Missing Data Basics

Utrecht University Winter School: Missing Data in R

Kyle M. Lang

Department of Methodology & Statistics
Utrecht University

2022-02-03

Outline

Missing Data Descriptives

Missing Data Mechanisms

Missing Data Treatments

What are Missing Data?

Missing data are empty cells in a dataset where there should be observed values.

- The missing cells correspond to true population values, but we haven't observed those values.

What are Missing Data?

Missing data are empty cells in a dataset where there should be observed values.

- The missing cells correspond to true population values, but we haven't observed those values.

Not every empty cell is a missing datum.

- Quality-of-life ratings for dead patients in a mortality study
- Firm profitability after the company goes out of business
- Self-reported severity of menstrual cramping for men
- Empty blocks of data following "gateway" items

A Little Notation

$$
\begin{aligned}
Y &:= \text{An } N \times P \text{ Matrix of Arbitrary Data} \\
Y_{mis} &:= \text{The } \textit{missing} \text{ part of } Y \\
Y_{obs} &:= \text{The } \textit{observed} \text{ part of } Y \\
R &:= \text{An } N \times P \text{ response matrix} \\
M &:= \text{An } N \times P \text{ missingness matrix}
\end{aligned}
$$

The R and M matrices are complementary.

- $r_{np} = 1$ means y_{np} is observed; $m_{np} = 1$ means y_{np} is missing.
- $r_{np} = 0$ means y_{np} is missing; $m_{np} = 0$ means y_{np} is observed.
- M_p is the *missingness* of Y_p.

Missing Data Descriptives

Missing Data Pattern

Missing data (or response) patterns represent unique combinations of observed and missing items.

- P items $\Rightarrow 2^P$ possible patterns.

	X	Y
1	x	y
2	x	.
3	.	y
4	.	.

Patterns for $P = 2$

	X	Y	Z
1	x	y	z
2	x	y	.
3	x	.	z
4	.	y	z
5	x	.	.
6	.	.	z
7	.	y	.
8	.	.	.

Patterns for $P = 3$

Missing Data Pattern

The concept of a “missing data pattern” can also be used to classify the spatial arrangement of missing cells on a data set.

- Univariate
  - Missing data occur on only one variable
- Monotone
  - The proportion of complete elements, in both rows and columns, decreases when traversing the data set.
  - The observed cells can be arranged into a “staircase” pattern.
- Arbitrary
  - Missing values are “randomly” scattered throughout the data set.

Example Missing Data Patterns

	X	Y	Z
1	x	y	z
2	x	y	z
3	x	y	z
4	x	y	z
5	x	y	z
6	x	.	z
7	x	.	z
8	x	.	z
9	x	.	z
10	x	.	z

Univariate Pattern

	X	Y	Z
1	x	y	z
2	x	y	z
3	x	y	z
4	x	y	.
5	x	y	.
6	x	y	.
7	x	.	.
8	x	.	.
9	x	.	.
10	.	.	.

Monotone Pattern

	X	Y	Z
1	x	.	z
2	x	y	z
3	x	y	z
4	x	.	z
5	x	y	z
6	x	.	z
7	.	y	z
8	x	y	z
9	x	.	.
10	x	y	.

Arbitrary Pattern

Nonresponse Rates

Proportion Missing

- The proportion of cells containing missing data
- Good early screening measure
- Should be computed for each variable, not for the entire dataset

Attrition Rate

- The proportion of participants that drop-out of a study at each measurement occasion

Nonresponse Rates

Proportion of Complete Cases

- The proportion of observations with no missing data
- Often reported but nearly useless quantity

Fraction of Missing Information

- Associated with an estimated parameter, not with an incomplete variable
- Like an R^2 for the missing data
- Most important diagnostic value for missing data problems
- Can only be computed after treating the missing data

Coverage Measures

Covariance Coverage

$$CC_{jk} = N^{-1} \sum_{n=1}^{N} r_{nj} r_{nk}$$

- The proportion of cases available to estimate a given pairwise relationship (e.g., a covariance between two variables)
- Very important to have adequate coverage of the parameters you want to estimate

Coverage Measures

Inbound Statistic

$$I_{jk} = \frac{\sum_{n=1}^{N}(1 - r_{nj})r_{nk}}{\sum_{n=1}^{N}(1 - r_{nj})}$$

- The proportion of missing cases in Y_j for which Y_k is observed

Outbound Statistic

$$O_{jk} = \frac{\sum_{n=1}^{N} r_{nj}(1 - r_{nk})}{\sum_{n=1}^{N} r_{nj}}$$

- The proportion of observed cases in Y_j for which Y_k is missing

Coverage Measures

Influx Coefficient

$$I_j = \frac{\sum_{k=1}^{P} \sum_{n=1}^{N} (1 - r_{nj}) r_{nk}}{\sum_{k=1}^{P} \sum_{n=1}^{N} r_{nk}}$$

- The proportion of observed cells in Y that exists in cases for which Y_j is missing
- How well the missing values in Y_j connect to the observed values in Y_{-j}

Coverage Measures

Outflux Coefficient

$$O_j = \frac{\sum_{k=1}^{P} \sum_{n=1}^{N} r_{nj}(1 - r_{nk})}{\sum_{k=1}^{P} \sum_{n=1}^{N}(1 - r_{nk})}$$

- The proportion of missing cells in Y that exists in cases for which Y_j is observed
- How well the observed values in Y_j connect to the missing values in Y_{-j}

Examples

1. What is the coverage for $cov(X, Y)$?
2. What is the coverage for $cov(W, Y)$?
3. What is the coverage for $cov(X, Z)$?
4. What is the outflux coefficient for W?
5. What is the influx coefficient for W?

	W	X	Y	Z
1	w	x	y	.
2	w	x	y	.
3	w	x	y	.
4	w	x	y	.
5	w	x	y	.
6	w	.	y	z
7	w	.	y	z
8	w	.	y	z
9	w	.	y	z
10	w	.	y	z

Examples

1. What is the percent missing at T2?
2. What is the attrition rate at T3?
3. What is the inbound statistic I_{32}?
4. What is the outbound statistic O_{42}?
5. What is the influx coefficient I_3?
6. What is the outflux coefficient O_2?

	T1	T2	T3	T4
1	x1	x2	x3	x4
2	x1	x2	x3	x4
3	x1	x2	x3	x4
4	x1	x2	x3	.
5	x1	x2	x3	.
6	x1	x2	.	.
7	x1	x2	.	.
8	x1	.	.	.
9	x1	.	.	.
10	x1	.	.	.

Missing Data Mechanisms

Missing Data Mechanisms

Missing Completely at Random (MCAR)

- $P(R|Y_{mis}, Y_{obs}) = P(R)$
- Missingness is unrelated to any study variables.

Missing at Random (MAR)

- $P(R|Y_{mis}, Y_{obs}) = P(R|Y_{obs})$
- Missingness is related to only the *observed* parts of study variables.

Missing not at Random (MNAR)

- $P(R|Y_{mis}, Y_{obs}) \neq P(R|Y_{obs})$
- Missingness is related to the *unobserved* parts of study variables.

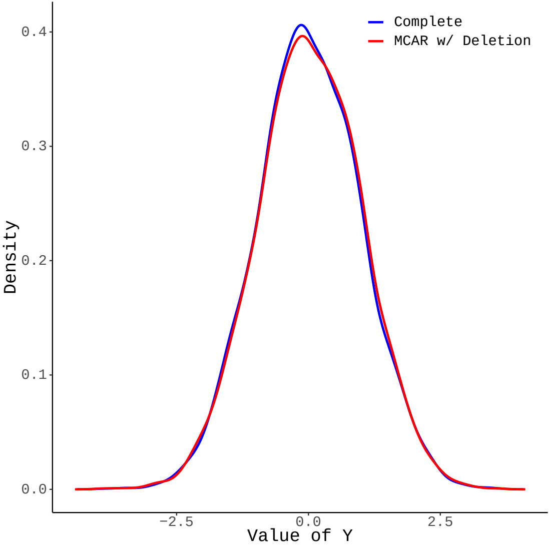
Simulate Some Toy Data

```
nObs <- 5000 # Sample Size
pm   <- 0.3  # Proportion Missing

sigma <- matrix(c(1.0, 0.5, 0.3,
                  0.5, 1.0, 0.0,
                  0.3, 0.0, 1.0),
                ncol = 3)
tmp <- rmvnorm(nObs, c(0, 0, 0), sigma)

x0 <- tmp[ , 1]
y0 <- tmp[ , 2]
z0 <- tmp[ , 3]

cor(y0, x0) # Check correlation between X and Y

[1] 0.5001822
```

MCAR Example

```r
## Simulate MCAR Missingness:
mVec <- sample(1 : length(y0), size = pm * length(y0))

yMcar       <- y0
yMcar[mVec] <- NA

cor(yMcar, x0, use = "pairwise") # Look at correlation

[1] 0.5197437
```

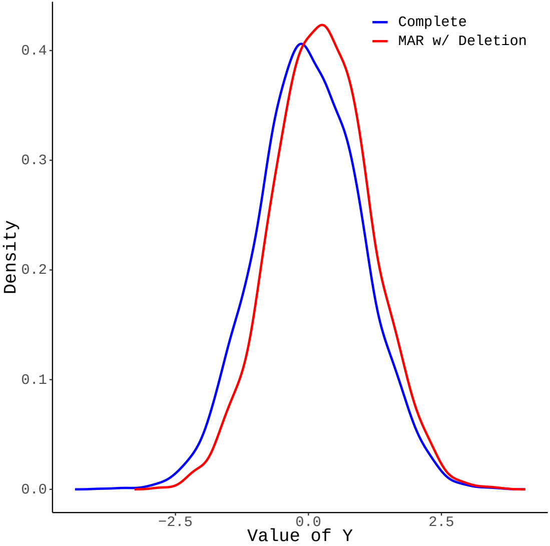
MCAR Example



MAR Example

```
## Simulate MAR Missingness:
mVec <- x0 < quantile(x0, probs = pm)
mean(mVec)

[1] 0.3

yMar       <- y0
yMar[mVec] <- NA

cor(yMar, x0, use = "pairwise") # Not looking so good :(

[1] 0.3825876
```

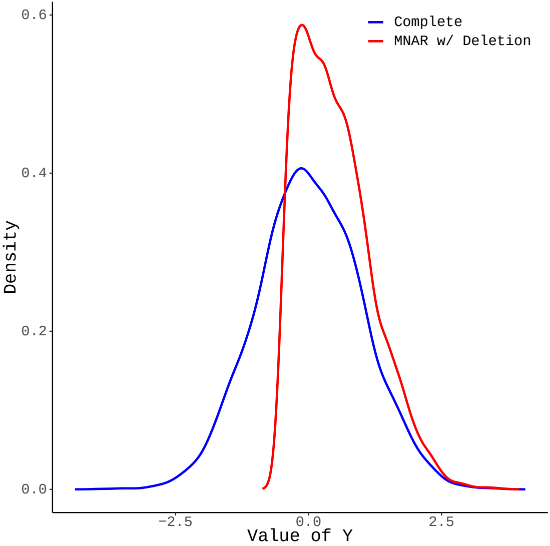
MAR Example


Complete
MAR w/ Deletion
Density
0.4
0.3
0.2
0.1
0.0
−2.5
0.0
2.5
Value of Y

MNAR Example

```
## Simulate MNAR Missingness:
mVec <- y0 < quantile(y0, probs = pm)
mean(mVec)

[1] 0.3

yMnar       <- y0
yMnar[mVec] <- NA

cor(yMnar, x0, use = "pairwise") # Hmm...looks pretty bad.

[1] 0.3901487
```

MNAR Example



Crucial Nuance

In our previous MAR example, ignoring the predictor of missingness actually produces *Indirect MNAR*.

Crucial Nuance

In our previous MAR example, ignoring the predictor of missingness actually produces *Indirect MNAR*.

Question: What happens if we ignore the predictor of missingness, but that predictor is independent of our study variables?

Crucial Nuance

In our previous MAR example, ignoring the predictor of missingness actually produces *Indirect MNAR*.

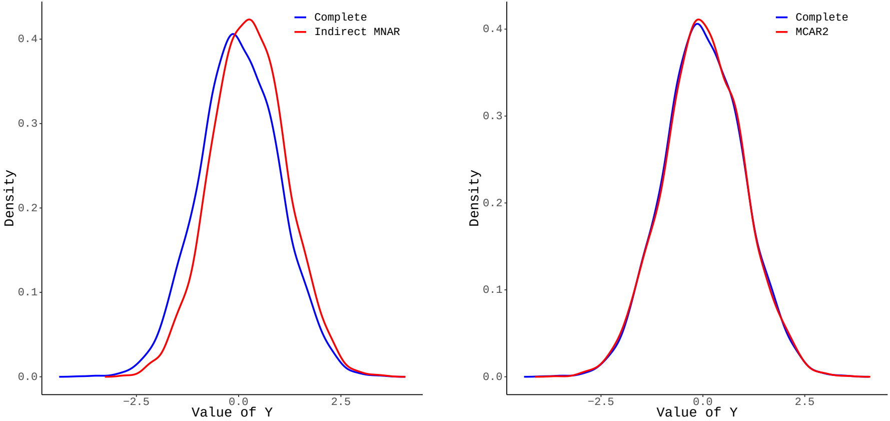
Question: What happens if we ignore the predictor of missingness, but that predictor is independent of our study variables?

```
mVec <- z0 < quantile(z0, probs = pm)

y       <- y0
y[mVec] <- NA

cor(y, x0, use = "pairwise")

[1] 0.5119953
```

Answer: We get back to MCAR :)

Crucial Nuance

The missing data mechanisms are not simply characteristics of an incomplete dataset; we also need to account for the analysis.



Testing the Missing Data Mechanism

We cannot fully test the MAR or MNAR assumptions.

- To do so would require knowing the values of the missing data.
- We can find observed predictors of missingness, but we can never know that we have them all.
- In practice, MAR and MNAR live on the ends of a continuum.
  - Our missing data problem exists at some unknown point along this continuum.
  - We can do a lot to nudge our problem towards the MAR side.

Testing the Missing Data Mechanism

We can test the MCAR assumption.

- With MCAR, the missing data and the observed data should have the same distribution.
- We can test for MCAR by testing the distributions of *auxiliary variables*, $\mathbf{Z}$.
    - Use a t-test to compare the subset of Z_p that corresponds to Y_{mis} to the subset corresponding to Y_{obs}.
    - The Little (1988) MCAR test is a multivariate version of this.

Example

Create some toy datasets from the variables we generated above.

```r
mcarData <- data.frame(y = yMcar, x = x0, z = z0,
                       m = as.numeric(is.na(yMcar))
                       )
marData  <- data.frame(y = yMar, x = x0, z = z0,
                       m = as.numeric(is.na(yMar))
                       )
mnarData <- data.frame(y = yMnar, x = x0, z = z0,
                       m = as.numeric(is.na(yMnar))
                       )
```

T-Test Example

Test for dependence between X and M_Y in MCAR data.

```
mcarData %$% t.test(x ~ m) %>% wrap()


Welch Two Sample t-test

data:  x by m
t = 0.68563, df = 2852.8, p-value = 0.493
alternative hypothesis: true difference in means between
group 0 and group 1 is not equal to 0
95 percent confidence interval:
 -0.03921499  0.08138543
sample estimates:
mean in group 0 mean in group 1
    0.013908816    -0.007176408
```

T-Test Example

Test for dependence between Z and M_Y in MCAR data.

```
mcarData %$% t.test(z ~ m) %>% wrap()


Welch Two Sample t-test

data:  z by m
t = 0.38865, df = 2841.9, p-value = 0.6976
alternative hypothesis: true difference in means between
group 0 and group 1 is not equal to 0
95 percent confidence interval:
 -0.04848298  0.07245421
sample estimates:
mean in group 0 mean in group 1
    0.009151786    -0.002833825
```

T-Test Example

Test for dependence between X and M_Y in MAR data.

```
marData %$% t.test(x ~ m) %>% wrap()


Welch Two Sample t-test

data:  x by m
t = 92.56, df = 3832.8, p-value < 2.2e-16
alternative hypothesis: true difference in means between
group 0 and group 1 is not equal to 0
95 percent confidence interval:
 1.614203 1.684066
sample estimates:
mean in group 0 mean in group 1
      0.5023237      -1.1468112
```

T-Test Example

Test for dependence between Z and M_Y in MAR data.

```
marData %$% t.test(z ~ m) %>% wrap()


Welch Two Sample t-test

data:  z by m
t = 16.913, df = 2832.1, p-value < 2.2e-16
alternative hypothesis: true difference in means between
group 0 and group 1 is not equal to 0
95 percent confidence interval:
 0.4491108 0.5669049
sample estimates:
mean in group 0 mean in group 1
      0.1579585      -0.3500494
```

T-Test Example

Test for dependence between X and M_{Y} in MNAR data.

```
mnarData %$% t.test(x ~ m) %>% wrap()


Welch Two Sample t-test

data:  x by m
t = 28.251, df = 2926.7, p-value < 2.2e-16
alternative hypothesis: true difference in means between
group 0 and group 1 is not equal to 0
95 percent confidence interval:
 0.7439001 0.8548632
sample estimates:
mean in group 0 mean in group 1
      0.2473977      -0.5519839
```

T-Test Example

Test for dependence between Z and M_Y in MNAR data.

```
mnarData %$% t.test(z ~ m) %>% wrap()


Welch Two Sample t-test

data:  z by m
t = -0.33313, df = 2778.5, p-value = 0.7391
alternative hypothesis: true difference in means between
group 0 and group 1 is not equal to 0
95 percent confidence interval:
 -0.07145430  0.05070098
sample estimates:
mean in group 0 mean in group 1
    0.002443105     0.012819764
```

Little (1988) MCAR Test Example

Use the Little (1988) MCAR test on MCAR data.

```
mcarData %>% select(-m) %>% mcar_test()

# A tibble: 1 x 4
  statistic    df p.value missing.patterns
      <dbl> <dbl>   <dbl>            <int>
1     0.504     2   0.777                2
```

Little (1988) MCAR Test Example

Use the Little (1988) MCAR test on MAR data.

```
marData %>% select(-m) %>% mcar_test()

# A tibble: 1 x 4
  statistic    df p.value missing.patterns
      <dbl> <dbl>   <dbl>            <int>
1     2862.     2       0                2
```

Little (1988) MCAR Test Example

Use the Little (1988) MCAR test on MNAR data.

```
mnarData %>% select(-m) %>% mcar_test()

# A tibble: 1 x 4
  statistic    df p.value missing.patterns
      <dbl> <dbl>   <dbl>            <int>
1      746.     2       0                2
```

Logistic Regression Example

```r
## Read in some data:
diabetes1 <- readRDS(paste0(dataDir, "diabetes.rds"))

## Generate MAR missingness:
diabetes1$m <- simLogisticMissingness0(data    = diabetes1,
                                       pm      = 0.25,
                                       preds   = c("bmi", "tc"),
                                       type    = "high",
                                       stdData = TRUE)$r

## Predict the missingness using logistic regression:
fit <- diabetes1 %>%
    select(-glu) %>%
    glm(m ~ ., data = ., family = "binomial")
```

Logistic Regression Example

```
partSummary(fit, 3)

Coefficients:
              Estimate Std. Error z value Pr(>|z|)
(Intercept) -1.459e+01  4.031e+00  -3.619 0.000296
age          1.205e-02  1.141e-02   1.056 0.290782
bmi          2.269e-01  4.054e-02   5.596 2.19e-08
bp          -1.213e-02  1.147e-02  -1.057 0.290292
tc           2.949e-02  2.897e-02   1.018 0.308696
ldl          2.703e-03  2.625e-02   0.103 0.917986
hdl         -5.961e-05  3.990e-02  -0.001 0.998808
tch         -3.160e-01  2.889e-01  -1.094 0.274049
ltg          5.588e-01  8.952e-01   0.624 0.532537
progress     2.501e-03  2.380e-03   1.051 0.293237
sexmale      4.336e-02  2.978e-01   0.146 0.884234
```

Missing Data Treatments

Bad Methods (These almost never work)

Listwise Deletion (Complete Case Analysis)

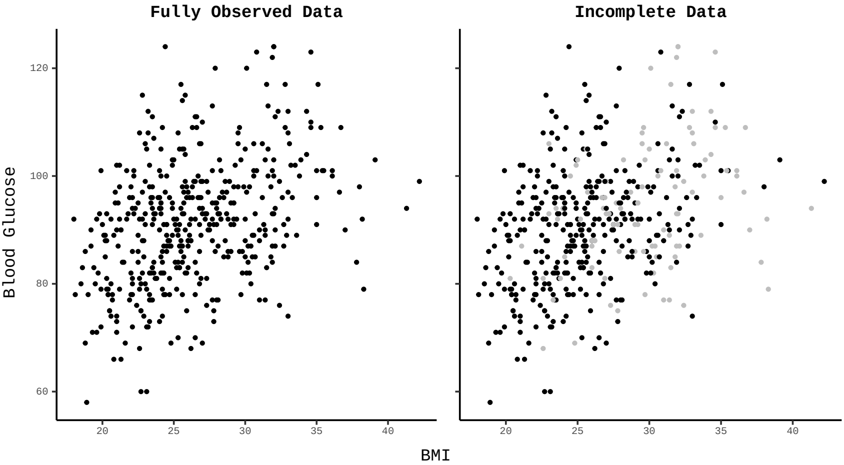
- Use only complete observations for the analysis
  - Very wasteful (can throw out lots of useful data)
  - Loss of statistical power

Pairwise Deletion (Available Case Analysis)

- Use only complete pairs of observations for analysis
  - Different samples sizes for different parameter estimates
  - Can cause computational issues

```r
diabetes2 <- diabetes1
mVec <- simLogisticMissingness0(data    = diabetes1,
                                pm      = 0.25,
                                preds   = "bmi",
                                stdData = TRUE)$r
diabetes2[mVec, "glu"] <- NA
```

Example


Fully Observed Data
Incomplete Data
Blood Glucose
120
100
80
60
20
25
30
35
40
BMI

Example

```
diabetes1 %>% select(bmi, glu, bp) %>% cor()

          bmi     glu        bp
bmi 1.0000000 0.38868 0.3954109
glu 0.3886800 1.00000 0.3904300
bp  0.3954109 0.39043 1.0000000

diabetes2 %>% select(bmi, glu, bp) %>% cor(use = "complete")

          bmi       glu        bp
bmi 1.0000000 0.3673707 0.3260986
glu 0.3673707 1.0000000 0.3662607
bp  0.3260986 0.3662607 1.0000000
```

Example

```
diabetes1 %>% select(bmi, glu, bp) %>% cor()

          bmi     glu        bp
bmi 1.0000000 0.38868 0.3954109
glu 0.3886800 1.00000 0.3904300
bp  0.3954109 0.39043 1.0000000

diabetes2 %>% select(bmi, glu, bp) %>% cor(use = "pairwise")

          bmi       glu        bp
bmi 1.0000000 0.3673707 0.3954109
glu 0.3673707 1.0000000 0.3662607
bp  0.3954109 0.3662607 1.0000000
```

Example

```
mean(diabetes1$glu)

[1] 91.26018

mean(diabetes2$glu, na.rm = TRUE)

[1] 90.18639

var(diabetes1$glu)

[1] 132.1657

var(diabetes2$glu, na.rm = TRUE)

[1] 125.4755
```

Example

```
s1 <- lm(glu ~ bmi + bp + age, data = diabetes1) %>% summary()
s2 <- lm(glu ~ bmi + bp + age, data = diabetes2) %>% summary()

s1$r.squared

[1] 0.2450996

s2$r.squared

[1] 0.2185308
```

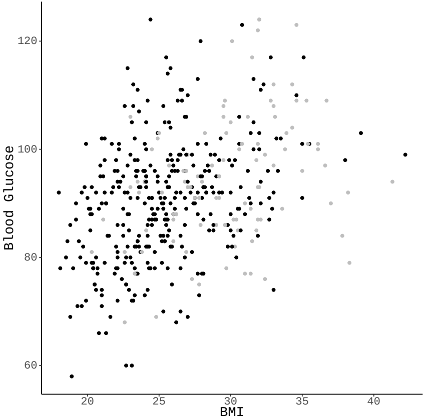
Example

```
s1$coef

              Estimate Std. Error   t value     Pr(>|t|)
(Intercept) 47.6809340 3.76076018 12.678536 1.351038e-31
bmi          0.6940756 0.11782779  5.890594 7.676778e-09
bp           0.1876015 0.03926201  4.778194 2.417752e-06
age          0.1549222 0.03871817  4.001279 7.396263e-05

s2$coef

              Estimate Std. Error   t value     Pr(>|t|)
(Intercept) 47.4726727 4.52293249 10.495994 1.837667e-22
bmi          0.7423454 0.14650653  5.066978 6.703498e-07
bp           0.1991158 0.04494852  4.429863 1.279770e-05
age          0.1132075 0.04384514  2.581985 1.024882e-02
```

Bad Methods (These almost never work)

(Unconditional) Mean Substitution

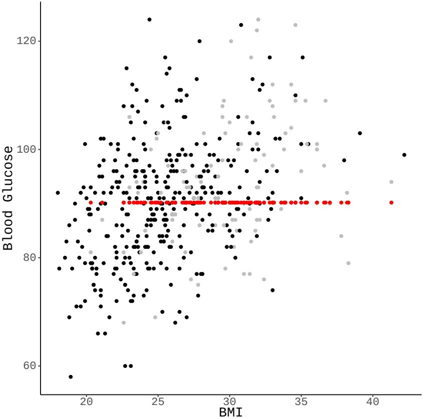
- Replace Y_{mis} with $\bar{Y}_{obs}$
  - Negatively biases regression slopes and correlations
  - Attenuates measures of linear association



Bad Methods (These almost never work)

(Unconditional) Mean Substitution

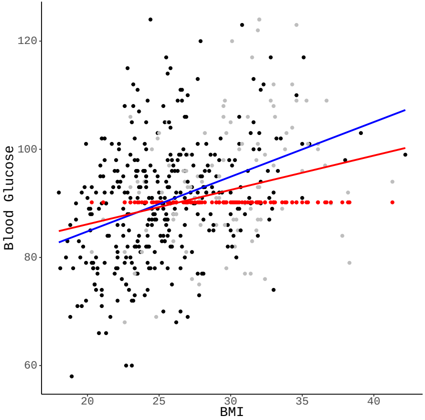
- Replace Y_{mis} with $\bar{Y}_{obs}$
  - Negatively biases regression slopes and correlations
  - Attenuates measures of linear association



Bad Methods (These almost never work)

(Unconditional) Mean Substitution

- Replace Y_{mis} with $\bar{Y}_{obs}$
  - Negatively biases regression slopes and correlations
  - Attenuates measures of linear association



Example

```
diabetes3              <- diabetes2
diabetes3[mVec, "glu"] <- mean(diabetes3$glu, na.rm = TRUE)

diabetes1 %>% select(bmi, glu, bp) %>% cor()

          bmi     glu        bp
bmi 1.0000000 0.38868 0.3954109
glu 0.3886800 1.00000 0.3904300
bp  0.3954109 0.39043 1.0000000

diabetes3 %>% select(bmi, glu, bp) %>% cor()

          bmi       glu        bp
bmi 1.0000000 0.2865045 0.3954109
glu 0.2865045 1.0000000 0.3079641
bp  0.3954109 0.3079641 1.0000000
```

Example

```
mean(diabetes1$glu)

[1] 91.26018

mean(diabetes3$glu, na.rm = TRUE)

[1] 90.18639

var(diabetes1$glu)

[1] 132.1657

var(diabetes3$glu, na.rm = TRUE)

[1] 95.88494
```

Example

```r
s1 <- lm(glu ~ bmi + bp + age, data = diabetes1) %>% summary()
s3 <- lm(glu ~ bmi + bp + age, data = diabetes3) %>% summary()

s1$r.squared

[1] 0.2450996

s3$r.squared

[1] 0.14431
```

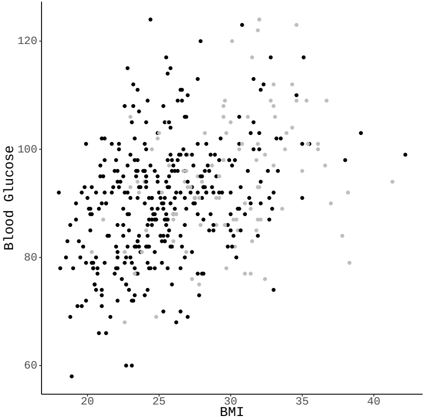
Example

```
s1$coef

              Estimate Std. Error  t value     Pr(>|t|)
(Intercept) 47.6809340 3.76076018 12.678536 1.351038e-31
bmi          0.6940756 0.11782779  5.890594 7.676778e-09
bp           0.1876015 0.03926201  4.778194 2.417752e-06
age          0.1549222 0.03871817  4.001279 7.396263e-05

s3$coef

              Estimate Std. Error  t value     Pr(>|t|)
(Intercept) 61.6619753 3.41039602 18.080591 5.453534e-55
bmi          0.4134903 0.10685058  3.869799 1.254941e-04
bp           0.1325834 0.03560424  3.723809 2.218616e-04
age          0.1044901 0.03511106  2.975988 3.082192e-03
```

Bad Methods (These almost never work)

Deterministic Regression Imputation (Conditional Mean Substitution)

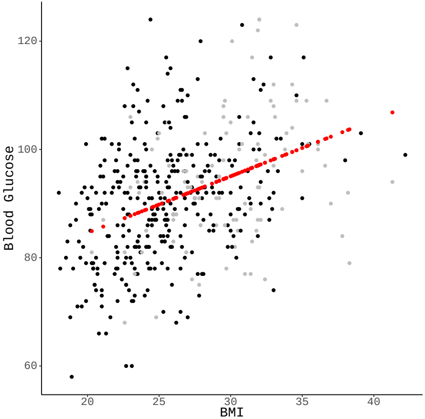
- Replace Y_{mis} with $\widehat{Y}_{mis}$ from some regression equation
    - Positively biases regression slopes and correlations
    - Inflates measures of linear association



Bad Methods (These almost never work)

Deterministic Regression Imputation
(Conditional Mean Substitution)

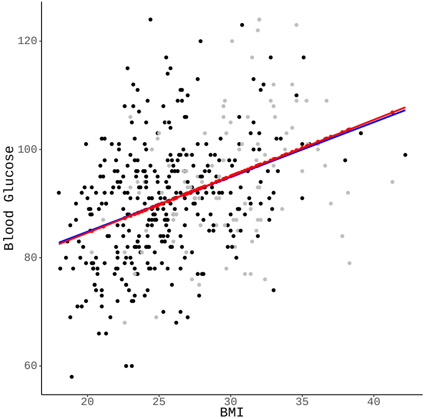
- Replace Y_{mis} with $\widehat{Y}_{mis}$ from some regression equation
  - Positively biases regression slopes and correlations
  - Inflates measures of linear association



Bad Methods (These almost never work)

Deterministic Regression Imputation
(Conditional Mean Substitution)

- Replace Y_{mis} with $\widehat{Y}_{mis}$ from some regression equation
  - Positively biases regression slopes and correlations
  - Inflates measures of linear association



Example

```r
diabetes3 <- mice(data      = diabetes2,
                  m         = 1,
                  maxit     = 1,
                  printFlag = FALSE,
                  method    = "norm.predict") %>%
    complete(1)
```

Example

```
diabetes1 %>% select(bmi, glu, bp) %>% cor()

          bmi     glu        bp
bmi 1.0000000 0.38868 0.3954109
glu 0.3886800 1.00000 0.3904300
bp  0.3954109 0.39043 1.0000000

diabetes3 %>% select(bmi, glu, bp) %>% cor()

          bmi       glu        bp
bmi 1.0000000 0.4214516 0.3954109
glu 0.4214516 1.0000000 0.4201420
bp  0.3954109 0.4201420 1.0000000
```

Example

```
mean(diabetes1$glu)

[1] 91.26018

mean(diabetes3$glu, na.rm = TRUE)

[1] 91.22398

var(diabetes1$glu)

[1] 132.1657

var(diabetes3$glu, na.rm = TRUE)

[1] 107.8749
```

Example

```
s1 <- lm(glu ~ bmi + bp + age, data = diabetes1) %>% summary()
s3 <- lm(glu ~ bmi + bp + age, data = diabetes3) %>% summary()

s1$r.squared

[1] 0.2450996

s3$r.squared

[1] 0.2737229
```

Example

```
s1$coef

              Estimate Std. Error   t value     Pr(>|t|)
(Intercept) 47.6809340 3.76076018 12.678536 1.351038e-31
bmi          0.6940756 0.11782779  5.890594 7.676778e-09
bp           0.1876015 0.03926201  4.778194 2.417752e-06
age          0.1549222 0.03871817  4.001279 7.396263e-05

s3$coef

              Estimate Std. Error   t value     Pr(>|t|)
(Intercept) 49.2273068 3.33259722 14.771454 2.347398e-40
bmi          0.6895143 0.10441308  6.603716 1.163618e-10
bp           0.1905886 0.03479202  5.477938 7.269262e-08
age          0.1189566 0.03431010  3.467101 5.780722e-04
```

Bad Methods (These almost never work)

General Issues with Deletion-Based Methods

- Biased parameter estimates unless data are MCAR
- Generalizability issues

General Issues with Simple Single Imputation Methods

- Biased parameter estimates even when data are MCAR
- Attenuates variability in any treated variables

Bad Methods (These almost never work)

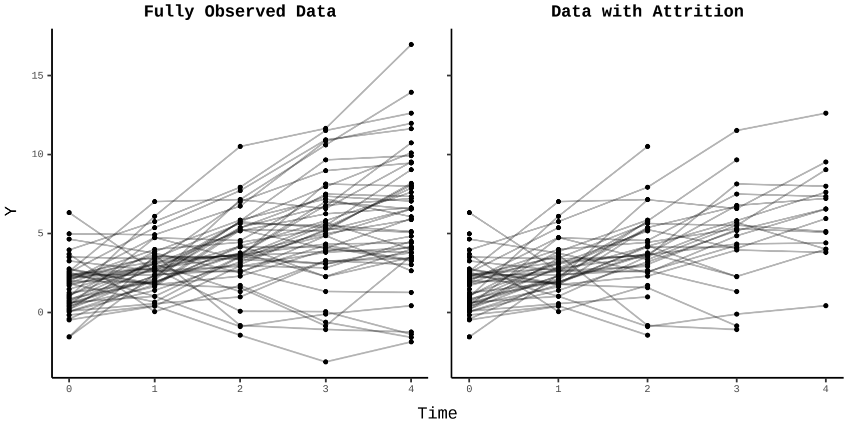
Averaging Available Items (Person-Mean Imputation)

- Compute aggregate scores using only available values
  - Missing data must be MCAR
  - Each item must contributes equally to the aggregate score

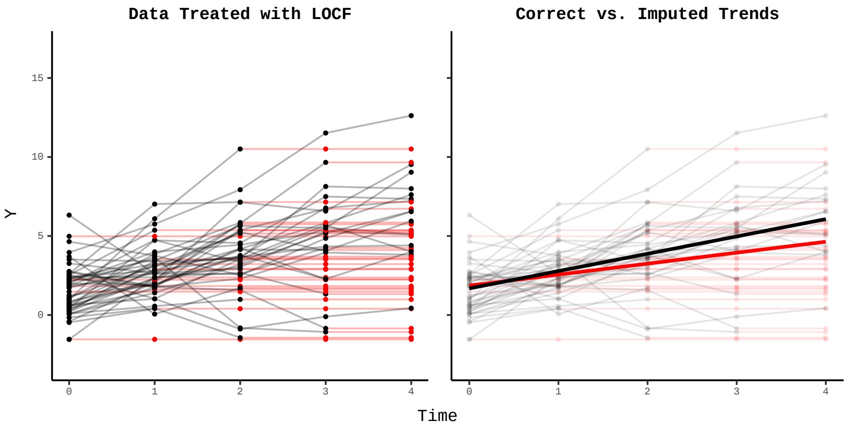
Last Observation Carried Forward (LOCF)

- Replace post-dropout values with the most recent observed value
  - Assume that dropouts would maintain their last known values
  - Attenuates estimates of growth/development

LOCF

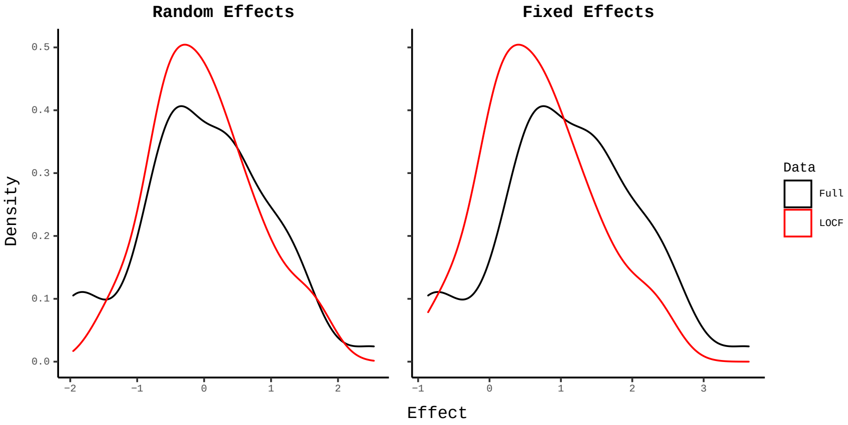

Fully Observed Data
Data with Attrition
Y
15
10
5
0
0
1
2
3
4
0
1
2
3
4
Time

LOCF


Data Treated with LOCF
Correct vs. Imputed Trends
Y
15
10
5
0
0
1
2
3
4
Time

Example

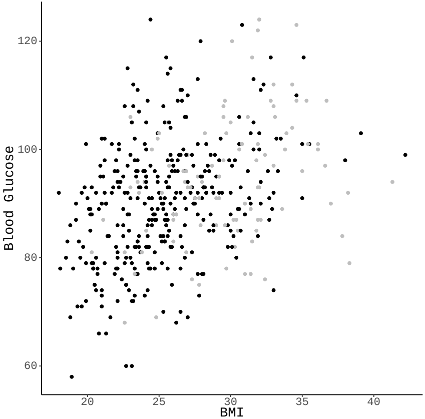
```
## Fit some multilevel regression models
fit1 <- lmer(y ~ t + (t | id), data = dat1) # Full data
fit2 <- lmer(y ~ t + (t | id), data = dat3) # LOCF data
```



OK Methods (These sometimes work)

Stochastic Regression Imputation

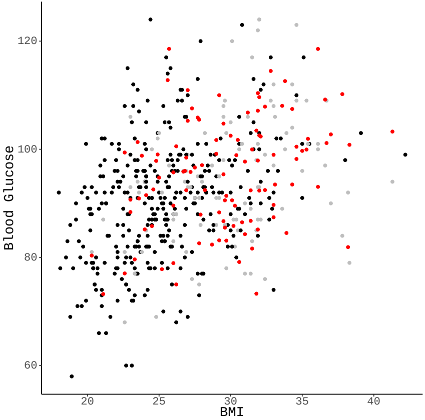
- Fill Y_{mis} with $\widehat{Y}_{mis}$ plus some random noise.
  - Produces unbiased parameter estimates and predictions
  - Computationally efficient
  - Attenuates standard errors
  - Makes CIs and prediction intervals too narrow



OK Methods (These sometimes work)

Stochastic Regression Imputation

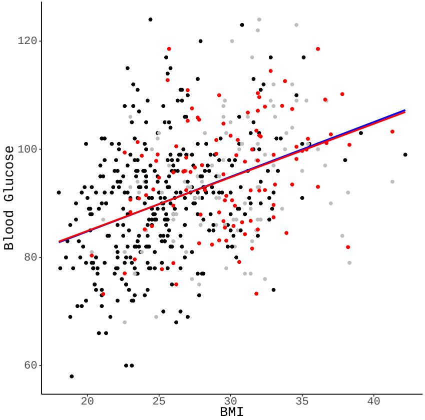
- Fill Y_{mis} with $\widehat{Y}_{mis}$ plus some random noise.
  - Produces unbiased parameter estimates and predictions
  - Computationally efficient
  - Attenuates standard errors
  - Makes CIs and prediction intervals too narrow



OK Methods (These sometimes work)

Stochastic Regression Imputation

- Fill Y_{mis} with $\widehat{Y}_{mis}$ plus some random noise.
  - Produces unbiased parameter estimates and predictions
  - Computationally efficient
  - Attenuates standard errors
  - Makes CIs and prediction intervals too narrow



Example

```
diabetes3 <- mice(data      = diabetes2,
                  m         = 1,
                  maxit     = 1,
                  printFlag = FALSE,
                  method    = "norm.nob") %>%
    complete(1)
```

Example

```
diabetes1 %>% select(bmi, glu, bp) %>% cor()

          bmi     glu        bp
bmi 1.0000000 0.38868 0.3954109
glu 0.3886800 1.00000 0.3904300
bp  0.3954109 0.39043 1.0000000

diabetes3 %>% select(bmi, glu, bp) %>% cor()

          bmi       glu        bp
bmi 1.0000000 0.3568889 0.3954109
glu 0.3568889 1.0000000 0.3653430
bp  0.3954109 0.3653430 1.0000000
```

Example

```
mean(diabetes1$glu)

[1] 91.26018

mean(diabetes3$glu)

[1] 91.19615

var(diabetes1$glu)

[1] 132.1657

var(diabetes3$glu)

[1] 125.404
```

Example

```
s1 <- lm(glu ~ bmi + bp + age, data = diabetes1) %>% summary()
s3 <- lm(glu ~ bmi + bp + age, data = diabetes3) %>% summary()

s1$r.squared

[1] 0.2450996

s3$r.squared

[1] 0.2103803
```

Example

```
s1$coef

             Estimate Std. Error   t value     Pr(>|t|)
(Intercept) 47.6809340 3.76076018 12.678536 1.351038e-31
bmi          0.6940756 0.11782779  5.890594 7.676778e-09
bp           0.1876015 0.03926201  4.778194 2.417752e-06
age          0.1549222 0.03871817  4.001279 7.396263e-05

s3$coef

             Estimate Std. Error   t value     Pr(>|t|)
(Intercept) 51.8038989 3.74658860 13.826951 2.442399e-36
bmi          0.6125714 0.11738378  5.218535 2.789768e-07
bp           0.1742313 0.03911406  4.454442 1.069164e-05
age          0.1390146 0.03857227  3.604005 3.493323e-04
```

OK Methods (These sometimes work)

Nonresponse Weighting

- Weight the observed cases to correct for nonresponse bias
  - Popular in survey research and official statistics
  - Only worth considering with *Unit Nonresponse*
  - Doesn’t make any sense with *Item Nonresponse*

Good Methods (These almost always work)

Multiple Imputation (MI)

- Replace the missing values with M plausible estimates
  - Essentially, a repeated application of stochastic regression imputation (with a particular type of regression model)
  - Produces unbiased parameter estimates and predictions
  - Produces “correct” standard errors, CIs, and prediction intervals
  - Very, very flexible
  - Computationally expensive

Good Methods (These almost always work)

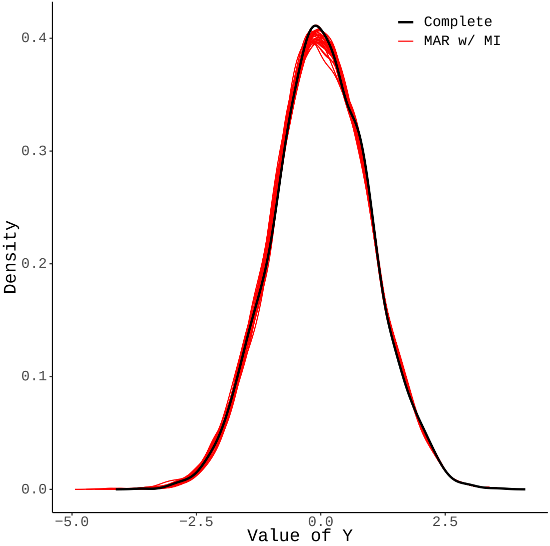
What happens when we apply MI to our previous MAR example?

```
## Estimate imputation model:
miceOut <- mice(data      = data.frame(y = yMar, x = x0),
                m         = 25,
                maxit     = 1,
                method    = "norm",
                printFlag = FALSE)

## Estimate and pool M correlations:
with(miceOut, cor(y, x))$analyses %>% unlist() %>% mean()

[1] 0.5041554
```

The MI-based parameter estimate looks good.

- MI produces unbiased estimates of the parameter when data are MAR.

Good Methods (These almost always work)



Good Methods (These *almost* always work)

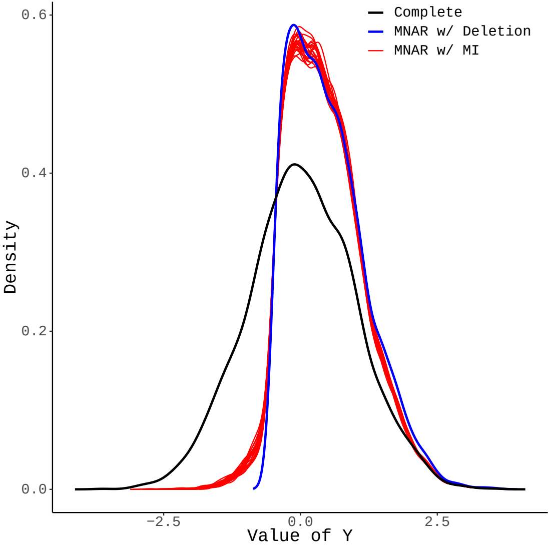
What about applying MI to our MNAR example?

```
## Estimate imputation model:
miceOut <- mice(data      = data.frame(y = yMnar, x = x0),
                m         = 25,
                maxit     = 1,
                method    = "norm",
                printFlag = FALSE)

## Estimate and pool M correlations:
with(miceOut, cor(y, x))$analyses %>% unlist() %>% mean()

[1] 0.4075215
```

The MI-based parameter estimate is still biased.

- MI cannot correct bias in parameter estimates when data are MNAR.

Good Methods (These *almost* always work)



Example

```r
miceOut <- mice(data      = diabetes2,
                m         = 25,
                maxit     = 1,
                printFlag = FALSE,
                method    = "norm")
```

Example

```
diabetes1 %>% select(bmi, glu, bp) %>% cor()

          bmi     glu        bp
bmi 1.0000000 0.38868 0.3954109
glu 0.3886800 1.00000 0.3904300
bp  0.3954109 0.39043 1.0000000

pooledCorMat(miceOut, c("bmi", "glu", "bp"))

          bmi       glu        bp
bmi 1.0000000 0.3827100 0.3954109
glu 0.3827100 1.0000000 0.3813741
bp  0.3954109 0.3813741 1.0000000
```

Example

```
mean(diabetes1$glu)

[1] 91.26018

with(miceOut, mean(glu))$analyses %>% unlist() %>% mean()

[1] 91.19761

var(diabetes1$glu)

[1] 132.1657

with(miceOut, var(glu))$analyses %>% unlist() %>% mean()

[1] 129.3954
```

Example

```
fit1 <- lm(glu ~ bmi + bp + age, data = diabetes1)
fit2 <- with(miceOut, lm(glu ~ bmi + bp + age))

summary(fit1)$r.squared

[1] 0.2450996

pool.r.squared(fit2)

          est     lo 95     hi 95 fmi
R^2 0.2283517 0.1404248 0.3241134 NaN
```

Example

```
summary(fit1)$coef

              Estimate Std. Error   t value     Pr(>|t|)
(Intercept) 47.6809340 3.76076018 12.678536 1.351038e-31
bmi          0.6940756 0.11782779  5.890594 7.676778e-09
bp           0.1876015 0.03926201  4.778194 2.417752e-06
age          0.1549222 0.03871817  4.001279 7.396263e-05

pool(fit2) %>% summary() %>% select(-df)

         term   estimate  std.error statistic      p.value
1 (Intercept) 49.3604551 4.94049286  9.990998 4.440892e-16
2         bmi  0.6856076 0.14561238  4.708443 6.867012e-06
3          bp  0.1881282 0.04595756  4.093520 6.698175e-05
4         age  0.1225923 0.04425146  2.770355 6.159043e-03
```

Good Methods (These almost always work)

Bayesian Modeling

- Treat missing values as just another parameter to be estimated
  - Models can be directly estimated in the presence of missing data
    - Essentially, runs MI behind-the-scenes during model estimation
  - The predictors of nonresponse must be included in the model, somehow
  - Computationally expensive

Good Methods (These almost always work)

Full Information Maximum Likelihood (FIML)

- Adjust the objective function to only consider the observed parts of the data
  - Models are directly estimated in the presence of missing data
  - The predictors of nonresponse must be included in the model, somehow
  - Unless you write your own optimization program, FIML is only available for certain types of models
  - In linear regression models, FIML cannot treat missing data on predictors (if the predictors are taken as fixed)

Example

```
fit <- diabetes2 %>%
    select(bmi, glu, bp) %>%
    lavCor(missing = "fiml", output = "sampstat")

mean(diabetes1$glu)

[1] 91.26018

fit$mean["glu"]

     glu
91.43176
```

Example

```
diabetes1 %>% select(bmi, glu, bp) %>% cov()

         bmi       glu        bp
bmi 19.51980  19.74191  24.16288
glu 19.74191 132.16571  62.08191
bp  24.16288  62.08191 191.30440

fit$cov

    bmi     glu     bp
bmi  19.476
glu  20.954 130.943
bp   24.108  63.330 190.872
```

Example

```r
mod <- "glu ~ 1 + bmi + bp + age"
fit <- sem(mod, data = diabetes2, missing = "fiml")

summary(fit1)$r.squared

[1] 0.2450996

inspect(fit, "r2")

  glu
0.248
```

Example

```
summary(fit1)$coef %>% round(3)

            Estimate Std. Error t value Pr(>|t|)
(Intercept)   47.681      3.761  12.679        0
bmi            0.694      0.118   5.891        0
bp             0.188      0.039   4.778        0
age            0.155      0.039   4.001        0

parameterEstimates(fit, ci = FALSE)[1:4, ]

  lhs op rhs    est    se      z pvalue
1 glu ~1     47.473 4.496 10.559  0.000
2 glu  ~ bmi  0.742 0.146  5.097  0.000
3 glu  ~  bp  0.199 0.045  4.456  0.000
4 glu  ~ age  0.113 0.044  2.597  0.009
```

References

Little, R. J. A. (1988). Missing-data adjustments in large surveys. *Journal of Business & Economic Statistics*, *6*(3), 287–296. doi: 10.1080/07350015.1988.10509663