乱数シミュレーションによる変数選択とロジスティック回帰分析を用いた欠損値データの補完法の提案

片所　強

1　序論

欠損値の補完法として平均値補完法、中央値補完法、k-NN 法（k-Nearest Neighbors : k 最近傍法）などがあげられる。また、これら 3 つの方法と、補完法とはいえないが、欠損値の処理法の 1 つである削除法を併せて比較検討したものとして文献 [1] がある。これをまとめて、文献 [6] ではランダム補完法という新たな方法を提案している。あるいは EM アルゴリズムに基づいた補完法（文献 [8]）や回帰の推定によって欠損値を補う方法（文献 [9]）などがあげられる。

ところが、例えば心理学の分野では質問紙調査によってデータを得ることが多いが、データ解析の際に欠損値が含まれる被験者（回答者）のデータは削除されること（削除法）も少なくない。しかし、SPSS Missing Value Analysis（文献 [4]）といったような欠損値の扱いに関する製品が提供されていることからして、方々の分野で需要があるものだと推察されるが、欠損値の取り扱いについての話題はとりわけ遺伝子解析（ゲノム解析）の分野で盛んなようである。

なお参考までに和文誌で欠損値の補完法について述べられたものとして、Fuzzy c-Varieties 法について議論したもの（文献 [3]）、混合主成分分析モデルを用いて欠損データを予測しようとするもの（文献 [7]）、ニューラルネットワークモデルを用いたもの（文献 [2]）をあげることができる。それから、分散分析における欠損値の影響度について議論されたもの（文献 [5]）は欠損値を補完した際の評価法として非常に参考になる。

本稿ではこれらいずれの方法とも異なる新たな欠損値の補完法を提案する。この方法には 2 つの手順を踏むことになるが、その 1 つ目の手順として変数単位でのランダムサンプリングによって決定係数を最大にする重回帰モデルを作成する。続く 2 つ目の手順として、重回帰分析によって得られた予測値を説明変数とし、補完する対象となる欠損値が含まれている変数を応答変数として多項ロジスティック回帰分析（Multinomial Logistic Regression）を行う。これによって得られる予測確率を参考にして適当な値を欠損値に割り当てるというものである。

そこで、まずは解析法として重回帰分析における変数選択法について紹介し、次いで多項ロジスティック回帰分析を利用して欠損値を補完する方法について述べる。それから解析例として、事例データへの適用とシミュレーション実験の結果を提示することにする。そして最後に、こうした方法についての問題点や改善点を提供して結びとする。

なお、今回の解析およびシミュレーション実験には R 2.10.1 を使用しており、重回帰分析には

lm() を、多項ロジスティック回帰分析には VGAM パッケージに含まれる vglm() を利用している。また、各解析には著者が定義した関数を用いているが、それらは下記 URL で観覧することができる。

http://homepage2.nifty.com/nandemoarchive/toukei_hosoku/RSLM/RSLM.htm

2 解析法

2.1 重回帰における変数選択

重回帰分析において変数選択（モデル選択）を行う方法は増加法、減少法、増減法（ステップワイズ法）などがある。またモデルから変数を増減させる際に基準とする指標として、単一の変数に対する p 値（H_0 : 推定されたパラメータは 0 である）、自由度調整済み決定係数、AIC などを参考にして行うことになるが、ここでは自由度調整済みの決定係数をモデル選択の基準とする。従来法と異なるのは、モデルに組み込まれる変数の選択をランダムに行うという点である。以下にそのステップを示す。

■ステップ 1　解析の対象となる p 個の説明変数（$X_1, X_2, ..., X_p$）の中からいくつの変数を抽出するか（モデルに組み込むか）、その抽出数 s をランダムに決定する。ただし抽出数である s は $1 \leq s \leq p$ の整数値である。

■ステップ 2　続いて X_1 から X_p までの、どの変数を抽出するかをランダムに決定する（変数を重複がないようにランダムサンプリングする）。

■ステップ 3　変数の抽出数と、どの変数を抽出するか決定されたら、それらを説明変数として重回帰分析を行う。

■ステップ 4　ステップ 1 からステップ 3 までを任意の数だけ繰り返し行い（例えば 1000 回）、最終的に最も大きな決定係数（自由度調整済み決定係数）を有していたモデル（変数の組み合わせ）を採用する。

ここでステップ 1 およびステップ 2 において、抽出数と、どの変数を抽出するかという変数の添え字番号をランダムに決定するためには R の sample() という関数を用いると便利である。例えば sample(1:10, 5, replace=FALSE) とコマンドすれば、1 から 10 までの整数値の中から重複を許さずに 5 つの整数値がランダムサンプリングされることになる。仮に解析の対象となる説明変数の数が $p = 9$ ならば、sample(1:9, sample(1:9, 1)) とコマンドすることでステップ 1 とステップ 2 の作業を行うことになる。

実際に以上の方法にて変数選択を行った結果の一部を以下に示す。なお、ここでは最終列に応答変数を含めた 100 行 12 列のデータ行列を用いている。つまり、第 1 列から第 11 列までの 11 個の変数が解析の対象となる説明変数群ということになる。また、繰り返し数は 100 回としている。最終的に（枠内の出力表示では Var となっているが）$X_9, X_5, X_4, X_{10}, X_{11}, X_1, X_2$ といった

7 個の変数を説明変数としたモデルが、100 回の繰り返しの中で最も大きな決定係数を有していたということになる。

```
> dat[1:5,]
     [,1] [,2] [,3] [,4] [,5] [,6] [,7] [,8] [,9] [,10] [,11] [,12]
[1,]    1    2    3    2    2    4    1    1    1     1     1     1
[2,]    2    2    2    3    2    1    2    1    1     1     2     2
[3,]    4    4    4    4    5    4    2    3    5     4     4     3
・・・  （以下省略）  ・・・
> rslm.simulate(dat, 100)
model.formula:
Var12 ~ Var2 + Var4 + Var1 + Var9 + Var10
Adj.R.Squared:
[1] 0.4277309
model.formula:
Var12 ~ Var3 + Var4 + Var9 + Var5 + Var1 + Var2
Adj.R.Squared:
[1] 0.466847
・・・  （途中省略）  ・・・
model.formula:
Var12 ~ Var9 + Var5 + Var4 + Var10 + Var11 + Var1 + Var2
Adj.R.Squared:
[1] 0.5327687
Complete!!
```

2.2　多項ロジットモデルによる欠損値補完

ここから紹介する多項ロジスティック回帰分析（多項ロジットモデル）による欠損値の補完法は、既に紹介した変数ランダムサンプリングによる重回帰分析の変数選択を応用している。便宜上、ここではこの変数選択法を RSLM（Random Sampling Linear Model）と呼ぶことにする。そしてこの RSLM によって採用されたモデル（最終的に最も決定係数が高かったモデル）から得られる予測値を RSLM 予測値と呼ぶことにする。

さて、多項ロジットモデルによる欠損値補完のステップは次のようになる。

■ステップ 1　補完したい欠損値の含まれる変数を応答変数とし、それ以外の変数を説明変数の候補とし、RSLM によって最適なモデルを探索する。なお、この段階では欠損値の含まれている

データ行は全て削除しておく（今回は、説明変数群には欠損値がないものとする)。ここで得られた最適なモデルを RSLM モデルと呼ぶことにする。

■ステップ 2　ステップ 1 で得られた RSLM 予測値を説明変数とし、同じくステップ 1 で用いられた応答変数を多項ロジットモデルの応答変数として解析する。ここで得られたモデルを MLogit モデルと呼ぶことにする。

■ステップ 3　ステップ 1 で削除されたデータ行列について、RSLM モデルを当てはめて欠損値を RSLM 予測値で補完する。これを RSLM 補完値と呼ぶことにする。

■ステップ 4　ステップ 3 で補完された欠損値、すなわち RSLM 補完値について MLogit モデルを当てはめて予測確率を求める。

■ステップ 5　ステップ 4 で求められた予測値に基づき、適当な値を最終的な欠損値の補完値として割り当てる。これを MLogit 補完値と呼ぶことにする。

以上のステップに従って、R で実行した結果を以下に示す。ここでは解析の対象となるデータ行列は 104 行 12 列で第 12 列に欠損値が 4 つ含まれているようなものである。ステップ 1 で使用するデータセット（欠損値のある 4 行を除いたもの）を `dat` とし、ステップ 3 で用いるデータセット（欠損値のある 4 行 12 列のデータ行列）を `dat2` としている。

```
> dat
     [,1] [,2] [,3] [,4] [,5] [,6] [,7] [,8] [,9] [,10] [,11] [,12]
[1,]    1    2    3    2    2    4    1    1    1     1     1     1
[2,]    2    2    2    3    2    1    2    1    1     1     2     2
・・・　（以下省略）　・・・
> dat2
     [,1] [,2] [,3] [,4] [,5] [,6] [,7] [,8] [,9] [,10] [,11] [,12]
[1,]    2    2    2    3    2    1    1    2    3     4     2    NA
[2,]    4    3    4    4    4    2    2    4    2     1     4    NA
[3,]    4    3    3    4    4    4    4    4    4     5     5    NA
[4,]    3    3    2    4    2    2    1    1    4     5     2    NA

# ステップ 1　RSLM モデルを作り、RSLM 予測値を得る
> RSLM.model <- rslm.simulate(dat, 100)
・・・　（途中省略）　・・・
model.formula:
Var12 ~ Var11 + Var6 + Var4 + Var5 + Var9 + Var1
Adj.R.Squared:
```

```
[1] 0.5328212
Complete!!
> RSLM.PV <- predict(RSLM.model)

# ステップ 2　MLogit モデルを作る
> library(VGAM)
> Y <- as.factor(dat[,12])
> MLogit.model <- vglm(Y ~ RSLM.PV, multinomial)
> predict(MLogit.model, type="response")[1:5,]
       1      2      3      4
1 0.9143 0.0856 0.0000 0.0000
2 0.8780 0.1219 0.0001 0.0000
3 0.0017 0.0549 0.7330 0.2104
・・・　（100 行にもわたるので最初の 3 行だけ表示）　・・・

# ステップ 3　欠損値を RSLM 補完値に置き換える
> colnames(dat2) <- paste("Var", 1:12, sep="")
> dat2 <- data.frame(dat2)
> (MV.RSLM.PV <- predict(RSLM.model, dat2))
       1        2        3        4
1.148406 2.211287 2.898929 1.530803

# ステップ 4　RSLM 補完値に対して MLogit モデルを当てはめる
> predict(MLogit.model, list(RSLM.PV=MV.RSLM.PV), type="response")
       1      2      3      4
1 0.7508 0.2483 0.0009 0.0000
2 0.1474 0.6260 0.2170 0.0097
3 0.0048 0.1074 0.7371 0.1507
4 0.5424 0.4494 0.0081 0.0001
```

R の実行例のステップ 4 において、欠損値 NA らをそれぞれ NA_1, NA_2, NA_3, NA_4 とすると、NA_1 が「1」である確率は 0.7508（75%）であり、「2」である確率は 0.2483（25%）、「3」である確率は 0.0009（0%）、「4」である確率は 0.0000（0%）となっていることが確認できる。すなわち NA_1 に割り当てるべき値は「1」となる。同様にして NA_2 には「2」を、NA_3 には「3」を、NA_4 には「1 が割り当てられることになる。

3　解析例

3.1　事例データへの適用

前述してきたような RSLM における 4 ステップ、および多項ロジットモデルによる欠損値補完における 5 つのステップをまとめて実行するための関数 `mlri2()` を用いて、事例データへの適用例を紹介する。なお、この `mlri2()` は序論で紹介した著者の web サイトでソースコードを観覧することができる。

前節では解析手順を示すために、欠損値を削除したデータセット `dat` と欠損値を含むデータセット `dat2` を別々に用意して解析した。しかし、関数 `mlri2()` ならば欠損値を混みにしたデータセットをそのまま指定すればよく、前節での結果と同じものが得られる。

使い方は簡単で、欠損値を含む応答変数を最終列に置いたデータ行列を第 1 引数に、RSLM における繰り返し数を第 2 引数に指定するだけでよい。この関数を実行すると大量のアウトプットが得られるが、その細かい見方はやはり著者の web サイト上で見ることができる。ここでは現場で用いる際に必要な部分のみを提示する。

```
$ANA.RESULT$TABLE # リザルトテーブル
 0  1
63 37

$ANA.RESULT$PROBABILITY # リザルト確率
   0    1
0.63 0.37

$MIS.RESULT$FITTED.VALUE # 当てはめ値
[1] 1 2 3 1
```

リザルトテーブルと称している部分は MLogit モデルによって得られた予測確率に基づいて割り当てられた値と実測値との差をとって、それを集計したものである。つまり「0」というのは割り当て値と実測値の差が 0 であることを意味しており、「1」以上の度数はなるべく少ない方が良い割り当て値を得ていることになる。

リザルト確率は全集計数（$63 + 37 = 100$）でそれぞれの度数を割ったものである（例えば $63/100 = 0.63$）。これによれば、完全に実測値と一致する値を補完できているのは 6 割程度であり、実測値と 1 だけずれている値を補完しているのが 4 割程度であることがわかる。

当てはめ値というのが、NA_1 から NA_{11} までそれぞれの欠損値に対して割り当てられるべき補完値である。実際にはこの当てはめ値がどれだけ的を射ているものかを、リザルト確率を参考に

して判断することになる。

3.2 シミュレーション実験

ここで改めて欠損値を除いたデータセット dat を用いて、応答変数の適当な値をいくつか伏せて、その値をこの方法でどれほど再現できるかをシミュレーションしてみる。試しに (a) 値を伏せる（欠損値とみなす）数と (b)RSLM の繰り返し数を変化させて、どれほど結果に違いが出るかを検証する。その結果を表 1 から表 4 までに示す。

各表の伏せた値と補完値の差の割合といのは、前述したリザルト確率にあたるものである。例えば表 1 の 1 回目では 70% が正しく元の値（伏せた値）を補完できていることを示している。いずれの結果の表からも、RSLM を実行した時点で決定係数が 0.5 から 0.6 程度ならば伏せた値への判別的中率はほぼ 0.5 から 0.6 の値を維持しているようであることが確認できる。

また、サンプルサイズが 100 のデータセットにおいて欠損数（NMV, Number of Missing Values）が 10 もしくは 20 ならば、判別的中率（DR, Discrimination Rates）に大きな影響はみられない（NMV=10 のとき DR=0.55、NMV=20 のとき DR=0.54)。RSLM における繰り返し数についても、100 回と 1000 回の繰り返し数（RN, Repeated Numbers）を比べて両者によって違いはないといってよい。NMV=10 のときに RN=100 でも RN=1000 でも変わらず、NMV=20 のときも同様である（NMV=20 のとき、それぞれ DR=0.54, DR=0.55 である)。

表 1　伏せた値の数＝ 10，RSLM の繰り返し数＝ 100

	伏せた値と補完値の差の割合			
	0	1	2	決定係数
1 回目	0.70	0.30	0.00	0.502
2 回目	0.50	0.50	0.00	0.537
3 回目	0.40	0.60	0.00	0.571
4 回目	0.50	0.50	0.00	0.523
5 回目	0.40	0.60	0.00	0.526
6 回目	0.40	0.50	0.10	0.534
7 回目	0.60	0.30	0.10	0.574
8 回目	0.50	0.50	0.00	0.537
9 回目	0.80	0.20	0.00	0.508
10 回目	0.70	0.30	0.00	0.536
平均値	0.55	0.43	0.02	0.535

表 2　伏せた値の数＝ 10，RSLM の繰り返し数＝ 1000

	伏せた値と補完値の差の割合			
	0	1	2	決定係数
1 回目	0.50	0.50	0.00	0.492
2 回目	0.60	0.40	0.00	0.554
3 回目	0.70	0.30	0.00	0.529
4 回目	0.50	0.50	0.00	0.524
5 回目	0.70	0.30	0.00	0.546
6 回目	0.60	0.40	0.00	0.547
7 回目	0.40	0.60	0.00	0.540
8 回目	0.60	0.30	0.10	0.564
9 回目	0.50	0.50	0.00	0.548
10 回目	0.40	0.60	0.00	0.575
平均値	0.55	0.44	0.01	0.542

表 3　伏せた値の数＝ 20，RSLM の繰り返し数＝ 100

	伏せた値と補完値の差の割合			
	0	1	2	決定係数
1 回目	0.45	0.45	0.10	0.566
2 回目	0.65	0.35	0.00	0.526
3 回目	0.55	0.45	0.00	0.562
4 回目	0.55	0.45	0.00	0.525
5 回目	0.65	0.35	0.00	0.567
6 回目	0.55	0.45	0.00	0.498
7 回目	0.50	0.50	0.00	0.566
8 回目	0.50	0.45	0.05	0.572
9 回目	0.50	0.45	0.05	0.593
10 回目	0.50	0.50	0.00	0.506
平均値	0.54	0.44	0.02	0.548

表 4　伏せた値の数＝ 20，RSLM の繰り返し数＝ 1000

	伏せた値と補完値の差の割合			
	0	1	2	決定係数
1 回目	0.35	0.60	0.05	0.571
2 回目	0.50	0.45	0.05	0.511
3 回目	0.60	0.40	0.00	0.535
4 回目	0.50	0.50	0.00	0.540
5 回目	0.70	0.30	0.00	0.534
6 回目	0.60	0.35	0.05	0.595
7 回目	0.60	0.40	0.00	0.523
8 回目	0.60	0.35	0.05	0.524
9 回目	0.55	0.40	0.05	0.544
10 回目	0.45	0.55	0.00	0.584
平均値	0.55	0.43	0.03	0.546

4　問題と課題

RSLM の実行について、今回は説明変数が連続型である場合のみ（重回帰モデル）を扱ったが、説明変数がカテゴリカル型である場合（分散分析モデル）や、あるいは両者が混在しているとき（共分散分析モデル）においても有効な方法であるかどうか検討する必要がある。また、いずれのモデルを扱う場合であっても、本稿で示した補完法は RSLM のモデル選択で決定係数を増加させるような説明変数が含まれていなければならない。変数をランダムサンプリングするという作業を行うことで、1 度の処理で大量の説明変数群を扱うことができるが、実際には説明変数の中にも欠損値が存在している場合が多い。したがって、説明変数の欠損値をどのように扱うかが問題となる。

一方で RSLM の変数選択（モデル選択）の際の指標として、今回は決定係数を採用したが、AIC によるモデル選択を行った場合や尤度比検定によるモデル選択とでの違いを比較検討してみる必要もあるだろう。

本稿で提示した事例データへシミュレーション実験を行った結果、決定係数が 0.6 程度のモデルが得られているならば、補完値のとして割り当てられた値の判別的中率も 0.6 程度を維持していた。これはあくまで事例なので、他の異なった性質のデータセットでどのような変化が生じるかは検討の余地がある。理論的に RSLM で低い決定係数（例えば $R^2 \leq 0.3$）しか有していないモデルしか選ばれなかったとしたら、そこから適当な補完値は得られない。特に心理学系の調査データでは、高い決定係数を有するモデルが得られることが多くはないので、この点を考慮してさらに適用の可能性を探っていく必要がある。

参考文献

[1] Edger Acuna and Caroline Rodriguez. The treatment of missing values and its effect in the classifier accuracy. http://academic.uprm.edu/eacuna/IFCS04r.pdf.

[2] 服部環. ニューラルネットワークを用いた欠損値の補完. 宇都宮大学教育学部紀要, Vol. 48(1), pp. 133–143, 1998.

[3] 本田克宏, 杉浦伸和, 市橋秀友, 荒木昭一, 九津見洋. 最小 2 乗基準を用いた fuzzy *c*-varieties 法における欠損値の処理法. 日本ファジィ学会誌, Vol. 13(6), pp. 680–688, 2001.

[4] 稲葉由之. Spss missing value analysis の紹介. 計算機統計学, Vol. 13(1), pp. 71–75, 2000.

[5] 岩崎学. 分散分析における欠損値の影響評価に有効なグラフィカル表現法. 計算機統計学, Vol. 7(1), pp. 29–36, 1994.

[6] 金子拓也. データマイニングにおける新しい欠損値補完方法の提案. 電子情報通信学会論文誌 D-II, Vol. J-88-D-II(4), pp. 675–686, 2005.

[7] 大羽成征, 佐藤雅昭, 石井信. 混合主成分分析モデルによる欠側データ予測. 電子情報通信学会技術研究報告 NC, Vol. 101(736), pp. 181–186, 2002.

[8] Ming Ouyang, William J. Welsh, Panos Georgopulos. Gaussian mixture clustering and imputation of microarray data. *Bioinformatics*, Vol. 20(6), pp. 917–923, 2000.

[9] Xiaobo Zhou, Xiaodong Wang, Edward R. Dougherty. Missing-value estimation using linear and non-linear regression with bayesian gene selection. *Bioinformatics*, Vol. 19(17), pp. 2302–2307, 2003.