EP16: Missing Values in Clinical Research: Multiple Imputation

11. Imputation with Non-linear Functional Forms

Nicole S. Erler

Department of Biostatistics, Erasmus Medical Center

n.erler@erasmusmc.nl

Imputation with mice

There is **no strategy** for MICE that can **guarantee valid imputations** when non-linear functional forms and/or interactions are involved, but **some settings** in **mice** **may help** to reduce bias in the resulting estimates.

For imputation of variables that have non-linear associations

- **PMM often works better** than imputation with a normal model,
- the **Just Another Variable** approach can reduce bias in interactions,
- passive imputation
- `quadratic` can help to impute variables with quadratic association.

Imputation with mice

For demonstration, we use a simulated example dataset `DFnonlin`:

- **y** continuous outcome
- **x** continuous (normal) covariate (50% missing values MCAR)
- **z** binary covariate (complete)

We assume a

- **quadratic effect** of x on y, and
- an **interaction** between x and z

```r
head(DF_nonlin)
```

```
##            y           x z
## 1 -0.4002016 -0.42298398 1
## 2  0.7883355 -1.54987816 0
## 3  0.1900922 -0.06442932 0
## 4  0.3321608  0.27088135 0
## 5  4.6146593  1.73528367 0
## 6  0.3705739          NA 0
```

```r
dim(DF_nonlin)
```

```
## [1] 200   3
```

Imputation with mice: JAV

Just Another Variable (JAV) approach:

- **pre-calculate** the non-linear form (or interaction term) in the incomplete data,
- add it as a column to the dataset, and
- impute it as if it was just another variable.

Imputation with mice: JAV

Just Another Variable (JAV) approach:

- **pre-calculate** the non-linear form (or interaction term) in the incomplete data,
- add it as a column to the dataset, and
- impute it as if it was just another variable.

```
DF2 <- DF_nonlin          # copy of the data, only for this example
DF2$xx <- DF2$x^2         # pre-calculate the quadratic term
DF2$xz <- DF2$x * DF2$z   # pre-calculate the interaction

# JAV imputation (using pmm and full predictor matrix)
impJAV <- mice(DF2, maxit = 20, printFlag = FALSE)
```

Imputation with mice: JAV

To relax the assumption of linear associations even more, we could introduce additional interactions with the outcome.

In this example, we can add an interaction between z and y:

```
DF3 <- DF2                   # make another copy of the data
DF3$yz <- DF3$y * DF3$z   # add interaction y and z

# JAV imputation with additional interaction
impJAV2 <- mice(DF3, maxit = 20, printFlag = FALSE)
```

Imputation with mice: Passive Imputation

Alternatively, we could impute all non-linear terms and interactions passively:

```r
# adapt the imputation method (we re-use the vector from impJAV2 here)
meth_passive <- impJAV2$method
meth_passive[c("xx", "xz", "yz")] <- c("~I(x^2)", "~I(x*z)", "~I(y*z)")


# adapt the predictor matrix (we re-use the matrix from impJAV2 here)
pred_passive <- impJAV2$predictorMatrix
pred_passive['x', 'xx'] <- 0
pred_passive[c('x', 'z'), 'xz'] <- 0
pred_passive[c('y', 'z'), 'yz'] <- 0

imp_passive <- mice(DF3, method = meth_passive,
                    predictorMatrix = pred_passive,
                    maxit = 20, printFlag = FALSE)
```

Imputation with mice: Polynomial Combination

The imputation method `quadratic` uses the **"polynomial combination" method** to impute covariates that have a **quadratic association** with the outcome (Van Buuren 2012 pp. 139–141; Vink and Buuren 2013).

- ➡ ensure the **imputed values** for x and x^2 are **consistent**
- ➡ **reduce bias** in the subsequent analysis that uses x and x^2

Imputation with mice: Polynomial Combination

The imputation method `quadratic` uses the **"polynomial combination" method** to impute covariates that have a **quadratic association** with the outcome (Van Buuren 2012 pp. 139–141; Vink and Buuren 2013).

- ➡ ensure the **imputed values** for x and x^2 are **consistent**
- ➡ **reduce bias** in the subsequent analysis that uses x and x^2

```
# adapt the imputation method (we re-use the vector from impJAV here)
methqdr <- impJAV$meth
methqdr[c("x", "xx", "xz")] <- c("quadratic", "~I(x^2)", "~I(x*z)")
```

➡ Here we use passive imputation for x^2 and the interaction.

Imputation with mice: polynomial combination

```r
# adapt the predictor matrix (we re-use the matrix from impJAV here)
predqdr <- impJAV$pred
predqdr['x', "xx"] <- 0              # prevent feedback
predqdr[c('x', 'z'), 'xz'] <- 0  # prevent feedback

impqdr <- mice(DF3, meth = methqdr, pred = predqdr,
               maxit = 20, printFlag = FALSE)
```

Imputation with mice: polynomial combination

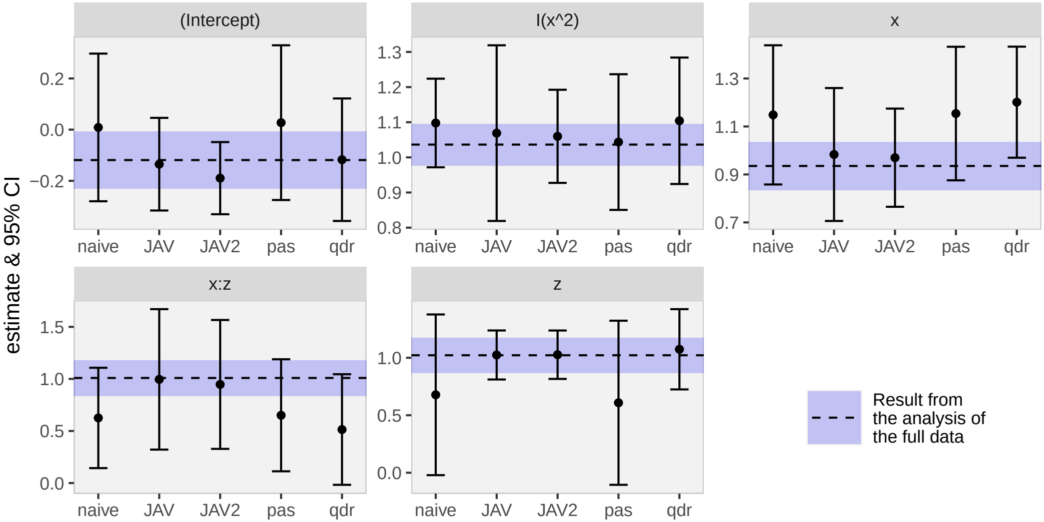
```r
# adapt the predictor matrix (we re-use the matrix from impJAV here)
predqdr <- impJAV$pred
predqdr['x', "xx"] <- 0              # prevent feedback
predqdr[c('x', 'z'), 'xz'] <- 0  # prevent feedback

impqdr <- mice(DF3, meth = methqdr, pred = predqdr,
               maxit = 20, printFlag = FALSE)
```

For comparison, we also run a naive version (using defaults):

```r
# naive imputation, using only y, x, z
impnaive <- mice(DF_nonlin, printFlag = FALSE)
```

Imputation with mice



Imputation with JointAI

The syntax we use to analyse and impute the current example using **JointAI** is similar to the specification of a standard linear model using `lm()`.

```
library("JointAI")
JointAI_nonlin <- lm_imp(y ~ x*z + I(x^2), data = DF_nonlin,
                         n.iter = 2500)
```

Imputation with JointAI

The syntax we use to analyse and impute the current example using **JointAI** is similar to the specification of a standard linear model using `lm()`.

```
library("JointAI")
JointAI_nonlin <- lm_imp(y ~ x*z + I(x^2), data = DF_nonlin,
                         n.iter = 2500)
```

Convergence of the Gibbs sampler can be checked using a `traceplot`.

```
traceplot(JointAI_nonlin, ncol = 3)
```

Imputation with JointAI

The syntax we use to analyse and impute the current example using **JointAI** is similar to the specification of a standard linear model using `lm()`.

```
library("JointAI")
JointAI_nonlin <- lm_imp(y ~ x*z + I(x^2), data = DF_nonlin,
                         n.iter = 2500)
```

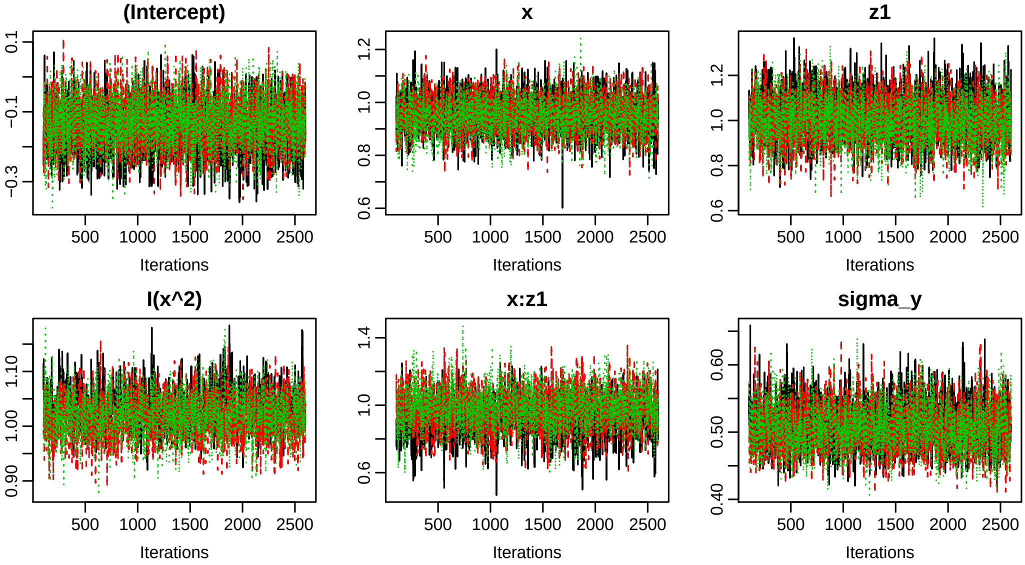
Convergence of the Gibbs sampler can be checked using a `traceplot`.

```
traceplot(JointAI_nonlin, ncol = 3)
```

Results (no separate analysis & pooling is necessary) can be obtained with the `summary()` function:

```
summary(JointAI_nonlin)
```

Imputation with JointAI: Convergence



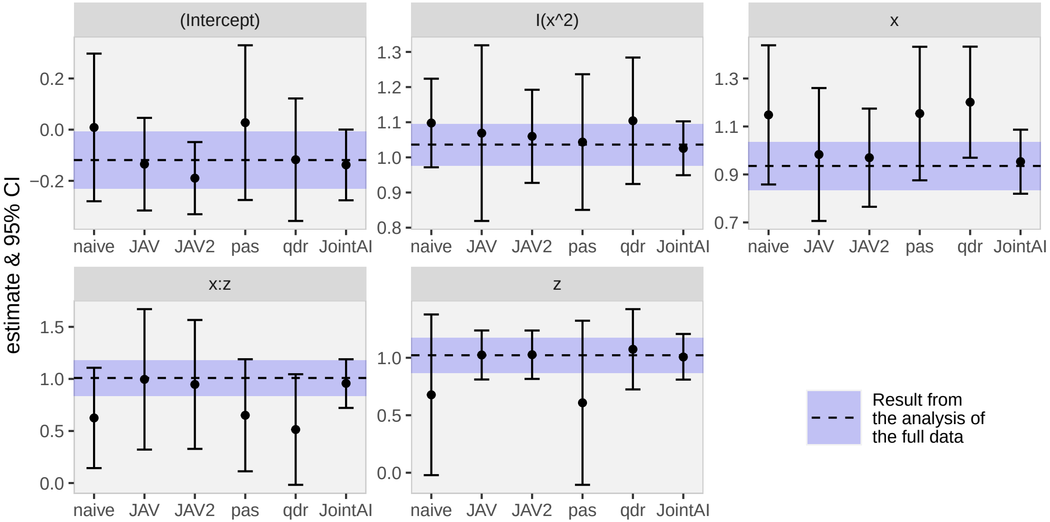
Imputation with JointAI: Model Summary

```
##
##  Linear model fitted with JointAI
##
## Call:
## lm_imp(formula = y ~ x * z + I(x^2), data = DF_nonlin, n.iter = 2500,
##     seed = 1234)
##
## Posterior summary:
##                Mean     SD   2.5%    97.5% tail-prob. GR-crit
## (Intercept) -0.138 0.0697 -0.276 0.000259     0.0512    1.09
## x            0.954 0.0683  0.820 1.086675     0.0000    1.02
## z1           1.007 0.1005  0.810 1.207309     0.0000    1.10
## I(x^2)       1.026 0.0393  0.949 1.102465     0.0000    1.32
## x:z1         0.957 0.1189  0.722 1.188642     0.0000    1.28
##
## Posterior summary of residual std. deviation:
##          Mean     SD  2.5% 97.5% GR-crit
## sigma_y 0.507 0.0334 0.447 0.576    1.01
##
##  [...]
```

Imputation with JointAI: Model Summary

```
## [...]
##
## MCMC settings:
## Iterations = 101:2600
## Sample size per chain = 2500
## Thinning interval = 1
## Number of chains = 3
##
## Number of observations: 200
```

Imputation with Non-linear Effects: Comparison



Imputation with Non-linear Effects: Comparison

Just Another Variable:

- Easy specification
- usually less bias than naive approach
- inconsistent imputed values

passive imputation

- easy specification
- consistent imputed values
- less flexible than JAV

polynomial combination

- consistent imputed values
- only available for quadratic association
- often numeric instabilities (warning messages)

Imputation with Non-linear Effects: Comparison

JointAI

- theoretically valid approach (= unbiased)
- similar specification to standard models
- simultaneous analysis & imputation instead of 3 steps

To use JointAI appropriately and to interpret the results correctly requires more knowledge about the underlying method than can be covered in this course.

Note:
The example used here only serves to demonstrate the different approaches. We cannot use these results to conclude which approach works better in general.

Your Turn!

Practical

Imputation with non-linear associations

html

References

Van Buuren, Stef. 2012. *Flexible Imputation of Missing Data*. Chapman & Hall/Crc Interdisciplinary Statistics. Taylor & Francis. https://stefvanbuuren.name/fimd/.

Vink, Gerko, and Stef van Buuren. 2013. “Multiple Imputation of Squared Terms.” *Sociological Methods & Research* 42 (4): 598–607.