MI in R

```
library(lavaan)
```

```
## This is lavaan 0.6-9
## lavaan is FREE software! Please report any bugs.
```

```
library(dplyr)
```

```
##
## Attaching package: 'dplyr'
```

```
## The following objects are masked from 'package:stats':
##
##     filter, lag
```

```
## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```
habit <- read.table("sem_categorical.dat", header = FALSE)
```

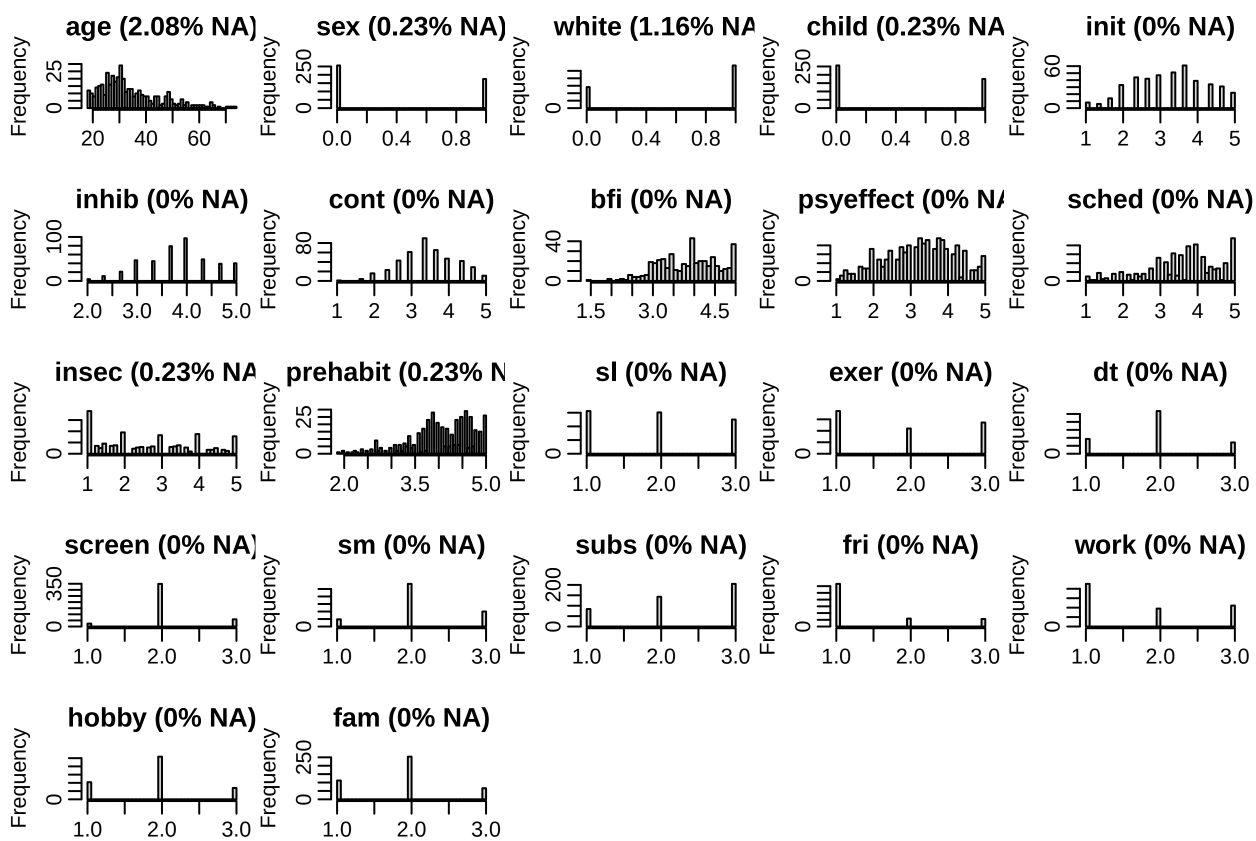
This is a longer dataset, so we'll add column names to the variables. Because we're using the same dataset as the one for Mplus, missing values are still "." At last line of code in this chunk tells R to turn all variables into numeric, which means that the "." (which are character) will be turned into NAs - and that's what we want.

```
colnames(habit) <- c("age", "sex", "white", "child", "init",
    "inhib", "cont", "bfi", "psyeffect", "sched", "insec", "prehabit",
    "sl", "exer", "dt", "screen", "sm", "subs", "fri", "work",
    "hobby", "fam")

habit <- data.frame(lapply(habit, function(x) as.numeric(as.character(x))))
```

```
## Warning in FUN(X[[i]], ...): NAs introduced by coercion

## Warning in FUN(X[[i]], ...): NAs introduced by coercion

## Warning in FUN(X[[i]], ...): NAs introduced by coercion

## Warning in FUN(X[[i]], ...): NAs introduced by coercion

## Warning in FUN(X[[i]], ...): NAs introduced by coercion

## Warning in FUN(X[[i]], ...): NAs introduced by coercion
```

First, a quick overview of the data. If you use your own dataset, you'll need to replace "habit" in the code with your dataset.

```r
nc <- max(5, ceiling(sqrt(ncol(habit))))
nr <- ceiling(ncol(habit)/nc)
par(mfrow = c(nr, nc), mgp = c(2, 0.6, 0), mar = c(2, 3, 3, 0.5))
for (i in 1:ncol(habit)) {
    if (is.numeric(habit[, i])) {
        hist(habit[, i], nclass = 50, xlab = "", main = paste0(names(habit[i]),
            " (", round(mean(is.na(habit[, i])) * 100, 2), "% NA)"))
    } else {
        barplot(table(habit[, i]), ylab = "Frequency", main = paste0(names(habit[i]),
            " (", round(mean(is.na(habit[, i])) * 100, 2), "% NA)"))
    }
}
```



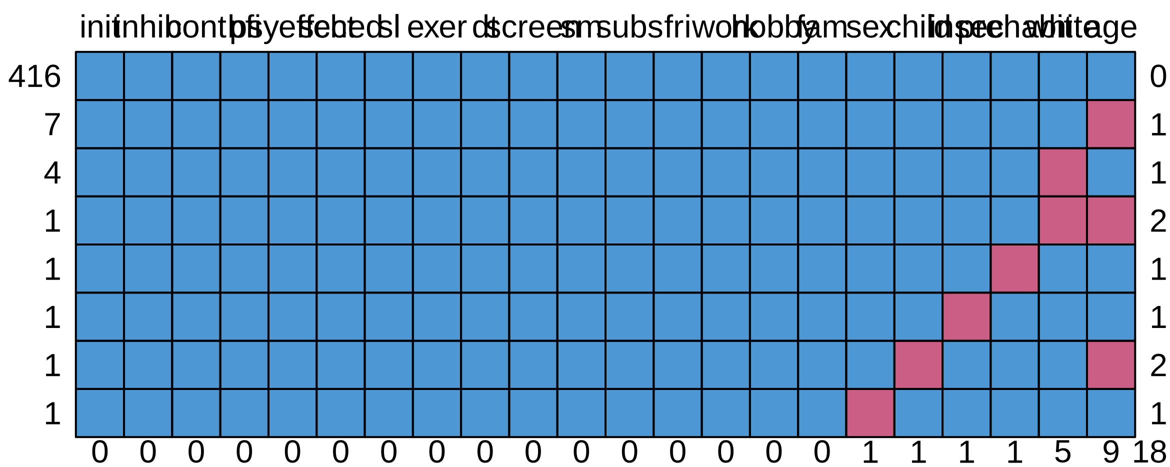
You can first visualize missing data in R using packages like MICE. There are many others you can choose from but MICE has nice visualizations. As usual with R, if you haven't installed a package yet, make sure you do. Once you have installed it, you can just load the package library (you need to load the package library every time you use a package, otherwise you will get an error saying that R doesn't recognize the function).

```r
# install.packages('mice')
library("mice")
```

```
##
```

```
## Attaching package: 'mice'

## The following object is masked from 'package:stats':
##
##     filter

## The following objects are masked from 'package:base':
##
##     cbind, rbind
```

```r
md.pattern(habit)
```




```
##     init inhib cont bfi psyeffect sched sl exer dt screen sm subs fri work
## 416    1     1    1   1         1     1  1    1  1      1  1    1   1    1
## 7      1     1    1   1         1     1  1    1  1      1  1    1   1    1
## 4      1     1    1   1         1     1  1    1  1      1  1    1   1    1
## 1      1     1    1   1         1     1  1    1  1      1  1    1   1    1
## 1      1     1    1   1         1     1  1    1  1      1  1    1   1    1
## 1      1     1    1   1         1     1  1    1  1      1  1    1   1    1
## 1      1     1    1   1         1     1  1    1  1      1  1    1   1    1
## 1      1     1    1   1         1     1  1    1  1      1  1    1   1    1
##         0     0    0   0         0     0  0    0  0      0  0    0   0    0
##     hobby fam sex child insec prehabit white age
## 416     1   1   1     1     1        1     1   1 0
## 7       1   1   1     1     1        1     1   0 1
```

```
## 4        1    1    1     1     1         1     0    1   1
## 1        1    1    1     1     1         1     0    0   2
## 1        1    1    1     1     1         0     1    1   1
## 1        1    1    1     1     0         1     1    1   1
## 1        1    1    1     0     1         1     1    0   2
## 1        1    1    0     1     1         1     1    1   1
##          0    0    1     1     1         1     5    9  18
```

This is a really cool output. Both windows show you the same information. On top, on the horizontal axis are the variables. On the left vertical axis are the sums of participants within a given missing data pattern across all variables. On the bottom horizontal axis are the sums of participants with missing data on each variable/column. A missing data pattern is basically each combination of missing data that we can find. That's represented on the right vertical axis. For example, you can see that there are 8 missing data patterns. The first is no missing data for 416 participants. The second is missing data on one 1 (right vertical axis) variable (age) for 7 participants (left vertical axis). The fourth missing data pattern is for 2 (right vertical axis) variables (age and race) for 1 participant (left vertical axis). And so on.

```
habit.sem <- "disrupt =~ psyeffect + sched + insec # Our latent variable 1 with 3 indicators
          sreg =~ init + inhib + cont + bfi # Our latent variable 2 with 4 indicators

          sl + exer + dt + subs ~ disrupt + sreg + prehabit + age + sex + white + child

          # Our regression with sl, exer, dt, and subs as endogenous variables and all else as exogenous
          # Note that we have a combination of latent and observed variables for exogenous variables
          # All endogeous are observed

          sreg ~~ disrupt + prehabit + age + sex + white + child # Covariances between exogenous variabl
          disrupt ~~ prehabit + age + sex + white + child
          prehabit ~~ age + sex + white + child
          age ~~ sex + white + child
          sex ~~ white + child
          white ~~ child
          "
```

Then we estimate the model as we did before

```
cont.fit <- sem(
  model = habit.sem, # tell R the model you're using (we specified it above and gave it a name)
  data = habit, # Tell R the dataset you're using (we gave it a name above)
  missing = "fiml", # Listiwse is the default, and we know there's a lot of missing data
  estimator = "mlr" # Tell R the estimator we're using
)

summary(cont.fit, # Tell R which model to output
        fit.measures = TRUE,# Tell R you want the fit measures (CFI, TLI etc.)
        standardized = TRUE) # Tell R you want the standardized solution
```

```
## lavaan 0.6-9 ended normally after 141 iterations
##
##   Estimator                                         ML
##   Optimization method                           NLMINB
##   Number of model parameters                        94
##
```

```
##   Number of observations                           432
##   Number of missing patterns                         8
##
## Model Test User Model:
##                                               Standard      Robust
##   Test Statistic                               164.661     166.366
##   Degrees of freedom                                58          58
##   P-value (Chi-square)                           0.000       0.000
##   Scaling correction factor                                  0.990
##        Yuan-Bentler correction (Mplus variant)
##
## Model Test Baseline Model:
##
##   Test statistic                              1223.810    1182.152
##   Degrees of freedom                               120         120
##   P-value                                        0.000       0.000
##   Scaling correction factor                                  1.035
##
## User Model versus Baseline Model:
##
##   Comparative Fit Index (CFI)                    0.903       0.898
##   Tucker-Lewis Index (TLI)                       0.800       0.789
##
##   Robust Comparative Fit Index (CFI)                         0.902
##   Robust Tucker-Lewis Index (TLI)                            0.798
##
## Loglikelihood and Information Criteria:
##
##   Loglikelihood user model (H0)              -8314.432   -8314.432
##   Scaling correction factor                                  0.984
##       for the MLR correction
##   Loglikelihood unrestricted model (H1)      -8232.102   -8232.102
##   Scaling correction factor                                  0.986
##       for the MLR correction
##
##   Akaike (AIC)                               16816.865   16816.865
##   Bayesian (BIC)                             17199.297   17199.297
##   Sample-size adjusted Bayesian (BIC)        16900.994   16900.994
##
## Root Mean Square Error of Approximation:
##
##   RMSEA                                          0.065       0.066
##   90 Percent confidence interval - lower         0.054       0.054
##   90 Percent confidence interval - upper         0.077       0.078
##   P-value RMSEA <= 0.05                          0.016       0.014
##
##   Robust RMSEA                                               0.065
##   90 Percent confidence interval - lower                     0.054
##   90 Percent confidence interval - upper                     0.077
##
## Standardized Root Mean Square Residual:
##
##   SRMR                                           0.036       0.036
##
```

```
## Parameter Estimates:
##
##   Standard errors                             Sandwich
##   Information bread                           Observed
##   Observed information based on                Hessian
##
## Latent Variables:
##                    Estimate  Std.Err  z-value  P(>|z|)   Std.lv  Std.all
##   disrupt =~
##     psyeffect         1.000                               0.680    0.724
##     sched             1.070    0.184    5.811    0.000    0.728    0.749
##     insec             1.022    0.145    7.029    0.000    0.695    0.538
##   sreg =~
##     init              1.000                               0.792    0.818
##     inhib             0.425    0.046    9.258    0.000    0.336    0.483
##     cont              0.616    0.047   13.146    0.000    0.488    0.654
##     bfi               0.781    0.043   18.342    0.000    0.619    0.879
##
## Regressions:
##                    Estimate  Std.Err  z-value  P(>|z|)   Std.lv  Std.all
##   sl ~
##     disrupt          -0.318    0.074   -4.271    0.000   -0.216   -0.269
##     sreg             -0.026    0.060   -0.433    0.665   -0.020   -0.025
##     prehabit         -0.028    0.061   -0.449    0.654   -0.028   -0.023
##     age              -0.001    0.004   -0.398    0.690   -0.001   -0.021
##     sex              -0.016    0.079   -0.209    0.834   -0.016   -0.010
##     white            -0.061    0.082   -0.745    0.456   -0.061   -0.036
##     child             0.136    0.083    1.646    0.100    0.136    0.083
##   exer ~
##     disrupt          -0.319    0.075   -4.235    0.000   -0.217   -0.254
##     sreg              0.028    0.061    0.464    0.643    0.022    0.026
##     prehabit          0.054    0.064    0.845    0.398    0.054    0.043
##     age              -0.001    0.004   -0.148    0.882   -0.001   -0.008
##     sex              -0.001    0.083   -0.016    0.987   -0.001   -0.001
##     white            -0.130    0.088   -1.474    0.141   -0.130   -0.071
##     child             0.093    0.089    1.051    0.293    0.093    0.054
##   dt ~
##     disrupt          -0.096    0.062   -1.562    0.118   -0.066   -0.107
##     sreg             -0.028    0.046   -0.604    0.546   -0.022   -0.036
##     prehabit          0.022    0.047    0.464    0.643    0.022    0.024
##     age               0.003    0.003    1.036    0.300    0.003    0.059
##     sex              -0.024    0.060   -0.395    0.693   -0.024   -0.019
##     white            -0.011    0.063   -0.167    0.868   -0.011   -0.008
##     child             0.096    0.065    1.472    0.141    0.096    0.077
##   subs ~
##     disrupt          -0.267    0.071   -3.781    0.000   -0.182   -0.236
##     sreg              0.019    0.058    0.332    0.740    0.015    0.020
##     prehabit          0.078    0.063    1.235    0.217    0.078    0.068
##     age               0.003    0.003    0.934    0.350    0.003    0.047
##     sex              -0.021    0.076   -0.276    0.783   -0.021   -0.013
##     white             0.162    0.081    2.008    0.045    0.162    0.099
##     child             0.028    0.082    0.344    0.731    0.028    0.018
##
## Covariances:
```

```
##                    Estimate  Std.Err  z-value  P(>|z|)   Std.lv  Std.all
##   disrupt ~~
##     sreg             -0.072    0.039   -1.837    0.066   -0.134   -0.134
##   sreg ~~
##     prehabit          0.119    0.030    4.002    0.000    0.150    0.223
##     age               1.731    0.501    3.458    0.001    2.185    0.190
##     sex              -0.046    0.020   -2.305    0.021   -0.059   -0.119
##     white             0.024    0.019    1.251    0.211    0.030    0.064
##     child             0.057    0.021    2.777    0.005    0.072    0.147
##   disrupt ~~
##     prehabit          0.074    0.028    2.653    0.008    0.108    0.161
##     age              -0.689    0.462   -1.490    0.136   -1.013   -0.088
##     sex              -0.021    0.022   -0.945    0.345   -0.030   -0.061
##     white            -0.028    0.019   -1.527    0.127   -0.042   -0.089
##     child             0.058    0.023    2.523    0.012    0.086    0.174
##   prehabit ~~
##     age               1.003    0.371    2.703    0.007    1.003    0.130
##     sex              -0.024    0.016   -1.503    0.133   -0.024   -0.072
##     white            -0.020    0.015   -1.342    0.180   -0.020   -0.064
##     child             0.025    0.016    1.571    0.116    0.025    0.075
##   age ~~
##     sex              -0.342    0.266   -1.287    0.198   -0.342   -0.061
##     white             0.746    0.244    3.054    0.002    0.746    0.138
##     child             1.072    0.260    4.123    0.000    1.072    0.190
##   sex ~~
##     white            -0.018    0.011   -1.632    0.103   -0.018   -0.080
##     child            -0.000    0.012   -0.033    0.973   -0.000   -0.002
##   white ~~
##     child            -0.011    0.011   -0.945    0.345   -0.011   -0.046
##  .sl ~~
##    .exer              0.017    0.034    0.515    0.607    0.017    0.027
##    .dt               -0.010    0.023   -0.433    0.665   -0.010   -0.021
##    .subs              0.059    0.028    2.101    0.036    0.059    0.103
##  .exer ~~
##    .dt                0.011    0.024    0.441    0.659    0.011    0.021
##    .subs             -0.033    0.030   -1.086    0.278   -0.033   -0.054
##  .dt ~~
##    .subs              0.011    0.023    0.454    0.650    0.011    0.024
##
## Intercepts:
##                    Estimate  Std.Err  z-value  P(>|z|)   Std.lv  Std.all
##    .psyeffect         3.251    0.045   71.940    0.000    3.251    3.461
##    .sched             3.641    0.047   77.870    0.000    3.641    3.747
##    .insec             2.661    0.062   42.751    0.000    2.661    2.060
##    .init              3.270    0.047   70.144    0.000    3.270    3.375
##    .inhib             3.826    0.033  114.233    0.000    3.826    5.496
##    .cont              3.448    0.036   96.077    0.000    3.448    4.623
##    .bfi               3.861    0.034  113.981    0.000    3.861    5.484
##    .sl                2.083    0.275    7.570    0.000    2.083    2.593
##    .exer              1.734    0.286    6.065    0.000    1.734    2.027
##    .dt                1.730    0.229    7.561    0.000    1.730    2.817
##    .subs              1.746    0.287    6.086    0.000    1.746    2.273
##     prehabit          4.056    0.032  125.295    0.000    4.056    6.035
##     age              34.647    0.557   62.200    0.000   34.647    3.017
```

```
##     sex               0.406    0.024   17.163    0.000    0.406    0.827
##     white             0.669    0.023   29.394    0.000    0.669    1.423
##     child             0.406    0.024   17.159    0.000    0.406    0.826
##     disrupt           0.000                               0.000    0.000
##     sreg              0.000                               0.000    0.000
##
## Variances:
##                    Estimate  Std.Err  z-value  P(>|z|)   Std.lv  Std.all
##    .psyeffect         0.420    0.077    5.435    0.000    0.420    0.476
##    .sched             0.415    0.093    4.458    0.000    0.415    0.439
##    .insec             1.186    0.096   12.299    0.000    1.186    0.711
##    .init              0.311    0.034    9.053    0.000    0.311    0.331
##    .inhib             0.371    0.025   14.936    0.000    0.371    0.766
##    .cont              0.318    0.029   11.123    0.000    0.318    0.572
##    .bfi               0.113    0.020    5.710    0.000    0.113    0.227
##    .sl                0.599    0.027   21.976    0.000    0.599    0.929
##    .exer              0.683    0.030   22.816    0.000    0.683    0.934
##    .dt                0.370    0.023   16.029    0.000    0.370    0.981
##    .subs              0.545    0.029   18.762    0.000    0.545    0.923
##     prehabit          0.452    0.032   13.991    0.000    0.452    1.000
##     age             131.838   10.579   12.462    0.000  131.838    1.000
##     sex               0.241    0.004   54.251    0.000    0.241    1.000
##     white             0.221    0.008   28.625    0.000    0.221    1.000
##     child             0.241    0.004   54.240    0.000    0.241    1.000
##     disrupt           0.463    0.091    5.095    0.000    1.000    1.000
##     sreg              0.628    0.058   10.808    0.000    1.000    1.000
```

We'll show you two options for MI in R. On the first, you'll create the imputed dataset using a flexible MI package called mice, then you'll fit your lavaan model to your datasets using semTools package.

In the second method, we'll just use the semTools package to do MI and fit the models. In both cases you'll need to download semTools (if your first time using it) and then load the library (every time you use it within an R session).

We'll first do the two-part method in which we mice to do multiple imputation with Bayesian regression, as in Mplus. This is the Rubin (1987) and Schafer (1997) method. The function for this algorithm is called "norm".

```r
# install.packages('semTools')
library(semTools)
```

```
##
```

```
## ###############################################################################
```

```
## This is semTools 0.5-5
```

```
## All users of R (or SEM) are invited to submit functions or ideas for functions.
```

```
## ###############################################################################
```

```r
habit.imp <- mice(habit, # the dataframe name
                  maxit = 0, # A scalar giving the number of iterations, default is 5
                  m = 10, # Number of imputed datasets, default is 5
                  defaultMethod = "norm") # The method we're choosing, which is norm
```

Now we use semTools package to fit a lavaan model using the imputed values we obtained in mice. To do this, we need to clean the mice output a little to create the 5 datasets it generated during imputation. That's the two lines of code below.

```r
mice.imp <- NULL
for (i in 1:10) mice.imp[[i]] <- complete(habit.imp, action = i,
    inc = FALSE)  # On this line, you'll need to change the 1:5 (one through 5) if you have more than 5
```

We then fit the model.

```r
impsem <- runMI(habit.sem, # Tell R the model we specified earlier
               data = mice.imp, # Tell R the imputed datasets that we created with mice
               fun = "sem" # tell MI the
               )
```

And we get our output

```r
summary(impsem, # Tell R which model to output
        fit.measures = TRUE,# Tell R you want the fit measures (CFI, TLI etc.)
        standardized = TRUE) # Tell R you want the standardized solution
```

```
## lavaan.mi object based on 10 imputed data sets.
## See class?lavaan.mi help page for available methods.
##
## Convergence information:
## The model converged on 10 imputed data sets
##
## Rubin's (1987) rules were used to pool point and SE estimates across 10 imputed data sets, and to cal
##
## Model Test User Model:
##
##   Test statistic                               162.218
##   Degrees of freedom                                58
##   P-value                                        0.000
##
## Model Test Baseline Model:
##
##   Test statistic                              1214.224
##   Degrees of freedom                               120
##   P-value                                        0.000
##
## User Model versus Baseline Model:
##
##   Comparative Fit Index (CFI)                    0.905
##   Tucker-Lewis Index (TLI)                       0.803
##
## Root Mean Square Error of Approximation:
```

```
##
##   RMSEA                                          0.064
##   90 Percent confidence interval - lower         0.053
##   90 Percent confidence interval - upper         0.076
##   P-value RMSEA <= 0.05                          0.021
##
## Standardized Root Mean Square Residual:
##
##   SRMR                                           0.038
```

```
##
## Parameter Estimates:
##
##   Standard errors                             Standard
##   Information                                 Expected
##   Information saturated (h1) model          Structured
##
## Latent Variables:
##                    Estimate  Std.Err  t-value       df  P(>|t|)   Std.lv
##   disrupt =~
##     psyeffect         1.000                                         0.680
##     sched             1.071    0.103   10.367      Inf    0.000    0.728
##     insec             1.022    0.114    8.966      Inf    0.000    0.695
##   sreg =~
##     init              1.000                                         0.793
##     inhib             0.425    0.044    9.752      Inf    0.000    0.336
##     cont              0.616    0.045   13.712      Inf    0.000    0.488
##     bfi               0.781    0.045   17.267      Inf    0.000    0.619
##   Std.all
##
##     0.724
##     0.749
##     0.538
##
##     0.818
##     0.483
##     0.655
##     0.879
##
## Regressions:
##                    Estimate  Std.Err  t-value       df  P(>|t|)   Std.lv
##   sl ~
##     disrupt          -0.317    0.073   -4.364      Inf    0.000   -0.216
##     sreg             -0.026    0.056   -0.469      Inf    0.639   -0.021
##     prehabit         -0.030    0.060   -0.496      Inf    0.620   -0.030
##     age              -0.001    0.003   -0.314      Inf    0.753   -0.001
##     sex              -0.016    0.079   -0.208      Inf    0.835   -0.016
##     white            -0.067    0.082   -0.815      Inf    0.415   -0.067
##     child             0.136    0.082    1.655      Inf    0.098    0.136
##   exer ~
##     disrupt          -0.319    0.077   -4.132      Inf    0.000   -0.217
##     sreg              0.029    0.060    0.479      Inf    0.632    0.023
##     prehabit          0.053    0.064    0.832      Inf    0.406    0.053
##     age              -0.001    0.004   -0.244      Inf    0.808   -0.001
```

```
##     sex               -0.002    0.084   -0.023      Inf    0.981   -0.002
##     white             -0.126    0.088   -1.436      Inf    0.151   -0.126
##     child              0.095    0.087    1.088      Inf    0.277    0.095
##   dt ~
##     disrupt           -0.096    0.055   -1.761      Inf    0.078   -0.066
##     sreg              -0.027    0.043   -0.622      Inf    0.534   -0.021
##     prehabit           0.022    0.047    0.461      Inf    0.645    0.022
##     age                0.003    0.003    1.121      Inf    0.262    0.003
##     sex               -0.025    0.061   -0.407      Inf    0.684   -0.025
##     white             -0.012    0.064   -0.181      Inf    0.857   -0.012
##     child              0.096    0.064    1.503      Inf    0.133    0.096
##   subs ~
##     disrupt           -0.267    0.069   -3.892      Inf    0.000   -0.182
##     sreg               0.020    0.053    0.379      Inf    0.704    0.016
##     prehabit           0.075    0.057    1.319      Inf    0.187    0.075
##     age                0.003    0.003    0.932      Inf    0.351    0.003
##     sex               -0.022    0.075   -0.290      Inf    0.772   -0.022
##     white              0.158    0.078    2.024      Inf    0.043    0.158
##     child              0.029    0.078    0.367      Inf    0.714    0.029
##   Std.all
##
##    -0.269
##    -0.026
##    -0.025
##    -0.016
##    -0.010
##    -0.039
##     0.083
##
##    -0.254
##     0.026
##     0.042
##    -0.012
##    -0.001
##    -0.069
##     0.055
##
##    -0.107
##    -0.035
##     0.024
##     0.057
##    -0.020
##    -0.009
##     0.076
##
##    -0.236
##     0.021
##     0.066
##     0.046
##    -0.014
##     0.097
##     0.018
##
## Covariances:
```

```
##                    Estimate  Std.Err  t-value       df  P(>|t|)   Std.lv
##   disrupt ~~
##     sreg            -0.072    0.033   -2.183      Inf    0.029   -0.134
##   sreg ~~
##     prehabit         0.118    0.029    4.130      Inf    0.000    0.149
##     age              1.691    0.482    3.512      Inf    0.000    2.134
##     sex             -0.047    0.020   -2.283      Inf    0.022   -0.059
##     white            0.023    0.019    1.172      Inf    0.241    0.029
##     child            0.057    0.020    2.788      Inf    0.005    0.072
##   disrupt ~~
##     prehabit         0.072    0.026    2.761      Inf    0.006    0.107
##     age             -0.694    0.441   -1.574      Inf    0.116   -1.021
##     sex             -0.020    0.019   -1.077      Inf    0.281   -0.030
##     white           -0.027    0.018   -1.516      Inf    0.130   -0.040
##     child            0.058    0.019    3.028      Inf    0.002    0.086
##   prehabit ~~
##     age              0.965    0.374    2.577 4024.431    0.010    0.965
##     sex             -0.024    0.016   -1.473      Inf    0.141   -0.024
##     white           -0.021    0.015   -1.374      Inf    0.170   -0.021
##     child            0.025    0.016    1.562      Inf    0.118    0.025
##   age ~~
##     sex             -0.331    0.272   -1.218      Inf    0.223   -0.331
##     white            0.732    0.262    2.788 3885.525    0.005    0.732
##     child            1.048    0.276    3.796      Inf    0.000    1.048
##   sex ~~
##     white           -0.018    0.011   -1.650      Inf    0.099   -0.018
##     child           -0.000    0.012   -0.035      Inf    0.972   -0.000
##   white ~~
##     child           -0.010    0.011   -0.921      Inf    0.357   -0.010
##  .sl ~~
##    .exer             0.017    0.032    0.547      Inf    0.584    0.017
##    .dt              -0.010    0.023   -0.442      Inf    0.659   -0.010
##    .subs             0.059    0.029    2.062      Inf    0.039    0.059
##  .exer ~~
##    .dt               0.011    0.025    0.435      Inf    0.663    0.011
##    .subs            -0.033    0.030   -1.095      Inf    0.273   -0.033
##  .dt ~~
##    .subs             0.011    0.022    0.488      Inf    0.625    0.011
##   Std.all
##
##    -0.134
##
##     0.221
##     0.186
##    -0.120
##     0.061
##     0.147
##
##     0.158
##    -0.089
##    -0.061
##    -0.086
##     0.175
##
```

```
##     0.125
##    -0.071
##    -0.066
##     0.076
##
##    -0.059
##     0.136
##     0.186
##
##    -0.080
##    -0.002
##
##    -0.044
##
##     0.027
##    -0.022
##     0.103
##
##     0.021
##    -0.054
##
##     0.024
##
## Variances:
##                    Estimate  Std.Err  t-value       df  P(>|t|)   Std.lv
##    .psyeffect         0.420    0.048    8.744      Inf    0.000    0.420
##    .sched             0.415    0.052    7.945      Inf    0.000    0.415
##    .insec             1.185    0.094   12.653      Inf    0.000    1.185
##    .init              0.311    0.034    9.011      Inf    0.000    0.311
##    .inhib             0.371    0.027   13.961      Inf    0.000    0.371
##    .cont              0.318    0.025   12.876      Inf    0.000    0.318
##    .bfi               0.113    0.018    6.256      Inf    0.000    0.113
##    .sl                0.599    0.042   14.252      Inf    0.000    0.599
##    .exer              0.683    0.048   14.297      Inf    0.000    0.683
##    .dt                0.370    0.025   14.591      Inf    0.000    0.370
##    .subs              0.545    0.038   14.340      Inf    0.000    0.545
##     prehabit          0.452    0.031   14.655      Inf    0.000    0.452
##     age             131.182    8.952   14.654      Inf    0.000  131.182
##     sex               0.241    0.016   14.655      Inf    0.000    0.241
##     white             0.221    0.015   14.655      Inf    0.000    0.221
##     child             0.241    0.016   14.655      Inf    0.000    0.241
##     disrupt           0.462    0.065    7.056      Inf    0.000    1.000
##     sreg              0.628    0.066    9.476      Inf    0.000    1.000
##   Std.all
##     0.476
##     0.439
##     0.711
##     0.331
##     0.766
##     0.572
##     0.227
##     0.929
##     0.934
##     0.981
```

```
##     0.924
##     1.000
##     1.000
##     1.000
##     1.000
##     1.000
##     1.000
##     1.000
```

Now let's do the second method, in which we use semTools to do MI and fit the model in one go.

```
impsem2 <- runMI(habit.sem, # tell which model to estimate
              data = habit, # tell which data to use
              m = 5, # how many datasets we're creating
              miPackage = "mice", # which package we're using to do MI
              fun = "sem", # the sem function we're using to estimate the model
              meanstructure = TRUE) # whether we also want the mean structure to be estimated

summary(impsem2, # Tell R which model to output
        fit.measures = TRUE,# Tell R you want the fit measures (CFI, TLI etc.)
        standardized = TRUE) # Tell R you want the standardized solution
```

```
## lavaan.mi object based on 5 imputed data sets.
## See class?lavaan.mi help page for available methods.
##
## Convergence information:
## The model converged on 5 imputed data sets
##
## Rubin's (1987) rules were used to pool point and SE estimates across 5 imputed data sets, and to cal
##
## Model Test User Model:
##
##   Test statistic                              165.643
##   Degrees of freedom                               58
##   P-value                                       0.000
##
## Model Test Baseline Model:
##
##   Test statistic                             1222.707
##   Degrees of freedom                              120
##   P-value                                       0.000
##
## User Model versus Baseline Model:
##
##   Comparative Fit Index (CFI)                   0.902
##   Tucker-Lewis Index (TLI)                      0.798
##
## Root Mean Square Error of Approximation:
##
##   RMSEA                                         0.066
##   90 Percent confidence interval - lower        0.054
##   90 Percent confidence interval - upper        0.077
##   P-value RMSEA <= 0.05                         0.015
```

```
##
## Standardized Root Mean Square Residual:
##
##   SRMR                                           0.036

##
## Parameter Estimates:
##
##   Standard errors                             Standard
##   Information                                 Expected
##   Information saturated (h1) model          Structured
##
## Latent Variables:
##                    Estimate  Std.Err  t-value       df  P(>|t|)   Std.lv
##   disrupt =~
##     psyeffect         1.000                                         0.680
##     sched             1.071    0.103   10.414      Inf    0.000    0.728
##     insec             1.027    0.114    9.016      Inf    0.000    0.698
##   sreg =~
##     init              1.000                                         0.792
##     inhib             0.424    0.043    9.757      Inf    0.000    0.336
##     cont              0.616    0.045   13.719      Inf    0.000    0.488
##     bfi               0.781    0.045   17.287      Inf    0.000    0.619
##   Std.all
##
##     0.724
##     0.749
##     0.540
##
##     0.818
##     0.483
##     0.654
##     0.879
##
## Regressions:
##                    Estimate  Std.Err  t-value       df  P(>|t|)   Std.lv
##   sl ~
##     disrupt          -0.318    0.073   -4.376      Inf    0.000   -0.216
##     sreg             -0.027    0.056   -0.475      Inf    0.635   -0.021
##     prehabit         -0.027    0.060   -0.449      Inf    0.654   -0.027
##     age              -0.001    0.003   -0.322 4596.669    0.747   -0.001
##     sex              -0.016    0.078   -0.209      Inf    0.834   -0.016
##     white            -0.062    0.082   -0.755 6207.121    0.450   -0.062
##     child             0.137    0.082    1.671      Inf    0.095    0.137
##   exer ~
##     disrupt          -0.319    0.077   -4.134      Inf    0.000   -0.217
##     sreg              0.027    0.060    0.460      Inf    0.646    0.022
##     prehabit          0.055    0.064    0.854      Inf    0.393    0.055
##     age              -0.000    0.004   -0.083      Inf    0.934   -0.000
##     sex              -0.001    0.084   -0.008      Inf    0.993   -0.001
##     white            -0.126    0.088   -1.435 6628.007    0.151   -0.126
##     child             0.092    0.087    1.051      Inf    0.293    0.092
##   dt ~
##     disrupt          -0.096    0.055   -1.757      Inf    0.079   -0.065
```

```
##     sreg              -0.027    0.043   -0.615      Inf    0.539   -0.021
##     prehabit           0.023    0.047    0.484      Inf    0.628    0.023
##     age                0.003    0.003    1.088      Inf    0.276    0.003
##     sex               -0.023    0.061   -0.382      Inf    0.703   -0.023
##     white             -0.009    0.064   -0.144      Inf    0.885   -0.009
##     child              0.094    0.064    1.474      Inf    0.140    0.094
##   subs ~
##     disrupt           -0.267    0.069   -3.891      Inf    0.000   -0.181
##     sreg               0.018    0.053    0.336      Inf    0.737    0.014
##     prehabit           0.079    0.057    1.378      Inf    0.168    0.079
##     age                0.004    0.003    1.066      Inf    0.287    0.004
##     sex               -0.019    0.075   -0.258      Inf    0.796   -0.019
##     white              0.164    0.078    2.105      Inf    0.035    0.164
##     child              0.029    0.078    0.373      Inf    0.710    0.029
##   Std.all
##
##    -0.269
##    -0.026
##    -0.023
##    -0.016
##    -0.010
##    -0.036
##     0.084
##
##    -0.254
##     0.025
##     0.043
##    -0.004
##    -0.000
##    -0.069
##     0.053
##
##    -0.107
##    -0.035
##     0.025
##     0.055
##    -0.019
##    -0.007
##     0.075
##
##    -0.236
##     0.018
##     0.069
##     0.053
##    -0.012
##     0.101
##     0.019
##
## Covariances:
##                    Estimate  Std.Err  t-value       df  P(>|t|)   Std.lv
##   disrupt ~~
##     sreg              -0.072    0.033   -2.173      Inf    0.030   -0.133
##   sreg ~~
##     prehabit           0.120    0.029    4.191      Inf    0.000    0.151
```

```
##     age                1.703    0.483    3.530      Inf    0.000    2.149
##     sex               -0.046    0.020   -2.282      Inf    0.022   -0.059
##     white              0.024    0.019    1.242      Inf    0.214    0.030
##     child              0.057    0.020    2.777      Inf    0.005    0.072
##   disrupt ~~
##     prehabit           0.075    0.026    2.846      Inf    0.004    0.110
##     age               -0.682    0.441   -1.545      Inf    0.122   -1.003
##     sex               -0.021    0.019   -1.096      Inf    0.273   -0.030
##     white             -0.029    0.018   -1.611      Inf    0.107   -0.043
##     child              0.058    0.019    3.009      Inf    0.003    0.085
##   prehabit ~~
##     age                0.999    0.375    2.663 9896.618    0.008    0.999
##     sex               -0.024    0.016   -1.516      Inf    0.129   -0.024
##     white             -0.020    0.015   -1.338      Inf    0.181   -0.020
##     child              0.024    0.016    1.493      Inf    0.135    0.024
##   age ~~
##     sex               -0.355    0.272   -1.301      Inf    0.193   -0.355
##     white              0.727    0.263    2.766 6906.285    0.006    0.727
##     child              1.061    0.277    3.835      Inf    0.000    1.061
##   sex ~~
##     white             -0.019    0.011   -1.709      Inf    0.087   -0.019
##     child             -0.001    0.012   -0.059      Inf    0.953   -0.001
##   white ~~
##     child             -0.012    0.011   -1.035 9084.367    0.301   -0.012
##  .sl ~~
##    .exer               0.017    0.032    0.549      Inf    0.583    0.017
##    .dt                -0.010    0.023   -0.440      Inf    0.660   -0.010
##    .subs               0.059    0.028    2.061      Inf    0.039    0.059
##  .exer ~~
##    .dt                 0.011    0.025    0.430      Inf    0.667    0.011
##    .subs              -0.033    0.030   -1.108      Inf    0.268   -0.033
##  .dt ~~
##    .subs               0.010    0.022    0.478      Inf    0.632    0.010
##   Std.all
##
##    -0.133
##
##     0.224
##     0.187
##    -0.119
##     0.065
##     0.146
##
##     0.163
##    -0.087
##    -0.062
##    -0.091
##     0.173
##
##     0.129
##    -0.073
##    -0.065
##     0.072
##
```

```
##    -0.063
##     0.135
##     0.188
##
##    -0.083
##    -0.003
##
##    -0.050
##
##     0.027
##    -0.022
##     0.103
##
##     0.021
##    -0.055
##
##     0.023
##
## Intercepts:
##                    Estimate  Std.Err  t-value       df  P(>|t|)   Std.lv
##    .psyeffect          3.251    0.045   71.789      Inf    0.000    3.251
##    .sched              3.641    0.047   77.706      Inf    0.000    3.641
##    .insec              2.663    0.062   42.711      Inf    0.000    2.663
##    .init               3.270    0.047   69.996      Inf    0.000    3.270
##    .inhib              3.826    0.034  113.993      Inf    0.000    3.826
##    .cont               3.448    0.036   95.875      Inf    0.000    3.448
##    .bfi                3.861    0.034  113.742      Inf    0.000    3.861
##    .sl                 2.069    0.273    7.570      Inf    0.000    2.069
##    .exer               1.719    0.291    5.898      Inf    0.000    1.719
##    .dt                 1.732    0.212    8.165      Inf    0.000    1.732
##    .subs               1.723    0.260    6.637      Inf    0.000    1.723
##     prehabit           4.056    0.032  125.202      Inf    0.000    4.056
##     age               34.661    0.554   62.603      Inf    0.000   34.661
##     sex                0.406    0.024   17.148      Inf    0.000    0.406
##     white              0.669    0.023   29.455      Inf    0.000    0.669
##     child              0.406    0.024   17.165      Inf    0.000    0.406
##     disrupt            0.000                                        0.000
##     sreg               0.000                                        0.000
##   Std.all
##     3.461
##     3.747
##     2.059
##     3.375
##     5.496
##     4.623
##     5.484
##     2.576
##     2.009
##     2.821
##     2.243
##     6.036
##     3.018
##     0.827
##     1.420
```

```
##     0.828
##     0.000
##     0.000
##
## Variances:
##                    Estimate  Std.Err  t-value       df  P(>|t|)   Std.lv
##    .psyeffect         0.420    0.048    8.772      Inf    0.000    0.420
##    .sched             0.414    0.052    7.966      Inf    0.000    0.414
##    .insec             1.185    0.094   12.647      Inf    0.000    1.185
##    .init              0.311    0.034    9.025      Inf    0.000    0.311
##    .inhib             0.371    0.027   13.973      Inf    0.000    0.371
##    .cont              0.318    0.025   12.889      Inf    0.000    0.318
##    .bfi               0.113    0.018    6.251      Inf    0.000    0.113
##    .sl                0.599    0.042   14.262      Inf    0.000    0.599
##    .exer              0.683    0.048   14.309      Inf    0.000    0.683
##    .dt                0.370    0.025   14.603      Inf    0.000    0.370
##    .subs              0.544    0.038   14.352      Inf    0.000    0.544
##     prehabit          0.452    0.031   14.666      Inf    0.000    0.452
##     age             131.870    8.992   14.666 8161.780    0.000  131.870
##     sex               0.241    0.016   14.666      Inf    0.000    0.241
##     white             0.222    0.015   14.666      Inf    0.000    0.222
##     child             0.241    0.016   14.666      Inf    0.000    0.241
##     disrupt           0.462    0.065    7.075      Inf    0.000    1.000
##     sreg              0.628    0.066    9.482      Inf    0.000    1.000
##   Std.all
##     0.476
##     0.439
##     0.708
##     0.331
##     0.767
##     0.572
##     0.227
##     0.929
##     0.934
##     0.981
##     0.922
##     1.000
##     1.000
##     1.000
##     1.000
##     1.000
##     1.000
##     1.000
```