OK)。程序将删除不符合频率阈值的行，并根据所选的一致性阈值将结果编码为 0 或 1。

★以下窗口将显示真值表

1.对数据设置频率阈值为 1，消除没有任何观测值的构型（8 种构型）

2.选择一致性阈值为 0.8，并为等于或大于 0.8 的一致性的构型（5 个构型）的 *weakvcy* 一列设置为 1，其他设置为 0（3 个构型）。

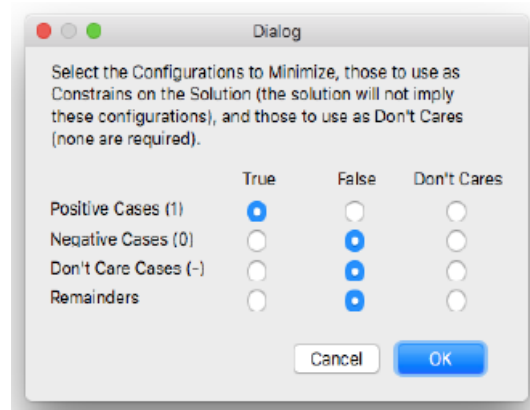


从这里，有两种可能的分析：指定分析和标准分析。**建议选择“标准分析”按钮，因为它是唯一可以生成中间解的方法。**

指定分析选项

★建好真值表后，选择指定分析打开真值表分析（对话）窗口。

★在指定面板将正面案例（Positive cases）设定为“真”（True），其他设定为“假”（False），得到复杂解。出现以下窗口：



★为了得到最为简洁的解，将正面案例（Positive cases）设定为“真”（True），负面案例（Negative cases）设置为“假”（False），逻辑余项设置为“不关心”（Don't Cares）。

★请注意：当选择质蕴涵算法无法减少真值表时，出现质蕴涵窗口（Prime Applicant Window），用户必须基于自身理论和实践知识选择应用的质蕴涵。当程序在推导简约解时，该窗口通常弹开，但也有可能出现在所有三个解中。（后面标准分析中会描述该窗口如何运行）。

标准分析选项

★当真值表全部建好后，选择标准分析（Standard Analyses）。标准分析会自动为用户提供复杂解、简洁解和中间解。建议使用“标准分析”而不是“指定分析”。

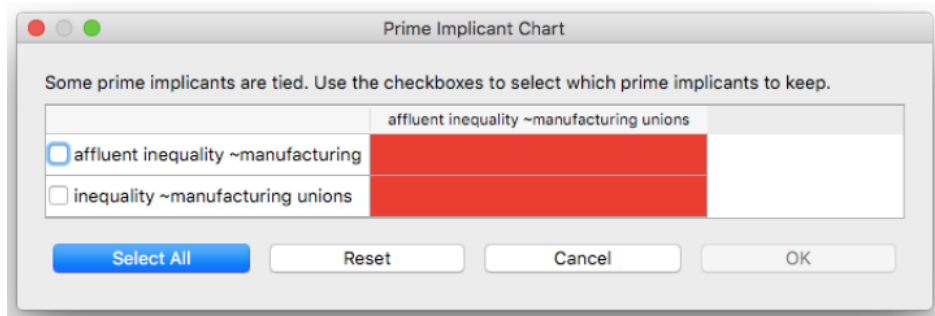
请参阅有关清晰集的“标准分析”过程的讨论。模糊集的步骤是相似的，可以得到三个解：复杂解、简约解和中间解。每一种解都是基于考虑不同逻辑余项的结果。

复杂解：将所有逻辑余项设置为“假”，没有反事实案例；

简约解：包含所有会产生逻辑更简洁解的逻辑余项，不管它是简单还是复杂反事实案例；

中间解：只包含简单反事实案例的逻辑余项。简单和复杂是基于用户提供的关于每个前因条件和结果关系的信息来决定的。

当选择质蕴含算法无法减少真值表时，出现质蕴含窗口（Prime Applicant Window），用户必须基于自身理论和实践知识选择应用的质蕴含。将出现以下窗口：



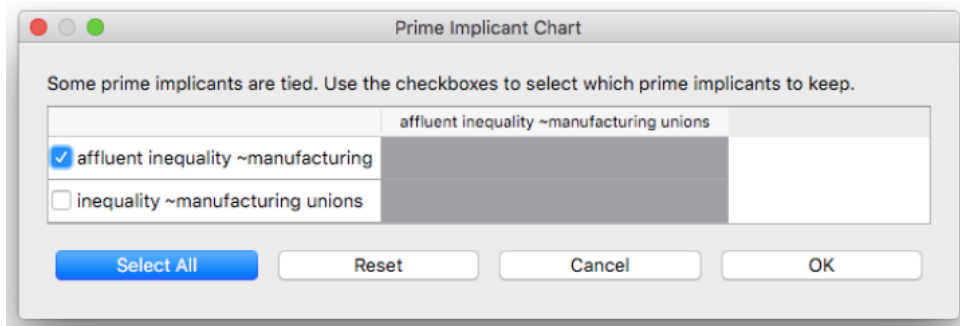
质蕴含（PIs）是使用最小化规则生成的产品术语（例如，前因组合行中只有一个因素不同但结果相同的规则）。例如，ABC 与 AbC 的组合生成 AC。因此，AC 是覆盖两个原始布尔表达式 ABC 和 AbC 的质蕴含。换句话说，ABC 和 AbC 是 AC 的子集，或 AC 暗含 ABC 和 AbC。

但是，通常情况下，覆盖所有原始表达的简化版质蕴含远比所需要的多，使用者可以运用质蕴含表格从中选择那些“逻辑相关”的质蕴含。（有关质蕴含应用的更全面的讨论，请参阅《The Comparative Method》第 95 页）。

★为了选择质蕴含（PIs），程序会采用一种算法来尝试简化表格，直到不能再进行简化为止。它起始于结果方案中必须要出现的必要 PIs（它能够唯一覆盖真值表中的特定行）。如果运行算法后，不能完全减少表格，使用者或许可以根据理论和实践知识选择要应用的 PIs。

★质蕴含表的选项卡展示了可供用户选择的可能质蕴含。表中的每一列代表了可以被一个以上质蕴含所覆盖的不同的真值表。顶部的“数据”字段显示了真值表中有问题的行（即需要被覆盖的行）。

★左侧的“Prime”字段描述了用户可以选择的PI。图表中的每一行代表了一个可供选择的PI。通过单击第一列中的单元格来选择一个PI。

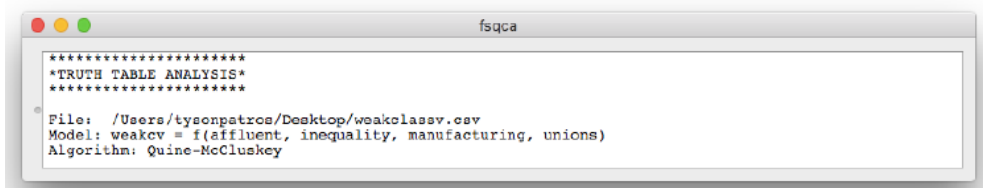


★点击所需的一个或多个单元格后，按“确定”（OK）运行分析。

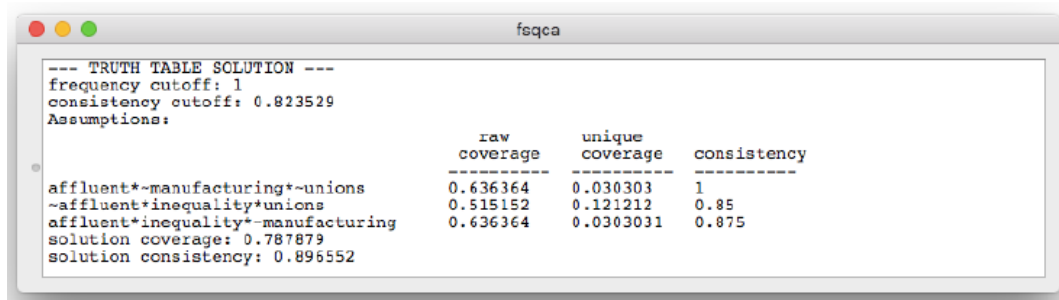
5.4 “指定分析”选项的输出

一旦真值表被简化后，主窗口将显示以下输出。所示输出是如上文所述的通过运行“指定选项”按钮获得的最复杂解。

输出的第一部分描述了真值表分析的数据。前两行表示文件目录和您指定分析的模型。接着，输出显示了您所使用的算法类型。



输出的最后一部分显示了分析的真值表解决方案（Truth Table Solution）。首先列出的是频率和一致性的截点值。接下来是解。当选择“指定分析”时，会出现一个真值表解决方案部分。当选择“标准分析”时，三个解的部分都会显示（复杂、简约和中间解）。在这个例子中，报告了复杂解的方案，其中 weak class voting 是三个构型的结果。



前两行列出了频率和一致性的截点值。一致性截点值列出的是用户指定选择的阈值之上最低的一致性值。在这里，0.8是一致性阈值，高于0.8的最低的实际值是0.823529。

结果方案为每一条得到结果的路径都提供了一行（在这个例子中有三条路径）。结果方案还计算了每一个方案和整个方案的一致性和覆盖率值（这些计算将在下面讨论）。

同样的真值表得到的最简约方案的输出如下：

```

*****
*TRUTH TABLE ANALYSIS*
*****
File: /Users/tysonpatros/Desktop/weakclassv.csv
Model: weakcv = f(affluent, inequality, manufacturing, unions)
Algorithm: Quine-McCluskey

--- PARSIMONIOUS SOLUTION ---
frequency cutoff: 1
consistency cutoff: 0.823529

```

	raw coverage	unique coverage	consistency
~unions	0.721273	0.0606061	1
~affluent	0.606061	0.121212	0.869565
inequality*~manufacturing	0.69697	0.030303	0.851852

```

solution coverage: 0.909091
solution consistency: 0.882353

```

结果方案显示得到弱类投票的三条路径。在拥有不强的工会的国家集合，或者是在不富裕国家集合，或者是收入不平等且制造业不强的国家集合中具有隶属度的国家都展示出弱类投票。

5.5 “标准分析” 选项的输出

标准分析的输入看起来会稍有不同：

```

*****
*TRUTH TABLE ANALYSIS*
*****
File: /Users/tysonpatros/Desktop/weakclassv.csv
Model: weakcv = f(affluent, inequality, manufacturing, unions)
Algorithm: Quine-McCluskey

--- INTERMEDIATE SOLUTION ---
frequency cutoff: 1
consistency cutoff: 0.823529
Assumptions:
  affluent (present)
  inequality (present)
  manufacturing (present)
  unions (present)

```

	raw coverage	unique coverage	consistency
affluent*~unions	0.69697	0.0606061	1
~affluent*inequality*unions	0.515152	0.0909091	0.85
affluent*inequality*~manufacturing	0.636364	0.0303031	0.875

```

solution coverage: 0.818182
solution consistency: 0.9

```

最值得注意的是，输入将显示复杂、中间以及简约解。上图展示的是中间解。输出的“假设”（Assumptions）部分显示的是先前“中间解”窗口中选择的选项（参见第 46 页）。在这里，每一个条件都是这样选择的，当案例出现时，该条件有助于结果出现。

在该解决方案中，弱类投票是三条路径的结果—在富裕国家集合中具有高隶属度且在强工会国家集合中具有低隶属度的国家；富裕国家具有低隶属度，不平等国家具有高隶属度，并且在强工会国家集合中具有高隶属度的国家；以及

富裕国家结合中具有高隶属度，不平等国家具有高隶属度，且在制造业国家具有低隶属度的国家。

5.6 一致性和覆盖率

输出包括每一个方案以及整个方案的覆盖率和一致性的测量。一致性（充分性）衡量了每个解决方案以及整个方案是结果集合的子集的程度。覆盖率衡量了每个解决方案和整个方案在多大程度上覆盖（或解释）了结果。这些测量是通过检验方案（有一个或多个方案组成）对应的原始模糊数据集合来计算的。原始数据集合中的案例在多大程度上出现在每个解决方案和结果中具有隶属度构成了一致性和覆盖率测量的基础。更具体地说，思考以下数据表，其中三个前因条件（A、B 和 C）和一个结果（Y）都是以模糊集合度量的。

Causal Condition Membership			Outcome Membership	Solution Membership			Consistency Calculations		
A	B	C	Y	A*B	A*C	A*B + A*C	C _{A*B}	C _{A*C}	C _{A*B+A*C}
.8	.9	.8	.9	.8	.8	.8	.8	.8	.8
.6	.7	.4	.8	.6	.4	.6	.6	.4	.6
.6	.7	.2	.7	.6	.2	.6	.6	.2	.6
.6	.6	.3	.7	.6	.3	.6	.6	.3	.6
.8	.3	.7	.8	.3	.7	.7	.3	.7	.7
.6	.1	.7	.9	.1	.6	.6	.1	.6	.6
.7	.4	.2	.3	.4	.2	.4	.3	.2	.3
.2	.9	.9	.1	.2	.2	.2	.1	.1	.1
.1	.6	.2	.2	.1	.1	.1	.1	.1	.1
.2	.1	.7	.3	.1	.2	.2	.1	.2	.2
.3	.1	.3	.3	.1	.3	.3	.1	.3	.3
.1	.2	.3	.2	.1	.1	.1	.1	.1	.1
Sum:			6.2	4.0	4.1	5.2	3.8	4.0	5.0

该分析的相关输出如下所示。结果方案包含两个部分： $A*B + A*C$ 。计算一致性和覆盖率之前，需要先计算出几个中间值。结果中的隶属度（ ΣY ）是数据中所有案例的结果隶属度得分的总和。一个案例在每个方案中的归隶属度取决于案例在方案组成中每一个前因条件的隶属度的最小值。第一个解（ Σ_{A*B} ）的隶属度是所有案例在这个方案组合中的隶属度的总和。同样，第二个解（ Σ_{A*C} ）的隶属度是所有案例在这个方案组合中的隶属度的总和。而解（ $\Sigma_{(A*B + A*C)}$ ）的隶属度被定义为案例在两个方案组合中的最大值。

```

              raw      unique
              coverage  coverage  consistency
-----
A*B+         0.612903    0.161290    0.950000
A*C          0.645161    0.193548    0.975610
solution coverage: 0.806452
solution consistency: 0.961538

```

一致性测量了每个解的归属度是结果的子集的程度。一致性通过首先计算每个案例的一致性得出。在一个解的方案中，如果一个案例在解的隶属度小与或等于其在结果的隶属度，那么这个案例就是一致的。如果一个案例在解的隶属度大于其在结果的隶属度（即不一致），那么这个案例会被赋予一个与其在结

果中的隶属度相同的分数。然后将这些分数求和（得到 ΣC_{A*B} ），再除以解（ Σ_{A*B} ）的隶属度的总和。因此，第一个解的一致性为 $\Sigma C_{A*B} / \Sigma_{A*B} = 3.8/4 = 0.95$ ，第二个解的一致性为 $4.0/4.1 = 0.976$ 。

*解的一致性*测量了解（解的集合）的隶属度是结果隶属度的子集的程度。将解的每个案例的隶属度的最大值 $\max(A*B + A*C)$ 与结果的隶属度进行对比。如果在解的隶属度小与或等于其在结果的隶属度，那么这个案例将被赋予与解的隶属度相同的分数。如果一个案例在解的隶属度大于其在结果的隶属度（即不一致），那么这个案例会被赋予一个与其在结果中的隶属度相同的分数（两个值的较小值）。将这些分数相加，然后除以解的隶属度的总和（ $\Sigma C_{(A*B + A*C)} / \Sigma_{(A*B + A*C)}$ ）。在本例中，解的一致性为 $5.0/5.2 = 0.962$ 。

*解的覆盖率*测量了完整解所解释结果的隶属度的比例。将所有案例的一致性隶属度分数相加，然后除以结果的隶属度的综合： $(\Sigma C_{(A*B + A*C)} / \Sigma_Y) = 5/6.2 = 0.806$ 。

*原始覆盖率*测量了每一个解所解释结果的隶属度的比例。通过计算解的一致性隶属度的和再除以结果隶属度的和，得到原始数据的每个解的原始覆盖率。第一个解的原始覆盖率是 $\Sigma C_{A*B} / \Sigma_Y = 3.8/6.2 = 0.613$ ，第二个解的原始覆盖率是 $4.0/6.2 = 0.645$ 。

*唯一覆盖率*测量了仅由每个解所解释的结果隶属度的比例（其他解没有涵盖的隶属度）。首先，将该解移除总解，计算解的覆盖率。在此案例中，移除第一个解（ ΣC_{A*B} ）的解的覆盖率仅为 ΣC_{A*C} （有 n 个解的方案中移除后的总解包含 $n-1$ 个解）。整个解的覆盖率减去原始覆盖率就得到了除去解的唯一覆盖率。第一个解（ ΣC_{A*B} ）的唯一覆盖率等于： $(\Sigma C_{(A*B + A*C)} / \Sigma_Y) - (\Sigma C_{A*C} / \Sigma_Y) = (5.0/6.2) - (4.0/6.2) = 0.161$ 。第二个解的唯一覆盖率等于 $(5.0/6.2) - (3.8/6.2) = 0.194$ 。