Lec 11: Missing Data

MATH 456 - Spring 2016

[Schedule] [Week 15]

Reading & References

Required Reading

- Afifi: Section 9.2
- mice: Multivariate Imputation by Chained Equations in R (skim) https://www.jstatsoft.org/article/view/v045i03

Additional References

- Little, R. and Rubin, D. Statistical Analysis with Missing Data, 2nd Ed., Wiley, 2002
  - Standard reference
  - Requires some math
- Allison, P. Missing Data, Sage, 2001
  - Small and cheap
  - Requires very little math
- Multiple Imputation.com http://www.stefvanbuuren.nl/mi/
- http://www.analyticsvidhya.com/blog/2016/03/tutorial-powerful-packages-imputing-missing-values/
- http://www.r-bloggers.com/imputing-missing-data-with-r-mice-package/

Packages used in this lecture

The following packages can be used to create (`missForest`), visualize(`VIM`) and impute (`mice`) missing data.

- VIM `install.packages("VIM")`
- MICE `install.packages("mice")`
- missForest `install.packages("missForest")`

```
lib <- c("VIM", "mice", "missForest", "ggplot2","scales", "lattice")
invisible(lapply(lib, library, character.only=T))
```

A more in-depth tutorial on these packages can be found at http://www.analyticsvidhya.com/blog/2016/03/tutorial-powerful-packages-imputing-missing-values/

dplyr warning! Many of the packages used here contain functions that are also contained in `dpyr`. If you want to use functions from the `dplyr` package in the same script fill you need to specify which function comes from which package. * Packages load in order that they were called * If you load MASS and then dplyr, when you call `select` it will use the version from `dplyr`. * If you load dplyr and then MASS, when you call `select` it will use the version from `MASS`. * Avoid this by calling specifically `MASS::select` or `dplyr::select` when using these functions.

Terminology

- General: Item non-response. Individual pieces of data are missing.

- Unit non-response: Records have some background data on all units, but some units don't respond to any question.
- Monotonone missing data: Variables can be ordered such that one block of variables more observed than the next.

Identifying missing data

Recap on NA's

- Missing data in `R` is denoted as `NA`
- Arithmetic functions on missing data will return missing

```
library(MASS)
head(survey$Pulse)
```

```
## [1]  92 104  87  NA  35  64
```

```
mean(survey$Pulse)
```

```
## [1] NA
```

The `summary()` function will always show missing.

```
summary(survey$Pulse)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's
##   35.00   66.00   72.50   74.15   80.00  104.00      45
```

The `is.na()` function is helpful to identify rows with missing data

```
table(is.na(survey$Pulse))
```

```
##
## FALSE  TRUE
##   192    45
```

The function `table()` will not show NA by default.

```
table(survey$M.I)
```

```
##
## Imperial   Metric
##       68      141
```

```
table(survey$M.I, useNA="always")
```

```
##
## Imperial   Metric     <NA>
##       68      141       28
```

What percent of the data set is missing?

```
round(prop.table(table(is.na(survey)))*100,1)
```

```
##
## FALSE  TRUE
##  96.2   3.8
```

4% of the data points are missing.

How much missing is there per variable?

```
prop.miss <- apply(survey, 2, function(x) round(sum(is.na(x))/NROW(x),4))
prop.miss
```

```
##    Sex Wr.Hnd NW.Hnd  W.Hnd   Fold  Pulse   Clap   Exer  Smoke Height
## 0.0042 0.0042 0.0042 0.0042 0.0000 0.1899 0.0042 0.0000 0.0042 0.1181
##    M.I    Age
## 0.1181 0.0000
```

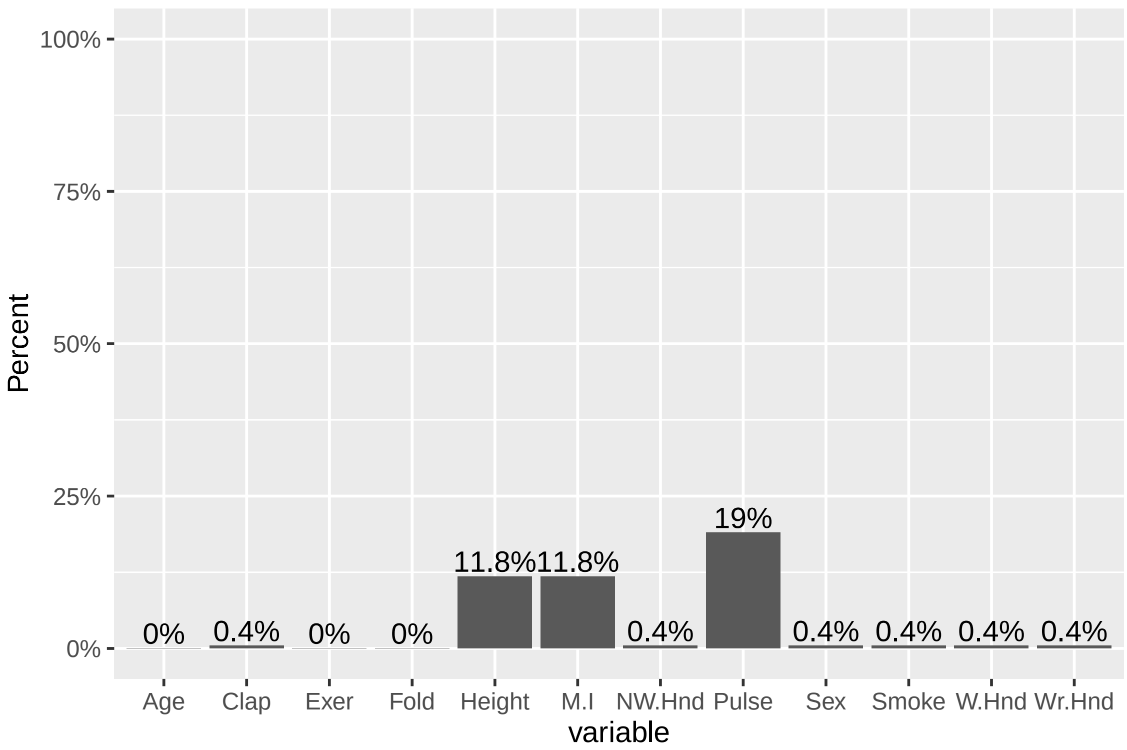
The amount of missing data per variable varies from 0 to 19%.

Can we visualize this?

- Using ggplot2

```
pmpv <- data.frame(variable = names(survey), pct.miss =prop.miss)

ggplot(pmpv, aes(x=variable, y=pct.miss)) +
  geom_bar(stat="identity") + ylab("Percent") + scale_y_continuous(labels=percent, limits=c(0,1)) +
  geom_text(data=pmpv, aes(label=paste0(round(pct.miss*100,1),"%"), y=pct.miss+.025), size=4)
```



Using `mice`

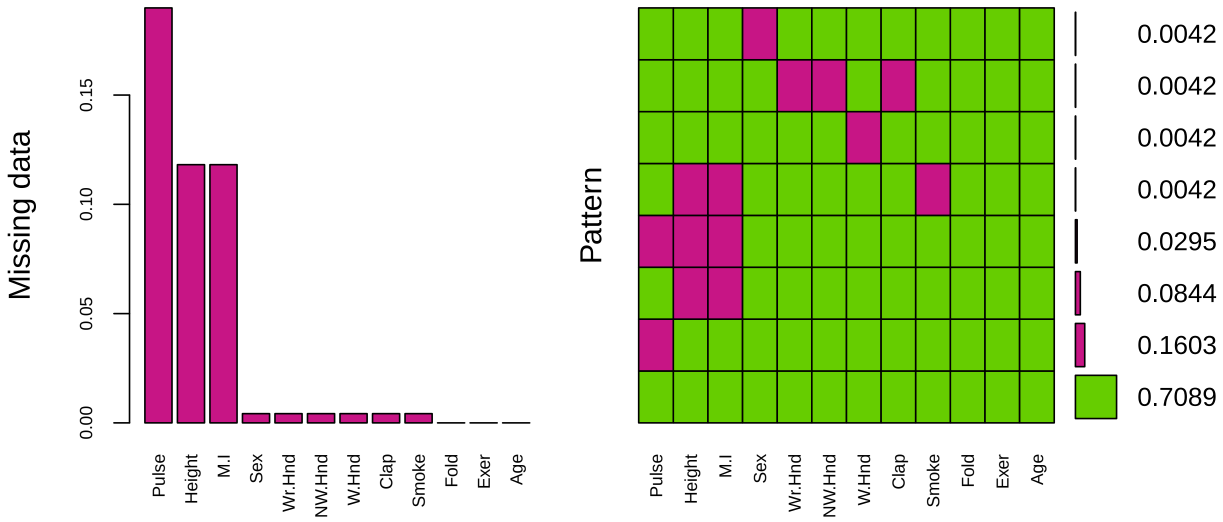
```
md.pattern(survey)
```

```
##     Fold Exer Age Sex Wr.Hnd NW.Hnd W.Hnd Clap Smoke Height M.I Pulse
## 168    1    1   1   1      1      1     1    1     1      1   1     1   0
##   1    1    1   1   0      1      1     1    1     1      1   1     1   1
##   1    1    1   1   1      1      1     0    1     1      1   1     1   1
##  38    1    1   1   1      1      1     1    1     1      1   1     0   1
##  20    1    1   1   1      1      1     1    1     1      0   0     1   2
##   1    1    1   1   1      0      0     1    0     1      1   1     1   3
##   7    1    1   1   1      1      1     1    1     1      0   0     0   3
##   1    1    1   1   1      1      1     1    1     0      0   0     1   3
##        0    0   0   1      1      1     1    1     1     28  28    45 107
```

This somewhat ugly output tells us that 168 records have no missing data, 38 records are missing only `Pulse` and 20 are missing both `Height` and `M.I`.

Or using the `VIM` package:

```
aggr(survey, col=c('chartreuse3','mediumvioletred'),
              numbers=TRUE, sortVars=TRUE,
              labels=names(survey), cex.axis=.7,
              gap=3, ylab=c("Missing data","Pattern"))
```



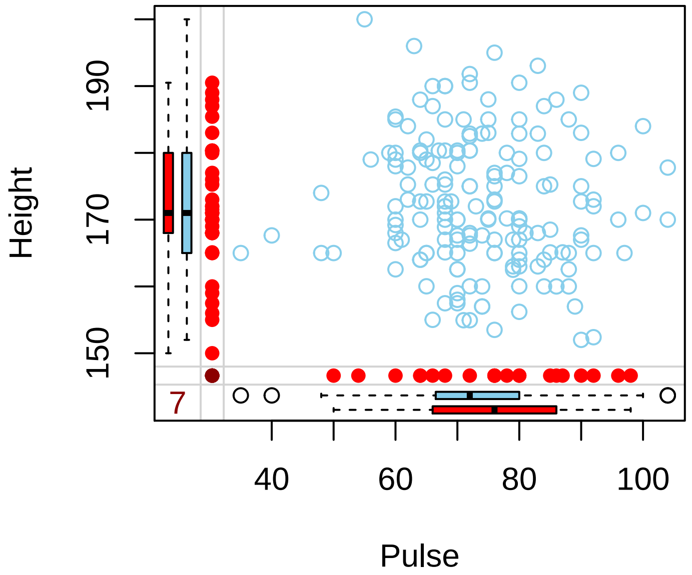
The plot on the left is a simplified, and ordered version of the ggplot from above, except the bars appear to be inflated because the y-axis goes up to 15% instead of 100%.

The plot on the right shows the missing data patterns, and indicate that 71% of the records has complete cases, and that everyone who is missing `M.I.` is also missing Height.

Another plot that can be helpful to identify patterns of missing data is a margin plot.

- Two continuous variables are plotted against each other.
- Blue bivariate scatterplot and univariate boxplots are for the observations where values on both variables are observed.
- Red univariate dotplots and boxplots are drawn for the data that is only observed on one of the two variables in question.
- The darkred text indicates how many records are missing on both.

```
marginplot(survey[,c(6,10)])
```



This shows us that the observations missing pulse have the same median height, but those missing height have a higher median pulse rate.

Effects of Nonresponse

Textbook example: Example reported in W.G. Cochran, Sampling Techniques, 3rd edition, 1977, ch. 13

> Consider data that come form an experimental sampling of fruit orchards in North Carolina in 1946. Three successive mailings of the same questionnaire were sent to growers. For one of the questions the number of fruit trees, complete data were available for the population. . .

Ave. # trees	# of growers	% of pop'n	Ave # trees/grower
1st mailing responders	300	10	456
2nd mailing responders	543	17	382
3rd mailing responders	434	14	340
Nonresponders	1839	59	290
	———	———	———
Total population	3116	100	329

- The overall response rate was very low.
- The rate of non response is clearly related to the average number of trees per grower.
- The estimate of the average trees per grower can be calculated as a weighted average from responders $\bar{Y}_1$ and non responders $\bar{Y}_2$.

Bias: The difference between the observed estimate $\bar{y}_1$ and the true parameter μ.

$$
\begin{aligned}
E(\bar{y}_1) - \mu &= \bar{Y}_1 - \bar{Y} \\
&= \bar{Y}_1 - \left[(1-w)\bar{Y}_1 - w\bar{Y}_2\right] \\
&= w(\bar{Y}_1 - \bar{Y}_2)
\end{aligned}
$$

Where w is the proportion of non-response.

- The amount of bias is the product of the proportion of non-response and the difference in the means between the responders and the non-responders.
- The sample provides no information about $\bar{Y_2}$, the size of the bias is generally unknown without information gained from external data.

Missing Data Mechanisms

Process by which some units observed, some units not observed

- Missing Completely at Random (MCAR): The probability that a data point is missing is completely unrelated (independent) of any observed and unobserved data or parameters.
  - P(Y missing| X, Y) = P(Y missing)
  - Ex: Miscoding or forgetting to log in answer
- Missing at Random (MAR): The probability that a data point is missing is independent can be explained or modeled by other observed variables.
  - P(Y missing|x, Y) = P(Y missing | X)
  - Ex: Y = age, X = sex
    - Pr (Y miss| X = male) = 0.2
    - Pr (Y miss| X = female) = 0.3
    - Males people are less likely to fill out an income survey - The missing data on income is related to gender - After accounting for gender the missing data is unrelated to income.

- Not missing at Random (NMAR): The probability that a data point is missing depends on the value of the variable in question.
  - P(Y missing | X, Y) = P (Y missing|X, Y)

  - Ex: Y = income, X = immigration status
    - Richer person may be less willing to disclose income

    - Illegal immigrant may be less willing to disclose income

Does it matter to inferences? **Yes!**

Demonstration via Simulation

MCAR

1. Draw a random sample of size 100 from a standard Normal distribution (Z) and calculate the mean.

```
z <- rnorm(100)
mean.z <- mean(z)
mean.z
```

```
## [1] 0.02552802
```

2. Delete data at a rate of p and calculate the complete case (available) mean.
    - Sample 100 random Bernoulli (0/1) variables with probability p.

    ```
    x <- rbinom(100, 1, p=.5)
    ```

    - Find out which elements are are 1's

    ```
    delete.these <- which(x==1)
    ```

    - Set those elements in `z` to `NA`.

    ```
    z[delete.these] <- NA
    ```

    - Calculate the complete case mean

    ```
    mean(z, na.rm=TRUE)
    ```

    ```
    ## [1] -0.04344224
    ```

3. Calculate the bias

```
mean.z - mean(z, na.rm=TRUE)
```

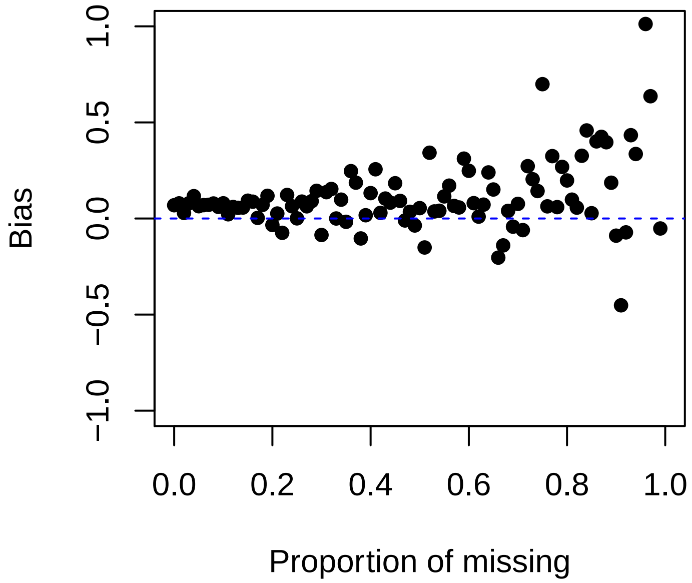
```
## [1] 0.06897026
```

How does the bias change as a function of the proportion of missing? Let p range from 0% to 99% and plot the bias as a function of p.

```
calc.bias <- function(p){
  mean.z - mean(ifelse(rbinom(100, 1, p)==1, NA, z), na.rm=TRUE)
}

p <- seq(0,.99,by=.01)

plot(c(0,1), c(-1, 1), type="n", ylab="Bias", xlab="Proportion of missing")
  points(p, sapply(p, calc.bias), pch=16)
  abline(h=0, lty=2, col="blue")
```



NMAR: Missing related to data

What if the rate of missing is related to the value of the outcome? 1. Randomly draw 100 random normal samples.

```
Z <- rnorm(100)
```

2. Randomly draw 100 uniform random values between 0 and 1 to serve as probabilities.

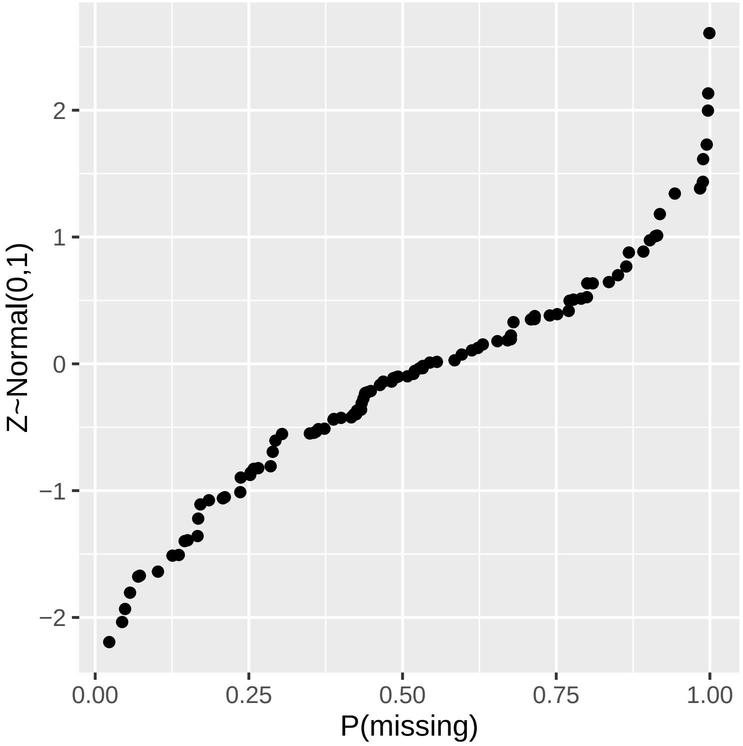
```
p <- runif(100, 0, 1)
```

3. Sort both the value of Z and the probability of missingness p in ascending order

```
dta <- data.frame(Z=sort(Z), p=sort(p))
head(dta)
```

```
##           Z          p
## 1 -2.194080 0.02288529
## 2 -2.036011 0.04382150
## 3 -1.933350 0.04860506
## 4 -1.804556 0.05667118
## 5 -1.677766 0.06991823
## 6 -1.670657 0.07265233
```

```
ggplot(dta, aes(x=p, y=Z)) + geom_point() + xlab("P(missing)") + ylab("Z~Normal(0,1)")
```



4. Set Z missing with probability equal to the p for that row. _Create a new vector `dta$z.miss` that is either 0, or the value of `dta$Z` with probability `1-dta$p`. Then change all the 0's to `NA`.

```
dta$Z.miss <- dta$Z * (1-rbinom(NROW(dta), 1, dta$p))
head(dta)
```

```
##           Z          p    Z.miss
## 1 -2.194080 0.02288529 -2.194080
## 2 -2.036011 0.04382150 -2.036011
## 3 -1.933350 0.04860506 -1.933350
## 4 -1.804556 0.05667118 -1.804556
## 5 -1.677766 0.06991823 -1.677766
## 6 -1.670657 0.07265233 -1.670657
```

```
dta$Z.miss[dta$Z.miss==0] <- NA
```

5. Calculate the complete case mean and the bias

```
mean(dta$Z.miss, na.rm=TRUE)
```

```
## [1] -0.710282
```

```r
mean(dta$Z) - mean(dta$Z.miss, na.rm=TRUE)
```

```
## [1] 0.5983707
```

NMAR: Pure Censoring

Consider a hypothetical blood test to measure a hormone that is normally distributed in the blood with mean $10\mu g$ and variance 1. However the test to detect the compound only can detect levels above 10.

```r
z <- rnorm(100, 10, 1)
y <- z
y[y<10] <- NA
mean(y, na.rm=TRUE)
```

```
## [1] 10.90831
```

When the data is not missing at random, the bias can be much greater.

Problem: Usually you don't know the missing data mechanism.

Degrees of difficulty

- MCAR is easiest to deal with.
- MAR: we can live with it.
- NMAR: most difficult to handle.

Evidence?

What can we learn from evidence in the data set at hand?

- May be evidence in the data rule out MCAR - test responders vs. nonresponders.
  - Example: Responders tend to have higher/lower average education than nonresponders by t-test
  - Example: Response more likely in one geographic area than another by chi-square test
- No evidence in data set to rule out MAR (although there may be evidence from an external data source)

What is plausible?

- Cochran example: when human behavior is involved, MCAR must be viewed as an extremely special case that would often be violated in practice
- Missing data may be introduced by design (e.g., measure some variables, don't measure others for reasons of cost, response burden), in which case MCAR would apply
- MAR is much more common than MCAR
- We cannot be too cavalier about assuming MAR, but anecdotal evidence shows that it often is plausible when conditioning on enough information

Ignorable nonresponse

- If missing-data mechanism is MCAR or MAR then nonresponse is said to be “ignorable”
- Origin of name: in likelihood-based inference, both the data model and missing-data mechanism are important but with MCAR or MAR, inference can be based solely on the data model, thus making inference much simpler

- “*Ignorability*” is a relative assumption: missingness on income may be NMAR given only gender, but may be MAR given gender, age, occupation, region of the country

General strategies

For handling missing data

- Complete-case/available-case analysis: drop cases that make analysis inconvenient

- Imputation procedures: fill in missing values, then analyze completed data sets using complete-date methods

- Weighting procedures: modify “design weights” (i.e., inverse probabilities of selection from sampling plan) to account for probability of response

- Model-based approaches: develop model for partially missing data, base inferences on likelihood under that model

Ignore it

Complete cases analysis

If not all variables observed, delete case from analysis

- Advantages:
  - Simplicity
  - Common sample for all estimates
- Disadvantages:
  - Loss of information, e.g., if there are 20 variables, each 10% missing MCAR, then expect 12% complete cases (0.92 -0.12)
  - Bias due to violation of MCAR

Available-case analysis

- Use all cases where the variable of interest is present
  - Potentially different sets of cases for means of X and Y
  - and complete pairs for r_{XY}

- Tempting to think that available-case analysis will be superior to complete-case analysis

- But it can distort relationships between variables by not using a common base of observations for all quantities being estimated.

Imputation

Fill in missing values, analyze completed data set

- Advantage:
  - Rectangular data set easier to analyze
- Disadvantage:
  - “Both seductive and dangerous”" (Little and Rubin)
  - Can understate uncertainty due to missing values.
  - Can induce bias if imputing under the wrong model.

Methods

- Unconditional mean substitution. **Never use**
  - Impute all missing data using the mean of observed cases
  - Highly biased
  - Artificially decreases the mean.
- Hot deck imputation
  - Impute values by randomly sampling values from observed data.

  - Good for categorical data
  - Reasonable for MCAR and MAR
- Model based imputation
  - Conditional Mean imputation: Use regression on observed variables to estimate missing values
  - Predictive Mean Matching: Fills in a value randomly by sampling observed values whose regression-predicted values are closest to the regression-predicted value for the missing point.
    - Cross between hot-deck and conditional mean
  - Categorical data can be imputed using classification models
  - Less biased than mean substitution
  - but SE’s could be inflated
- Adding a residual
  - Impute regression value ± a randomly selected residual based on estimated residual variance
  - Over the long-term, we can reduce bias, on the average

. . . but we can do better.

Multiple Imputation (MI)

Goals

- Accurately reflect available information
- Avoid bias in estimates of quantities of interest
- Estimation could involve explicit or implicit model
- Accurately reflect uncertainty due to missingness

Technique

1. For each missing value, impute m estimates (usually $m = 5$)
    - Imputation method must include a random component
2. Create m complete data sets
3. Perform desired analysis on each of the m complete data sets
4. Combine final estimates in a manner that accounts for the between, and within imputation variance.

MI as a paradigm

- Logic: "Average over" uncertainty, don't assume most likely scenario (single imputation) covers all plausible scenarios
- Principle: Want nominal 95% intervals to cover targets of estimation 95% of the time
- Simulation studies show that, when MAR assumption holds:
    - Proper imputations will yield close to nominal coverage (Rubin 87)
    - Improvement over single imputation is meaningful
    - Number of imputations can be modest - even 2 adequate for many purposes, so 5 is plenty

Rubin 87: Multiple Imputation for Nonresponse in Surveys, Wiley, 1987).

Inference on MI

Consider m imputed data sets. For some quantity of interest Q with squared $SE = U$, calculate $Q_1, Q_2, \ldots, Q_m$ and $U_1, U_2, \ldots, U_m$ (e.g., carry out m regression analyses, obtain point estimates and SE from each).

Then calculate the average estimate $\bar{Q}$, the average variance $\bar{U}$, and the variance of the averages B.

$$\bar{Q} = \sum_{i=1}^{m} Q_i/m$$
$$\bar{U} = \sum_{i=1}^{m} U_i/m$$
$$B = \frac{1}{m-1}\sum_{i=1}^{m}(Q_i - \bar{Q})^2$$

Then $T = \bar{U} + \frac{m+1}{m}B$ is the estimated total variance of $\bar{Q}$.

Significance tests and interval estimates can be based on

$$\frac{\bar{Q} - Q}{\sqrt{T}} \sim t_{df}, \text{ where } df = (m-1)(1 + \frac{1}{m+1}\frac{\bar{U}}{B})^2$$

- df are similar to those for comparison of normal means with unequal variances, i.e., using Satterthwaite approximation.
- Ratio of (B = between-imputation variance) to (T = between + within-imputation variance) is known as the fraction of missing information.

Practical Considerations

- Typically desirable to include many predictors, both to
  - improve precision of imputed values
  - make MAR assumption more plausible
- But the number of covariance parameters goes up as the square of the number of variables in the model,
- implying practical limits on the number of variables for which parameters can be estimated well

Bottom Line for MI

Little & Rubin (Book, p. 218):

> "In our experience with real and artificial data…, the practical conclusion appears to be that multiple imputation, when carefully done, can be safely used with real problems even when the ultimate user may be applying models or analyses not contemplated by the imputer."

Read more about this here: https://support.sas.com/documentation/cdl/en/statug/63033/HTML/default/viewer.htm#statug_mi_sect032.htm

Imputation methods for complex survey data and data not missing at random is an open research topic.

Multiple Imputation using Chained Equations (MICE)

- Generates multiple imputations for incomplete multivariate data by Gibbs sampling.
- Missing data can occur anywhere in the data.
- Impute an incomplete column by generating 'plausible' synthetic values given other columns in the data.
- For predictors that are incomplete themselves, the most recently generated imputations are used to complete the predictors prior to imputation of the target column.
- A separate univariate imputation model can be specified for each column.
- The default imputation method depends on the measurement level of the target column.

Consider a data matrix with 3 variables y_1, y_2, y_3, each with missing values. At iteration (ℓ):

1. Fit a model on $y_1^{(\ell-1)}$ using current values of $y_2^{(\ell-1)}, y_3^{(\ell-1)}$
2. Impute missing y_1, generating $y_1^{(\ell)}$
3. Fit a model on $y_2^{(\ell-1)}$ using current versions of $y_1^{(\ell)}, y_3^{(\ell-1)}$
4. Impute missing y_2, generating $y_2^{(\ell)}$
5. Fit a model on y_3 using current versions of $y_1^{(\ell)}, y_2^{(\ell)}$
6. Impute missing y_3, generating $y_3^{(\ell)}$
7. Start next cycle using updated values $y_1^{(\ell)}, y_2^{(\ell)}, y_3^{(\ell)}$

(Select) Built-in elementary imputation methods are:

- *pmm*: Predictive mean matching (any) **DEFAULT FOR NUMERIC**
- *norm.nob*: Linear regression ignoring model error (numeric)
- *norm.boot*: Linear regression using bootstrap (numeric)
- *norm.predict*: Linear regression, predicted values (numeric)

- *mean*: Unconditional mean imputation (numeric)
- *logreg*: Logistic regression (factor, 2 levels) **DEFAULT**
- *logreg.boot*: Logistic regression with bootstrap
- *polyreg*: Polytomous logistic regression (factor, >= 2 levels) **DEFAULT**
- *lda*: Linear discriminant analysis (factor, >= 2 categories)
- *cart*: Classification and regression trees (any)
- *rf*: Random forest imputations (any)

Example: Prescribed amount of missing.

We will demonstrate using Fisher's Iris data (pre-built in with R) where we can artificially create a prespecified percent of the data missing. This allows us to be able to estimate the bias incurred by using these imputation methods.

For the `iris` data we set a seed and use the `prodNA()` function from the `missForest` package to create 10% missing values in this data set.

```r
prop.table(table(is.na(iris)))
```

```
##
## FALSE
##     1
```

```r
set.seed(12345) # Note to self: Change the combo on my luggage
iris.mis <- prodNA(iris, noNA=0.1)
prop.table(table(is.na(iris.mis)))
```

```
##
## FALSE  TRUE
##   0.9   0.1
```

Multiply impute the missing data using `mice()`

```r
imp_iris <- mice(iris.mis, m=5, maxit=20, meth="pmm", seed=500, printFlag=FALSE)
summary(imp_iris)
```

```
## Multiply imputed data set
## Call:
## mice(data = iris.mis, m = 5, method = "pmm", maxit = 20, printFlag = FALSE,
##     seed = 500)
## Number of multiple imputations:  5
## Missing cells per column:
## Sepal.Length  Sepal.Width Petal.Length  Petal.Width      Species
##           16           15           13           16           15
## Imputation methods:
## Sepal.Length  Sepal.Width Petal.Length  Petal.Width      Species
##        "pmm"        "pmm"        "pmm"        "pmm"        "pmm"
## VisitSequence:
## Sepal.Length  Sepal.Width Petal.Length  Petal.Width      Species
##            1            2            3            4            5
## PredictorMatrix:
##              Sepal.Length Sepal.Width Petal.Length Petal.Width Species
## Sepal.Length            0           1            1            1       1
## Sepal.Width             1           0            1            1       1
## Petal.Length            1           1            0            1       1
## Petal.Width             1           1            1            0       1
## Species                 1           1            1            1       0
## Random generator seed value:  500
```

- The method can be either a single string, or a vector of strings with length `ncol(data)`
  - specifying the elementary imputation method to be used for each column in data.
  - If specified as a single string, the same method will be used for all columns.
  - Columns that need not be imputed have the empty method ”
- `printFlag`: Use print=FALSE for silent computation.

Read 4.3: Assessing Convergence in the JSS article on mice

Check the imputation method used on each variable.

```
imp_iris$meth
```

```
## Sepal.Length  Sepal.Width Petal.Length  Petal.Width      Species
##        "pmm"        "pmm"        "pmm"        "pmm"        "pmm"
```

Look at the values generated for imputation

This just shows what values were imputed for this variable at each imputation.

```
imp_iris$imp$Sepal.Length
```

```
##       1   2   3   4   5
## 1   5.0 5.0 5.0 5.8 5.2
## 5   5.6 5.0 5.4 4.8 5.8
## 26  5.1 4.9 5.1 5.0 5.5
## 31  5.1 5.0 5.1 4.7 5.5
## 33  5.5 5.7 5.5 5.8 4.9
## 39  4.9 4.4 4.4 4.8 4.6
## 43  5.0 4.9 4.3 4.6 4.6
## 56  6.6 5.8 6.1 5.6 6.1
## 96  6.2 6.2 5.8 6.7 6.6
## 103 6.3 6.3 6.7 6.7 7.2
## 113 6.3 6.7 6.9 6.9 6.7
## 124 6.6 5.6 6.4 6.2 6.2
## 132 7.2 7.7 7.7 7.7 7.7
## 135 7.7 6.9 6.3 6.7 6.3
## 149 7.0 6.7 6.7 6.3 6.7
## 150 6.3 6.4 6.7 7.2 6.4
```

Create a complete data set by filling in the missing data using the imputations

```
iris_1 <- complete(imp_iris, action=1)
```

Action=1 returns the first completed data set, action=2 returns the second completed data set, and so on.

Alternative - Stack the imputed data sets in *long* format.

```
iris_long <- complete(imp_iris, 'long')
```

By looking at the `names` of this new object we can confirm that there are indeed 5 complete data sets with $n = 150$ in each.

```
names(iris_long)
```

```
## [1] ".imp"         ".id"          "Sepal.Length" "Sepal.Width"
## [5] "Petal.Length" "Petal.Width"  "Species"
```

```
table(iris_long$.imp)
```

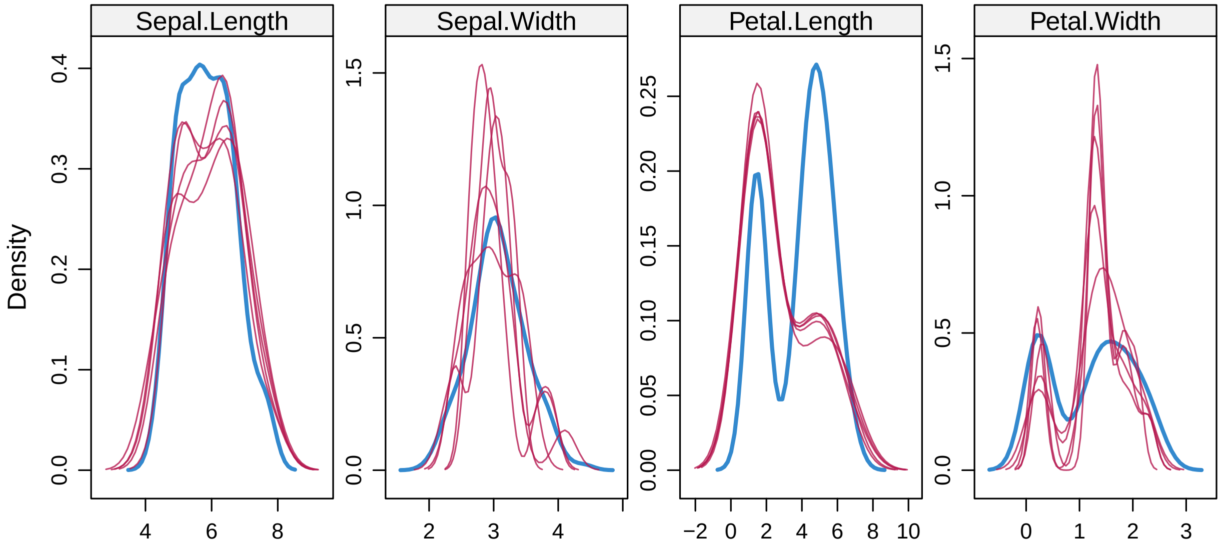
```
##
##   1   2   3   4   5
## 150 150 150 150 150
```

Visualize Imputations

Let's compare the imputed values to the observed values to see if they are indeed "plausible" We want to see that the distribution of of the magenta points (imputed) matches the distribution of the blue ones (observed).

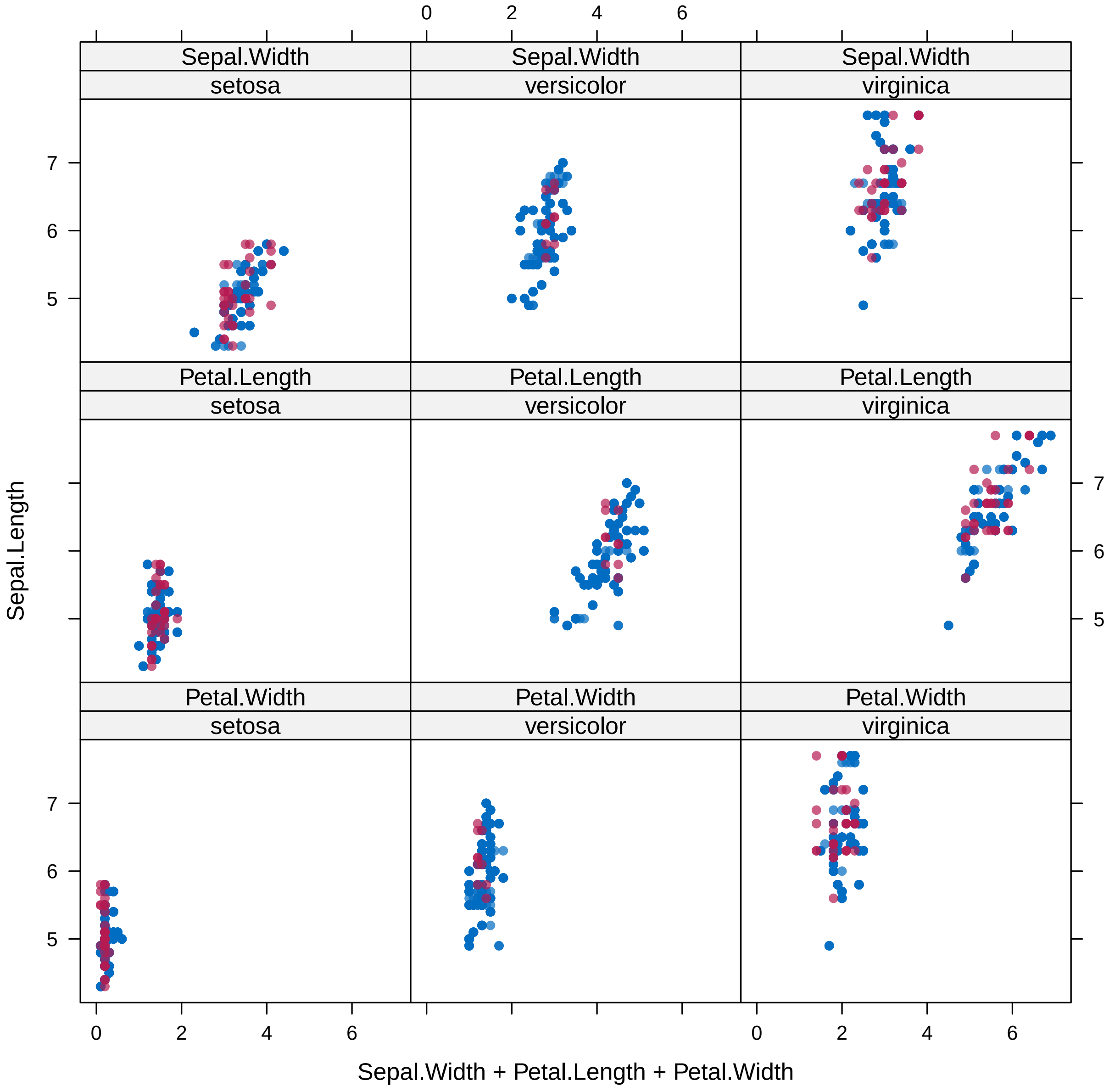
Univariately

```
densityplot(imp_iris)
```



Multivariately

```
xyplot(imp_iris, Sepal.Length ~ Sepal.Width + Petal.Length + Petal.Width | Species, cex=.8, pch=16)
```



Analyze and pool

All of this imputation was done so we could actually perform an analysis!

Let's run a simple linear regression on `Sepal.Length` as a function of `Sepal.Width`, `Petal.Length` and `Species`.

```
model <- with(imp_iris, lm(Sepal.Length ~ Sepal.Width + Petal.Length + Species))
summary(pool(model))
```

```
##                      est         se         t       df     Pr(>|t|)
## (Intercept)   2.4424312 0.29430137  8.299082 81.52267 1.851852e-12
## Sepal.Width   0.4574809 0.09086928  5.034495 85.27283 2.636929e-06
## Petal.Length  0.7058246 0.07676480  9.194639 39.26328 2.448841e-11
## Species2     -0.7728084 0.24720667 -3.126163 56.48034 2.796938e-03
## Species3     -1.1255913 0.34481260 -3.264357 36.94475 2.368406e-03
##                   lo 95      hi 95 nmis       fmi    lambda
## (Intercept)   1.8569206  3.0279417   NA 0.1506633 0.1300790
## Sepal.Width   0.2768168  0.6381451   15 0.1424999 0.1226211
## Petal.Length  0.5505864  0.8610627   13 0.2890890 0.2537759
## Species2     -1.2679300 -0.2776869   NA 0.2176870 0.1904668
## Species3     -1.8242833 -0.4268994   NA 0.3017982 0.2649972
```

Pooled parameter estimates $\bar{Q}$ and their standard errors $\sqrt{T}$ are provided, along with a significance test (against $\beta_p = 0$), and a 95% interval.

Additional information included in this table is the number of missing values, the *fraction of missing information* (`fmi`) as defined by Rubin (1987), and `lambda`, the proportion of total variance that is attributable to the missing data ($\lambda = (B + B/m)/T$).

You can easily create a simpler table using xtable and selecting specific columns to display.

```
print(xtable(summary(pool(model))[,c(1:3, 5:7, 9)], digits=3), type="html")
```

est

se

t

Pr(>|t|)

lo 95

hi 95

fmi

(Intercept)

2.442

0.294

8.299

0.000

1.857

3.028

0.151

Sepal.Width

0.457

0.091

5.034

0.000

0.277

0.638

0.142

Petal.Length

0.706

0.077

9.195

0.000

0.551

0.861

0.289

Species2

-0.773

0.247

-3.126

0.003

-1.268

-0.278

0.218

Species3

-1.126

0.345

-3.264

0.002

-1.824

-0.427

0.302

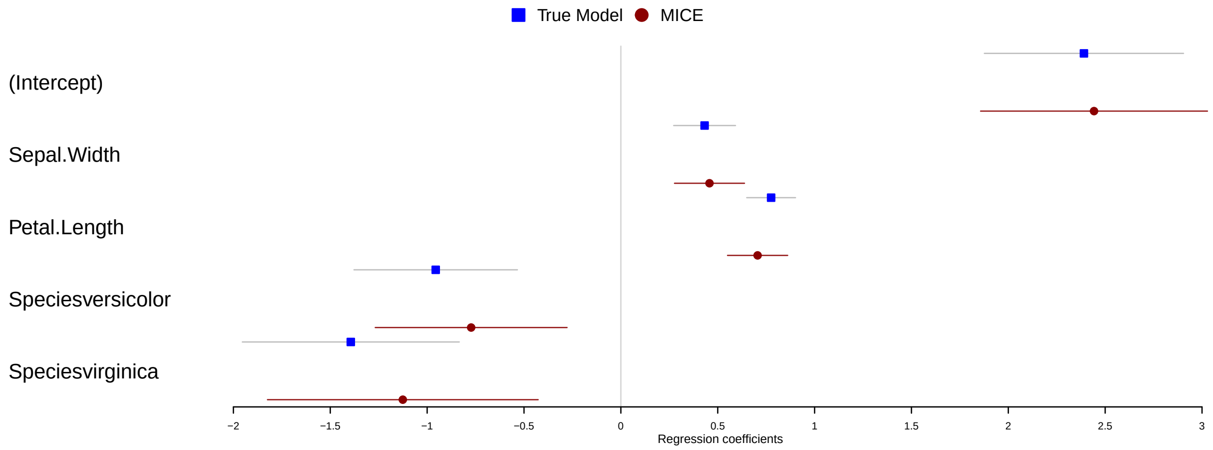
Calculating bias

The iris data set had no missing data to begin with. So we can calculate the “true” parameter estimates…

```r
true.model <- lm(Sepal.Length ~ Sepal.Width + Petal.Length + Species, data=iris)
```

and find the difference in coefficients.

The variance of the multiply imputed estimates is larger because of the between-imputation variance.



Final thoughts

Take home messages:

1. Don't ignore missing data.
2. Impute sensibly and multiple times.

Here are some other reasons for missing data that we do not have time to cover. So I will leave you with some final thoughts.

- MI applies to subjects who have a general missingness pattern, i.e., they have measurements on some variables, but not on others.
- But, subjects can be lost to follow up due to death or other reasons (i.e., attrition).
- Here we have only baseline data, but not the outcome or other follow up data.
- If attrition subjects are eliminated from the sample, they can produce non-response or attrition bias.
- Use attrition weights.
  - Given a baseline profile, predict the probability that subject will stay and use the inverse probability as weight.
  - e.g., if for a given profile all subjects stay, then the predicted probability is 1 and the attrition weight is 1. Such a subject "counts once".
  - For another profile, the probability may be 0.5, attrition weight is $1/.5 = 2$ and that person "counts twice".
- For differential drop-out, or self-selected treatment, you can consider using Propensity Scores.

On Your Own

On Your Own

1. For each distribution listed below, draw a random sample of $n = 100$ and delete 20% of the data completely at random and calculate the bias. State if the observed mean over- or under-estimates the true mean.

   a. $X \sim \mathcal{N}(10, 2)$
   b. $Y \sim \mathcal{P}\wr\rangle\int(10)$
   c. $B \sim \mathcal{B}\rangle\backslash\wr\Updownarrow\rangle\dashv\updownarrow(10, .5)$
   d. $F \sim \mathcal{B}\rangle\backslash\wr\Updownarrow\rangle\dashv\updownarrow(10, .9)$
   Does the effect of MCAR on the bias differ for the different distributions?

2. Repeat #1 but set the missing data mechanism to b NMAR, where p is negatively correlated with the data.
3. Using the Parental HIV data set, consider **only** the following variables: `Age`, `Gender`, `livwith`, `BSI_overall`, `Frnds`, and `Hookey`.

   a. What percent of the data set overall is missing?
   b. How much missing data is there per variables?
   c. Describe the missing data pattern. (_Hint: Use `aggr` from the `VIM` package for part b and c)
   d. Describe the relationship of missing data between Age and BSI overall.

4. Multiply impute this data set $m = 5$ times.

   a. State the imputation models used for each variable.

5. After controlling for age, who the student lives with, their overall friendships, and overall BSI score, what is the effect of gender on the likelihood a student will skip school? (I.e. Fit a logistic regression model using `hookey` as the outcome and all other covariates as predictors. Calculate the OR and 95% CI for the effect of gender)

   a. Fit this model on the complete cases (no imputation).
   b. Fit this model on the multiply imputed data sets and pool the results.
   c. Compare these two intervals. Which one is wider? Why?
   d. Discuss the amount of information lost (in terms of sample size and varance) using the complete case method.