複雑な社会調査における

データ・クリーニング技法の開発 *[1]

保田時男
（関西大学）

【論文要旨】

筆者は、複雑性を増す社会調査に対応するために、Fellegi and Holt の原則に従いつつ実践的に利便性の高いデータ・クリーニングの手続きを定め、粘土細工アプローチと名付けた。本稿は、粘土細工アプローチの理論的背景、開発過程、具体的なツール類について解説したものである。2005 年 SSM 調査のクリーニングにおける困難を経験したことは、筆者がクリーニング技法の開発の必要性を考える端緒となった。その結果として開発された粘土細工アプローチは、2015 年 SSM 調査のクリーニングに適用され、一定の成果を収めたといえる。粘土細工アプローチは他の調査のクリーニングにも適用され、それぞれがもつ困難に対応するために発展してきた。粘土細工アプローチを採用するかどうかは別としても、社会調査のデータ・クリーニングの携わる調査実践者には、本論考は広く役立つものと考える。

キーワード： データ・クリーニング、社会調査、Fellegi-Holt、方法論

１．目的

本稿の目的は筆者がおよそ 10 年間にわたって取り組んできたデータ･クリーニング技法の開発について、その総括をすることにある。近年の社会調査の複雑化に伴ってクリーニングの難度は格段に高まっている。繰り返し横断調査やパネル調査、あるいは 2 者以上から多面的に情報を収集するマルチアクターの調査など、同時に関連しあうデータ量が大きくなるこのような調査方法は、調査データの複雑性を格段に高めている。また、大規模な調査では単純に質問項目の分量や枝分かれの量も多くなる傾向にあり、単体の横断調査でも複雑なデータを形成することがある。

「社会階層と社会移動に関する全国調査（以下、SSM 調査）」における職歴の回顧は、もちろんそのような複雑な社会調査の典型であり、クリーニングの難度も相当に高いといってよい。筆者は 2005 年 SSM 調査で主に教育関連の質問項目についてクリーニング作業の末端を担ったが、率直に言って非常な混乱を経験した。中心的にクリーニング作業に取り組んでいた研究メンバーの混乱はその比ではなかったものと想像する。ただし、そこで行われていた手続きは、日本で行われている一般的なクリーニング手続きであり、決して特殊なものではなかった。筆者は、現代の複雑化した社会調査においては従来の因習的な手法の適用は限

[1] 本研究は、JSPS 科研費 JP25000001 の助成を受けたものです。

界にあり、クリーニングにも確固とした理論的後付けに根差した実践が必要と感じられた。

後述するように、クリーニングの理論と実践には応用統計学の分野で一定の蓄積があることがわかった。そこで、筆者はクリーニングの古典的理論と現代日本の社会調査における背景をミックスしながら実践的なクリーニング技法の開発に取り組み、この技法を「粘土細工アプローチ（Clay Modeling Approach）」と名付けた。粘土細工アプローチは SSM 調査などの実践的取り組みの中で改善され、一応の到達点までたどり着いたつもりである。

粘土細工アプローチについてこれまで散発的にその思想や手法を紹介してきたが（保田 2010, 2011, 2012）、まとまった論考は存在しない。本稿はこれまでの粘土細工アプローチの発展を総括するものである。クリーニングの理論的背景（2 節）、粘土細工アプローチの手順（3 節）、手法の開発過程（4 節）、ツールの概説（5 節）、今後の課題（6 節）について順に論じる。

２．クリーニングの理論的背景

2.1　一般的なクリーニング法の問題点

まず日本の社会調査で一般的に用いられているクリーニング方法と、その問題点を確認しておこう。多分に因習的なものであるが、通常、クリーニングは次のような手続きでなされることが多い。1）調査票の原票を確認して異常な個所があれば赤字で修正する*[2]。2）データの度数分布表で各変数に異常値がないか点検する。3）クロス表で異常な組み合わせのパターンがないか点検する（いわゆる論理エラーの点検）。4）異常が見つかった点検事項については修正方針を決めてデータを修正する（場合によっては原票を確認して赤字を入れる）。

複雑性の低い社会調査や小規模な社会調査では、このような手続きでクリーニングを行ったとしてもとくに問題は生じないし、実際に効率的である。しかしながら、大規模で複雑性の高い調査になると、問題が続出する。というのは、この手続きでは異常個所を特定して修正を施す作業を点検事項ごとに行うので、ある点検事項に対応するために施した修正のせいで別の点検事項で矛盾が生じる場合があるからである。そうすると、すでに点検を終えている事項についても繰り返し見直しをする必要が生じる。また、大規模な調査では点検事項を分担して並行的にクリーニング作業を進めることも多い。この場合、それぞれの点検事項の視点から相反する修正がなされ、どれが最新のデータなのか混乱することもある。原票への

[2] 原票を赤字で修正することを指して「エディティング」と呼ぶことがあるが、クリーニング研究ではこの語法は一般的ではない。通常、data editing はデータの点検から修正全体を指す用語、つまりデータ・クリーニングを指す用語として用いられる（data cleaning も用いるが、data editing の方が一般的である）。また、狭義の data editing は調査票やデータを点検して異常個所を特定する作業を指し、異常個所について具体的な修正値を決める作業と区別する場合にも用いる。日本のデータ・クリーニングで原票の点検を「エディティング」と呼ぶのは、この狭義の意味から派生したのではないかと想像される。

赤字の修正を重視した場合には書き込みだらけで判読が困難になることすらある。あるいは、調査の複雑性が増すほどに修正時のミス*[3]による新たな異常値の発生も頻出する。このような事情があるために、全体としてどの程度までクリーニングが進んでいるのか客観的に判断することが難しく、気が付けば膨大な時間をクリーニングに費やしている場合がある。

具体的には、たとえば 2005 年 SSM 調査では次のような問題を経験した。筆者は香川めい氏らとともに教育項目（学歴）を点検する視点から在学年齢の異常などを修正した。ところが、そうすると、続く職業経歴の開始年齢との矛盾が生じる。もともとは学卒後にすぐに仕事に就いたように回答していたものが 1 年ずれてしまったりする。そこでその点も修正が必要になるが、一方で職歴の点検をしていたグループからは別の視点から修正が施されていたり、思いもよらない矛盾が生じたりする。この問題は上記の典型的な例であり、偶然に筆者が身近に経験したものであるが、同様の問題は各所で起こっていたものと思われる。

このような問題が生じる根本的な問題は、因習的に用いられているクリーニングの手続きが統計学的なクリーニングの理論から逸脱していることによる。応用統計学のなかでデータ・クリーニング（通常は、data editing と称されることが多い）は一定の蓄積がある研究分野である。そのなかでもっとも重視されている古典的理論が Fellegi and Holt（1976）で示されている原則である。これについては次節で整理するが、欧米の大規模な政府系調査ではこのような原則に従ったクリーニングの手続きが取られることが一般化しており、クリーニングにかかる金銭的コストは調査費用の 20%以上を占め、一般的な調査ではその比率は 40%にものぼると言われている*[4]（Weisberg 2005; Groves et al. 2009）。1990 年代以降とくに UNECE（United Nations Economic Commission for Europe）の Statistical Data Editing 作業部会で急速に再展開され、一定の成果を収めている（United Nations 2006; De Wall et al. 2011; Herzog et al. 2007）*[5]。複雑な社会調査のクリーニング問題に対応するためには、このような研究成果を活かすことが必要不可欠である。

---

[3] 経験則であるが、修正時のミスとして比較的多いのは、無回答・非該当コードに関わるものである。たとえば、無回答コードが 999 なのに誤って 99 と修正してしまう場合や、枝分かれ質問に関するデータを修正したのに、他の項目を非該当コードに修正するのを忘れていたりするような場合である。その他、複数回答の項目で 0/1 のダミーコードをあてているのに、誤って選択肢番号を入力してしまうような初歩的なミスも意外と多い。このようなミスを目視だけで完全になくすことは難しく、後述するように edit ルールによる機械的な点検が必須である。

[4] このことからもわかるように、クリーニングの技法を整備すればそのコストが激減するということではない。クリーニング技法を整備することの第一の利点は、混沌とした手続きを管理・理解可能なものにすることにある。また、その経験は、将来の調査設計に役立つとともに（Granquist and Kovar 1997）、その知見の共有が広く他の調査の負担軽減にもつながる（Weisberg 2005）。その意味で、クリーニングを整備することは、現在の利益を追求する以上に投資の意味が大きい。

[5] 2017 年の作業部会の報告内容はここに掲載されている。https://www.unece.org/index.php?id=43887（2018 年 1 月 31 日取得）

2.2　Fellgi-Holt のクリーニング哲学

応用統計学におけるクリーニング研究において Fellegi and Holt（1976）の研究はもっとも重要な古典的理論である。この研究で示されているのは、「自動クリーニング」の理論的な考え方である。つまり、人間の判断でクリーニングを行うのではなく入力されたデータと点検事項をもとにして一定のアルゴリズムでコンピューターに修正値を決定させる方法について論じている。ここで前提とされているのは、たとえば国勢調査のような超大規模データであり、我々が行うような（せいぜい 10,000 程度のサンプルサイズの）相対的に規模の小さい社会調査にはそぐわない。しかしながらコンピューターによる自動クリーニングを行う上では当然、クリーニングの背景にある考え方やあるべき手続きを明確にする必要がある。このクリーニングの哲学とでも言うべき原則は、手動か自動かということにかかわらず有効である。

若干遠回りになるが、Fellegi and Holt（1976）で示されている自動クリーニングの理論を概観しよう。まずクリーニングの手続きは大きく editing（異常値の検出）と imputation（修正値の推定）の 2 段階から成り立っており、これらはまったく独立した別々の過程と考える。つまり、どのようにして異常が検出されたかということによってあるべき修正値が左右されることはない。

これは一見すると不思議な考え方であるが、理にかなっている。我々が因習的に行っているクリーニング手続きではそれぞれの点検事項に引っかかった異常値について、その点検事項をクリアできるように修正を考える。しかしながら、修正しようとしている個所は他の点検事項にも引っかかる可能性があり、どの点検事項で最初に異常が発見されるかはいわば偶然にすぎない。したがって、そのような偶然に左右されないようにクリーニングを行うには、まずすべての点検事項について徹底的に異常値の個所を洗い出す editing を行ったうえで、次にすべての条件をクリアできるような異常値の修正を考える imputation を行うというように完全に 2 つの段階を切り分ける必要がある。

次に具体的な自動クリーニングの手続きについて説明する。Fellegi-Holt が示す理論的な手続きは以下の 5 つのステップにまとめられる。このうち、（1）～（2）は editing の段階であり、（3）～（5）は imputation の段階といえる。

（1）edit ルールを normal form で作成する（フローチャート等は使わない）。
（2）edit ルールを適用し、違反箇所を特定する。
（3）explicit edit から implicit edit を演繹的に生成する。
（4）ケース単位で、修正する最小限の変数を決定する。
（5）修正することになった変数の値をホットデッキ法などで imputation する

繰り返すが、この自動クリーニングの理論的手続きがそのまま我々が扱うような社会調査

データにも有効というわけではない。しかしながら、後で示す粘土細工アプローチはこの手続きの一部を手動に置き換えたものなので、それぞれのステップを簡単に説明しておきたい。

ステップ（1）の「edit ルール」とはどのような回答パターンを異常とみなすかというルールのことである。たとえば、「未婚者は結婚年齢が非該当でなければならない」とか「回答者年齢は 70 歳以下でなければならない」といったルールである。あるべきルールは当然コンピューターには判断できないので、外から示してやる必要がある。問題はその提示方法である。よくあるのはフローチャート式に異常値の条件を示す方法であり、具体的には IF 文を用いたプログラミングで異常値を検出するということである。しかし、Fellegi-Holt はこれでは不十分と考える。彼らが normal form と呼ぶのは、要するに一定のルールに従って集合論的な論理式に整理するということである。後のステップ (3) (4) のためにこの手続きが必要になる。

ステップ（2）はどのケースがどの edit ルールに違反しているか特定するということである。ここでは異常値の可能性がある変数が特定されるが、厳密にはどの変数の数値を修正すべきかは特定されない。たとえば、未婚者が結婚年齢を回答している場合、「未婚者」の方を修正しても「結婚年齢」の方を修正してもその edit ルールの違反は回避できるからである。しかしながら、ともかくすべての edit ルールについて点検を行い、各ケースがどの edit ルールに違反しているのかというリストを作ることができ、editing の段階は完了する。

ステップ（3）以降は imputation、すなわち修正値を推定するためのステップである。まず explicit edit からすべての implicit edit を導出する。ここで述べられているのはステップ（1）で明示的に定めた edit ルールから考えるとこのパターンも違反となるという暗示的な edit ルールを明らかにするということである。たとえば「未婚者は結婚年齢が非該当でなければならない」と「結婚年齢が非該当ならば離婚経験を尋ねる質問も非該当でなければならない」という 2 つの explicit edit が存在すれば、演繹的に「未婚者は離婚経験が非該当でなければならない」という implicit edit が成り立つ。ただし、このような edit ルールは設定していなくとも異常個所の検出には差し支えない。2 つの explicit edit だけで確実に異常が検出されるからである。しかしながら、すべての implicit edit を明示的に導出することが次のステップのために必要になる。導出自体は edit ルールが normal form で表現されていれば数学的な演算で自動的に可能である。

ステップ（4）では各ケースの edit ルール違反に対応するためにどの変数の数値を修正するかを決定する。先ほども述べたとおり「未婚者は結婚年齢が非該当でなければならない」に違反している場合、未婚者の方を修正すべきか結婚年齢の方を修正すべきかは明らかではない。ケースによって異なってくる。そこで、Fellegi-Holt の自動クリーニングでは最小限の変数を修正することですべての edit ルールをクリアするように修正対象の変数を決定する。この最小限の変数を特定するのは見た目ほど簡単ではない。ある変数を修正することで新たに他の edit ルールに違反することもあるからである。そこで、ある変数を修正することが edit

ルール群全体に対してどれだけ影響があるかを演算するために、ステップ（3）で求めた implicit edit が必要になる。implicit edit がすべて明らかになっていればある変数の値を修正することのデータ全体への影響が正確に評価できる。これをもとに、どの変数の値を修正することがもっとも最小の修正に収まるかを特定する。

ステップ（5）では前のステップで修正することになった変数を具体的にどのような値に修正するかを決定する。本来あるべき数値を正確に知ることはできないので、修正値を推定するわけだが、Fellegi-Holt ではホットデッキ法などでの推定を想定している。つまり、同じデータの中から問題のない類似のケースをランダムに参照して修正値として採用する。自動クリーニングでは国勢調査のような超大規模データを想定しているので、このような方法が有効に機能する。

2.3　Fellegi-Holt からの教訓

以上のように Fellegi-Holt の自動クリーニングを概観すると、いくつかの点で我々のような社会調査データのクリーニングにはそぐわない点があることがわかる。原則を活かしつつ手続きに手を加えなければならない。

Fellegi-Holt の原則からとくに我々が学ぶべき教訓は以下の 3 点と考えた。第 1 に、edit ルールは必ず明確にプログラミングしなければならない。度数分布表やクロス表の目視で点検を済ませることは簡単な調査では効率的であるが、複雑な調査では見落としが生じる。また、より問題なことは修正が他の edit ルールへの違反を引き起こしていることに気づきにくいという点である。簡単なルールを含めてすべての edit ルールをプログラミングしなければならない。

第 2 に、editing（異常の検出）と imputation（修正）の切り分けはやはり重要である。この間を行き来してしまうことが各種のクリーニングの混乱を引き起こす根本的な問題といってもよい。もちろん現実的には imputation の段階に入ってから editing の見落とし（edit ルールの不足や間違い）に気づき訂正することはあるが、少なくとも理念的には徹底的に edit ルールの作成・適用を終えてから修正の作業に入るべきである。

第 3 に、異常の修正は edit ルールごとに行うのではなく、ケースごとに全 edit ルールを考慮した修正を考えなければならない。同じ edit ルールに引っかかったケース群でも他にどのような異常個所があるかはケースごとに異なる。自動クリーニングではそれが自動的に考慮されるわけであるが、事情は手動クリーニングの場合でも変わらない。また、異常個所以外の情報も適切な修正値を考えるうえで役に立つ。ケースごとの事情を勘案せずに一括した修正ルールを適用することは、かなり単純な水準のクリーニングを除いてはかえって混乱を増幅する原因にもなる。

一方で、Fellegi-Holt の手続きのうち、normal form や implicit edit といった手続きは自動ク

リーニングを志向するためにのみ必要なもので、我々が行うクリーニングでは必要ないと考えた*[6]。これらの手続きはステップ（4）で修正すべき変数を決定するためのもので手動のクリーニングにはそぐわない。また、当然ながらステップ（5）のような自動修正は我々の規模の調査では現実的に不可能で、どの変数をどのように修正することが適切かは、基本的に個別に判断するしかない。

３．粘土細工アプローチの手続き

Fellegi-Holt から得られた教訓をもとにして、我々が扱っているような（複雑性が高く、サンプルサイズが極端に大きくはない）社会調査における実践的なクリーニング方法を考慮した結果、表 1 のような手続きにたどり着いた。この手続きは、決めたルールで型にはめた修正を行うのではなく、1 ケースごとに生じている異常な凸凹を（そのケースの全体的なバランスを見ながら）修正していくことをイメージしている。形がいびつな粘土細工を 1 つずつ修正していくことをメタファーとして、これを「粘土細工アプローチ（Clay Modeling Approach）」と名付けた。粘土細工アプローチでとくに意識していることは、Fellegi-Holt のクリーニングの基礎原則から逸脱しないことに加えて、クリーニングの進度が明確に把握しやすく、分担作業で混乱が生じないということである。因習的なクリーニング方法にはいろいろな問題があるが、その大きな原因は分担作業が管理しにくいことから生じる混乱にあると考えたからである。

表１　粘土細工アプローチの基本手順

手順	分担
1. edit ルールを論理式で作成 a) 単独変数の range edit b) 枝分かれの filter edit c) その他の general edit	a) b)はとくに素養がなくても分担可能 c)はクリーニング熟練者＋調査領域の専門家が望ましい
2. ケースごとに、引っかかった edit をリストする（各 edit のフラグを立てる）	自動
3. ケースごとに、2 の結果と原票を見ながら修正の決定	ケース単位で分担可能
4. 修正後のデータに edit を再適用	自動
5. 全 edit をクリアすれば完了	全体がほぼ同時に終了

3.1　手順 1：edit ルールの作成

表 1 にそって、粘土細工アプローチの手順を順に説明する。粘土細工アプローチでも、手順 1 のとおり、最初に edit ルールを作成することが重要である。Fellegi-Holt の言うように厳

[6] normal form が不要というのは、論理式を一定のルールに従わせることまでは不要という意味で、論理式で edit ルールを示すこと自体は必要である。

密な normal form である必要はなく、通常の論理式になっていればよい。つまり、ルールに違反している場合には 1、違反していない場合には 0 となる（プログラミングにおけるいわゆる「フラグを立てる」）式を書けばよい。

edit ルールの作成は range edit、filter edit、general edit の 3 種類にわけて考える。これは Delgado-Quintero and Salazar-González（2008）の提唱に従ったもので、一般的な社会調査で考えるとこの 3 分類は実践的である*[7]。range edit は値範囲の点検とでもいうべきもので、それぞれの変数で定められている範囲以外の値を取っていないかを確認するものである。因習的なクリーニングでいえば、度数分布表から異常値を探すことと対応する。range edit で認める値範囲には質問紙の選択肢だけでなく、無回答や非該当のコードもあることに注意が必要である*[8]（図 1）。

```
（spss シンタックスの場合）
compute ed1037=~any(q1_1,1,2).
compute ed1045=~any(q2_a,1,2,3,4,5,6,7,8,9,10,11,12,999990,999999).
compute ed1046=~any(q2_b,721,791,821,822,999990,888888,999999) & ~range(q2_b,1,99).

（Excel 関数の場合）
[ed1037] =NOT(OR(OR(q_2_b={721,791,821,822,999990,888888,999999}), AND(q_2_b>=1,q_2_b<=
    99)))
[ed1045] =NOT(OR(q_1_1={1,2}))
[ed1046] =NOT(OR(q_2_a={1,2,3,4,5,6,7,8,9,10,11,12,999990,999999}))
```

図 1　2015 年 SSM 調査での range edit の例*[9]

また、filter edit はスクリーニングの点検である。枝分かれ質問などによって「非該当」が生じる変数について、非該当の条件と整合しているかを点検する。因習的なクリーニングで論理エラーのチェックと呼ばれ、クロス表から異常値を探すものは多くが filter edit に対応する。filter edit には「非該当のはずはないのに、非該当になっている」（図 2 の ed...x）と「非該当のはずなのに、非該当になっていない」（図 2 の ed...z）の 2 種類があることに注意が必要である。また、違反が見つかった際に、枝分かれ質問の値が間違っている場合と枝分かれ後の変数の値が間違っている場合の両方があるので、修正は慎重にしなければならない。さ

[7] 概念的にいえば、Herzog et al.（2007）が示している分類の方が包括的であるが、実践面で言えばあまり意味のない分類となる。ここで示している 3 分類の方が実践作業においては明らかに利便性が高い。

[8] 変数によって非該当や無回答のコードが異なることは、edit ルールを作成する際に大きな負担になる。そのため、クリーニング時のデータはすべての変数について非該当と無回答のコードを統一し、クリーニングの完了後に元のコードに戻してやる方がよい。2015 年 SSM 調査では最大の桁数を取って、非該当を 888888 に、無回答を 999999 に統一してクリーニングを行った（「わからない」も 999990 に統一した）。

[9] Excel 関数の場合、「セルに名前を付ける」機能で変数名を用いる。ただし、この場合たとえば「q1」という名前はセル参照の表現（Q1 セルを参照）と重なり使用できない。このため、Excel 関数の場合には q_1 のようにセル参照と重ならない変数名を付ける必要がある。

らに、枝分かれ質問が無回答だった場合に、枝分かれ後の変数をどのように処理するのかは、質問紙の構造やデータ作成の方針にもよるが混乱のもとになるポイントである。

```
（spss シンタックスの場合）
compute ed2131x=(any(q25,3,4)) & (q34=888888).
compute ed2131z=~(any(q25,3,4)) & ~(q34=888888).
compute ed2132x=(any(q25,3,4)) & (q35=888888).
compute ed2132z=~(any(q25,3,4)) & ~(q35=888888).

（Excel 関数の場合）
[ed2131x] =and(or(q_25={3,4}),q_34=888888)
[ed2131z] =and(not(or(q_25={3,4})),not(q_34=888888))
[ed2132x] ==and(or(q_25={3,4}),q_35=888888)
[ed2132z] =and(not(or(q_25={3,4})),not(q_35=888888))
```

図２　2015 年 SSM 調査での filter edit の例

最後に、general edit はその他のすべての edit ルールを指す総合的な点検である。論理的に考えてありえない回答の組み合わせを考え、それぞれに論理式を立てる。複雑な調査ほど general edit の対象は多くなるが、一般的な調査では作成すべきものはそう多くはない。たとえば、複数回答の「いずれもあてはまらない」と他の選択肢が同時に選ばれていないかという点検、あるいは本人収入と配偶者収入の合計が世帯収入を超えていないかという点検などがこれにあたる（図 3)。また、厳密に「ありえない」わけではないが誤答であることが強く疑われるようなパターンも general edit として作成しておいた方がよい。実際にそれが修正をすべきケースなのかどうかは後で個別に判断されることになる。たとえば、回答者の同居世帯員が、「子ども」と「祖母」などとなっていた場合、多くのケースでは「子ども」と「母（あるいは配偶者の母)」の誤答である。家庭内で回答者の母のことを「おばあちゃん」と呼んでいるために誤っているパターンである。世帯員の年齢などの周辺的な情報がある調査では、どのケースが誤答かを高い蓋然性で判断できるので、点検項目に含めておいた方がよい。

```
（spss シンタックスの場合）
compute ed3017=(q11_98=1 & sum(q11_1 to q11_19, q11_99)~=0).
（問 11 の MA で、「どれもない」と他の選択肢に同時に○がある）
compute ed3095=(q26>q1_2_5 & q26<888888).
（問 26 の結婚年齢が現年齢よりも大きい）

（Excel 関数の場合）
[ed3017] =AND(q_11_98=1, NOT(SUM(q_11_1, q_11_2, q_11_3, q_11_4, q_11_5, q_11_6, q_11_7,
    q_11_8, q_11_9, q_11_10, q_11_11, q_11_12, q_11_13, q_11_14, q_11_15, q_11_16, q_11_17,
    q_11_18, q11_19, q11_99)=0))
[ed3095] =AND(q_26>q_1_2_5, q_26<888888).
```

図３　2015 年 SSM 調査での general edit の例

range edit や filter edit は定式的なものなのでとくに素養がなくても作成作業が可能と考えられるが、general edit を正しく作成するには一定のクリーニングの熟練が必要となる。また、そもそも調査対象としている領域についてどのような点が問題になってくるのかを判断するためには、その調査の領域について専門的な知識を必要とする。両方を兼ね備えた者、あるいはそれぞれの役割を果たせる者がチームで general edit の作成に望むことが望ましい。

3.2　手順 2：edit ルールの適用

edit ルールの作成が終われば、データに対してそれを適用する。具体的には、一つひとつの edit ルールは 1（違反）か 0（違反していない）の値を取る変数として作成されているはずであるから、SPSS などの統計分析ソフトで各ケースについてその値を算出させればよい。また、後で示すように、最終的な汎用ツールでは統計分析ソフトを使わずに Excel のみで演算する形を取っているが、考え方は同じである。

たとえば、SPSS のデータで図 4 のように SPSS シンタックスで edit ルールを適用すれば、違反ケースを示す変数が作成される。当然ながら、（とくに複雑な general edit については）edit ルールの論理式を書き間違えることは頻繁に起こるので、適用結果を見て edit ルールが意図通りに機能しているか点検しなければならない。ただし、書き間違えが残ってしまった場合でも、実際にケースごとの修正を考える際に異常に気付く機会は頻繁にあるので、最終的に致命的なことになる可能性は低い。

```
*range edit.
……
compute ed1037=~any(q1_1,1,2).
compute ed1038=~any(q1_2_1,999999) & ~range(q1_2_1,1935,1994).
compute ed1039=~any(q1_2_2,1,2,999999).
……
*filter edit.
compute ed2002x=(any(q2_a,1,2,3,4,5,6,7,8,999990,999999)) & (q2_b=888888).
compute ed2003x=(any(q2_a,1,2,3,4,5,6,7,8,999990,999999)) & (q2_c=888888).
compute ed2004x=(any(q2_a,1,2,3,4,5,6,7,8,9,999990,999999)) & (q2_d=888888).
……
*general edit.
compute ed3001=((meibo_1=1)*(meibo_2+1925)+(meibo_1=2)*(meibo_2+1988) ~= ((q1_2_2=1)*
    (q1_2_3+1925)+(q1_2_2=2)*(q1_2_3+1988))) & ((meibo_1=1)*(meibo_2+1925)+(meibo_1=2)*
    (meibo_2+1988) ~= q1_2_1).
compute ed3002=(meibo_3~=q1_2_4).
compute ed3003=(q1_2_1~=(q1_2_2=1)*(q1_2_3+1925)+(q1_2_2=2)*(q1_2_3+1988)) & q1_2_1~
    =999999 & q1_2_3~=999999.
……
```

図４　edit ルール適用シンタックスの例

3.3　手順 3：ケースごとの修正

次に edit ルールの適用結果を見ながら、修正を行う。Fellegi-Holt の原則どおり editing（異常値の検出）の段階と imputation（修正）の段階は明確に区別され、ここからが imputation の段階となる。繰り返すが、修正は「ケースごとに」すべての edit ルールの適用結果を見ながら行う。具体的には、図 5 のようにそのケースのデータと違反があった edit ルールの一覧を見ながら修正を判断する。

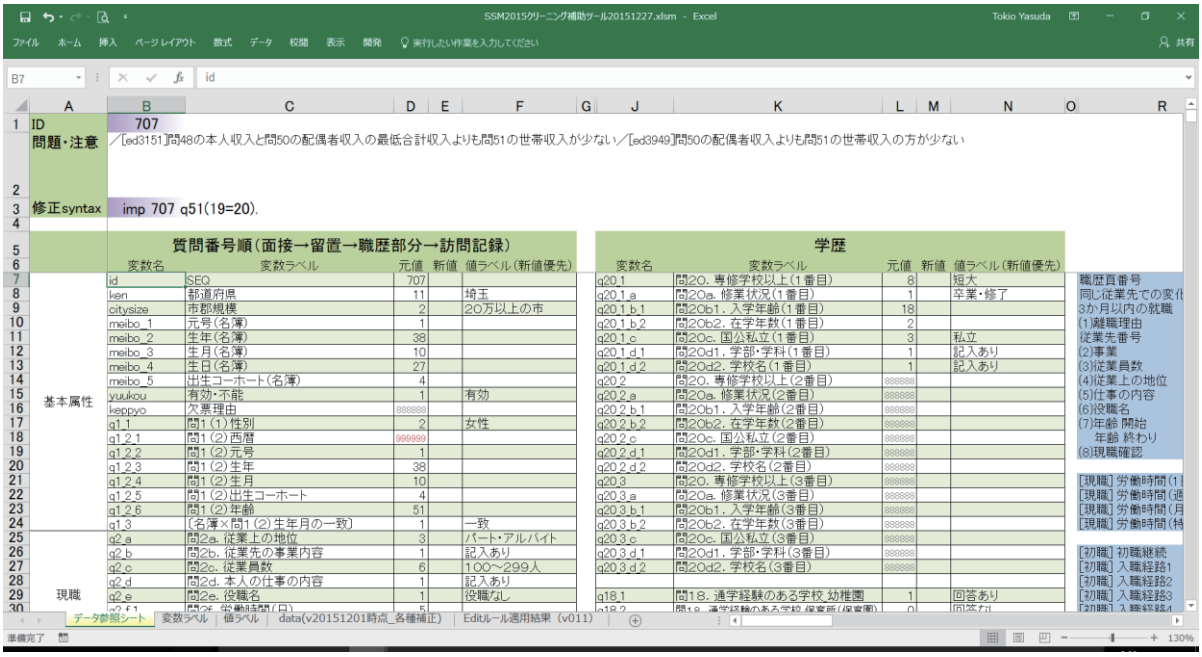

図５　ケースごとの修正の例

ここで用いている Excel のツールでは、ID の欄に ID 番号を入力すると、該当ケースのデータが「元値」の欄に読み込まれる。また、違反があった edit ルールはその形式名（ed3151 など）とともに「Q1 の値が範囲外」「Q4 は非該当のはず」など、違反の内容がわかるラベルを示している。このラベルと既存のデータ（および調査票の原票*[10]）を見ながら、修正したい変数の値を「新値」の欄に入力する。すると、修正内容に応じて修正用のプログラム（ここでは SPSS シンタックス*[11]）が生成されるようになっている。この修正用のプログラムをケースごとに記録して積み上げれば、全体のデータを修正するためのプログラムができあがる。ただし、修正コマンドだけを見てもどのような判断でその修正に決定したのかがわかり

[10] この修正作業にあたる際には、可能な限り調査票の原票を参照できる環境を整えた方がよい。画面上のデータから回答の流れを理解するのは意外と難しい。また、原票からのデータ入力のミスなどが疑われる際には、原票が確認できない環境では修正を保留しなければいけなくなる。修正作業にあたっては可能な限り紙の原票を用意し、それが難しい場合でもスキャニングした原票の画像データは用意した方がよい。

[11] シンタックスの中で用いられている imp コマンドはマクロで作成した新コマンドである。詳細は 5 節で後述する。

にくい場合があるので、意図がわかるようなメモ書きを残しておいた方がよい。

この方式での大きなメリットは、ケース単位で修正作業の分担が可能なので、クリーニング作業の進度の予測・管理が容易になるということがあげられる。粘土細工アプローチで実際に修正作業を行った際には、ほとんどの場合で数 10〜100 ケース程度のまとまりをもって作業を分担し、作業者の進度に応じて分担の割り当てを調整する方法を取った。また、点検事項ごとに分担する因習的なクリーニング方法と異なり、修正の結果が他の分担者に影響を与えることはないので、完全に並行して作業を進めることができる。極論すれば、分担者の数を増やせば増やすほど全体の修正作業は早期に終わることになる。ただし、実際的には修正作業には一定の慣れ（熟練）が必要なので、分担者の数を増やしすぎると効率は悪くなる。また、修正作業中に edit ルールの見直しの必要が生じることもあり、情報共有をスムースにするには、ある程度少人数（10 人程度まで）の分担者が 1 か所に集まって集中的に作業を行った方がよい。

3.4　手順 4：修正後のデータに edit ルールを再適用

全ケースについて修正コマンドの作成が終われば、その修正コマンドをデータに適用すれば修正が完了したことになる。しかし、多くの場合、修正の漏れやミスが発生しているので、異常値がないか再度確認する必要がある。これは、単純にすべての edit ルールをもう一度適用することで確認できる。完全に修正されていれば、全 edit について値が 0 になるはずである。

実際的には、複雑性の高い調査ではクリーニングに相当に熟練していない限り修正ミスは頻出する。修正ミスの頻度を抑えるには、ある程度のケース数を終えた段階で、修正コマンドが作成できたケースについてだけ修正を適用し、どのようなミスが生じているかを分担者に早期に伝えることである。このような機会があれば、その後の同様の修正ミスは抑制される。また、後で示すように修正の実行までを 1 つの Excel ファイルで完結させるシステムを用意した場合には、修正後のデータに edit ルールを適用した結果をリアルタイムに知ることができ、分担者がより効率的に修正ミスの発生を知ることができる。

3.5　手順 5：完了の判断

再修正と edit ルールの再適用を繰り返し、最終的に修正を要する異常値が 0 になれば、クリーニングを完了したと判断できる。もちろん、点検すべき個所を見落としている可能性はあるが、少なくとも点検の必要性を認識し適切に edit ルールを作成した個所については完全に問題がなくなったことを確信できる。

実際的には完了したと思ったが後で見落としが見つかることは少なくない。しかしながら、その場合も柔軟に修正の追加ややり直しができることが粘土細工アプローチの特徴である。

どのような判断でどの変数の値をどのように修正したかはすべて記録されているので、記録から修正を戻すことは容易である。ケースごとに修正を行っているので、特定のケースだけについて修正を破棄したり手直ししたりすることも行いやすい。

3.6 粘土細工アプローチの副次的なメリット

当初から意図したわけではなかったが、実践を通して粘土細工アプローチには様々な副次的メリットがあることがわかった。いくつかのメリットは、手順 1 で最初に edit ルールを論理式で作成することと関わる。この作業は調査票の構造とデータの形式（変数名やコードなど）が決まっていればデータを収集する前から準備することができる。つまり、粘土細工アプローチには実査の前からクリーニングの作業に取りかかれるというメリットがある。調査データの最終的な完成時期を早める意味でこの点は大きい。また、事前にクリーニングのやり方について十分に考慮することになるので、調査票の構造的な不備を発見するきっかけになる場合もある。調査会社との間でデータ納品の仕方（どこまでの修正を調査会社に依頼するか）について起こるトラブルを避けることができる。

また、いくつかの副次的メリットは、手順 3 でケースごとに修正作業を分担することと関わる。ケースごとの分担ではケース番号の順番どおりに進めていくため、原票を確認するための物理的な出し入れ作業の負担が少ない。これに対して、因習的な点検事項ごとのクリーニングではその事項に引っかかったケースを調査票原票の束からばらばらに探してきて出し入れしなければならない。作業的な手間から考えると、意外とこのメリットは大きい。加えて、ケース番号の順番は調査地点（あるいはその地点を担当する調査員）ごとに固まっているので、調査地点（調査員）における偏りを考慮しながらクリーニング作業ができるメリットもあることがわかった。

その他の大きなメリットとして、クリーニング作業自体が研究上の有意義な過程となる可能性が示唆された。粘土細工アプローチでは 1 ケースずつの修正に向かい合うので、疑似的に面接調査員として一人ひとりから回答を得ているような感覚を得ることができる。このような感覚は、研究者が調査員を経験する機会が減っている近年の調査環境では貴重なものと考える。後でデータ分析を行ううえでも、現実的な感覚を失わずに分析を進めるための一助になるだろう。

3.7 粘土細工アプローチのその他の注意点

粘土細工アプローチによるクリーニングを実践するうえで、いくつか典型的に懸念される注意点をまとめておく。第 1 に、edit ルールの作成についてである。edit ルールを論理式で作成することは、通常の社会調査で求められる技術とは異なり、プログラミングの技術に近い。そのため、誰がどのようにこれを担うのかという問題が生じる。しかしながら、典型的

に求められる edit ルールはある程度パターンが決まっている。また、極論すれば、因習的なクリーニングで行われるように集計表などを頼りにして異常ケースの ID を特定し、手入力で異常ケースを示す変数を作成しても、後で行う作業は同じことになるので、非常にややこしい場合にはそのような手段もある（ただし、クリーニングの完了を確認するためには、修正データに対してもこれを繰り返さなければならなくなる）。

第 2 に、ケースごとに修正の仕方を決めることで、ケース間で修正方針に齟齬が出るのではないか、という懸念がもたれる。これはある程度そのとおりであり、分担者の間で修正方針の情報はなるべく共有されるほうがよい。ただし、その場合でも注意しなければならないことは、「同じ異常だったとしても、すべてのケースにまったく同じ修正をする必要はない」ということである。実際には、同じ異常でもケースによって違う理由で異常が生じているはずだからである。たとえば、「未婚者なのに結婚年齢を答えている」という同じ状況でも、未婚者が誤答のケースと結婚年齢を答えていることが間違いのケースは混在している。これを画一に「結婚年齢を非該当に修正する」といった方針を決めてしまうと、不当にデータの分散を小さくしてバイアスを産む原因ともなる。したがって、根本的にいえば修正方針にはある程度のぶれがあってよい、と考えるべきである*[12]。どの修正がよいかどうしても選択できない場合には該当の変数に無回答（不明）のコードをあてるしかないが、実際的には周辺的な回答の様子からある程度高い蓋然性で修正すべき内容が推定できることがほとんどである。したがって、ある程度信頼のおける分担者が担当している限り、ケース間の修正のずれは過度に心配しない方がよい。

第 3 にクリーニングの分担作業をコントロールする統括者について言及しておきたい。粘土細工アプローチは、並行的な分担作業を前提としているので、統括者の存在は重要である。統括者の働きは、粘土細工アプローチの成否を大きく左右するであろう。その意味で、統括者の作業負担を軽減する工夫は重要に思える。edit ルールの作成は統括者が直接行うことが望ましいものの、必ずしもそうでなければならないわけではない。また、ケースごとに分担した修正結果を一つにまとめることは、意外と負担になる。外に出せる作業は外に出して、なるべく統括作業に専念できる環境を整える方がよいであろう。

4．粘土細工アプローチの開発過程

粘土細工アプローチは、複雑性の高い社会調査ではこれまで 4 つの調査データのクリーニングに適用されてきた（表 2）*[13]。基本的なアプローチ方法に違いはないものの、具体的な

[12] Fellegi-Holt の自動クリーニングの場合には、ホットデッキ法でランダム性をもたせることにより、画一的な修正が行われることを避け、データの分散を適正に保持している。このことからも、理論的にある程度ぶれのある修正の方が望ましいことがわかる。

[13] 日本版 General Social Survey 2009 ライフコース調査（JGSS-2009LCS）は、大阪商業大学 JGSS 研究センター（文部科学大臣認定日本版総合的社会調査共同研究拠点）が実施している

運用方法や使用するツールは、その都度発展してきた。ここでは、その発展過程をまとめておく。

表２　粘土細工アプローチを適用した調査

作業年	調査名	調査方法
2008～2009 年	JGSS-2009 ライフコース調査	面接回顧調査
2009～2013 年	NFRJ-08Panel	郵送パネル調査（wave-1,5 のみ留置）
2015～2017 年	2015 年 SSM 調査	面接回顧調査
2017 年	NFRJ-16R	郵送回顧調査

4.1.　2015 年 SSM 調査より前

明確に粘土細工アプローチを意識したクリーニング作業をはじめに行ったのは、JGSS-2009 ライフコース調査である。この調査は SSM 調査と同じように細かく職歴を尋ねる複雑な回顧調査である。ただし、2005 年 SSM 調査の経験から、クリーニングの作業に混乱が起こらないことを意識して調査票を再設計したことや、対象年齢が抽出時で 28～42 歳に限られていたことなどから比較的大きな問題を起こさずにクリーニングシステムを構築することができた。この時点で粘土細工アプローチの大枠はほぼ固まっており、以降、ほぼ同じツール群を改良しながらシステムの開発を進めることになった。とくに大きな収穫は、Fellegi and Holt の原則にもとづいて editing 段階と imputation 段階を明確に区別することやケースごとに修正を進めることが理念的なことのみでなく実践的に可能であることが確認できたことである。ただし、ここでのクリーニングがうまく進められたのは、環境面に恵まれた効果が大きかった。調査原票の管理がしっかりしていたことや、少人数の専属研究員で集中的に作業に取り組めたこと、general edit の作成に際してあらかじめどのような誤答のパターンがありえるのか研究メンバーで洗い出しができていたことなどがあげられる。

粘土細工アプローチがうまく機能することが確かめられたので、次に関わった NFRJ-08Panel のクリーニングにも同じシステムを適用することを試みた。NFRJ-08Panel は 1 年おき 5 時点のパネル調査で家族関係を研究対象とする調査である。この調査のクリーニングは様々な意味で制約や困難が多く、逆の見方をすればシステムを発展させるきっかけとなった。まず一般的に家族調査はクリーニングが複雑になりやすい。家族成員やその関係性に

研究プロジェクトである。共同研究拠点の推進事業と大阪商業大学の支援を受けた。
NFRJ-08Panel は、JSPS 科研費 JP 21243034 の助成を受けて実施され、日本家族社会学会全国家族調査委員会がデータを管理いている。
2015 年 SSM 調査については注 1 のとおり。
NFRJ-16R は、JSPS 科研費 JP26285124 の助成を受けて実施した。

よって回答構造が異なってくるため点検すべき事項が多く、回答間違いも生じやすいからである。さらに、パネル調査のため 1 年おきに調査データが増加していく。パネル調査では単純に点検事項が加速度的に増加することに加えて、前年までのデータで加えた修正を後で見直さなければならない、といったことも生じてくる。さらにこの調査では 5 時点の最初と最後は留置調査で中間 3 時点は郵送調査という変則的な調査方法を用いている。郵送調査では本人以外が回答している可能性なども発生する。そして、この調査のクリーニングの一番の問題点は、研究体制の都合から約 30 名の分担者が 1 か所に集まっての作業はできず、また調査票の原票を参照できない環境で作業をしなければならなかったことにある。

このような困難があったが、その結果としてシステムはより効率的になり、作業の進め方についていくつかの知見を得ることができた。まず、しっかりとしたマニュアルを用意すれば分担者がばらばらに作業を行うことは意外と可能なことがわかった。ただし、やはり調査票原票は参照できることが望ましく、後回しにしていた原票の確認を最後にまとめて行うことは予想以上に手間がかかったとともに、最初から作業中に原票が参照できれば修正方針に悩む時間はもっと短縮できたであろうことがわかった。また、このパネル調査では 5 時点間のすべての組み合わせで整合性を検討した。つまり、wave-5 は wave-4 との整合性だけでなく、wave-1～3 との整合性も検討するといった具合で、5 時点間だと 10 通りの組み合わせを検討したことになる。調査内容の複雑さにもよるが、すべての組み合わせについて手作業で general edit を作成することは、5 時点程度が限界であると感じられた。総じて、パネル調査のクリーニングにかかる負担は横断調査に比べて甚大である。対策としては、途中期間でのクリーニングはある程度制限し基本的なクリーニングだけに留めて、数年分のデータがまとまってから徹底的なクリーニングを行う方が効率はよいと考えられる。また、この調査では最後の wave-5 の調査票で、修正方針を決めるために役立つ質問項目（養子や養父母の有無など）をいくつか挿入した。パネル調査のクリーニングに困難が生じた際には、このような方策も有効に感じられた。

4.2　2015 年 SSM 調査

2015 年 SSM 調査への粘土細工アプローチの適用については、菅澤・保田（2018）に詳述しているのでそちらを参照してほしい。ここでは、その特殊性を簡単に整理しておく。2015 年 SSM 調査では、職業や産業のアフターコーディングを終える前にクリーニングを始めたために、職歴以外のクリーニングと職歴のクリーニングの大きく 2 つに分けて作業を行わなければならなかった。また、その他の部分でもデータの変更や追加が随所でなされ、クリーニング作業は複雑なものになった。このような手続きは言うまでもなく望ましいものではない。完全にそろったデータに対してすべての edit ルールを適用し、ケースごとに全体にわたるクリーニングが行えることが望ましい。ただ、それでも粘土細工アプローチは柔軟に対応する

ことが可能であり、最終的に全 edit で異常値をなくすという明確なゴールに向かうことができた。

職業に関していえば、今回のクリーニングでは職歴全体の中での職業の各段階（いわゆる職業段数）は、単純に複数の質問項目のように扱ってクリーニングを行った。つまり、1 行が 1 回答者のデータである状態でクリーニングを行った。NFRJ-08Panel における各 wave も同じように扱った。しかしながら、これだけ職業段数が多くなるとこのやり方は煩雑であり、職業段数ごとに 1 行となるデータ（パネルデータでいうところのいわゆる long data）を形成した方が効率的であったかもしれない。

また、2015 年 SSM 調査では一部で自由記述の回答（事業内容や仕事内容など）を直接読み取り edit ルールを作成した。これは一定程度役に立ったが、データを修正した後も自由記述内容までは修正しないので、修正後のデータで不必要に edit ルールに引っかかってしまうといった問題が生じた。自由記述内容を含めた対応にはまだ課題が多そうである。

2015 年 SSM 調査は粘土細工アプローチを適用した中で最大のサンプルサイズであった。基本的には、そのぶんだけ修正作業にかかる時間が増えること以外に違いはないのであるが、各ケースにかかる時間がわずかに変わるだけで総作業時間が相当に変わってくるために、少しでもシステムをスリムにすることの重要性を痛感するとともに作業時間の見積もりの難しさが感じられた。

また、この調査では調査実施の中心メンバー以外としてクリーニングを統括するという初めての経験となった。この点はやはり難しさを感じたところである。修正の最終的な判断について責任をもつことができないという点や、調査票の設計についてクリーニングの視点からの意見を通しにくい点などがあげられる。可能な限り、クリーニングの統括者は調査実施の中心メンバーで担われるべきであろう。その意味でも、粘土細工アプローチを汎用的に使用できる環境を整える必要性が強く感じられた。

4.3　2015 年 SSM より後

2015 年 SSM 調査での反省もふまえて、もっとも最近に粘土細工アプローチを適用した調査が NFRJ-16R である。この調査は 15 歳から現在（最長で 45 歳）までの家族経験を郵送調査で尋ねており、該当の選択肢の期間に線を引いてもらうというやや変則的な調査票を用いている（保田 2017）。この調査では時点数が多いため形式的な変数の数が膨大になる。そのため、より効率的にツールを改修する必要があった。具体的には、これまで SPSS と Excel を使い分けて両者の間でやり取りをしていたのを、Excel のみで作業が完結するように変更した。

その結果として、次のような利点が得られた。第 1 に、edit ルールを SPSS シンタックスではなく Excel 関数で作成したことによって Excel 上でデータに修正を加えた際に edit ルール

の適用結果がリアルタイムに参照できるようになった。つまり、データの修正によって当該の edit ルールをクリアできているのか、あるいは新たな問題が生じていないか、といったことが分担者に即座にわかるようになった。このことによって、分担者の負担感が減じられるとともに、統括者による点検・再修正の作業負担も大幅に軽減された。第 2 に、Excel 上で修正済みのデータも記録するようにしたため、修正シンタックスをデータに適用する必要が基本的になくなった。ただし、どのような修正を施したのかという記録を分かりやすく残すことは重要なため、これまでどおり SPSS の imp コマンドによる修正シンタックスは作成・記録するようにした。第 3 に Excel のみで完結しているので、一連のツール群を 1 つのファイルに収めることができるようになった。作業管理上は簡便になる。

一方で Excel のみでの作業にデメリットがないわけでもない。第 1 に、利便性との裏返しでもあるが、リアルタイムに edit ルールの適用結果を示すことによりファイルの動作が相当に重くなってしまった。このため、すべてのケースについて edit ルールの適用結果を演算させることは諦め、現在参照しているケースに限って演算結果を示すように仕様を変更した。ただし、この対策が第 2 のデメリットを産んでいる。この方式では全体に対して edit ルールを適用した結果、どの程度の違反がどのケースに生じているのかといった分布が把握できない。そのため、全体的な分布は別途手作業で確認するようなこととなった。この点は何らかの形で改善しなければならないだろう。第 3 に、やや些末なことかもしれないが、1 つの Excel ファイルにすべてを収めたことでシートの数がかなり多くなり、全体的な構造が把握しにくくなった。

以上のようなデメリットはあるものの、全体的にいえば Excel のみに集約したことのメリットの方が大きく、粘土細工アプローチの一般的な利用のハードルは大きく下がったものと思われる。これまでやや複雑性が高すぎる調査ばかりにシステムを適用してきたが、今後はもう少し一般的な水準での利用を視野に入れてシステムの改修に取り組みたい。

５．粘土細工アプローチのツール

粘土細工アプローチによるクリーニングを実際的に適用するためには、対応したツール群の存在が重要になる。現在、NFRJ-16R のクリーニングに利用した Excel ファイルを基礎として汎用ツールの公開を準備中である*[14]。ここでは、これまで保田が使用してきたツール群の概略を示す。

公開用の汎用ツールは Excel に集約する予定であるが、SPSS との併用で数年間にわたり使

[14] 一方で、同じように粘土細工アプローチを参考としてクリーニングシステムの研究を進めていた羅一等氏（羅 2017）は一つのアプリケーションの形で DCSS というシステムを開発し、公開している（https://www.hepokiki.com/）。ただ、Excel でシステムを用意することには、独立したアプリケーションに比べて誰もがカスタマイズをしやすいという利点がある。複雑性の高い社会調査ではカスタマイズの必要性は比較的高いと考えている。

用してきたツール群の方が、その発想はわかりやすい。ここでは、まず従来のツール群について説明したうえで、それを Excel にどう集約したのかを解説することにする。粘土細工アプローチで用いてきたツール群は表 3 の 4 つのファイルである。名称がわかりにくいが、用いてきた通称のままに示す。

表３　粘土細工アプローチのツール群（従来のもの）

通称	ファイルの種類	用途
edit ルール	Excel（.xlsx）	入力されたコード情報から各変数の range edit と filter edit の SPSS シンタックスを自動生成するとともに、各種の edit ルールを整理する。
補助ツール	マクロ付 Excel（.xlsm）	imputation 段階で分担作業をする際に、指定された ID のデータを呼び出し参照するとともに、修正に対応した SPSS シンタックスを生成する。
imp コマンド	SPSS マクロ・シンタックス（.sps）	SPSS 上で、特定の ID の特定の変数を簡便に修正する imp コマンドを使用できるようにするマクロ・シンタックス。
整理シート	Excel（.xlsx）	editing 段階で各ケースに edit ルールを適用した結果や、imputation 段階の分担作業で生成した修正シンタックスやメモ書きを記録するためのシート。

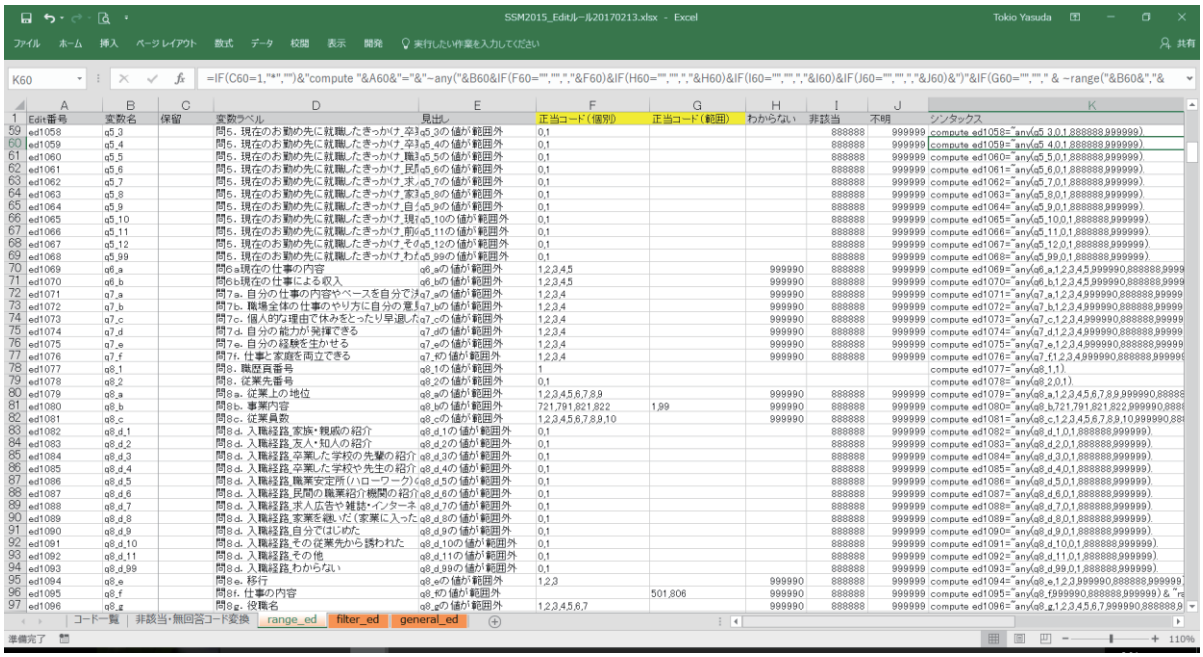

図６　edit ルールの SPSS シンタックスを作成する Excel シートの例（range edit）

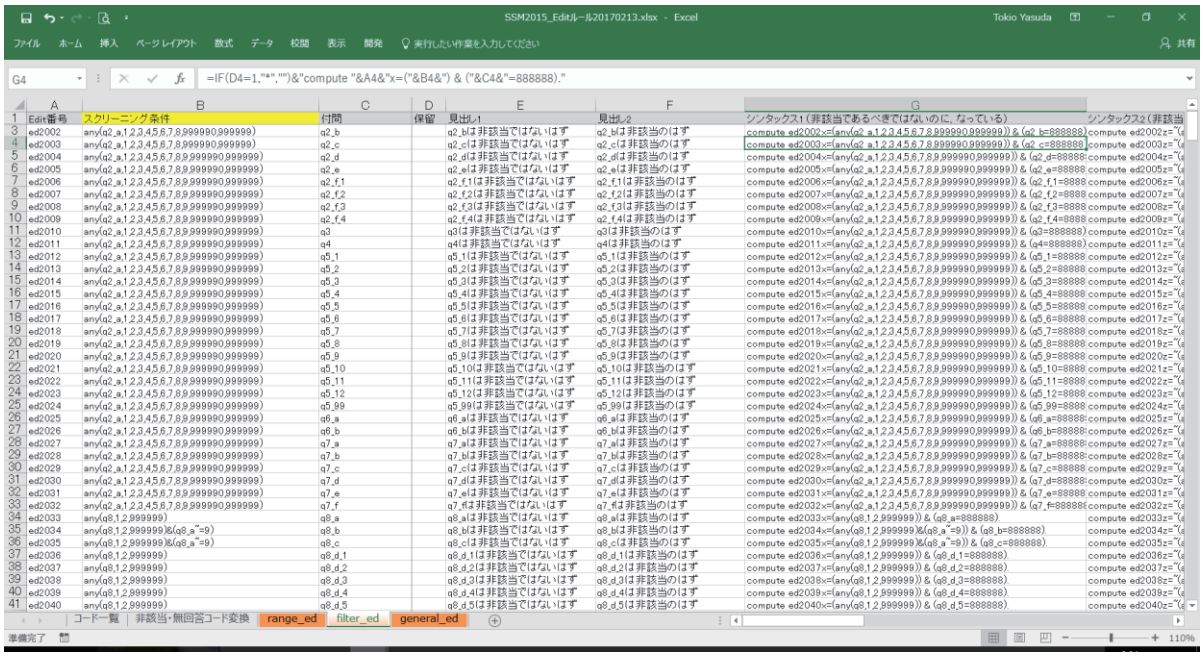

図 7　edit ルールの SPSS シンタックスを作成する Excel シートの例（filter edit）

「edit ルール」の Excel ファイルは、手順 1（表 1 参照）で作成する各種の edit ルールを整理するためのものである。（従来のツールでは）edit ルールは SPSS シンタックスで記述している。general edit は基本的に一つずつ考えて書くしかないが、range edit は回答コードの範囲（および非該当・無回答コード）さえわかれば生成できるので、情報から Excel 関数で SPSS シンタックスを自動生成する（図 6）。同様に、filter edit も非該当の条件から SPSS シンタックスを自動生成する（図 7）。作り出された edit ルールの SPSS シンタックスは、手順 2 でまとめてデータに適用される。

「補助ツール」は手順 3 で分担者が修正の判断をするために使用する。ロー・データおよび変数や値のラベルをすべて格納し、ID を入力すれば参照したいケースのデータを呼び出すことができる。「新値」の欄に修正値を入力すると、それに応じた修正シンタックスが自動生成される。分担者はこれを記録する（前掲の図 5 を参照）。

「imp コマンド」は SPSS の新コマンドを作るためのマクロ・シンタックスである。これを実行すると「imp␣ID 番号␣変数名(元値=新値)␣変数名(元値=新値).....」という書式で特定の ID 番号のケースについて複数の変数のリコードを 1 つのコマンドで行えるようになる。同じことは既存のコマンドで複数行のプログラムを書けば実現できるが（do if ID=3095.[改行] recode q1(4=1).[改行]. end if.など）、1 ケースの修正コマンドが複数行にまたがることは作業を煩雑にするので、このようなマクロコマンドを用意した。また、既存コマンドでも「if (id=3095) q1=1.」のように if コマンドで 1 行に収めることもできるが、この場合は修正前の値が記入されないので、後からの点検がやりにくいことと、(元の値を勘違いした)誤った修正を誘発する問題があるため、使用を避けた。他の言語であれば（たとえば Stata）、

既存のコマンドのみで 1 行に収まるものもある。

分担作業で生成した修正シンタックスを適用すれば、データが修正されるが、修正漏れや修正ミスは頻繁に発生するので、点検・修正のプロセスを何度か繰り返す必要がある。そのため分担作業を含めたこのプロセスを時系列的に記録しておくために Excel の「整理シート」を用意した。ここではケースごとに edit ルールの適用結果がどうであったのか、どのような修正を施したのか、あるいはメモ書き等をすべて記録していく。必要に応じて記録は右に伸びる考えである（図 8）。

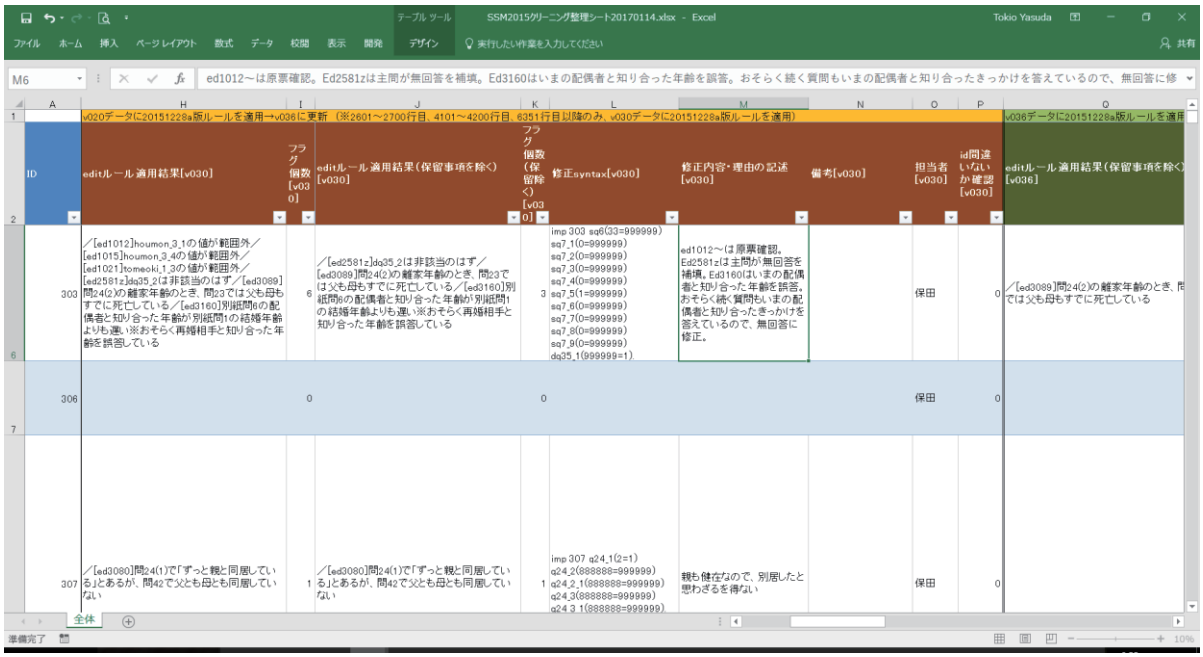

図８　整理シートの例

2015 年 SSM 調査までの調査では、多少のカスタマイズは加えながらも基本的にこの 4 つのファイル群を利用して粘土細工アプローチの理念を実現してきた。すでに記したとおり、続く NFRJ-16R のクリーニングにおいては、1 つの Excel ファイルにこれらのツールを集約した。そのファイルの構成シートは表 4 のように従来のツール群と対応している。全体として意図していることは旧ツール群と同様であるが、クリーニングの手順に沿って a）から順番に作業を行っていくように再編成している。旧ツールと大きく異なる点は、g）ケース参照シートにおいて、データの修正に対応して edit ルールの適用結果が即時に反映されること、およびそこで分担作業した結果が i）クリーニング結果整理シートに自動的に格納されることである（旧ツールでは手作業でコピーしていた）。

表４　粘土細工アプローチの統合的ツール

シート名	用途	対応する旧ツール群
a）修正前 data シート	修正前のロー・データを格納する。	補助ツール
b）変数ラベルシート	各変数のラベルを格納する。	
c）値ラベルシート	各変数の値ラベルを格納する。	
d）range edit シート	range edit を生成・格納する。	edit ルール
e）filter edit シート	filter edit を生成・格納する。	
f）general edit シート	general edit を格納する。	
g）ケース参照シート	特定のケースのデータを呼び出し、修正作業を行う。予備的に imp コマンドを生成する。	補助ツール
h）修正後 data シート	修正後のデータを格納する。	（imp コマンドを不要に）
i）クリーニング結果整理シート	ケースごとに修正前の edit ルールの適用結果、修正シンタックス、修正時のメモ書き、修正後の edit ルールの適用結果、を格納する。	整理シートを自動化

６．今後の課題

粘土細工アプローチによるクリーニング・システムは、2015 年 SSM 調査をはじめとしていくつかの調査で一定の成果を収めたが、課題も少なくない。第 1 に、統括者をどう育てるかという問題がある。整理されていないノウハウをどうまとめて伝達するかは一つの大きな課題である。それとも関係するが、第 2 に edit ルール作成の外部化の問題がある。edit ルールの作成は基本的にはプログラミングの問題なので、情報技術の専門家に外注することは可能なはずである。そのようなルートがうまく作られれば統括者の負担はいくぶん軽くなるであろう。第 3 に、実際的な適用事例が偏っている問題があげられる。異なった様式の複雑な調査や､逆にもっと単純な調査においてシステムがうまく機能するのか検証する必要がある。

［文献］

Delgado-Quintero, Sergio, and Juan-Jose Salazar-Gonzalez. 2008. “A new approach for data editing and imputation,” *Mathematical Methods of Operations Research* 68(3): 407-428.

Fellegi, I.P., and Holt, D. 1976. “A Systematic Approach to Automatic Edit and Imputation,” *Journal of the American Statistical Association* 71(No. 353): 17-35.

Granquist, Leopold, and John G. Kovar. 1997. “Editing of Survey Data: How Much is Enough?” pp. 415-35 in *Survey Measurement and Process Quality*, edited by Lars Lyberg, Paul Biemer, Martin Collins, Edith de Leeuw, Cathryn Dippo, Norbert Schwarz, and Dennis Trewin. John Wiley & Sons.

Groves et al. 2009. *Survey Methodology, second edition*. Wiley.

Herzog, Thomas N., Fritz J. Scheuren, William E. Winkler. 2007. *Data Quality and Record Linkage Techniques*. Springer.
羅一等．2017.「社会調査データの統合データクリーニングシステム開発の研究：DCSS の開発と試用」『第 90 回日本社会学会大会報告要旨集』
菅澤貴之・保田時男．2018.「データ・クリーニング時期別にみたエラー検出傾向に関する基礎的分析」保田時男編『2015 年 SSM 調査報告書 1　調査方法・概要』2015 年 SSM 調査研究会: 143-175.
United Nations. 2006. *Statistical Data Editing Volume 3: Impact on data quality*. United Nations Publication.
Waal, Ton de, and Jeroen Pannekoek, and Sander Scholtus. 2011. *Handbook of Statistical Data Editing and Imputation*. Wiley.
Weisberg, Herbert F. 2005. *The Total Survey Error Approach: A Guide to the New Science of Survey Research*. The University of Chicago Press.
保田時男．2010.「調査データのクリーニング方法に関する提言: Fellegi-Holt の原則に立ち返る」 第 49 回数理社会学会大会.
保田時男．2011.「NFRJ-08Panel における調査票の設計: 研究課題とクリーニングを視野に」『家族社会学研究』23(1): 89-95.
保田時男. 2012.「パネルデータの収集と管理をめぐる方法論的な課題」『理論と方法』vol.27, no.1, pp.85-98.
保田時男. 2017.「回顧式家族調査 NFRJ-16R のねらいと経過」『家族社会学研究』29(2): 216-222.

Development of the Data Editing System for Complicated Social Surveys *

Tokio YASUDA
(Kansai University)

Abstract

In order to deal with complicated social surveys, I developed the practical and effective data editing system which followed the Fellegi-Holt's principle and named it as Clay Modeling Approach. This paper describes the theoretical background, development process, and technical tools of this system. I experienced the difficulties in the data editing of the SSM survey in 2005, and the experience caused my project to develop a new data editing system. The system was applied to the data editing for the SSM survey in 2015, and it gave certain success. The Clay Modeling Approach has also been applied to some other surveys and has evolved to address each difficulty. This paper will be of great use to survey practitioners involved in data editing of social surveys.

Keywords: data cleaning, social survey, Fellegi-Holt, methodology.

* The study was supported by JSPS KAKENHI Grant Number JP25000001.