Modern Methods in Biostatistics and Epidemiology
Missing data in observational and randomized studies
Lab 3 Sample Solution

Nicholas J. Horton
Amherst College

June 10, 2014

Part A: Assessing the impact of different missing data mechanisms

We've undertaken simulation studies assuming MCAR and assuming MAR (depending on X_1). In this lab, you will extend this in two ways:

MAR-Y simulating 50% missingness of X_2 related to the value of Y (hint: recall that under the model I posited, the expected value for Y is equal to 0.5), and

NINR simulating 50% missingness of X_2 that is related to the unobserved value of X_2.

For each of these models, you will compare results from 200 simulations each of 250 observations for each of 10 imputations (for the imputation model) as well as the complete case estimator.

1. Before you begin, what would you expect in terms of bias and efficiency for the two estimators (imputation and complete case) for each of the two new scenarios?

2. Use the following program as a template for the four programs you will create (suggested naming scheme `simmaryimpute, simmarycc, simninrimpute, simninrcc`)

```
. capture program drop simmiss
. program define simmiss
. syntax [, obs(integer 250)]
.         drop _all
.         matrix c = (1, 0.7378648, 1, 0.1054093, 0.6, 1)
.         matrix m = (0.5, 0, 0)
.         matrix sd = (1.897, 1, 1)
.         drawnorm y x1 x2, n(`obs') corr(c) cstorage(lower) means(m) sds(sd)
.         gen mynorm=rnormal()
.         * generates 50% missingness on average since E[X1]=0
.         replace x2=. if x1 < mynorm
.         mi set mlong
.         mi register imputed x2
.         mi register regular x1 y
```

```
.         mi impute regress x2 x1 y, add(10)
.         mi estimate, post: regress y x1 x2
. end
.
. simulate _b, reps(100): simmiss
. summarize
```

(Hint: to save typing, this can be copied from `https://www.amherst.edu/~nhorton/data/simmarx1.do`).

Here are the programs: SIMMARYCC

```
. capture program drop simmarycc

. program define simmarycc
  1. syntax [, obs(integer 250)]
  2. drop _all
  3. matrix c = (1, 0.7378648, 1, 0.1054093, 0.6, 1)
  4. matrix m = (0.5, 0, 0)
  5. matrix sd = (1.897, 1, 1)
  6. drawnorm y x1 x2, n(`obs') corr(c) cstorage(lower) means(m) sds(sd)
  7. gen mynorm=rnormal()
  8. * censor values with P(missing X_2)=0.5 (since E[Y]=0.5)
. replace x2=. if y < mynorm + 0.5
  9.         regress y x1 x2
 10. end

.
. simulate _b, reps(100): simmarycc

      command:  simmarycc

Simulations (100)
----+--- 1 ---+--- 2 ---+--- 3 ---+--- 4 ---+--- 5
..................................................    50
..................................................   100

. summarize

    Variable |       Obs        Mean    Std. Dev.       Min        Max
-------------+--------------------------------------------------------
       _b_x1 |       100    1.598135    .1265214   1.313072   1.853166
       _b_x2 |       100   -.8086218    .1093638  -1.031775  -.5507689
     _b_cons |       100     1.05925    .0899044    .779412   1.243547

.
```

SIMMARYIMPUTE

```
. capture program drop simmaryimpute

. program define simmaryimpute
  1. syntax [, obs(integer 250)]
  2. drop _all
  3. matrix c = (1, 0.7378648, 1, 0.1054093, 0.6, 1)
  4. matrix m = (0.5, 0, 0)
  5. matrix sd = (1.897, 1, 1)
  6. drawnorm y x1 x2, n(`obs') corr(c) cstorage(lower) means(m) sds(sd)
  7. gen mynorm=rnormal()
  8. * censor values with P(missing X_2)=0.5 (since E[Y]=0.5)
. replace x2=. if y < mynorm + 0.5
  9.        mi set mlong
 10.        mi register imputed x2
 11.        mi register regular x1 y
 12.        mi impute regress x2 x1 y, add(10)
 13.        mi estimate, post: regress y x1 x2
 14. end

. simulate _b, reps(100): simmaryimpute

      command:  simmaryimpute

Simulations (100)
----+--- 1 ---+--- 2 ---+--- 3 ---+--- 4 ---+--- 5
..................................................    50
..................................................   100

. summarize

    Variable |       Obs        Mean    Std. Dev.       Min        Max
-------------+--------------------------------------------------------
       _b_x1 |       100    2.000456     .097344   1.775537   2.246564
       _b_x2 |       100   -.9874669    .1080835  -1.228736   -.714845
     _b_cons |       100    .4983182    .0939299   .2622764   .6989068

.
```

SIMNINRCC

```
. capture program drop simninrcc

. program define simninrcc
  1. syntax [, obs(integer 250)]
```

```
  2. drop _all
  3. matrix c = (1, 0.7378648, 1, 0.1054093, 0.6, 1)
  4. matrix m = (0.5, 0, 0)
  5. matrix sd = (1.897, 1, 1)
  6. drawnorm y x1 x2, n(`obs') corr(c) cstorage(lower) means(m) sds(sd)
  7. gen mynorm=rnormal()
  8. * censor values with P(missing X2)=0.5 (since E[X2]=0)
. replace x2=. if x2 < mynorm
  9. regress y x1 x2
 10. end

.
. simulate _b, reps(100): simninrcc

      command:  simninrcc

Simulations (100)
----+--- 1 ---+--- 2 ---+--- 3 ---+--- 4 ---+--- 5
..................................................    50
..................................................   100

. summarize

    Variable |       Obs        Mean    Std. Dev.       Min        Max
-------------+--------------------------------------------------------
       _b_x1 |       100     2.00859    .1220782   1.660506   2.265304
       _b_x2 |       100   -1.006355    .1396343  -1.335865  -.6780321
     _b_cons |       100    .4942574    .0976302   .2506156   .7316667
```

SIMNINRIMPUTE

```
. capture program drop simninrimpute

. program define simninrimpute
  1. syntax [, obs(integer 250)]
  2. drop _all
  3. matrix c = (1, 0.7378648, 1, 0.1054093, 0.6, 1)
  4. matrix m = (0.5, 0, 0)
  5. matrix sd = (1.897, 1, 1)
  6. drawnorm y x1 x2, n(`obs') corr(c) cstorage(lower) means(m) sds(sd)
  7. gen mynorm=rnormal()
  8. * censor values with P(missing X2)=0.5 (since E[X2]=0)
. replace x2=. if x2 < mynorm
  9.         mi set mlong
 10.         mi register imputed x2
```

```
 11.         mi register regular x1 y
 12.         mi impute regress x2 x1 y, add(10)
 13.         mi estimate, post: regress y x1 x2
 14. end

.
. simulate _b, reps(100): simninrimpute

      command:  simninrimpute

Simulations (100)
----+--- 1 ---+--- 2 ---+--- 3 ---+--- 4 ---+--- 5
..................................................    50
..................................................   100

. summarize

    Variable |       Obs        Mean    Std. Dev.       Min        Max
-------------+--------------------------------------------------------
       _b_x1 |       100    1.938135    .0988907   1.629458   2.196218
       _b_x2 |       100   -1.065563    .1298198   -1.38654  -.7334127
     _b_cons |       100    .7877831    .0869752   .5722428   .9939342
```

3. How do you summarize your results? Where did you see bias? Where did you see efficiency gains?
4. What do you conclude?

Part B: Accounting for missingness with a categorical missing value

We will continue work started in the first lab to analyse the `routine` dataset to shed light on the question of whether it is possible to predict the length of stay (in days, `los`) for these subjects as a function of whether it was a routine discharge (`routine`), age (in years), weekend admission (`aweekend`), gender `female`), number of medical diagnoses (`ndx`) and subject race (partially observed, `race`, where 1=white, 2=black, 3=hispanic, 4=other).

But this time, rather than just excluding the subjects missing `race`, we will model them using two separate approaches: predictive mean matching as well as through a multinomial logit model.

We begin by reading in the dataset and keeping only these 6 variables.

```
. use https://www.amherst.edu/~nhorton/data/routine
. keep routine age aweekend female los ndx race

. summarize

    Variable |       Obs        Mean    Std. Dev.       Min        Max
-------------+--------------------------------------------------------
```

```
         age |     13477    16.32196    2.709657         10         20
    aweekend |     13477    .1964087    .3972959          0          1
      female |     13477    .5362469    .4987029          0          1
         los |     13477    6.459375    11.89629          0        339
         ndx |     13477    3.452697    1.994336          1         16
-------------+---------------------------------------------------------
        race |     11268    1.523518    .8767465          1          4
     routine |     13477    .8645841    .3421799          0          1
```

1. As before, add labels to ensure that the `race` variable is clearer (hint: use the `label define` and `label values` commands).

```
. label define racegrp 1 "white" 2 "black" 3 "hispanic" 4 "other"
. label values race racegrp
```

2. Fit and save the regression coefficients for the complete case model:

```
. regress los routine age female ndx i.race
. estimates store cc
```

```
      Source |       SS       df       MS              Number of obs =   11268
-------------+------------------------------           F(  7, 11260) =   43.78
       Model |  46082.4603     7  6583.20861           Prob > F      =  0.0000
    Residual |  1693284.39 11260  150.380496           R-squared     =  0.0265
-------------+------------------------------           Adj R-squared =  0.0259
       Total |  1739366.85 11267  154.377106           Root MSE      =  12.263

------------------------------------------------------------------------------
         los |      Coef.   Std. Err.      t    P>|t|     [95% Conf. Interval]
-------------+----------------------------------------------------------------
     routine |  -1.880481   .3366137    -5.59   0.000    -2.540303    -1.22066
         age |  -.4676682   .0426833   -10.96   0.000     -.551335   -.3840014
      female |  -1.030675   .2322176    -4.44   0.000    -1.485862   -.5754881
         ndx |   .2994255   .0583796     5.13   0.000     .1849912    .4138597
             |
        race |
          2  |   3.103983   .3209864     9.67   0.000     2.474794    3.733173
          3  |   1.000233   .3862419     2.59   0.010     .2431309    1.757334
          4  |   2.567519    .525042     4.89   0.000     1.538345    3.596693
             |
       _cons |   14.67111   .7963618    18.42   0.000     13.11011    16.23212
------------------------------------------------------------------------------
```

3. Generate 25 imputations using a predictive mean matching (PMM) algorithm, with random seed set to 2001, then fit the linear regression model using these imputed values. What is the largest fraction of missing information? Be sure to save the results as `pmm`.

```
. mi set wide
. mi register imputed race
. mi register regular los routine age female ndx

. mi impute pmm race los routine age female ndx, add(25) rseed(2001)

Univariate imputation                     Imputations =       25
Predictive mean matching                        added =       25
Imputed: m=1 through m=25                     updated =        0

                                    Nearest neighbors =        1

------------------------------------------------------------------
                   |               Observations per m
                   |----------------------------------------------
          Variable |   Complete   Incomplete   Imputed |     Total
-------------------+-----------------------------------+----------
              race |      11268         2209      2209 |     13477
------------------------------------------------------------------
(complete + incomplete = total; imputed is the minimum across m
 of the number of filled-in observations.)

. mi estimate, post: regress los routine age female ndx i.race
. estimates store pmm

Multiple-imputation estimates                   Imputations     =         25
Linear regression                               Number of obs   =      13477
                                                Average RVI     =     0.0640
                                                Largest FMI     =     0.1548
                                                Complete DF     =      13469
DF adjustment:   Small sample                   DF:     min     =     939.09
                                                        avg     =    8819.25
                                                        max     =   13435.57
Model F test:       Equal FMI                   F(   7, 9696.7) =      46.81
Within VCE type:          OLS                   Prob > F        =     0.0000

------------------------------------------------------------------------------
         los |      Coef.   Std. Err.      t    P>|t|     [95% Conf. Interval]
-------------+----------------------------------------------------------------
     routine |   -1.73198   .2976549    -5.82   0.000    -2.315425   -1.148534
         age |  -.4494534    .037678   -11.93   0.000    -.5233075   -.3755993
      female |  -1.001961   .2036661    -4.92   0.000    -1.401175   -.6027469
         ndx |    .260339   .0513417     5.07   0.000     .1597021     .360976
             |
        race |
          2  |   2.876088   .2998337     9.59   0.000     2.287969    3.464207
```

```
           3  |   .9198051   .3650181     2.52   0.012     .2035629    1.636047
           4  |   2.384354   .5005394     4.76   0.000     1.402049     3.36666
              |
       _cons |   14.25307   .7039073    20.25   0.000     12.87331    15.63283
------------------------------------------------------------------------------
```

4. Generate 25 imputations using the multinomial (mlogit) function, with random seed set to 2002, then fit the linear regression model using these imputed values. What is the largest fraction of missing information? Be sure to save the results as `mlogit`.

```
. mi impute mlogit race los routine age female ndx, replace rseed(2002)

Univariate imputation                        Imputations =       25
Multinomial logistic regression                    added =        0
Imputed: m=1 through m=25                        updated =       25

------------------------------------------------------------------
                   |               Observations per m
                   |----------------------------------------------
          Variable |   Complete   Incomplete   Imputed |     Total
-------------------+-----------------------------------+----------
              race |      11268         2209      2209 |     13477
------------------------------------------------------------------
(complete + incomplete = total; imputed is the minimum across m
 of the number of filled-in observations.)

. mi estimate, post: regress los routine age female ndx i.race
. estimates store mlogit

Multiple-imputation estimates                     Imputations     =         25
Linear regression                                 Number of obs   =      13477
                                                  Average RVI     =     0.0521
                                                  Largest FMI     =     0.1687
                                                  Complete DF     =      13469
DF adjustment:   Small sample                     DF:     min     =     801.67
                                                          avg     =    9064.56
                                                          max     =   13444.71
Model F test:       Equal FMI                     F(   7,10658.6) =      47.15
Within VCE type:          OLS                     Prob > F        =     0.0000

------------------------------------------------------------------------------
         los |      Coef.   Std. Err.      t    P>|t|     [95% Conf. Interval]
-------------+----------------------------------------------------------------
     routine |  -1.735757   .2976168    -5.83   0.000    -2.319128   -1.152387
         age |  -.4496698   .0376826   -11.93   0.000     -.523533   -.3758066
      female |  -1.006967   .2036503    -4.94   0.000     -1.40615   -.6077837
```

```
         ndx |   .2619443   .0513733     5.10   0.000     .1612455    .3626432
             |
        race |
          2  |    2.85375   .2962195     9.63   0.000     2.272845    3.434656
          3  |   .9665843   .3544798     2.73   0.006     .2714819    1.661687
          4  |   2.333734   .5037762     4.63   0.000     1.344857     3.32261
             |
       _cons |   14.25709   .7036235    20.26   0.000     12.87789    15.63629
------------------------------------------------------------------------------
```

How big is the Monte-Carlo error for this problem? Let's investigate:

```
. mi estimate, mcerror: regress los routine age female ndx i.race

Multiple-imputation estimates                     Imputations     =         25
Linear regression                                 Number of obs   =      13477
                                                  Average RVI     =     0.0521
                                                  Largest FMI     =     0.1687
                                                  Complete DF     =      13469
DF adjustment:   Small sample                     DF:     min     =     801.67
                                                          avg     =    9064.56
                                                          max     =   13444.71
Model F test:       Equal FMI                     F(   7,10658.6) =      47.15
Within VCE type:          OLS                     Prob > F        =     0.0000

------------------------------------------------------------------------------
         los |      Coef.   Std. Err.      t    P>|t|     [95% Conf. Interval]
-------------+----------------------------------------------------------------
     routine |  -1.735757   .2976168    -5.83   0.000    -2.319128   -1.152387
             |   .0018864   .0000534     0.01   0.000     .0018468    .0019309
             |
         age |  -.4496698   .0376826   -11.93   0.000     -.523533   -.3758066
             |   .0003014   9.57e-06     0.01   0.000     .0003058    .0002981
             |
      female |  -1.006967   .2036503    -4.94   0.000     -1.40615   -.6077837
             |   .0013096   .0000271     0.01   0.000     .0013112    .0013101
             |
         ndx |   .2619443   .0513733     5.10   0.000     .1612455    .3626432
             |   .0005759   .0000252     0.01   0.000     .0005805    .0005755
             |
             |
        race |
          2  |    2.85375   .2962195     9.63   0.000     2.272845    3.434656
             |   .0179681   .0045598     0.16   0.000     .0206672    .0195992
             |
          3  |   .9665843   .3544798     2.73   0.006     .2714819    1.661687
```

```
             |   .0205214   .0051831     0.07   0.001     .0215093    .0243492
             |
           4 |   2.333734   .5037762     4.63   0.000     1.344857     3.32261
             |    .040342   .0128774     0.09   0.000     .0320194    .0598407
             |
             |
       _cons |   14.25709   .7036235    20.26   0.000     12.87789    15.63629
             |   .0068864   .0002402     0.01   0.000     .0067549     .007047
------------------------------------------------------------------------------
Note: values displayed beneath estimates are Monte Carlo error estimates.
```

The largest MC errors are for the race/ethnicity variable: more imputations may be helpful here to stabilize the results.

5. Using your multinomial logit model, carry out a test that after controlling for the other factors in the model, there is no difference between the average length of stay for black subjects and hispanic subjects (hint: use the `mi test` function).

```
. mi estimate (mydiff: _b[2.race] - _b[3.race]), nocitable : ///
. regress los routine age female ndx i.race

. mi testtransform mydiff

note: assuming equal fractions of missing information

       mydiff: _b[2.race] - _b[3.race]

 ( 1)  mydiff = 0

       F(  1,1758.1) =   19.36
            Prob > F =    0.0000
```

We conclude that after controlling for the other factors, blacks have higher predicted values than hispanic subjects.

6. Using your multinomial logit model, carry out a test that after controlling for the other factors in the model, there is no difference between the average length of stay for black subjects, hispanic and other subjects (hint: recoding may be helpful).

```
. mi estimate (mydiff1: _b[2.race] - _b[3.race]) (mydiff2: _b[3.race] - _b[4.race]), nocita
. regress los routine age female ndx i.race

. mi testtransform mydiff1 mydiff2

note: assuming equal fractions of missing information

      mydiff1: _b[2.race] - _b[3.race]
```

```
    mydiff2: _b[3.race] - _b[4.race]

( 1)  mydiff1 = 0
( 2)  mydiff2 = 0

      F(  2,1942.0) =    9.48
           Prob > F =    0.0001
```

We conclude that there are statistically significant differences between the non-white subgroups.

7. Compare and contrast the results from the three models. What do you conclude?

```
. estimates table cc pmm mlogit, b se

----------------------------------------------------
    Variable |     cc          pmm         mlogit
-------------+--------------------------------------
     routine | -1.8804813   -1.7319796   -1.7357575
             |  .33661373    .29765492    .29761676
         age | -.46766823   -.44945341   -.44966984
             |  .04268333    .03767796     .0376826
      female | -1.0306753    -1.001961   -1.0069668
             |  .23221764    .20366606    .20365026
         ndx |  .29942547    .26033905    .26194434
             |  .05837963     .0513417    .05137325
             |
        race |
          2  |  3.1039831    2.8760882    2.8537502
             |  .32098644    .29983368    .29621948
          3  |  1.0002325    .91980507    .96658428
             |  .38624189    .36501811    .35447985
          4  |  2.5675194    2.3843543    2.3337337
             |  .52504202    .50053944    .50377621
             |
       _cons |  14.671114    14.253068    14.257093
             |  .79636175    .70390732    .70362348
----------------------------------------------------
                                         legend: b/se
```

As has been the case previously, the differences between the results for the two imputation models are smaller than the difference between these and the complete case model. Also here we see a greater recovery of information, with max FMI and average RVI in the range of 0.17 and 5% increase, respectively.