BST 210 Lab: Week 15
Missing Data

Missing Data

Missing data is a common occurrence in many studies, not just in long-term longitudinal studies. Most people tend to think of missing data as a problem related to a subject dropping out or withholding information in a study, but it can also occur due to probe malfunction, machine error, or even human/operator mistakes. Important to analysis, missing data usually always leads to some loss of efficiency, but in some cases, can also lead to bias. There are are three main mechanisms of missing data, classified by Rubin, that we're interested in. First, let's review some notation.

We will assume that we have a fully observed covariate X and a partially observed outcome Y, where R will be an indicator equal to 1 if the outcome is missing.

Missing Completely at Random (MCAR)

The missing mechanism is said to be missing completely at random if the missingness in our outcome is completely independent of both Y and X. Thus, if we look at the probability of experiencing missingness, it does not depend on Y or X:

$$P(R=1|X,Y) \overset{MCAR}{=} P(R=1)$$

Q: What are some ways you could experience missingness completely at random?

MCAR is a relatively strong assumption, which might occur if missingness is due to a random event like some outcomes were randomly deleted from a database.

Q: How might check whether Y is MCAR?

If we regress our missingness indicator R on our covariate X and saw that they had a significant relationship, that is evidence that our data are *not* MCAR.

However, we cannot test the assumption that missingness R is independent of Y because that would require we know the missing Y values in the first place.

So this approach can only help us to confirm that our data are not MCAR by demonstrating lack of independence between R and X.

In general, we cannot use a test to confirm that our missingness meets any particular assumption

Missing at Random (MAR)

The missing mechanism is said to be missing at random if the missingness in our outcome is independent of Y given our covariate X. So, among those subjects with the same value of the covariate, missingness in Y is independent of the value of Y. Thus, if we look at the probability of experiencing missingness, it depends solely on X:

$$P(R=1|X,Y) \overset{MAR}{=} P(R=1|X)$$

Q: What are some ways you could experience missingness at random?

Consider opinion polls. Many people refuse to answer. If you assume that the reasons people refuse to answer are entirely based on demographics (e.g., young people are less likely to answer), and if you have those demographics on each person, then the data is MAR. It is known that some of the reasons why people refuse to answer can be based on demographics (for instance, people at both low and high incomes are less likely to answer than those in the middle), but there's really no way to know if that is the full explanation.

Q: Can we check whether our outcome is MAR?

As stated above, there is no way to assess the relationship between R and Y, so MAR is not a testable assumption from observed data. However, it may seem more or less plausible depending on context.

Missing Not at Random (MNAR)

The missing mechanism is said to be missing not at random if the missingness in our outcome is neither MCAR nor MAR. This means that our chance of observing Y depends on the value of Y, even after conditioning on our observed covariates X. Thus, if we look at the probability of experiencing missingness:

$$P(R = 1|X, Y) \overset{MNAR}{=} P(R = 1|X, Y)$$

Q: What are some ways you could experience missingness not at random?

A few examples: sicker people being more likely to drop out of a study, only high-achieving students submitting test scores to a database, a data error that deletes the largest outcomes values, to name a few.

Analysis Under Missingness

Many datasets will contain some amount of missing data. So, how do we proceed?

Complete Case Analysis

Often by default, statistical packages will drop observations with missing data from regression analyses. This is called **complete case analysis** because we restrict our analysis only to those observations with complete data.

Q: When is complete case analysis appropriate?

If our missingness is MCAR, meaning that it is completely unrelated to either covariate or outcome values, then complete case analysis is perfectly valid.

It will be less efficient than if we had all of the data, because we have a smaller 'effective' sample size, but it will not be subject to bias.

Moreover, if our data is MAR conditionally on X, and we run a complete case analysis that adjusts for X, then this will also avoid bias due to missingness.

Q: When is complete case analysis inappropriate?

If our data is MAR conditionally on X, and we run a complete case analysis that does *not* adjust for X, then our results will still be biased by the missingness. That is, our results will not be representative of the population within levels of X.

Moreover, if our data is MNAR, then complete case analysis will also be biased, because the complete cases will not be representative of the broader population even within strata of covariates we adjust on.

Thus, it is important to be aware when your analysis tools are dropping observations from your dataset, to ensure that you are not introducing selection bias into your results.

Naive Imputation

There are other naive approaches you could take, such as 'filling in' missing values with reference values. Examples would include

- If you know that your database has introduced NAs where you know there should be 0's, filling in missing values with 0.
- Considering 'missing' to be its own category of a categorical outcome, and including it in the analysis
- If you want to assess a 'worst case' result, replacing NAs with the 'worst' outcome
- Filling in missing values with the mean value of the variable
- etc.

Clearly, these approaches all introduce additional 'strong' assumptions about what is causing your missing data, and the results will reflect those assumptions. Next, we introduce a more sophisticated form of imputation that specifies a model for missingness, and can lead to more efficient estimation than complete case analysis.

Multiple Imputation

Our course does not get into details about the multiple imputation procedure, but it proceeds along the following steps:

- Impute or 'fill in' each missing value multiple times, resulting in multiple completed data sets
- Analyze each completed data set individually
- Combine the estimates appropriately to get overall estimates and appropriate measures of variability

This analysis approach now leverages data from incomplete cases as well as complete cases. Thus, multiple imputation tends to be more efficient than complete case analysis. It may also better address bias than complete case analysis, if the assumed missingness type is incorrect.

However, the results of multiple imputation depend on a correct model for imputation. We will now look at a simple example of multiple imputation in action.

R Example - MICE packages

Below are results using the MICE package in R to analyze a sample of 25 data points from the National Health and Nutrition Examination Survey (NHANES).

We have data on age, BMI, hypertension status (hyp=1), and cholesterol level.

```
library(mice)
data(nhanes)
nhanes
---------------------
   age  bmi hyp chl
1    1   NA  NA  NA
2    2 22.7   1 187
3    1   NA   1 187
4    3   NA  NA  NA
5    1 20.4   1 113
6    3   NA  NA 184
7    1 22.5   1 118
8    1 30.1   1 187
9    2 22.0   1 238
10   2   NA  NA  NA
11   1   NA  NA  NA
12   2   NA  NA  NA
13   3 21.7   1 206
14   2 28.7   2 204
15   1 29.6   1  NA
16   1   NA  NA  NA
17   3 27.2   2 284
18   2 26.3   2 199
19   1 35.3   1 218
20   3 25.5   2  NA
21   1   NA  NA  NA
22   1 33.2   1 229
23   1 27.5   1 131
24   3 24.9   1  NA
25   2 27.4   1 186
--------------------

md.pattern(nhanes)
----------------------
 age hyp bmi chl
13   1   1   1   1  0
1   1   1   0   1  1
3   1   1   1   0  1
1   1   0   0   1  2
7   1   0   0   0  3
0   8   9  10 27
----------------------

tempData = mice(nhanes, m = 5, maxit = 25, seed = 210)
summary(tempData)
------------------------------------------------------
```

```
Multiply imputed data set
Call:
mice(data = nhanes, m = 5, maxit = 25, seed = 210)
Number of multiple imputations:  5
Missing cells per column:
age bmi hyp chl
0   9   8  10
Imputation methods:
age   bmi   hyp   chl
"" "pmm" "pmm" "pmm"
VisitSequence:
bmi hyp chl
2   3   4
PredictorMatrix:
     age bmi hyp chl
age   0   0   0   0
bmi   1   0   1   1
hyp   1   1   0   1
chl   1   1   1   0
Random generator seed value:  210
------------------------------------------------------

complete(tempData,action=1)
---------------------
age  bmi hyp chl
1    1 25.5   1 187
2    2 22.7   1 187
3    1 27.2   1 187
4    3 22.7   1 186
5    1 20.4   1 113
6    3 22.7   1 184
7    1 22.5   1 118
8    1 30.1   1 187
9    2 22.0   1 238
10   2 30.1   1 186
11   1 25.5   1 187
12   2 22.5   2 186
13   3 21.7   1 206
14   2 28.7   2 204
15   1 29.6   1 187
16   1 22.0   2 187
17   3 27.2   2 284
18   2 26.3   2 199
19   1 35.3   1 218
20   3 25.5   2 284
21   1 27.5   1 187
22   1 33.2   1 229
23   1 27.5   1 131
24   3 24.9   1 184
25   2 27.4   1 186
---------------------
```

```
# Fitting the model with the imputed sets and pooling results
y = rbinom(25,1,0.5)
model.imp = with(tempData,  glm(y ~ as.factor(age) + bmi + hyp + chl, family = binomial))
summary(pool(model.imp))
----------------------------------------------------------------------------------------
                        est         se            t        df  Pr(>|t|)
(Intercept)     -1.06220534 3.87027114 -0.27445244 13.439602 0.7879104
as.factor(age)2 -1.23618170 1.60991245 -0.76785647 10.030787 0.4602662
as.factor(age)3 -2.31862998 2.55866883 -0.90618604  6.531153 0.3970360
bmi             -0.09478422 0.22899922 -0.41390631  5.879295 0.6936170
hyp              0.06291103 1.44907158  0.04341472 12.983882 0.9660315
chl              0.02184611 0.02334234  0.93590030  5.632199 0.3876947
                      lo 95     hi 95 nmis       fmi    lambda
(Intercept)     -9.39570156 7.2712909   NA 0.2571695 0.1542815
as.factor(age)2 -4.82179837 2.3494350   NA 0.3894415 0.2787408
as.factor(age)3 -8.45808987 3.8208299   NA 0.5600923 0.4432689
bmi             -0.65792560 0.4683572    9 0.5990853 0.4825283
hyp             -3.06801288 3.1938349    8 0.2740660 0.1702417
chl             -0.03618699 0.0798792   10 0.6146638 0.4984621
```

Q: Based on the output of `md.pattern`, how many rows have complete data?

This matrix shows different missingness patterns, and counts the number of each. The last column gives the number of missing values in each pattern.

So, the first row corresponds with complete data, and shows that there are 13 such observations.

Q: How many imputed datasets did we create?

in our specification of the `mice` function, we set the number of imputations to be $m = 5$. This means that our final analysis will fit five regression models, and average the results into our final estimate.

Lastly, we simulate a binary outcome y, and fit a logistic model based on our newly imputed results.

Q: Using the standard multiple imputation method defined by the package, when we run a logistic regression for a what is our estimate and 95% CI for BMI?

We estimate that $\hat{\beta}_{BMI} = -0.0947$, with a 95% CI of $(-0.658, 0.468)$.

Best Advice: Recognize and Account for Missingness!