1. **注意事項と分析内容**

**I-1. 注意事項**

これは、RのQCApackageの取り扱い説明書である。この説明書以外に、サンプル・スクリプト（QCA）とサンプル・データセット（LIP）の３つがセットになっている。サンプル・スクリプトはWordで書かれており。これを、コピーして、そのままRあるいは、Rstudioに張り付ければ、Rのスクリプトとなる。すぐにQCAを実行することができる。この説明書を読みながら、各段階で、Rを実行していけば、QCAが何をしているのかが、具体的にわかるように、説明書は構成されている。これをｋ実行する際に、具体的に分析対象になっているのが、サンプル・データ（LIP）である。これは、Excel CSVで書かれているので、LIPという名前で保存しておけば、そのまま、Rに読み込むことができる。解説を読みながら、何をしているのか分析内容を理解して、Rを実行しいけば、Rの使い方も、QCAも両方理解できる。ただし、RのQCAには、致命的なバグがある。それを理解していないと、RのQCApackageは使えない。説明書ではそれについても、詳しく解説している。

なお、Wordで書かれたスクリプトを、Rにコピーする際に、行番号が、もとのWordの行番号とは違ってしまう。これは、WordとRで１行の文字数が異なっているためである。それを調整する方法を考えてみたが、良い方法を思いつかなかった。そのため、本説明書では、筆者が、サンプル・スクリプトを自分のコンピューターのRstudioにコピーした時の、行番号で内容を解説している。行番号が違っているため、該当する箇所がわからない場合には、Wordの行番号の最大50以内の行に該当する部分があるので、説明文を頼りに、該当箇所を見つけてもらいたい。

I-2. 分析内容

分析事例として使ったのは、QCAのパッケージに内蔵されているデータである。このデータは、QCAを立ち上げて

library QCA、

data(LR)

LR

を実行すれば、

> data(LR)

> LR

DEV URB LIT IND STB SURV

AU 720 33.4 98.0 33.4 10 -9

BE 1098 60.5 94.4 48.9 4 10

CZ 586 69.0 95.9 37.4 6 7

EE 468 28.5 95.0 14.0 6 -6

FI 590 22.0 99.1 22.0 9 4

FR 983 21.2 96.2 34.8 5 10

DE 795 56.5 98.0 40.4 11 -9

GR 390 31.1 59.2 28.1 10 -8

HU 424 36.3 85.0 21.6 13 -1

IE 662 25.0 95.0 14.5 5 8

IT 517 31.4 72.1 29.6 9 -9

NL 1008 78.8 99.9 39.3 2 10

PL 350 37.0 76.9 11.2 21 -6

PT 320 15.3 38.0 23.1 19 -9

RO 331 21.9 61.8 12.2 7 -4

ES 367 43.0 55.6 25.5 12 -8

SE 897 34.0 99.9 32.3 6 10

UK 1038 74.0 99.9 49.9 4 10

と出てくる

このデータフレームの名前、LRはおそらくLipsetのRawデータの略号だろう。Seymour Martin Lipset(1922-2006)は、「リプセットの仮説：経済発展は中間層を形成し、政治の民主化をもたらす」を唱えた、高名なアメリカの政治社会学者であり、データフレームLRを使ったQCA分析の目的は、この仮説の検証である。現在の中国、イラン革命、さらに現在のトランプ政権の強権的な政策をみれば、民主主義の維持が、経済の発展のみによって支えられているのではないことは明らかだ。したがって、Liopsetの仮設そのものは、すでに破綻している。あるいは修正を迫られていると言えるのだろう。この分析事例は、Lipsetが研究対象とした戦間期（1919年第一次世界大戦の終了から1939年第二次世界大戦の勃発）のヨーロッパについて、当時の経済指標や政治情勢のデータを用いて、Lipsetの仮説を県書するものである。データは、おそらく、QCAの開発者であるCharles C.Raginらが、当時の経済指標のデータ等を拾い上げたものである。分析の視点として重要なのは、たとえば、第一次世界後のベルサイユ条約により、莫大な賠償金の支払い義務を負わされたとはいえ、フィンランド、アイルランド、チェコよりも経済的に発展し工業化が進んでドイツにおいて、何故、民主主義が崩壊し、ヒトラーが台頭したのか、あるいは、ドイツとソ連に挟まれて、ドイツとも戦い、ソ連とも戦って枢軸国に含まれたフィンランドにおいて、どうして、民主主義が維持されたのかという視点である。自然科学においては、数値のみを客観的なデータとして、ここで例に挙げたような予見を持たずに分析をすることが多く、そうした態度を良しとすることが多いが、社会科学の分野において、そうした問題意識を持たずに分析をすることはほとんど不可能だろう。ということを念頭において、解説を読んでいただきたい。

データフレームは、5つの項目（列）と１８の事例（行）からなる。１列目の列名は、DEV

である。これは、おそらくdevelopmentの略であろう。具体的には、1930年時点の一人当たりの国民総生産（GNP/capita）である。２列目の列名は、URBである。これは、urbanizationで、都市化の程度を表していて、具体的には全人口にしめる人口２万人以上の都市に住む住人の割合である。3行目のLITは(識字率:literacy)である（以上の詳細は、Berg-Schlosser, D. and Michell, J. 2000, 2003を参照）。５列目のSTBはstability政治の安定性であるが、これは戦間期30年間に国を統治した内閣の数であり、この数値が大きければ、政治が不安定であり、政治が安定していたと判断する。この数値は、Charles Raginらが、いくつかのQCAの解説本（たとえば、Qualitative Competitive Analysis and Related Techniques 2009）の中で加えたものである。６列目のSURVはsurvivalで、民主主義の維持状態であるが、これがどのようなデータに基づいて、誰がどのように数値化したのか、説明を探したが見つからなかった。民主主義の維持と崩壊の程度を10点法のscoreで表したもので、民主種具が完璧に維持されていれば１０、民主主義が完全に崩壊していればー１０のスコアとなっている。政治社会学の分野では、その評価の妥当性がまず問題になるのだろうが、この文書を書いている筆者には、このスコアの妥当性を論ずるだけに知識能力がないので、そのまま受け止めておく。このスコアが、一般的な数理統計の分析で対象になる被説明変数にあたるものであり、他の列の数値が説明変数に当たるものということになる。QCAでは、このスコアーによって、分けられる集合とその補集合と他のスコアーによって振り分けれる集合、補集合との包含関係を論ずる。これが、QCAによる分析の作業であることを理解しておくと、スクリプトの理解が容易だろう。

列名のアルファベットは、分析対象の事例となる、ヨーロッパの国の国名IDである。一列目のAUはオーストリア（Republic of Austria）、BEはベルギー（Kingdom of Belgium）、CZは当時のチェコスロバキア（Czechoslovakia）、EEはエストニア（Republic of Estonia）、FIはフィンランド（Finland）、FRはフランス（French Republic）、DEはドイツ（Federal Republic of Germany：DEはDeutschlandの略）、GRはギリシャ（Greece）、HUはハンガリー（Kingdom of Hungary）、IEはアイルランド（Irish Free State）、ITはイタリア（Kingdom of Italy）、ＮＬはオランダ（Kingdom of Netherland）、PLはポーランド（Second Republic of Poland）、PTはポルトガル（Portuguese Republic）、ROはルーマニア（Kingdom of Romania）、ESはスペイン（Kingdom of Spain）、SEはスエーデン（Kingdom of Sweden）、UKが大英帝国（United Kingdom）である。

このデータを、従来の数値解析的方法で分析することはもちろん可能である。筆者はこのデータでいくつかの数値解析を試みた。行った分析は、総当たり相関分析、多次元尺度構成法、主成分分析、クラスター分析、因子分析、重回帰分析である。（本ブログの「やさしい水産学/QCA参照」その結果、得られた結論は、ここで対象とする18の国は、経済的な尺度（いわゆる豊かさ）で２分することができ、それらの尺度を説明変数として、民主主義の崩壊の程度を説明することは出来る。すなわちLipsetの仮説は支持されるのであるが、チェコ（CZ）、フィンランド（FI）、アイルランド（IE）オーストリア（AU）、ドイツ（GE）、エストニア、（EE）は、中間的な位置にある。このうち、３か国（CZ,FI,IE）は民主主義維持国であり、オーストリア、ドイツ、エストニア（AU,DE,EE）は民主主義崩壊国である。これらの国々の運命を分けたものは、経済的発停では説明できない。戦間期のヨーロッパという時間的空間的に限られた事例では、Lipsetの仮説は全体として支持される。すでに述べたように、Lipsetの仮説を検証するには、戦間期のヨーロッパを離れて、第二次世界大戦後の政治史を分析すべきであるが、当時のデータで、この６か国における民主主義の維持の違いがどのような原因によって生じたのかを論ずることにも、Lipsetの仮説の成り立ちを確認するという、別の意味でぇｓある問題である。このような視点を持って、RのQCA package による。分析事例を見ていくと、QCAの特徴と優位性、分析対象などについて、理解が深まるだろうと思う。

1. QCApacageの導入と立ちあげ

II-1.　導入と起動や

Rの立ち上げ

　Rstudioを立ち上げる。普通は、画面は４分割または３分割されている。左画面袴鈴が上下２つに分かれていない場合は、それぞれの画面の右上に、画面分割を指定する、よくわからない画面の絵があるから、それで、分割を指定する。４分割された画面の内容は、分割の指定に仕方で違ってくっるのだが、標準的には、時計回りに、左上：スクリプトを書く画面。右上：データファイルを読みこんだり、できたデータファイルを整理する画面、右下、スクリプトで実行して作られた、図を見たり、ファイルとうぃて書き出す画面、左下、スクリプトの実行結果が出る画面になっている。

スクリプトのコピー

スクリプトの画面、左上のFileをクリック→New File→Rscriptをクリック→コピーしたスクリプトを張り付ける。

スクリプトに名前を付ける。

Fileをクリック→Save as→file名（QCA）と保存場所を指定する、

QCApackageの立ち上げ

１#初めにRのQCAパッケージを導入

２install.packages("QCA", dependencies = TRUE)

4 #QCAを起動する

5 library(QCA)

一度、２行目のように、packageを導入すれば、毎回導入する必要はない。以後は５行目の、libraryの起動から始める。

II-２．データの読み込み。（7－9行目）

Rstudioにデータを読み込む。

右上のデータを読み込む画面で箒のシンボルマークをクリック→本当に、すべてのデータを消去してよいかと聞いてくる→データ消去される。→Import datasetをクリック→From texst(base)….クリック→ファイルとデータを選ぶ、この場合はLIT→右下のopenクリック→データファイルの読み込み方を指定する画面：Encoding―Automatic, Heading （１列目を列名として使うかという意味）―Yes, Row names(デフォルトはAutomaticだが下に下がって変更)―use first column. 以上を指定して、右下のimportをクリック→見左上のスクリプトの画面の上部に、LIPというファイルができるので、内容を確認後、QCAの画面に戻る

7データ読み込み。

8 LR<-LIP

9 LR

このデータの内容は、QCApackageが内蔵しているLRというデータと同じなのだが、テンプレートなので、読者が自分のデータを導入することを想定してスクリプトを作った。読み込んだデータファイルは以下の通り



QCApackageのLRとの違いはドイツのケースコードをDEからGEに変えたことである。スクリプトの９行目は不要だが、念のため内容を確認した。

III．csQCA

閾値の決定（11行目 から42行目）

III-1. 各項目（列）の取り出し）

　11 #csQCA

12 #すべての項目をthresholdを決めて、DEV, ~DEV(not DEV)のように２分する

13 #データを取り出す。データフレームとして取り出したいので、列番号号で指定する。

14 data<-LR[1]

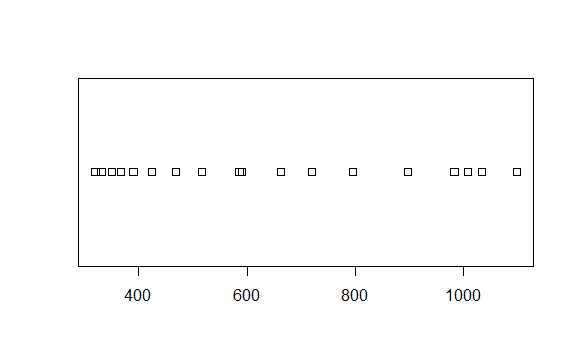
15 data

16 plot(data)

14行目は、データフレームとして１行目のDEVを読み込でいる。行番号で指定するとデータフレームとして読み込むのでIDが読み込めて、クラスター分析等が見やすくなる。

15行目　データの確認

16行目　データ分布の視覚化



貧困側に偏っている。中央部に不連続点がない。

III-2 距離行列を作って、クラスター分析をする。

III-2-1. 距離行列を作る

18 #総当たりの距離行列を作る。

19 edist<-dist(data)

20 edist

1次元のデータだからユークリッド距離もマハラノビス距離も同じ、distで

距離行列を作る。

edist

AU BE CZ EE FI FR GE GR HU IE IT NL PL PT RO ES SE

BE 378

CZ 134 512

EE 252 630 118

FI 130 508 4 122

FR 263 115 397 515 393

GE 75 303 209 327 205 188

GR 330 708 196 78 200 593 405

HU 296 674 162 44 166 559 371 34

IE 58 436 76 194 72 321 133 272 238

IT 203 581 69 49 73 466 278 127 93 145

NL 288 90 422 540 418 25 213 618 584 346 491

PL 370 748 236 118 240 633 445 40 74 312 167 658

PT 400 778 266 148 270 663 475 70 104 342 197 688 30

RO 389 767 255 137 259 652 464 59 93 331 186 677 19 11

ES 353 731 219 101 223 616 428 23 57 295 150 641 17 47 36

SE 177 201 311 429 307 86 102 507 473 235 380 111 547 577 566 530

UK 314 64 448 566 444 51 239 644 610 372 517 26 684 714 703 667 137

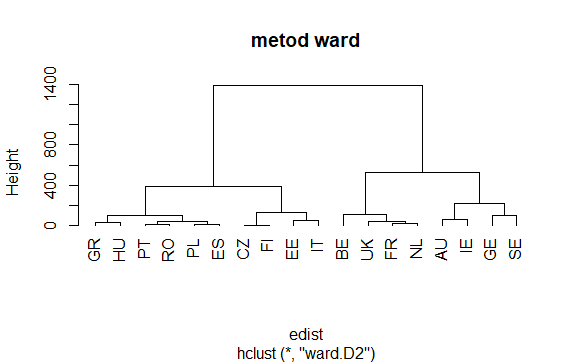
III-2-2 階層的クラスター分析をする。

　21: #階層的クラスター分析（Ward法やってみる。）

22 edc<-hclust(edist,method="ward.D2")

23 edc

24 plot(edc,hang=-1,main="method ward")



25 #樹形図にしたがって２分する

26 result<-cutree(edc,k=2)

27 result

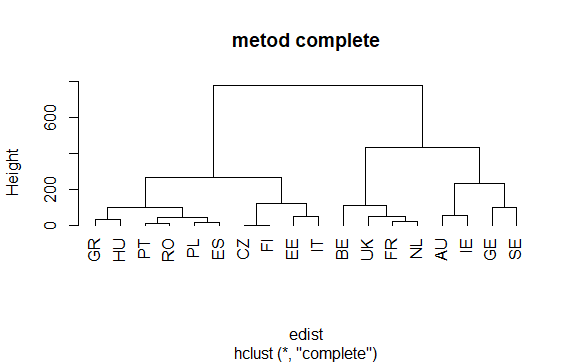
　　　　AU BE CZ EE FI FR GE GR HU IE IT NL PL PT RO ES SE UK

　　　　1 　1 　2 　2 2 1　 1 2　 2 　1　 2 1 2　 2 　2　 2 1 1

　 29 #最長距離法でもやってみる。

30 edc2<-hclust(edist,method="complete")

31 plot(edc2,hang=-1,main="metod complete")



32 #樹形図にしたがって２分する

33 result2<-cutree(edc2,k=2)

34 result2

AU BE CZ EE FI FR GE GR HU IE IT NL PL PT RO ES SE UK

1 1 2 2 2 1 1 2 2 1 2 1 2 2 2 2 1 1

36 #kmeans法によるクラスター分析。

37 km<-kmeans(data,2)

38 km

AU BE CZ EE FI FR GE GR HU IE IT NL PL PT RO ES SE UK

1. 1 2 2 2 1 1 2 2 2 2 1 2 2 2 2 1 1

40 RのQCAパケージにもthresholdを提案する機能があるので、やってみる

41 findTh(LR$DEV)

findTh(LR$DEV)

[1] 626

III-2-3 閾値の選択

これらのデータを参考にして、閾値を決める。3種の、クラスター分析の結果を表に取りまとめると、下記の表になる。



この表は、一人当たりのGNPの低い国から並べるという形で整理した。この表からわかるようにWard 法と距離法による階層的クラスター分析の結果は同じだが、Kmenas法の結果は、わずかに異なっている。だがいずれにしても、アイルランド（IE）が、2つのグループを分ける境界になっていて、階層的クラスター分析では、アイルランド（IE）とそれよりもGNP/CAPの少ないフィンランド（FI）との間、Kmeans法では、アイルランド（IE）と、それよりもGNP/CAPの大きなおオーストリア（AU）の間に、豊かな国と（DEV）と豊かでない国（not DEV, ~DEV）を分ける閾値を設けることが適切だろうという推測ができる。QCAのパッケージも閾値として、626を提案している。これらを、参考にすると、閾値として、620あるいは690を採用することが考えられる。しかし、このデータを使ったQCAの解説本では、閾値値として、550を選択している。QCAパッケージのサンプルデータでも、550が閾値として選択されている。理由として様々なデータを検討した結果と書かれている。聞きたいのは、様々なデータをどのように考えて、選択したのかであるが、書かれていない。こちらで推測するしかない。以下は筆者の推測である。もし、IEとAUの間に、閾値を設けると、それ以上のGNP/CAPがある豊かな国(DEV)は、７か国で、その中に含まれる民主主義崩壊国は、オーストリア（ＡＵ）とドイツ（ＧＥ）の２か国になってしまう。この２つの国は、第二次世界大戦の原因を作った国であり、この閾値を採用すると、豊かな国の中で、例外的な国がなぜ例外なのかを論ずることになりかねないし、さらに、もっとGNP/CAPTが高い国との境界、つまり、ドイツ(GE)とスエーデン(SE)の間の閾値を設定すれば、民主主義を維持した国しか豊かな国に含まれていないのだから、明快に豊かな国は民主主義が維持されるという結論が出るだろう。だがこれはLipsetの仮説の追認であって、検証にはなっていない。分析の目的を考慮すれば、閾値をもっと低く、AU, GEよりもGN/CAPが低いながらも、民主主義を維持した、アイルランド（IE）、フィンランド（FI）、チェコスロバキア（CZ）を含めて富裕国（DEV）とするために、イタリア（IT）とチェコスロバキア（CZ）の間、550とするのは、政治社会学として一つの見識だろう。これを恣意的チューニングだととらえるか、分析の視点をより明確にするための、妥当な操作ととらえるかは、読者の判断であろう。また、こうした以前の判断材料がないとすれば、クラスター分析等の数量的分析に頼らざるを得ないだろう。

42 #すべての項目について、上記の操作を行う。

43 #以上の分析結果と既往の知見から、分析者の判断で、閾値（threshold)を決めて

２分する。

III-3. cs(QCA)の実施

　Scriptの目的はfsQCA(fuzzy set QCA)の実行なのがだ、fsQCAをやる前に、csQCAを行っておくことは、分析の見通しを立てたり、閾値の妥当性を確認するためにも有効である。csQCAをやるためには、すべての項目について、データを、集合に属する（A）と集合の属さない（補集合 not A, ~A）、1-0で分ける。これには、fsQCAで使われる、calibrateという関数を用いる。

III-3-1 csQCAの準備１（データフレーム）

43 #すべての項目について、上記の操作を行う。

44 #以上の分析結果と既往の知見から、分析者の判断で、閾値（threshold)を決めて２分する。

45 #閾値を決めて、TRUE＝１，FALSE=0の表を作る。デフォルトがfuzzyなので、csQCAの場合は、type="crisp", threshold値を入力

46 DEV<-calibrate(LR$DEV, type = "crisp", thresholds = 550)

47 URB<-calibrate(LR$URB, type = "crisp", thresholds = 50)

48 LIT<-calibrate(LR$LIT, type = "crisp", thresholds = 75)

49 IND<-calibrate(LR$IND, type = "crisp", thresholds = 40)

50 STB<-calibrate(LR$STB, type = "crisp", thresholds = 9.5)

51 SURV<-calibrate(LR$SURV, type = "crisp", thresholds = 0)

52 dfc<-data.frame(DEV,URB,LIT,IND,STB,SURV)

53 #項目名（列名）を入力

54rownames(dfc)<-c("AU","BE","CZ","EE","FI","FR","GE","GR","Hu","IE","IT","NL","PL","PT","RO","ES","SE","UK")

55 #確認

56 dfc

46行目から51行目までは、それぞれの項目について、指定された閾値にもとづいて、DEV＝１、~DEV（not DEV）=0の様に振り分けている。関数はcalibrateで、（振り分けるデータ、QCAのタイプの指定、閾値）のように、変数を指定する。Calibrateではタイプのデフォルトがfizzy setなので、csQCAの時は、crispと指定する。52行目は、それぞれの項目で降り分けた１－０の値の列を、並べて、dfcというデータフレームに格納している。54行目はデータフレームの各列名の挿入。56行目は作られてデータフレームの確認。

　　　　　> dfc

　　　　　　 DEV URB LIT IND STB SUR

AU 1 0 1 0 1 0

BE 1 1 1 1 0 1

CZ 1 1 1 0 0 1

EE 0 0 1 0 0 0

FI 1 0 1 0 0 1

FR 1 0 1 0 0 1

GE 1 1 1 1 1 0

GR 0 0 0 0 1 0

Hu 0 0 1 0 1 0

IE 1 0 1 0 0 1

IT 0 0 0 0 0 0

NL 1 1 1 0 0 1

PL 0 0 1 0 1 0

PT 0 0 0 0 1 0

RO 0 0 0 0 0 0

ES 0 0 0 0 1 0

SE 1 0 1 0 0 1

UK 1 1 1 1 0 1

III-3-2 csQCAの準備２（真理表の作成）

　　QCApackaではtruthTabeという関数がで真理表を作る。truthTabeでは（）内に、真理表にするデータフレーム、outcome, conditions, incl.cut, n.cut, pri.cut, exclude, をデータとして入力する。入力するデータフレームは、上記の操作で作ったdfc、outcome＝“SURV”、包含される要因はDEV,URB,LIT, IND,STBだが、デフォルトの設定が表の列の残りのすべてになっているので、この場合、conditionsは入力の必要がない。incl.cut,はoutcomeに占める割合の切り捨て値なのだが、デフォルトで１になっており、csQCAでは入力の必要がない。以下、n.cut、pri.cut,　Exdlude, も入力の必要はない。Completeは真理表の全体が見たいときはTRUEを入力する。空集合（集合に含まれる事例がない）を含まない、省略された真理表を見る場合はFALSEを入力する。show.caseは含まれる事例のケースIDが見たい場合にTRUEを入力する。58行目は省略されていない真理表を表示する。59行目は省略された真理表を指定している。

58 #真理表を作る。

59 truthTable(dfc, outcome = "SURV", complete = TRUE, show.cases = TRUE)

60 truthTable(dfc, outcome = "SURV", conditions = "DEV,URB,LIT,IND,STB", incl.cut = 1, n.cut = 1, pri.cut = 0,

exclude = NULL, complete = FALSE, show.cases=TRUE)

省略された真理表

DEV URB LIT IND STB OUT n incl PRI cases

1 0 0 0 0 0 0 2 0.000 0.000 IT,RO

2 0 0 0 0 1 0 3 0.000 0.000 GR,PT,ES

5 0 0 1 0 0 0 1 0.000 0.000 EE

6 0 0 1 0 1 0 2 0.000 0.000 Hu,PL

21 1 0 1 0 0 1 4 1.000 1.000 FI,FR,IE,SE

22 1 0 1 0 1 0 1 0.000 0.000 AU

29 1 1 1 0 0 1 2 1.000 1.000 CZ,NL

31 1 1 1 1 0 1 2 1.000 1.000 BE,UK

32 1 1 1 1 1 0 1 0.000 0.000 GE

**真理表の読み方**、左の列の数字は、DEV,URB,LIT,IND, STBの組み合わせ（condition）につけられたコード番号。たとえば、コード１は、DEV＝０（FALSE:豊かでない）、URB＝０（FALSE：都市化されていない）、LIT＝０（FALSE:識字率が低い）、IND＝０，（FALSE:工業化されていない）、STAB=0（FALSE:政治が不安定）の組み合わせ。OUTは結果（民主主義が維持された）でこの場合は事実として、民主主主義が維持されなかった国が多いので０、n=2 (そのような国は２か国ある)。incl=0.000 (それらの国のうち、民主主義が維持された国は０)、PRI＝０（csQCAではPRIはinclと等しい。）、cases=IL,RO)、そのような国は（イタリアとルーマニア）である。完全心理表では、Condition コード3のところでｎ=?となっているが、これはそのような事例がない（要素がない：空集合）という意味である。

III-3-3-(1)　csQCAの実施

　真理表を使って、csQCAを実施すればよいだけなのだが、様々な組み合わせがあって、いちいちそれらを検討するのは、面倒で、煩わしい。RのQCAパケージには、QCAの結果として、提案すべき、節約化された条件（集合の組み合わせ）を提案する

62 #csQCAを実施

63 #心理表をTTBCPCに格納

64 TTBCPC<-truthTable(dfc, outcome = "SURV", complete = TRUE, show.cases = TRUE)

65 #最節約解

66 minimize(TTBCPC,include="?", details = TRUE)

67 minimize(TTBCPC, details = TRUE)

68 minimize(TTBCPC,details=TRUE)$PIchart

69 superSubset(dfc, outcome= SURV,conditions="DEV,URB,LIT,IND,STB")

70 SSCP<-superSubset(dfc,outcome= "SURV",incl.cut=0.9, ron.cut=0.6)

結果　最節約解　（66 minimize(TTBCPC,include="?", details = TRUE)）

M1: DEV\*~STB <-> SURV

inclS PRI covS covU cases

------------------------------------------------------------------

1 DEV\*~STB 1.000 1.000 1.000 - FI,FR,IE,SE; CZ,NL; BE,UK

------------------------------------------------------------------

M1 1.000 1.000 1.000

66行の minimize(TTBCPC,include="?", details = TRUE)を実行すれば、上記のような結果が得られる。これは、最終的な節約解の提案である。この例では、M1:として、１つだけの解が提案されている、その内容はDEV\*~STB <-> SURV　すなわち、DEVと~STB(not STB)の積集合が、結果、SURVの必要十分条件（）になっているということである。つまり、豊かで、政治が安定していれば、民主主義が維持され、民主主義が維持されていれば、豊かで政治が安定するということである、注意しなければならないのは、~STB(not STB)の解釈である。STBに用いたのは、政権交代の回数である。だから、STBが高いほど、政治が不安定だと解釈すべきなのである。csQCAの、calibrate関数では、閾値だけを指定して、正負の方向性を指定できないので、このようになる。これを避けるには、raw dataにマイナスをかけておくとか、あらかじめ、何らかの工夫をしてお必要がある。なお、fsQCAでは、方向性も含めてcalibrateできる。

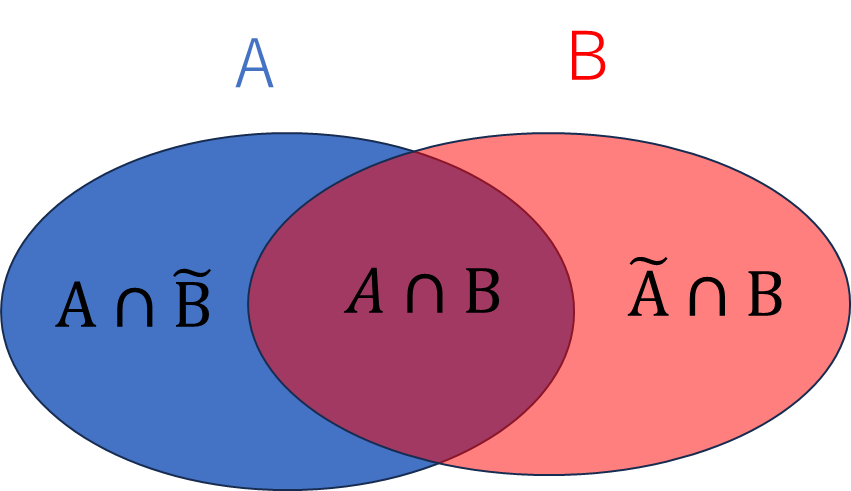
表の値を見ると、inclSとcovSの値が両方１になっている。計算式と使われ方から判断して、inclSは素整合度　(Raw consistency)である。covSは素被覆度(Raw coverage)

である。何故、最後の大文字がＲでなくて、Ｓなのかというと、一番下段が（Ｍ１）を解とするという表の形式になっていて、こちらは解（solution）だから、それに合わせて、Ｓになっている。Consistencyをinclusion（包含）というのは、QCApackage独特の用語の使い方なのだが、これは、実は、深刻な混乱を招く結果になっている。何が何を包含しているのかわからないのである

少し、回り道になるが、重要なことなので、説明しておく。

（重要：RのQCApackageには、致命的なバグがあります。これを知らないと、正しくQCAの結論を導くことができません。どのように間違っているのかを理解して、上手に使うために、基本的なことから説明しますので、必ず読んでください。）

被覆度(coverage)とは、ある集合Xがある集合Yに内包される場合、集合Xが集合Yの中で、占めている割合のことである。だが、現実には、ある集合が完全に他の集合の内包されることはまれだ。一つの集合が、完全にある集合に内包されずに、下図のように部分的に重なり合っていることが多いだろう。このような状態を、集合の式で書き表してみる。集合の書き表し方には、いろいろなやり方があるが、AとBの積集合（A\*B）を、AとBの和集合（A＋B）を、Aの補集合（not A, ~A）をと表すのが最も一般的だと思う。



この図からわかるように、式で表すと次のようになっている。

集合の大きさとは何か（なんで定義するか）という問題があるが、いろいろなやり方で大きさを定義できるだろう。集合に含まれる要素の数もその一つだし、メンバーシップ値の大きさもその一つになるだろう。だがとりあえず、その大きをSと表すと、

となることは、わかるだろう。集合Aの中のBのcoverage(被覆度)とは、BならばAであるといったときに、Aの中でBであるものが占めている割合である。だが同時にこれは、AならばBである（といった時の,Aであること全体に占める、Bであることの割合（consistency:整合度：AならばBということの正しさ）でもある。

このことは、との関係についてもいえる。

BならばA（だと主張したときに、その主張（あるいは予言）の正確性、信頼性を知りたくなるのは当然のことだろう。その場合、Bであったことの中で、実際にAであったことがどのくらいなのかを調べるだろう。それが、の意味である。の方は、実際にAであったことの中で、Bであったことの割合だから、当然、その予測からは予測できない理由でAであったことはたくさんあるから、BからAを予測することの、全体に対する重要性のようなものを表している。その意味で、Coverage（被覆度）というのである。予言者がいて、その予言者が指摘する人は、３か月以内に死ぬと予言したときに、その予言の正しさを評価するには、預言者が死ぬと予測した人（B）の中で、実際に死んだ人の割を調べる。３か月以内に死んだ人（B）の中で、予言者に死ぬ宣告された人の割合を調べる「馬鹿」はいない。予言されたら死ぬ（予言される→死ぬ）ということの正しさを調べているのだから、

=

の値を調べるに決まっている。

紛らわしいのだが、因果関係の矢印の向きで判断する。

Coverageの時は、矢印の終点の方（結果）で割る。Consistencyの時は、矢印の起点の方（原因）で割ると覚えればよい。

関数mobilizeではinclはconsistency、covはcoverageの意味でつかわれている。これは正しい。だが、関数superSubsetでは、この関係が逆になっている。詳細は、superSubsetのところで、説明する。

それはそれとして、計算のプロセスを見ないで、いきなり、最終的な節約解が出てきても、すぐに納得がいかないだろう。そこで、その結果に至るプロセスをみておく必要がある、67行目 minimize(TTBCPC, details = TRUE)を実行すると、下記の初期解が出てくる。

結果　初期解　66 minimize(TTBCPC, details = TRUE)

M1: DEV\*URB\*LIT\*~STB + DEV\*LIT\*~IND\*~STB <-> SURV

inclS PRI covS covU cases

--------------------------------------------------------------------

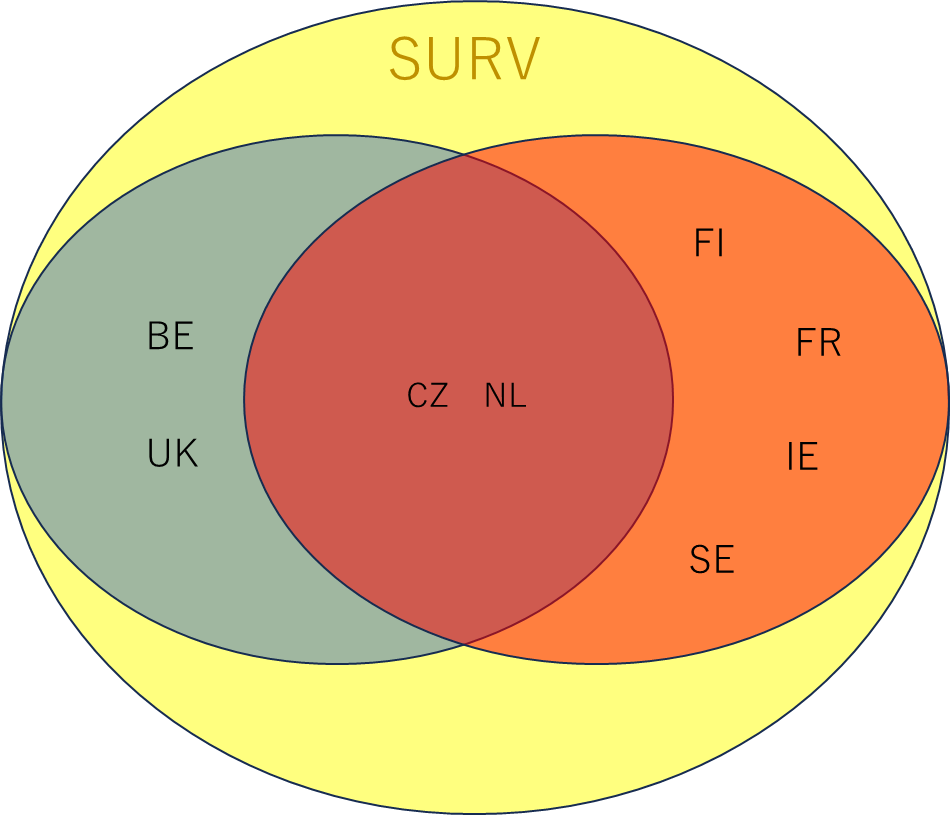
1 DEV\*URB\*LIT\*~STB 1.000 1.000 0.500 0.250 CZ,NL; BE,UK

2 DEV\*LIT\*~IND\*~STB 1.000 1.000 0.750 0.500 FI,FR,IE,SE; CZ,NL

--------------------------------------------------------------------

M1 1.000 1.000 1.000

この結果の読み方だが、「M1: DEV\*URB\*LIT\*~STB + DEV\*LIT\*~IND\*~STB <-> SURVのところは、初期解の提案で、DEV、URB、LIT、notSTBの積集合と、DEV、LIT、IND、not STBの積集合の、和集合がSURVと必要十分条件（）になっている。」と読む。確かに、真理表を見て、民主主義維持国のみが含まれるのは、condition番号の21,29,31しかなくて、しかも、民主主義維持国はそれ以外にないので、それをそのまま、まとめればこのような初期解になる。しかし、もっと、節約的に簡略化しないと、言語化して解釈できない。DEV\*URB\*LIT\*~STBのcovSが0.500になっている。これは、DEV\*URB\*LIT\*~STBSURVとしたときに、SURV（民主主義維持国）は8か国あるのに対し、その中でDEV\*URB\*LIT\*~STBにに属する国は4か国だから、coverageの時は、矢印の終点で、積集合を割るというルールで、＝０.500と計算している。covUはunique coverage(固有被覆度)である。この場合、DEV\*URB\*LIT\*~STB とDEV\*LIT\*~IND\*~STB の和集合が、SURVを構成する要素となっていて、２つの集合は、下図のように、重なり合って積集合となっている部分と、重なり合っていない部分がある、Unique coverageとは、



それぞれの集合の中で重なり合っていない部分が独自に、結果に占めている大きさの割合のことである。この場合、DEV\*URB\*LIT\*~STBについていえば、BE,UK,CZ,NLの４か国が、DEV\*URB\*LIT\*~STBの要素であるが、積集合に含まれていないのは、BE,UKの２か国だから、となる。

結果　PIchart　68 minimize(TTBCPC,details=TRUE)$PIchart

> minimize(TTBCPC,details=TRUE)$PIchart

　　　　　　 21 29 31

DEV\*URB\*LIT\*~STB - x x

DEV\*LIT\*~IND\*~STB x x -

68 minimize(TTBCPC,details=TRUE)$PIchartを実行すると、上記の表が示される。これは、初期解のconditionの成り立ち方を表している。

　　　結果　superSubset 69 superSubset(dfc, outcome= SURV,conditions="DEV,URB,LIT,IND,STB")

superSubset(dfc, outcome= SURV,conditions="DEV,URB,LIT,IND,STB")

inclN RoN covN

------------------------------------

1 DEV 1.000 0.800 0.800

2 LIT 1.000 0.500 0.615

3 ~STB 1.000 0.700 0.727

4 DEV\*LIT 1.000 0.800 0.800

5 DEV\*~STB 1.000 1.000 1.000

6 LIT\*~STB 1.000 0.900 0.889

7 DEV\*LIT\*~STB 1.000 1.000 1.000

8 URB + ~IND 1.000 0.000 0.444

------------------------------------

superSubset(dfc, outcome= SURV,conditions="DEV,URB,LIT,IND,STB")を実行すると、上記の表が提示される。これはなかなか便利な機能で、節約解の候補を示している。だが、この表には致命的な誤りがある。Minimizeのところで説明したように、inclはconsistencyの意味なのだが、この表では、DEVのincl値は1.000となっている。ここは、という解釈についてのconsistencyなのだから、同じ、inclsionでも、起点の値で、積集合の値を割らなくてはいけない。つまり、DEVの要素は全部で１０ケースあり、そのうちSURVの要素となっているのは８ケースなので、

同様に、という解釈についてのcoverageなのだから、終点の値で、積集合を割らなくてはいけない。SURVの要素は８個なのだから、

となる。つまり、superSubsetの結果の表の、inclとcovの値は、反対になっているのである。これは、包含関係の矢印を反対に取り違えたためだと考えられる。念のため、RのQCApackageの解説の文書を読んだところ、inclはconsistencyだという説明があり、その説明の因果関係の矢印も、この文章の説明の通り、正しい方向を向いていた。したがって、superSubset関数を作った人が間違えたのだと思う。単純な間違いのように思うかもしれないが、実は、深刻な内容を含んでいる、superSubsetの表は、incl(実はcoverage)の高いものが選ばれているのである。より重要なのはconsistencyだから、consistencyの高い解の中で、coverageを参考に、より妥当な解を選ばなくてはならない。superSubsetの結果表示の仕方では、重要な情報が隠されてしまう可能性がある、

次に、RoNの説明をする。csQCAの場合、RoNは、conditionが結果OUTに内包されないケースの数に対する、conditionにされないケース数の割合である。

III-3-3-(2)　詳細な確認（csQCA）

　上記のバグもあり、念のため、詳細な確認作業をしたほうが良いだろう。ここでかかわっているのは、DEV,LIT,STBのようだから、この３つに絞って検討する。

72 #DEV,LIT,STBについて確認

73 truthTable(dfc, outcome = "SURV", complete = TRUE, conditions = "DEV,LIT,STB", show.cases = TRUE)

75 TTBCPC1<-truthTable(dfc, outcome = "SURV", conditions = "DEV,LIT,STB", complete = TRUE, show.cases = TRUE)

77 minimize(TTBCPC1,include="?", details = TRUE)

78 minimize(TTBCPC1, details = TRUE)

79 minimize(TTBCPC1,details=TRUE)$PIchart

80 superSubset(dfc, outcome= SURV,conditions="DEV,LIT,STB")

81 SSCP1<-superSubset(dfc,outcome= "SURV",incl.cut=0.9, ron.cut=0.6)

　真理表

　　　　　　　　DEV LIT STB OUT n incl PRI cases

1 0 0 0 0 2 0.000 0.000 IT,RO

2 0 0 1 0 3 0.000 0.000 GR,PT,ES

3 0 1 0 0 1 0.000 0.000 EE

4 0 1 1 0 2 0.000 0.000 Hu,PL

5 1 0 0 ? 0 - -

6 1 0 1 ? 0 - -

7 1 1 0 1 8 1.000 1.000 BE,CZ,FI,FR,IE,NL,SE,UK

8 1 1 1 0 2 0.000 0.000 AU,GE

結果　最節約解　77 minimize(TTBCPC1,include="?", details = TRUE)

minimize(TTBCPC1,include="?", details = TRUE)

M1: DEV\*~STB <-> SURV

　　　　　 inclS PRI covS covU cases

----------------------------------------------------------------

1 DEV\*~STB 1.000 1.000 1.000 - BE,CZ,FI,FR,IE,NL,SE,UK

---------------------------------------------------------------

　　　　　 M1 1.000 1.000 1.000

結果　初期解

> minimize(TTBCPC1, details = TRUE)

M1: DEV\*LIT\*~STB <-> SURV

inclS PRI covS covU cases

-----------------------------------------------------------------

1 DEV\*LIT\*~STB 1.000 1.000 1.000 - BE,CZ,FI,FR,IE,NL,SE,UK

----------------------------------------------------------------

　　　　M1 1.000 1.000 1.000

　　　結果PIchart

minimize(TTBCPC1,details=TRUE)$PIchart

7

DEV\*LIT\*~STB x

結果　superSubset

superSubset(dfc, outcome= SURV,conditions="DEV,LIT,STB")

inclN RoN covN

------------------------------------

1 DEV 1.000 0.800 0.800

2 LIT 1.000 0.500 0.615

3 ~STB 1.000 0.700 0.727

4 DEV\*LIT 1.000 0.800 0.800

5 DEV\*~STB 1.000 1.000 1.000

6 LIT\*~STB 1.000 0.900 0.889

7 DEV\*LIT\*~STB 1.000 1.000 1.000

------------------------------------

5項目で分析した場合と、結果は大きく変わらない。しかし、初期解として提案されている、M1: DEV\*LIT\*~STB <-> SURVは注目に値する。Consistency、coverageとも最節約解、M1: DEV\*~STB <-> SURVと違わないうえに、どちらもcondition 7の要因の重なりになっている。DEV\*~STB <-> SURVでは、 豊かで、政治が安定していれば、民衆主義は維持されるとなるが、DEV\*LIT\*~STB <-> SURVでは、豊かで、教育レベルが高く、政治が安定していれば、民主主義が維持されるという、2つの結論があり得ることになる。論理学的のルールに従えば、同じ評価であれば、より節約的な表現を用いるべきであり、DEV\*~STB <-> SURVを採用されるのだが、そもそも、豊かさと、教育レベルの高さと、政治の安定は、それぞれが原因でもあり結果でもあるという、複雑な関係を持っている。そのことを考えると、単に制節約的に最終解を決めればよいというものでもないようにも思う。それを判断することは、筆者にはできない。実際に分析に当たる、政治社会学の研究者が、知識・経験と見識によって、判断すべきものであろう。

III-4. 結果の補集合についての分析

通常の数値分析では、あることが送る原因の裏返しが、あることが起こらないことの原因であるというように考えるのが自然なのだが、よく考えると、社会現象では、しばしば、あることが起こらないことが、あることが起こることの原因の裏返しでないことがある。たとえば、A君の成績が上がったのは、よく勉強したからかもしれないが、B君の成績が下がったのは、勉強しなかったからではなくて、体調が悪かったためかもしれない。こういう非対称性は社会科学では重要な視点だ。そいうリアリティーを持った分析がQCAの魅力である。QCAの特徴を生かすには、あることが起こる原因の組み合わせに続いて、あることが起こったという集合に対する補集合、つまりあることが起こらなかったことについても、分析しておくほうが良い。サンプル・スクリプトでは、outcome=~SURVについての分析についても、scriptを書いておいた。

84 #~SURV(not SURV)となるケースについての真理表を作る。

85 truthTable(dfc, outcome = ~SURV, complete = TRUE, show.cases = TRUE)

86 TTBCNC<-truthTable(dfc, outcome = ~SURV, complete = TRUE, show.cases = TRUE)

87 truthTable(dfc, outcome = ~SURV, conditions = "DEV,URB,LIT,IND,STB", incl.cut = 1, n.cut = 1, pri.cut = 0, exclude = NULL, complete = FALSE, show.cases=TRUE)

90 minimize(TTBCNC,include="?", details = TRUE)

91 minimize(TTBCNC, details = TRUE)

92 minimize(TTBCNC,details=TRUE)$PIchart

93 superSubset(dfc,outcome=~SURV,incl.cut=0.9, ron.cut=0.6)

94 SSCN<-superSubset(dfc,outcome=~SURV,incl.cut=0.9, ron.cut=0.6)

結果　真理表　（省略された真理表）　87 truthTable(dfc, outcome = ~SURV, conditions = "DEV,URB,LIT,IND,STB", incl.cut = 1, n.cut = 1, pri.cut = 0, exclude = NULL, complete = FALSE, show.cases=TRUE)

DEV URB LIT IND STB OUT n incl PRI cases

1 0 0 0 0 0 1 2 1.000 1.000 IT,RO

2 0 0 0 0 1 1 3 1.000 1.000 GR,PT,ES

5 0 0 1 0 0 1 1 1.000 1.000 EE

6 0 0 1 0 1 1 2 1.000 1.000 Hu,PL

21 1 0 1 0 0 0 4 0.000 0.000 FI,FR,IE,SE

22 1 0 1 0 1 1 1 1.000 1.000 AU

29 1 1 1 0 0 0 2 0.000 0.000 CZ,NL

31 1 1 1 1 0 0 2 0.000 0.000 BE,UK

32 1 1 1 1 1 1 1 1.000 1.000 GE

結果　最節約解の提案

minimize(TTBCNC,include="?", details = TRUE)

M1: ~DEV + STB <-> ~SURV

inclS PRI covS covU cases

---------------------------------------------------------------

1 ~DEV 1.000 1.000 0.800 0.300 IT,RO; GR,PT,ES; EE; Hu,PL

2 STB 1.000 1.000 0.700 0.200 GR,PT,ES; Hu,PL; AU; GE

---------------------------------------------------------------

M1 1.000 1.000 1.000

結果　初期解

> minimize(TTBCNC, details = TRUE)

M1: ~DEV\*~URB\*~IND + ~URB\*LIT\*~IND\*STB + DEV\*URB\*LIT\*IND\*STB <-> ~SURV

inclS PRI covS covU cases

----------------------------------------------------------------------

1 ~DEV\*~URB\*~IND 1.000 1.000 0.800 0.600 IT,RO; GR,PT,ES; EE; Hu,PL

2 ~URB\*LIT\*~IND\*STB 1.000 1.000 0.300 0.100 Hu,PL; AU

3 DEV\*URB\*LIT\*IND\*STB 1.000 1.000 0.100 0.100 GE

----------------------------------------------------------------------

　　M1 1.000 1.000 1.000

結果　PIchart

minimize(TTBCNC,details=TRUE)$PIchart

1 2 5 6 22 32

~DEV\*~URB\*~IND x x x x - -

~URB\*LIT\*~IND\*STB - - - x x -

DEV\*URB\*LIT\*IND\*STB - - - - - x

結果　 superSubset

superSubset(dfc,outcome=~SURV,incl.cut=0.9, ron.cut=0.6)

inclN RoN covN

----------------------------------

1 ~DEV + IND 0.900 0.778 0.818

2 ~DEV + STB 1.000 1.000 1.000

3 ~LIT + STB 0.900 1.000 1.000

----------------------------------

最節約解は、和集合になっていて、~DEV + STB <-> ~SURV、すなわち　「貧しいか、政治が不安定であれば、民主主義が崩壊する。」となる。superSubsetの結果を見ると、３つの組み合わせが提案されている、このうち、３の集合は、２の集合と同様に、covN（実際にはconsistency）が1.000である。２に比べて、coverage(表ではinclN)の値がわずかに低いが、この解（教育レベルが低いか、政治が不安定ならば、民主主義が崩壊する）も、また、含蓄のある解のような気がする。詳細な議論は、政治学の研究者に任せるしかないが、重要なことは、この分析では、民主主義が維持される条件が、豊かで、政治が安定していることであり、それらが同時に満たされないと（すなわち、貧しいか、政治が不安定であるか）、民主主主義が維持されないという、表裏をなす、対照的な結果が得られたということである。

IV-6.fsQCAの実行

IV-1．メンバーシップ値の決定。

fsQCAは、ファジー理論を応用したQCAで、まず、それぞれのケースが、ある条件の含まれる度合い（程度）を０から１の値で表現する。これをメンバーシップ値という。ファジー理論では条件の重なり合い（積集合）のメンバーシップ値は元の条件（集合の）メンバーシップ値の内の最小値。条件の和（同タラの受験を満たせばよい（和集合）のメンバーシップ値は元の集合のメンバーシップ値の最大値になる。

fsQCAでは、初めに各ケースのメンバーシップ値を項目ごとに決めなくてはならない。メンバーシップ値は、データの分布をみて、分析者の判断で決めればよい。例えば、csQCAで行ったクラスター分析データ分布の形を見て、以下のように、決めてもよい。

104 recode(LR$DEV, cuts = "370, 450, 550, 600,750,900,1050", values = "0, 0.2, 0.4, 0.45, 0.55, 0.6, 0.8, 1")

元データLR$DEV

[1] 720 1098 586 468 590 983 795 390 424 662 517 1008 350 320

[15] 331 367 897 1034

結果、

[1] 0.55 1.00 0.45 0.40 0.45 0.80 0.60 0.20 0.20 0.55 0.40 0.80 0.00 0.00

[15] 0.00 0.00 0.60 0.80

この結果は、わかりやすくて、操作も簡単だが、項目数が多い場合すべての項目にいついて、妥当なメンバーシップ値を決めるのは知識量が追い付かないだろう。QCApackage、は統計的な確率曲線に当てはめて、メンバーシップ値を決める方法を用意している。

109 #操作上簡単なのは、既往の確率密度曲線に当てははめる方法である。

110 #QCApackageはcalibrateという関数を用意している。

111 #calibrate(データセット、type="",threshold="e=,C=,i=")と入力する。

112 #デフォルトはfuzzy setなので、typeの指定は不要

113 DEV<-calibrate(LR$DEV,thresholds = "e=400,c=550,i=900")

114 URB<-calibrate(LR$URB, thresholds = "e=25,c=50,i=65")

115 LIT<-calibrate(LR$LIT, thresholds = "e=50,c=75,i=95")

116 IND<-calibrate(LR$IND, thresholds = "e=30, c=40,i=50")

117 STB<-calibrate(LR$STB, thresholds = "e=15, c=9.5, i=5")

118 SURV<-calibrate(LR$SURV, thresholds = "e=-9, c=0, i=10")

120 dff<-data.frame(DEV,URB,LIT,IND,STB,SURV)

122 rownames(dff)<-c("AU","BE","CZ","EE","FI","FR","GE","GR","Hu","IE","IT","NL","PL","PT","RO","ES","SE","UK")

124 dff

結果

DEV URB LIT IND STB SURV

AU 0.80692563 0.12399751 0.96726744 0.1252825070 0.433477656 0.05000000

BE 0.99014771 0.88706515 0.94563344 0.9321725332 0.973370852 0.95000000

CZ 0.57514068 0.97656070 0.95593140 0.3174416200 0.908055403 0.88706515

EE 0.16664046 0.07363025 0.95000000 0.0004731895 0.908055403 0.12314714

FI 0.58334186 0.03564771 0.97202442 0.0049668503 0.581068173 0.76454761

FR 0.97448622 0.03254662 0.95775497 0.1778315906 0.950000000 0.95000000

GE 0.88706515 0.78175508 0.96726744 0.5294104003 0.309378000 0.05000000

GR 0.04145790 0.09744022 0.13459988 0.0292019668 0.433477656 0.06803461

Hu 0.07774921 0.16609597 0.81339450 0.0044174400 0.133110271 0.41893183

IE 0.71954796 0.05000000 0.95000000 0.0005482014 0.950000000 0.91337445

IT 0.34349076 0.10059213 0.41543182 0.0446929713 0.581068173 0.05000000

NL 0.97922495 0.99650640 0.97505515 0.4486539613 0.992662542 0.95000000

PL 0.01934250 0.17783159 0.56947799 0.0002075389 0.002115061 0.12314714

PT 0.01082721 0.01651422 0.01264498 0.0068536146 0.006145632 0.05000000

RO 0.01340171 0.03524504 0.17441369 0.0002785762 0.836962555 0.21271459

ES 0.02679927 0.30482137 0.09238257 0.0137966533 0.207776370 0.06803461

SE 0.94878749 0.13187961 0.97505515 0.0938745867 0.908055403 0.95000000

UK 0.98378370 0.99108533 0.97505515 0.9485827149 0.973370852 0.95000000

>

113行目から118行目で、個々の項目のメンバーシップ値を計算している。120行目でそれらの値を、データフレームとして表にまとめ、122行目で、各ケースのIDを列名として書き加えている。124行目は内容の確認

IV-2. 真理表を作る。

126 #fsQCAの実行

127 #真理表を作る。

128 truthTable(dff, outcome = SURV, complete = TRUE, show.cases = TRUE)

129 #Complete＝FALSEにすれば、要約した真理表が出力される。

130 truthTable(dff, outcome = "SURV", exclude = NULL, complete = FALSE, show.cases=TRUE)

結果　省略された真理表

> truthTable(dff, outcome = "SURV", exclude = NULL, complete = FALSE, inxl.cut=1, show.cases=TRUE)

DEV URB LIT IND STB OUT n incl PRI cases

1 0 0 0 0 0 0 3 0.267 0.000 GR,PT,ES

2 0 0 0 0 1 0 2 0.335 0.000 IT,RO

5 0 0 1 0 0 0 2 0.528 0.108 Hu,PL

6 0 0 1 0 1 0 1 0.529 0.215 EE

21 1 0 1 0 0 0 1 0.530 0.187 AU

22 1 0 1 0 1 0 4 0.799 0.744 FI,FR,IE,SE

30 1 1 1 0 1 0 2 0.831 0.735 CZ,NL

31 1 1 1 1 0 0 1 0.466 0.070 GE

32 1 1 1 1 1 0 2 0.898 0.877 BE,UK

fsQCAにおけるincl値について、説明する、表のcondition 1では、DEV＊URB＊LIT＊IND＊STBの積集合とSURBの積集合のメンバーシップ値をDEV＊URB＊LIT＊IND＊STBの積集合のメンバーシップ値の合計値で割っている。数式で表すと、

となる。PRI値は、論理矛盾するメンバーシップ値を除いた、inclの値である。論理矛盾するとは、相補的な集合が積集合を持つ（ありえないこと）ということである。メンバーシップ値はそういうことを考慮しないで作っているから、集合と補集合の積集合が、メンバーシップ値を持ってしまうことがある、その部分を取り除いいたincl値という意味で、おそらく、Proportional reduction in inconsistencyだと思われる。この部分をRとすると、

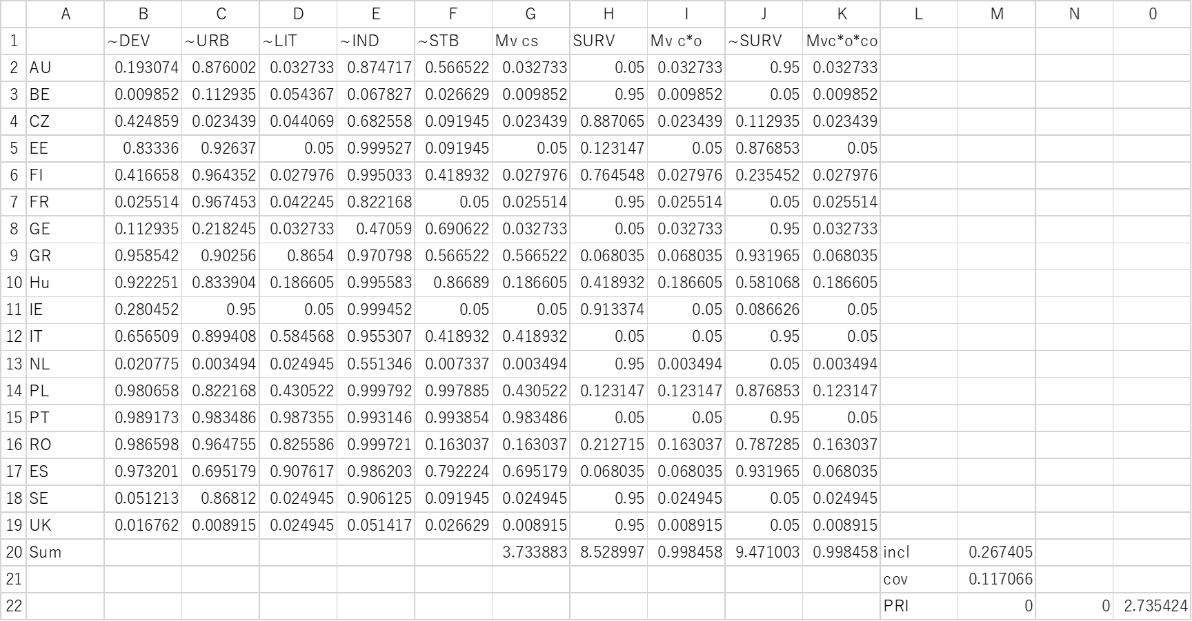
　　　　　R=

である。したがって、

　　　PRI=

となる。

エクセル・シートを使って具体的な計算を表に示す。



G列は、のメンバーシップ値で、BからFまでの列の最小値。たとえば、condition 番号１について、G2（0.032733）はB2,C2,D2,E2,F2の最小値D2の値（0.032733）、G20はその合計値（3.733883）Hの列は、SURVのメンバーシップ値で、H20はその合計（8.528997）、I列は、のメンバーシップ値で、G列H列の最小値たとえば、I２の値は、G2（0.032733）とH2(0.05)の小さい方の値でI2=0.032733、I20その合計で、I20=0.998458、Ｊ列は、 (not SURV)のメンバーシップ値、Ｋ列は、のメンバーシップ値で、Ｉ列とＪあ列の小さい方。K20がその合計で数式で表したＲの値になる。

これらの表の行列の記号であらわすと、

ところで、この真理表で、OUTの列を見ると、すべてが、０になっている。ある一定以上のincl値（consistency）をとる、conditionの組み合わせがないことになる。これは、truthTable関数のincl.cutのデフォルト値が１だからである。fsQCAの場合、incl値（consistency）が１になることはほとんどない。そのまま、解を求めると、エラーが出る。だから、出てきた、真理表のincl値を参考にincl.cutの値を入れる必要がある（この場合は、condition番号22, 30, 32のincl値を参考に、例えば、incl.cut=0.75という制約を入れる。）。ここでは、この状態で、実際に実行し、エラーが出たところで、修正する。

IV-3. fsQCAの実行

132 #真理表をTTBFPCに格納する

133 TTBFPC<-truthTable(dff, outcome = SURV,complete = FALSE, show.cases = TRUE)

134 #最節約解候補を提示させる。

135 minimize(TTBFPC,inclde="?")

結果　135 minimize(TTBFPC,inclde="?",details=TRUE)の実行

> minimize(TTBFPC,inclde="?",details=TRUE))

 Show Traceback

 Rerun with Debug

エラー: None of the values in OUT is explained. Please check the truth

table.

このエラーは、真理表Oとの条件を満たす値がないので、何も説明できないと言っている。デフォルトincl.cut=1となっているので、その条件を満たすものがないのである。149 truthTable(dff, outcome = SURV,complete = FALSE, ,incl.cut=0.75, show.cases = TRUE

のように、incl.cut=0.75を挿入して、真理表を作り直す

136 #真理表を作り直す。

137 truthTable(dff, outcome = SURV,complete = FALSE, ,incl.cut=0.75, cov.cut=0.9, show.cases = TRUE)

139 TTBFPC<-truthTable(dff, outcome = SURV,complete = FALSE, ,incl.cut=0.75, cov.cut=0.5,show.cases = TRUE)

141 minimize(TTBFPC,inclde="?",details=TRUE)

142 minimize(TTBFPC,details=TRUE)

143 minimize(TTBFPC,include="?")$PIchart

144 superSubset(dff,outcome=SURV,incl.cut=0.75, ron.cut=0.6)

145 SSFP<-superSubset(dff,outcome=SURV,incl.cut=0.75, ron.cut=0.6)

　　　結果　修正された真理

#真理表を作り直す。

> truthTable(dff, outcome = SURV,complete = FALSE, ,incl.cut=0.75, show.cases = TRUE)

OUT: output value

n: number of cases in configuration

incl: sufficiency inclusion score

PRI: proportional reduction in inconsistency

DEV URB LIT IND STB OUT n incl PRI cases

1 0 0 0 0 0 0 3 0.267 0.000 GR,PT,ES

2 0 0 0 0 1 0 2 0.335 0.000 IT,RO

5 0 0 1 0 0 0 2 0.528 0.108 Hu,PL

6 0 0 1 0 1 0 1 0.529 0.215 EE

21 1 0 1 0 0 0 1 0.530 0.187 AU

22 1 0 1 0 1 1 4 0.799 0.744 FI,FR,IE,SE

30 1 1 1 0 1 1 2 0.831 0.735 CZ,NL

31 1 1 1 1 0 0 1 0.466 0.070 GE

32 1 1 1 1 1 1 2 0.898 0.877 BE,UK

Condition 番号22, 30,32のOUTの値が１になっている。

結果　141 minimize(TTBFPC,inclde="?",details=TRUE)を実行

minimize(TTBFPC,details=TRUE)

M1: DEV\*URB\*LIT\*STB + DEV\*LIT\*~IND\*STB -> SURV

inclS PRI covS covU cases

-------------------------------------------------------------------

1 DEV\*URB\*LIT\*STB 0.903 0.880 0.469 0.248 CZ,NL; BE,UK

2 DEV\*LIT\*~IND\*STB 0.828 0.786 0.553 0.332 FI,FR,IE,SE; CZ,NL

-------------------------------------------------------------------

M1 0.869 0.847 0.801

最節約解として、M1: DEV\*URB\*LIT\*STB + DEV\*LIT\*~IND\*STB -> SURVという、複雑な和集合が提案される、これを、言語化すると「豊かで、都市化されて、教育レベルが高く、政治が安定しているか、豊かで、教育レベルが高く、工業化されていなくて、政治が安定していれば、民主主義が維持される。」なる。確かに、実際に民主主義が維持された8か国が含まれているし、incl値、PRI値、covS値も、0.869, 0.847, 0.801と高い。しかし、これでは、言語的な解釈が複雑すぎて、何を結論として述べるべきかわからない。和集合を形成している２つの積集合に着目して、DEV、LIT、STBの組み合わせで、説明することを考えるべきかもしれない。

　　結果　初期解の提示　152 minimize(TTBFPC,details=TRUE)を実行

> minimize(TTBFPC,details=TRUE)

M1: DEV\*URB\*LIT\*STB + DEV\*LIT\*~IND\*STB -> SURV

inclS PRI covS covU cases

-------------------------------------------------------------------

1 DEV\*URB\*LIT\*STB 0.903 0.880 0.469 0.248 CZ,NL; BE,UK

2 DEV\*LIT\*~IND\*STB 0.828 0.786 0.553 0.332 FI,FR,IE,SE; CZ,NL

-------------------------------------------------------------------

M1 0.869 0.847 0.801

最節約解の提案と内容は変わらない。つまり、最節約解はたまたま、incl値等が最も高かったので、選択されたと思われる。

　　　結果　PIchart 153 minimize(TTBFPC,include="?")$PIchartを実行

> minimize(TTBFPC,include="?")$PIchart

22 30 32

DEV\*STB x x x

URB\*~IND - x -

URB\*STB - x x

I IND\*STB - - x

Chartをよく見ると、DEV\*STBという集合はcondition22,30,32に共通して含まれている。

　　　　結果　superSubsetによる、解の提示

154 superSubset(dff,outcome=SURV,incl.cut=0.75, ron.cut=0.6)を実行

> superSubset(dff,outcome=SURV,incl.cut=0.75, ron.cut=0.6)

　　　　　　 inclN RoN covN

-----------------------------------

1 DEV 0.831 0.811 0.775

2 STB 0.921 0.681 0.708

3 DEV\*LIT 0.831 0.818 0.781

4 DEV\*STB 0.824 0.904 0.869

5 LIT\*STB 0.916 0.805 0.797

6 DEV\*LIT\*STB 0.823 0.906 0.872

-----------------------------------

superSubsetでは、inclNがcoverageでcovNがconsistencyなので、注意が必要だが、DEV\*STB→SURVのconsistencyは0.869で、minimizeが提案する、M1: DEV\*URB\*LIT\*STB + DEV\*LIT\*~IND\*STB -> SURVと変わらない。また、coverageは、DEV\*STB→SURVの方が高い。さらに、DEV\*LIT\*STB→SURVはわずかだが、さらに、coverageが高い。以上のことから、DEV\*STB→SURVまたは、DEV\*LIT\*STB→SURVを最終的な節約解とすべきだと結論できる。しかし、DEV, LIT, STBをconditionとする分析を確認のために行ってみるべきである。

IV-3 (補足　詳細確認

147 #３条件（DEV,（DEV,LIT,STB)で、真理表を作り直す。

148 truthTable(dff,outcome=SURV,condition="DEV,LIT,STB",incl.cut=c(0.75,0.5), show.cases=TRUE)

150 TTBF<-truthTable(dff,outcome=SURV,conditions="DEV,LIT,STB",incl.cut=0.75, show.cases=TRUE)

152 minimize(TTBF,incllude="?",details=TRUE)

153 minimize(TTBF,include="?")$PIchart

154 minimize(TTBF,details=TRUE)

155 superSubset(dff,outcome=SURV,conditions ="DEV,LIT,STB",incl.cut=0.75, ron.cut=0.6)

156 SSFP1<-superSubset(dff,outcome=SURV,conditions ="DEV,LIT,STB",incl.cut=0.75, ron.cut=0.6)

結果　真理表

truthTable(dff,outcome=SURV,condition="DEV,LIT,STB",incl.cut=c(0.75,0.5), show.cases=TRUE)

OUT: output value

n: number of cases in configuration

incl: sufficiency inclusion score

PRI: proportional reduction in inconsistency

DEV LIT STB OUT n incl PRI cases

1 0 0 0 0 3 0.267 0.000 GR,PT,ES

2 0 0 1 0 2 0.347 0.000 IT,RO

3 0 1 0 C 2 0.540 0.108 Hu,PL

4 0 1 1 C 1 0.590 0.334 EE

7 1 1 0 0 2 0.452 0.134 AU,GE

8 1 1 1 1 8 0.872 0.851 BE,CZ,FI,FR,IE,NL,SE,UK

民主主義維持国は、condition 8にまとまっている。

結果　最節約解　> minimize(TTBF,incllude="?",details=TRUE)

M1: DEV\*LIT\*STB -> SURV

inclS PRI covS covU cases

-------------------------------------------------------------------

1 DEV\*LIT\*STB 0.872 0.851 0.823 - BE,CZ,FI,FR,IE,NL,SE,UK

-------------------------------------------------------------------

M1 0.872 0.851 0.823

　　結果　PIchart

minimize(TTBF,include="?")$PIchart

8

DEV\*STB x

結果　初期解

minimize(TTBF,details=TRUE)

M1: DEV\*LIT\*STB -> SURV

　　　 inclS PRI covS covU cases

-------------------------------------------------------------------

1 DEV\*LIT\*STB 0.872 0.851 0.823 - BE,CZ,FI,FR,IE,NL,SE,UK

-------------------------------------------------------------------

M1 0.872 0.851 0.823

　　結果　superSubset

superSubset(dff,outcome=SURV,conditions ="DEV,LIT,STB",incl.cut=0.75, ron.cut=0.6)

inclN RoN covN

-----------------------------------

1 DEV 0.831 0.811 0.775

2 STB 0.921 0.681 0.708

3 DEV\*LIT 0.831 0.818 0.781

4 DEV\*STB 0.824 0.904 0.869

5 LIT\*STB 0.916 0.805 0.797

6 DEV\*LIT\*STB 0.823 0.906 0.872

-----------------------------------

最節約解として、M1: DEV\*LIT\*STB -> SURVが提案されていることを除くと、DEV, URV, LIT、IND, STBをconditionsとした結果と、内容的に同じで、: DEV\*LIT\*STB -> SURV、DEV\*STB -> SURVをのどちらかを最終解とすべきことは明らか。どちらを選択すべきかは、分析者の判断。（節約原則を優先させれば、DEV\*STB -> SURV。わずかな違いでもconsisitencyを優先させれば、DEV\*LIT\*STB -> SURV。）

IV-4. outcomeを補集合とした分析

　fsQCAでも結果を補集合とした分析（この場合は、outcome=~SURV）を行うべきである。

160 #~SURV(not surv)について

161 truthTable(dff,outcome=~SURV,incl.cut=0.8,ccomplete=TRUE, show.cases=TRUE)

162 #省略形

163 truthTable(dff,outcome=~SURV,complete=FALSE,incl.cut=0.6, show.cases=TRUE)

164 #真理表をTTBNFCに格納

165TTBFNC<-truthTable(dff,outcome=~SURV,incl.cut=0.6,complete=TRUE, show.cases=TRUE)

167 minimize(TTBFNC,include="?", incl.cut=0.6,details = TRUE)

168 minimize(TTBFNC, incl.cut=0.6,details = TRUE)

169 minimize(TTBFNC,incl.cut=0.6,details=TRUE)$PIchart

170 superSubset(dff,outcome=~SURV,incl.cut=0.6,cov.cut=0.5)

171 SSFN<-superSubset(dff,outcome=~SURV,incl.cut=0.8,cov.cut=0.5)

結果

truthTable(dff,outcome=~SURV,complete=FALSE,incl.cut=0.6, show.cases=TRUE)

OUT: output value

n: number of cases in configuration

incl: sufficiency inclusion score

PRI: proportional reduction in inconsistency

DEV URB LIT IND STB OUT n incl PRI cases

1 0 0 0 0 0 1 3 0.991 0.988 GR,PT,ES

2 0 0 0 0 1 1 2 0.985 0.977 IT,RO

5 0 0 1 0 0 1 2 0.870 0.754 Hu,PL

6 0 0 1 0 1 1 1 0.871 0.785 EE

21 1 0 1 0 0 1 1 0.892 0.813 AU

22 1 0 1 0 1 0 4 0.415 0.256 FI,FR,IE,SE

30 1 1 1 0 1 0 2 0.531 0.265 CZ,NL

31 1 1 1 1 0 1 1 0.960 0.930 GE

32 1 1 1 1 1 0 2 0.271 0.123 BE,UK

　　　結果　最節約解

minimize(TTBFNC,include="?", incl.cut=0.6,details = TRUE)

M1: ~DEV + ~STB -> ~SURV

inclS PRI covS covU cases

---------------------------------------------------------------

1 ~DEV 0.837 0.797 0.783 0.230 GR,PT,ES; IT,RO; Hu,PL; EE

2 ~STB 0.902 0.871 0.658 0.106 GR,PT,ES; Hu,PL; AU; GE

---------------------------------------------------------------

M1 0.848 0.814 0.889

　　　結果　初期解168 minimize(TTBFNC, incl.cut=0.6,details = TRUE)を実行

minimize(TTBFNC, incl.cut=0.6,details = TRUE)

M1: ~DEV\*~URB\*~IND + ~URB\*LIT\*~IND\*~STB + DEV\*URB\*LIT\*IND\*~STB -> ~SURV

inclS PRI covS covU cases

----------------------------------------------------------------------

1 ~DEV\*~URB\*~IND 0.893 0.864 0.737 0.444 GR,PT,ES; IT,RO; Hu,PL; EE

2 ~URB\*LIT\*~IND\*~STB 0.878 0.793 0.348 0.041 Hu,PL; AU

3 DEV\*URB\*LIT\*IND\*~STB 0.960 0.930 0.105 0.042 GE

---------------------------------------------------------------------

M1 0.899 0.874 0.834

結果　PIchart 169 minimize(TTBFNC,incl.cut=0.6,details=TRUE)$Pichartを実行

> minimize(TTBFNC,incl.cut=0.6,details=TRUE)$PIchart

　1 2 5 6 21 31

~DEV\*~URB\*~IND x x x x - -

~URB\*LIT\*~IND\*~STB - - x - x -

DEV\*URB\*LIT\*IND\*~STB - - - - - x

　　　結果　superSubset

uperSubset(dff,outcome=~SURV,incl.cut=0.6,cov.cut=0.5)

inclN RoN covN

---------------------------------------

1 ~DEV 0.783 0.864 0.837

2 ~URB 0.857 0.605 0.675

3 ~IND 0.941 0.405 0.623

4 ~STB 0.658 0.943 0.902

5 ~DEV\*~URB 0.737 0.924 0.893

6 ~DEV\*~IND 0.783 0.864 0.837

7 ~URB\*~IND 0.857 0.624 0.686

8 ~IND\*~STB 0.635 0.944 0.899

9 ~DEV\*~URB\*~IND 0.737 0.924 0.893

10 DEV + ~LIT 0.766 0.428 0.542

11 URB + ~LIT 0.647 0.684 0.621

12 ~LIT + IND 0.618 0.783 0.690

13 ~LIT + STB 0.741 0.393 0.513

---------------------------------------

最節約解はM1: ~DEV + ~STB -> ~SURVである。初期解は、M1: ~DEV\*~URB\*~IND + ~URB\*LIT\*~IND\*~STB + DEV\*URB\*LIT\*IND\*~STB -> ~SURVであるが、2つの解の、

Consistencyを比べるとほとんど差がない。このことから、最終解は、M1: ~DEV + ~STB -> ~SURVとすべきだと思われるが、念のために、superSubsetを確認すると、~DEV + ~STBについて情報がない。Consistencyの高い集合の順にデータを集約していないという、superSubsetの致命的な欠陥のために、そうなってしまう。こうなってくると、他に検討すべき、解がありそうな気がしてきて、不安になる。念のため、ありそうな、集合の組み合わせについて、確認しておく。

とりあえず、２要因の組み合わせについて検討した。それで、大丈夫かという気もするが、感覚的には、積集合をつくると、集合の大きさは小さくなって、consistencyは高くなるが、coverageは低くなる。和集合を作ると、大きくなるので、coverageは大きくなるが、consistencyは小さくなる。そこで、consistencyの大きな和集合を探せばよいと割り切ってしまう。

173 #2要因の組み合わせについて検討する。

174 #DEV,URB

175 truthTable(dff, outcome = "~SURV", conditions = "DEV,URB", incl.cut = 0.8, n.cut = 1, pri.cut = 0, exclude = NULL, complete = FALSE, show.cases=TRUE)

177 TTBFNC1<-truthTable(dff, outcome = "~SURV", conditions = "DEV,URB", incl.cut = 0.6, n.cut = 1, pri.cut = 0, exclude = NULL, complete = TRUE, show.cases=TRUE)

179 minimize(TTBFNC1,include="?", details = TRUE)

180 minimize(TTBFNC1, details = TRUE)

181 minimize(TTBFNC1,details=TRUE)$PIchart

182 superSubset(dff, outcome=~SURV, incl.cut=0.6, conditions="DEV,URB")

結果　まとめて

　　　　　真理表

> truthTable(dff, outcome = "~SURV", conditions = "DEV,URB", incl.cut = 0.8, n.cut = 1, pri.cut = 0, exclude = NULL, complete = FALSE, show.cases=TRUE)

DEV URB OUT n incl PRI cases

1 0 0 1 8 0.893 0.864 EE,GR,Hu,IT,PL,PT,RO,ES

3 1 0 0 5 0.445 0.312 AU,FI,FR,IE,SE

4 1 1 0 5 0.344 0.209 BE,CZ,GE,NL,UK

　　　　最節約解

> TTBFNC1<-truthTable(dff, outcome = "~SURV", conditions = "DEV,URB", incl.cut = 0.6, n.cut = 1, pri.cut = 0, exclude = NULL, complete = TRUE, show.cases=TRUE)

> minimize(TTBFNC1,include="?", details = TRUE)

M1: ~DEV -> ~SURV

inclS PRI covS covU cases

------------------------------------------------------------

1 ~DEV 0.837 0.797 0.783 - EE,GR,Hu,IT,PL,PT,RO,ES

------------------------------------------------------------

M1 0.837 0.797 0.783

　　　初期解

minimize(TTBFNC1, details = TRUE)

M1: ~DEV\*~URB -> ~SURV

inclS PRI covS covU cases

-----------------------------------------------------------------

1 ~DEV\*~URB 0.893 0.864 0.737 - EE,GR,Hu,IT,PL,PT,RO,ES

-----------------------------------------------------------------

M1 0.893 0.864 0.737

　　　　　PIchart

> minimize(TTBFNC1,details=TRUE)$PIchart

1

~DEV\*~URB x

superSubset

superSubset(dff, outcome=~SURV, incl.cut=0.6, conditions="DEV,URB")

inclN RoN covN

---------------------------------

1 ~DEV 0.783 0.864 0.837

2 ~URB 0.857 0.605 0.675

3 ~DEV\*~URB 0.737 0.924 0.893

---------------------------------

最節約解は、~DEV -> ~SURVで和集合の解はない。

以下省略（183～205）

206 #DEV,STB

207 truthTable(dff, outcome = "~SURV", conditions = "DEV,STB", incl.cut = 0.6, n.cut = 1, pri.cut = 0, exclude = NULL, complete = FALSE, show.cases=TRUE)

209 TTBFNC4<-truthTable(dff, outcome = "~SURV", conditions ="DEV,STB", complete=TRUE,show.cases=TRUE)

211 minimize(TTBFNC4,include="?", incl.cut=0.5,details = TRUE)

212 minimize(TTBFNC4, incl.cut= 0.5,details = TRUE)

213 minimize(TTBFNC4,incl.cut=0.5,details=TRUE)$PIchart

214 superSubset(dff, outcome=~SURV, conditions="DEV,STB", incl.cut=0.6)

**結果　まとめて**

#DEV,STB

　　　真理表

> truthTable(dff, outcome = "~SURV", conditions = "DEV,STB", incl.cut = 0.6, n.cut = 1, pri.cut = 0, exclude = NULL, complete = FALSE, show.cases=TRUE)

DEV STB OUT n incl PRI cases

1 0 0 1 5 0.895 0.855 GR,Hu,PL,PT,ES

2 0 1 1 3 0.839 0.774 EE,IT,RO

3 1 0 1 2 0.915 0.866 AU,GE

4 1 1 0 8 0.261 0.141 BE,CZ,FI,FR,IE,NL,SE,UK

　　　最節約解

> TTBFNC4<-truthTable(dff, outcome = "~SURV", conditions ="DEV,STB", complete=TRUE,show.cases=TRUE)> minimize(TTBFNC4,include="?", incl.cut=0.5,details = TRUE)

M1: ~DEV + ~STB -> ~SURV

inclS PRI covS covU cases

-------------------------------------------------------------

1 ~DEV 0.837 0.797 0.783 0.230 GR,Hu,PL,PT,ES; EE,IT,RO

2 ~STB 0.902 0.871 0.658 0.106 GR,Hu,PL,PT,ES; AU,GE

-------------------------------------------------------------

M1 0.848 0.814 0.889

　　　初期解

minimize(TTBFNC4, incl.cut= 0.5,details = TRUE)

M1: ~DEV + ~STB -> ~SURV

inclS PRI covS covU cases

-------------------------------------------------------------

1 ~DEV 0.837 0.797 0.783 0.230 GR,Hu,PL,PT,ES; EE,IT,RO

2 ~STB 0.902 0.871 0.658 0.106 GR,Hu,PL,PT,ES; AU,GE

-------------------------------------------------------------

M1 0.848 0.814 0.889

PIchart

minimize(TTBFNC4,incl.cut=0.5,details=TRUE)$PIchart

1 2 3

~DEV x x -

~STB x - x

SuperSubset

superSubset(dff, outcome=~SURV, conditions="DEV,STB", incl.cut=0.6)

inclN RoN covN

----------------------------

1 ~DEV 0.783 0.864 0.837

2 ~STB 0.658 0.943 0.902

----------------------------

最節約解として、M1: ~DEV + ~STB -> ~SURVが提案されている

以下省略（215～255）

256 #LIT,STB

257 truthTable(dff, outcome = "~SURV", conditions = "LIT,STB", incl.cut = 0.6, n.cut = 1, pri.cut = 0, exclude = NULL, complete = FALSE, show.cases=TRUE)

259 TTBFNC9<-truthTable(dff, outcome = "~SURV", conditions = "LIT,STB",

incl.cut = 0.6, n.cut = 1, pri.cut = 0, exclude = NULL, complete = TRUE, show.cases=TRUE)

261 minimize(TTBFNC9,include="?", incl.cut=0.6,details = TRUE)

262 minimize(TTBFNC9, incl.cut= 0.6,details = TRUE)

263 minimize(TTBFNC9,incl.cut=0.6,details=TRUE)$PIchart

264 superSubset(dff, outcome=~SURV, conditions="LIT,STB", incl.cut = 0.6)

結果　まとめて

　　　　　真理表

#LIT,STB

> truthTable(dff, outcome = "~SURV", conditions = "LIT,STB", incl.cut = 0.6, n.cut = 1, pri.cut = 0,

+ exclude = NULL, complete = FALSE, show.cases=TRUE)

OUT: output value

n: number of cases in configuration

incl: sufficiency inclusion score

PRI: proportional reduction in inconsistency

LIT STB OUT n incl PRI cases

1 0 0 1 3 0.990 0.987 GR,PT,ES

2 0 1 1 2 0.984 0.974 IT,RO

3 1 0 1 4 0.894 0.829 AU,GE,Hu,PL

4 1 1 0 9 0.333 0.221 BE,CZ,EE,FI,FR,IE,NL,SE,UK

> TTBFNC9<-truthTable(dff, outcome = "~SURV", conditions = "LIT,STB", incl.cut = 0.6, n.cut = 1, pri.cut = 0,

+ exclude = NULL, complete = TRUE, show.cases=TRUE)

> minimize(TTBFNC9,include="?", incl.cut=0.6,details = TRUE)

M1: ~LIT + ~STB -> ~SURV

inclS PRI covS covU cases

----------------------------------------------------------

1 ~LIT 0.985 0.980 0.540 0.131 GR,PT,ES; IT,RO

2 ~STB 0.902 0.871 0.658 0.250 GR,PT,ES; AU,GE,Hu,PL

----------------------------------------------------------

M1 0.912 0.888 0.790

minimize(TTBFNC9, incl.cut= 0.6,details = TRUE)

M1: ~LIT + ~STB -> ~SURV

inclS PRI covS covU cases

----------------------------------------------------------

1 ~LIT 0.985 0.980 0.540 0.131 GR,PT,ES; IT,RO

2 ~STB 0.902 0.871 0.658 0.250 GR,PT,ES; AU,GE,Hu,PL

----------------------------------------------------------

M1 0.912 0.888 0.790

minimize(TTBFNC9,incl.cut=0.6,details=TRUE)$PIchart

1 2 3

~LIT x x -

~STB x - x

superSubset(dff, outcome=~SURV, conditions="LIT,STB", incl.cut = 0.6)

inclN RoN covN

----------------------------------

1 ~STB 0.658 0.943 0.902

2 ~LIT + STB 0.741 0.393 0.513

3 LIT + STB 0.701 0.344 0.471

----------------------------------

最節約解として、M1: ~LIT + ~STB -> ~SURVが提案されている。

以下省略(265～276)

２要因についての検討結果をまとめると、 ~DEV + ~STB -> ~SURV　(consistency= 0.848 PRI= 0.814 coverage=0.889)と ~LIT + ~STB -> ~SURV　(consistency= 0.912 PRI=0.888 coverage=0.790 が提案されたことになる。興味があるのは、~DEV+~LITだがこちらは、特に候補に挙げられていないのだが、~DEV,~LIT,~STBが大きくかかわってることがわかる。そのように考えると、この３つがかかわりあって、outcome=~SURVが生じる可能性について、検討しておきたくなる。

277 #3条件（DEV,LIT,STB）について、検討する。

278 #DEV,LIT,STB

279 truthTable(dff, outcome = "~SURV", conditions = "DEV,LIT,STB", incl.cut = 0.6, n.cut = 1, pri.cut = 0, exclude = NULL, complete = FALSE, show.cases=TRUE)

281 TTBFNC11<-truthTable(dff, outcome = "~SURV", conditions = "DEV,LIT,STB", incl.cut = 0.6, n.cut = 1, pri.cut = 0, exclude = NULL, complete = TRUE, show.cases=TRUE)

283 minimize(TTBFNC11,include="?", incl.cut=0.6,details = TRUE)

284 minimize(TTBFNC11, incl.cut= 0.6,details = TRUE)

285 minimize(TTBFNC11,incl.cut=0.6,details=TRUE)$PIchart

286 superSubset(dff, outcome=~SURV, conditions="DEV,LIT,STB", incl.cut = 0.6)

結果

真理表

> truthTable(dff, outcome = "~SURV", conditions = "DEV,LIT,STB", incl.cut = 0.6, n.cut = 1, pri.cut = 0,

+ exclude = NULL, complete = FALSE, show.cases=TRUE)

DEV LIT STB OUT n incl PRI cases

1 0 0 0 1 3 0.990 0.987 GR,PT,ES

2 0 0 1 1 2 0.985 0.977 IT,RO

3 0 1 0 1 2 0.873 0.754 Hu,PL

4 0 1 1 1 1 0.794 0.666 EE

7 1 1 0 1 2 0.915 0.866 AU,GE

8 1 1 1 0 8 0.262 0.142 BE,CZ,FI,FR,IE,NL,SE,UK

　最節約解

> TTBFNC11<-truthTable(dff, outcome = "~SURV", conditions = "DEV,LIT,STB", incl.cut = 0.6, n.cut = 1, pri.cut = 0, exclude = NULL, complete = TRUE, show.cases=TRUE)

> minimize(TTBFNC11,include="?", incl.cut=0.6,details = TRUE)

M1: ~DEV + ~STB -> ~SURV

inclS PRI covS covU cases

---------------------------------------------------------------

1 ~DEV 0.837 0.797 0.783 0.230 GR,PT,ES; IT,RO; Hu,PL; EE

2 ~STB 0.902 0.871 0.658 0.106 GR,PT,ES; Hu,PL; AU,GE

---------------------------------------------------------------

M1 0.848 0.814 0.889

　　　初期解

minimize(TTBFNC11, incl.cut= 0.6,details = TRUE)

M1: ~DEV + LIT\*~STB -> ~SURV

inclS PRI covS covU cases

-------------------------------------------------------------------

1 ~DEV 0.837 0.797 0.783 0.481 GR,PT,ES; IT,RO; Hu,PL; EE

2 LIT\*~STB 0.894 0.829 0.407 0.106 Hu,PL; AU,GE

-------------------------------------------------------------------

M1 0.850 0.816 0.889

　　PIchart

minimize(TTBFNC11,incl.cut=0.6,details=TRUE)$PIchart

1 2 3 4 7

~DEV x x x x -

LIT\*~STB - - x - x

SuperSubset

superSubset(dff, outcome=~SURV, conditions="DEV,LIT,STB", incl.cut = 0.6)

inclN RoN covN

----------------------------------

1 ~DEV 0.783 0.864 0.837

2 ~STB 0.658 0.943 0.902

3 DEV + ~LIT 0.766 0.428 0.542

4 ~LIT + STB 0.741 0.393 0.513

5 LIT + STB 0.701 0.344 0.471

----------------------------------

3要因で分析しても、最節約解は~DEV + ~STB -> ~SURVである。初期解として~DEV + LIT\*~STB -> ~SURVという解も得られ、こちらの方が、consistency, coverageともに、わずかに高いが、その差はわずかである。superSubsetの結果では、３要因のうち、２つの要因の和集合となっている解が３つ提案されているが、いずれもconsistencyが小さい。以上の理由で、最終解として~DEV + ~STB -> ~SURVとするとのが、最も妥当だと判断できる。

以上の検討の結果として、分析の結果をまとめると、

1. 豊かで、政治的に安定している国は、民主主義を維持することができた。
2. 貧しいか、政治が不安定な国は、民主主義が維持できなかった。
3. おそらく、そのようになるメカニズムには、豊かで政治的に安定していることによる、教育レベルの向上が関係している可能性がある。

となる。