Missing data simulation

```
#Download and load the necessary packages
install.packages("mice")
install.packages("tidyverse")
install.packages("lavaan")

library(mice)
library(tidyverse)
library(lavaan)
```

#Under this URL, you find a presenation of the ampute() function within MICE that allows #creating missing data

```
https://rdrr.io/cran/mice/man/ampute.html

#Data creating (similar to your data set, N=8000)
set.seed(123)
e1 = rnorm(8000) #Three random error variables are created
e2 = rnorm(8000)
e3 = rnorm(8000)
X = e1 #
M = .5*X + e2 #Effecs are .5
Y = .5*M + e3
data = tibble(X,M,Y) #Creating a data set
```

#SEM with complete data set

```
mod.full <- '
  Y~M
  M~X'

summary(sem(mod.full, data=data))
```

```
  Number of observations                            8000

  Estimator                                           ML
  Model Fit Test Statistic                         0.239
  Degrees of freedom                                   1
  P-value (Chi-square)                             0.625

(...)

Regressions:
                   Estimate  Std.Err  z-value  P(>|z|)
  Y ~
    M                 0.484    0.010   47.976    0.000
  M ~
    X                 0.498    0.011   44.269    0.000
```

#You see, the effects are (of course) almost exactly as created (within the margin of sampling error)

#Missing data generation: I delete **60%** in all of the three variables!

```
results = ampute(data, prop=.6, mech="MAR")
```

#Extraction of the data set with missing data

```
data.ms <- results[11]$amp
data.ms <- tbl_df(data.ms)
```

#Rename variables (has to be done, as ampute eliminates them)
```
data.ms <- data.ms %>%
  rename(Xmis=X, Ymis=Y, Mmis=M)
```

#Show a part of the dataset with missings
```
data.ms %>% print(n=30)
```

```
# A tibble: 8,000 x 3
      Xmis     Mmis    Ymis
     <dbl>    <dbl>   <dbl>
 1  -0.560   -1.15    -0.315
 2  NA       -0.672    0.581
 3   1.56     1.76     0.159
 4   0.0705  -0.273   -0.945
 5   0.129    0.198   NA
 6  NA        0.988    2.75
 7   0.461    1.08    -1.84
 8  -1.27    -0.141   -0.525
 9  -0.687   -0.250   -0.185
10  NA       -0.276    0.723
11  NA       -0.496   -2.03
12   0.360    1.95    NA
13   0.401   -0.216   -0.478
14   0.111   NA        0.179
15  NA        0.911   -0.594
16   1.79     0.712    0.734
17   0.498   -0.719   NA
18  -1.97    -0.353   NA
19  NA       -0.0600   0.364
20  -0.473    0.250   -1.57
21  -1.07    -2.66    -2.79
22  -0.218   NA        2.20
23  -1.03    -0.526   -0.175
24  -0.729   -1.03     1.22
25  -0.625   NA        1.26
26  NA       -0.271    0.164
27  NA       -0.544   -0.444
28  NA       -0.361    0.952
29  -1.14    NA        0.132
30   1.25     2.26    NA
# ... with 7,970 more rows
```

#SEM with listwise deletion
```
mod.LD <- '
  Ymis ~ Mmis
  Mmis ~ Xmis
 '

summary(sem(mod.LD, data=data.ms))
```

```
                                                Used       Total
  Number of observations                        3237        8000

  Estimator                                       ML
  Model Fit Test Statistic                     0.119
  Degrees of freedom                               1
  P-value (Chi-square)                         0.730

Parameter Estimates:
```

```
  Information                                   Expected
  Information saturated (h1) model            Structured
  Standard Errors                               Standard

Regressions:
                   Estimate  Std.Err  z-value  P(>|z|)
  Ymis ~
    Mmis              0.439    0.016   26.818    0.000
  Mmis ~
    Xmis              0.426    0.018   23.911    0.000
```

#You see, the used data is N=3237, effects are downward biased. One could create a systematically affected data set where missingness of the DV is affected my missing common causes and the bias would be larger

#SEM with FIML
#The model structure is the same)

```
summary(sem(mod.LD, data=data.ms, missing="FIML", fixed.x=FALSE))
```

```
  Estimator                                           ML
  Model Fit Test Statistic                         2.005
  Degrees of freedom                                   1
  P-value (Chi-square)                             0.157

Parameter Estimates:

  Information                                   Observed
  Observed information based on                  Hessian
  Standard Errors                               Standard

Regressions:
                   Estimate  Std.Err  z-value  P(>|z|)
  Ymis ~
    Mmis              0.490    0.012   40.134    0.000
  Mmis ~
    Xmis              0.497    0.013   37.283    0.000

Intercepts:
                   Estimate  Std.Err  z-value  P(>|z|)
   .Ymis              0.005    0.013    0.404    0.686
   .Mmis              0.013    0.013    1.013    0.311
    Xmis              0.006    0.012    0.503    0.615
```

#You see, the effects are exactly as in the model with the full data set—although only 40% is available!