Comparing Imputation Methods

Kayleigh Ryherd

9/11/2018

In this document, I compare the use of the `missForest` and `mice` packages to impute some data for the definition of poor comprehenders project.

Background

We are using the findings from Eckert et al. (2018) to inform our imputation methods. They found the most success with the `missForest` package (as compared to mean replacement and predictive mean matching using the `mice` package).

In this paper they do **explicit multiple imputation**, where 10 imputed datasets are generated and then pooled to for point and variance estimates. However, personal communication with the creator of `missForest`, Daniel Stekhoven, suggests that this method is not necessary. In his words:

> [...] randomForest provides an implicit multiple imputation by averaging over many decision/regression trees [...] When we use the different imputation methods, missForest was so much better (while at the same time underestimating the standard deviation of the CIs) that my intermediate hypothesis is; we do not need multiple imputation if we have the right data (when the data is right, I have not yet figured out).

So, it seems like we can just use `missForest` without having to worry about multiple imputation. To check this, we compare explicit multiple imputation (using `mice`) to `missForest`.

```
# read in libraries, data
library(missForest)
library(mice)
library(fBasics)
library(dplyr)
library(ggplot2)
library(gridExtra)
setwd("~/definitionofPCs")
# this has all the raw data, no overlapping subjects
data <- read.csv("All_IMPUTE_COMP.csv")
```

```
# look at missingness
data %>% summarize_all(funs(sum(is.na(.)) / length(.)))
```

```
##   SubjectID Project age.tested towre.w.ipm towre.nw.ipm wj3.wid.raw
## 1         0       0          0  0.06651885   0.06651885  0.07427938
##   wj3.watt.raw  ppvt.raw wasi.matr.raw wj3.rcomp.raw ktea2.raw
## 1   0.07538803 0.1441242    0.09645233     0.1862528 0.5365854
##   gm.rcomp.raw nd.rcomp.raw
## 1    0.5720621    0.8070953
```

```
# subset to just predictors and WJ3
dat_imp <- data[,-c(11:13)]
```

Comparing `mice` to `missForest`

Full dataset

First, we will use mice and missForest to impute the missing values we have and see how similar the values they create are.

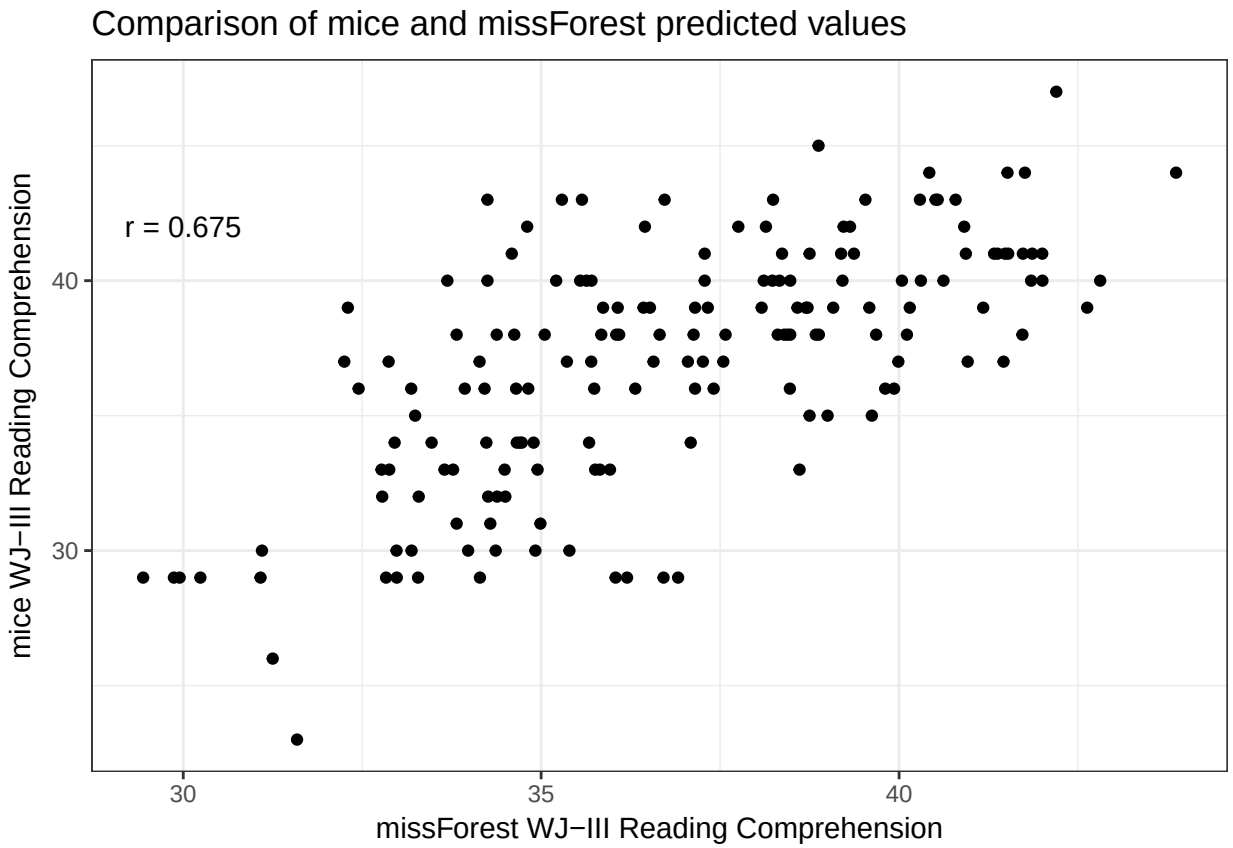
```
## impute all missing values for all variables
# missForest -- max iterations of 20
missForest <- missForest(dat_imp[,-c(1,2)], maxiter = 20)
```

```
##   missForest iteration 1 in progress...done!
##   missForest iteration 2 in progress...done!
##   missForest iteration 3 in progress...done!
##   missForest iteration 4 in progress...done!
##   missForest iteration 5 in progress...done!
##   missForest iteration 6 in progress...done!
```

```
# mice -- 10 imputations, predictive mean matching method, don't print all output, 20 max iterations
mice <- mice(dat_imp[,-c(1,2)], m = 10, method = "pmm", printFlag = FALSE, maxit = 20)
# create mice dataset
mice_complete <- complete(mice)

# create dataset to compare predictions from missForest and mice
compare <- data.frame(data$SubjectID)
compare$MF_wj3.rcomp.raw <- missForest$ximp$wj3.rcomp.raw
compare$M_wj3.rcomp.raw <- mice_complete$wj3.rcomp.raw
compare$real <- dat_imp$wj3.rcomp.raw
# select only rows that had missing data in original dataset
compare_missings <- compare[!complete.cases(compare$real),]

# test correlation
cor_test <- cor.test(compare_missings$MF_wj3.rcomp.raw, compare_missings$M_wj3.rcomp.raw)
ggplot(compare_missings, aes(MF_wj3.rcomp.raw, M_wj3.rcomp.raw)) +
  geom_point() + theme_bw() +
  labs(title = "Comparison of mice and missForest predicted values",
       x = "missForest WJ-III Reading Comprehension",
       y = "mice WJ-III Reading Comprehension") +
  annotate("text", x = 30, y = 42, label = paste0("r = ", round(cor_test$estimate, 3)))
```



The predictions from `missForest` and `mice` for `wj3.rcomp.raw` are significantly correlated, but not to an extreme extent. This suggests that the two methods are in fact producing different results. Let's try using data where we know the actual `wj3.rcomp.raw` to see how close the methods get.

Half of dataset removed

```
# make complete-cases dataset
cc_dat <- dat_imp[complete.cases(dat_imp),]
# randomly select half of the rows
randsample <- sample_frac(cc_dat, size = .5)
# set wj3.rcomp.raw in those rows to NA
cc_dat$wj3.rcomp.raw[cc_dat$SubjectID %in% randsample$SubjectID] <- NA

# sanity check -- is half of rcomp missing?
cc_dat %>% summarize_all(funs(sum(is.na(.)) / length(.)))
```

```
##   SubjectID Project age.tested towre.w.ipm towre.nw.ipm wj3.wid.raw
## 1         0       0          0           0            0           0
##   wj3.watt.raw ppvt.raw wasi.matr.raw wj3.rcomp.raw
## 1            0        0             0     0.4992743
```

```
## impute wj3.rcomp.raw
# xtrue provides actual dataset, 20 max iterations
missForest_imputeTest <- missForest(cc_dat[,-c(1,2)], xtrue = dat_imp[,-c(1,2)], maxiter = 20)
```

```
##   missForest iteration 1 in progress...done!
##   missForest iteration 2 in progress...done!
##   missForest iteration 3 in progress...done!
```

```
# mice -- 10 imputations, predictive mean matching method, don't print all output, 20 max iterations
mice_imputeTest <- mice(cc_dat[,-c(1,2)], m = 10, method = "pmm", printFlag = FALSE, maxit = 20)
# create mice dataset
mice_imputeTest_data <- complete(mice_imputeTest)

# create dataframe to compare the two methods
compare_test <- data.frame(cc_dat$SubjectID,
                           missForest_imputeTest$ximp$wj3.rcomp.raw,
                           mice_imputeTest_data$wj3.rcomp.raw)
names(compare_test) <- c("SubjectID", "missForest_wj3.rcomp.raw", "mice_wj3.rcomp.raw")
compare_test <- merge(compare_test, dat_imp[,c(1,10)], by = "SubjectID")
compare_test$missForest_error <- compare_test$missForest_wj3.rcomp.raw - compare_test$wj3.rcomp.raw
compare_test$mice_error <- compare_test$mice_wj3.rcomp.raw - compare_test$wj3.rcomp.raw

# sum the total error for each method
paste("missForest total error:", round(sum(abs(compare_test$missForest_error)),4))
```

```
## [1] "missForest total error: 748.0765"
```

```
paste("mice total error:", round(sum(abs(compare_test$mice_error)),4))
```

```
## [1] "mice total error: 898"
```

These data suggest that for our dataset, `missForest` is the best option, even without explicit multiple imputation.