Imputing missing data

MEVM04

Johan Larsson

November 23, 2015

1 Introduction

Missing values are common in clinical research and can occur for many reasons. Crudely, they can be separated into missing values on the predictor(s) (x, also known as *independent variables*) and on the outcome(s) (y, also known as *dependent variables*).

In the first case, missing data might occur because a participant did not wish to disclose information on their weight (here, a predictor); in the second case, observations might be missing because the blood pressure gauge had malfunctioned when the participant's blood pressure was measured. Whether a value is missing on x or y affect our ability to 1) do something about it, and 2) the consequences for our results. In this text, we will deal with each instance of missing values in turn.

A common choice when data is missing is *case deletion*: deleting each observation for which a value is missing, which is also known as doing a complete cases analysis. Many common statistical models, analysis of variance for instance, requires complete cases to work properly, whilst some, such as tree-based models, do not [1]; such models, however, are beyond the scope of this text. Whilst it is frequent, case deletion is generally bad practice since it can lead to unpredictable bias [2] and inefficiency [3].

A better alternative to case deletion is *imputation*, which is when you fill in (impute) missing values based on information given by other variables. In its simplest form, missing data on a person's weight might be substituted with the mean weight of the entire group (though as we shall see there are much better options available).

This text is an assignment for a class in my master's programme: *MEVM04*, wherein I have attempted to outline the basics of missing data with a focus on how to handle missing valued on predictor variables using imputation. The text is based heavily on books by Steyerberg [4] and Harrell [5]; as a final step, I have provided a worked-through example given in the latter book.

As a secondary objective, this text is an attempt of reproducible research using *knitR* for *R* (R Core Team). All of the files used to produce this text (including plots and statistical analyses) will therefore be provided publicly on Github at https://github.com/Hjalt/mevm04imput.

2 Types of missing data

Apart from whether predictor or outcome variables are missing, there are three types of missing data:

MCAR Missing Completely At Random

MAR Missing At Random

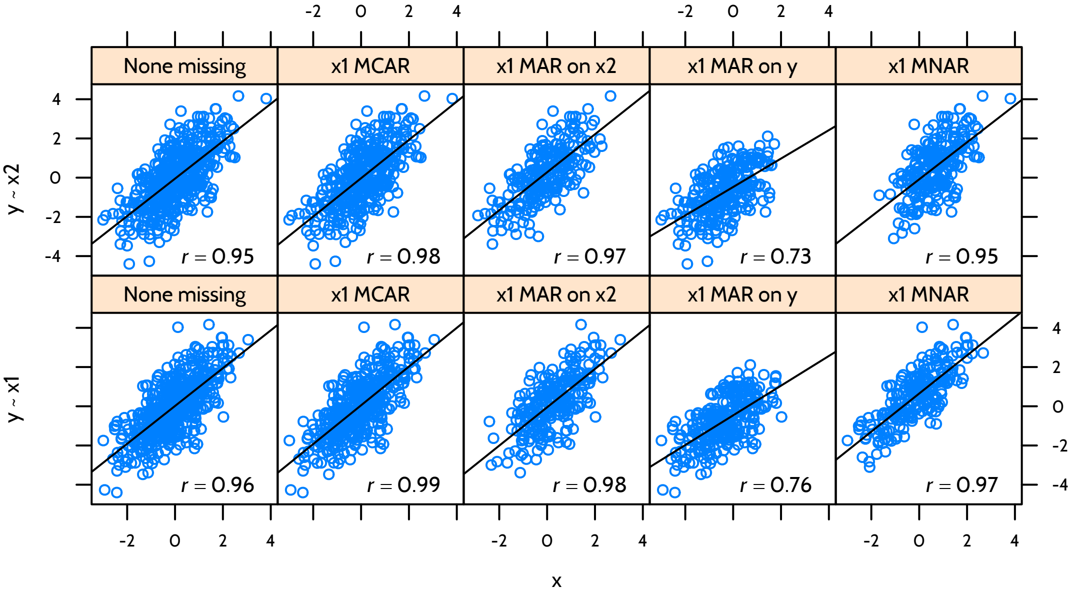


Figure 1: Different types of missing data and their effects on the correlation coefficient. $x1$ and $x2$ are normally distributed. The true correlation coefficients (r) are 1. Only when data is MAR on y is the correlation coefficient noticeably erroneous. This example has been adapted from Steyerberg [4]

MNAR Missing Not At Random

MCAR means that there is no pattern to why the data is missing; perhaps the scale had malfunctioned and weight data had to be discarded. When data is MAR there is conversely a pattern to its absence, although that pattern is apparent from the data – younger people might for example have refused to be weighed more frequently (and we recorded age). Lastly, when data is MNAR there is a pattern that cannot be deciphered from the available data. Perhaps people of lower income did not want to step up on the scale but we did not measure income. MNAR is a particularly problematic form since it is by definition impossible to understand if and how it might be affecting the results.

3 Missing data on independent variables

The situation where data is missing on predictors deserves most attention as it is both common and manageable, whilst being detrimental to our results (if not taken care of).

In Figure 1 various scenarios of missing predictor values have been simulated from a simple linear model

$$y = x1 + x2 + \text{error}$$

Here, $x1$, $x2$, and the error term are generated from a normal distribution with a mean of 0, and a standard deviation of 1, 1, and 0.1 respectively. Thus, the true slope of the relationship is $r = 1$, which can be seen from the first panel. This example has been adapted and revised from the code excerpts from Steyerberg [4].

When data is MCAR on $x1$, the foremost consequence is that we lose statistical power. If for instance every other participant had one variable missing out of 5, this would mean that half of the sample was dropped from the analysis, even though only 10% of the data was missing. As we see on the slope, however, data that is MCAR does not actually affect the estimate, simply because it is random and not systematic missingness.

In the third panel, $x1$ is missing for lower levels of $x2$, which is a case of MAR. Here, too, is the slope unbiased (but would not be if the relationship was different for lower, compared with higher, values of $x2$.

In the fourth panel, $x1$ is missing for higher levels of y – also a case of MAR. We can see here that the slope estimate is biased low ($r = 0.73$ for $x1$ and $r = 0.76$ for $x2$).

Finally, in the fifth panel, $x1$ is missing for higher values of $x1$ – a pattern that cannot be deduced from the data. This is an instance of MNAR. The slope is unbiased in this instance, which is related to missingness of $x1$ being related to $x1$ itself. If there was a third variable in the mix, to which $x1$ owed its missingness, the results would be different.

4 Single imputation

4.1 Simple mean imputation

A simple, but flawed, option is to replace missing values with the mean of the variable, which is suboptimal because it fails to take advantage of patterns in the data, i.e., correlations between variables that otherwise can improve the accuracy of imputations. Moreover, it underestimates the variance (the randomness) of the estimates, which leads to overconfidence in the estimates.

Better options exist in *conditional mean imputation*, *stochastic regression imputation*, and *multiple imputation*.

4.2 Conditional mean imputation

In conditional mean imputation (or regression imputation), missing data on a variable are imputed using the patters of how the variable correlates with other variables. As with simple mean imputation, however, conditional mean imputation underestimates the variance of the imputations; if, for example, we were to impute missing values in Figure 1 with conditional mean imputation, all those values would lie along the imputation slope and this would not reflect the true nature of variance.

4.3 Stochastic regression imputation

Stochastic regression imputation alleviates the issue of underestimated variance by randomly sampling from the distribution of predicted values, that is, the distribution that in conditional mean imputation would have been used to compute the mean, whereby it simulates the random error that is inherent to all real measurements.

5 Multiple imputation

Though single imputation techniques are efficient and sometimes adequate, they present one drawback in that the randomness inherent to the imputation step itself is not explicitly covered in the analysis. The simplest way of looking at this is to consider that only after we are done imputing do we conduct

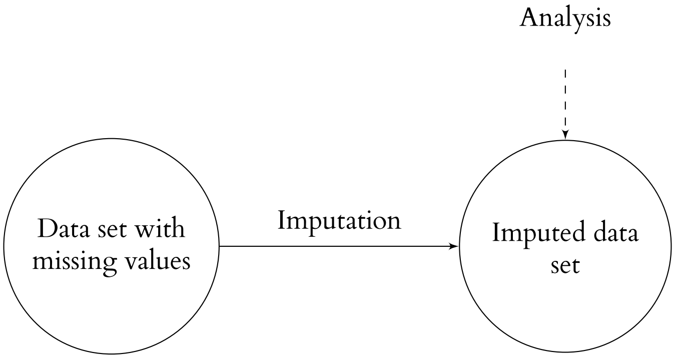


Figure 2: Flow chart of single imputation models.

our analysis 2, which means that any bias we introduce with our imputation model is disregarded – this poses a serious problem for the external validity of our model.

This issue is solved with *multiple imputation*, a statistical technique that was designed by Rubin [6] as a flexible method for dealing with missing data. In multiple imputation, several data sets are produced by random sampling from the original data set. In each of these sets an imputation model is fit (as in conditional mean imputation) and a estimate is established as an aggregate of all of these data sets. Because we are then including the variance of imputation in the aggregate, we directly take bias from the imputation step into account.

5.1 Chained equations

If data is missing on several variables, it might be necessary to impute in steps, which is known as chained equations multiple imputation. It works by first imputing data on one variable, then creating a new sample from the original data plus the imputations, and thereafter repeating the imputation step. This iterative process can proceed for as long as desired, but most software use a stopping-rule to drop out of the loop at some preestablished point.

6 Missing data on dependent variable

Observations with missing data on the dependent variable (outcome) are commonly excluded from the analysis. And if they are MCAR, there is an argument to do so since no additional information can be gathered from the remaining data if the variable of interest is missing [5].

Yet, when data on Y is MAR, bias might occur. Harrell recommends that researchers should as a minimum characterize any patterns of missingness, and simulation studies show that there are benefits to imputation also on y [7].

7 Issues with imputation

In imputation we assume that data is MAR, i.e. we are able to figure out from the rest of the variables why data is missing. However, this assumption is by definition not testable [8] (but becomes increasingly tenable as more variables are measured).

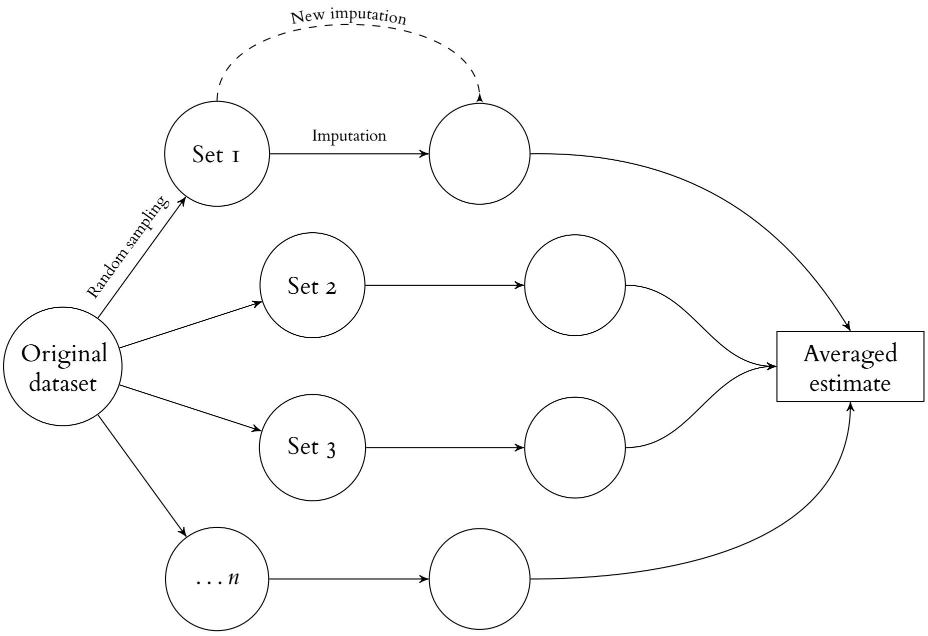


Figure 3: Flow chart of multiple imputation model with chained equations. The entire situation is only visualized for *Set 1*.

If the assumption does not hold and data is instead MNAR then imputation can only lead to increased bias, and it would be advisable to stick to case deletion. However, as many empirical as well as simulation experiments have shown, imputation generally leads to less bias and more efficiency with relatively little bias.

8 A worked-through example

8.1 Explore the variables and patterns of missing data in the SUPPORT dataset.

8.1.1 Print univariate summaries of all variables. Make a plot (showing all variables on one page) that describes especially the continuous variables.

We start with some descriptive variables

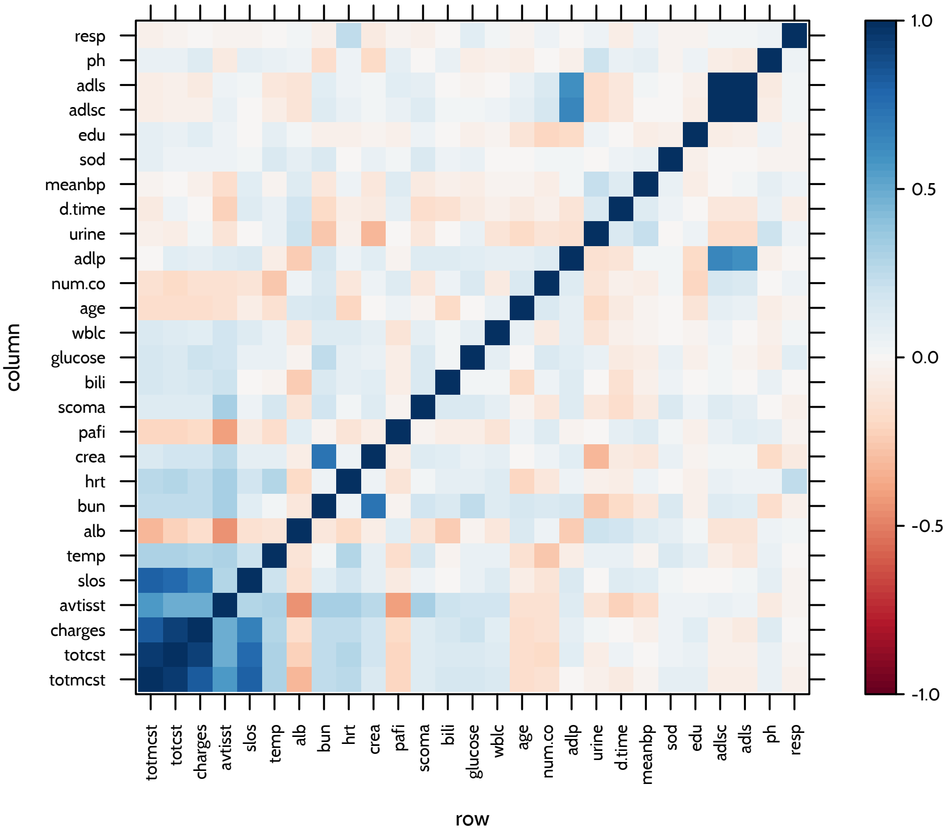
```
library(Hmisc)

# Load the SUPPORT data set.
getHdata(support)

# Total proportion of missing values
sum(sapply(support, is.na))/length(sapply(support, is.na))

## [1] 0.1375429

# Print some summary statistics of every variable
describe(support)
```

```
## support
##
##  35  Variables      1000  Observations
## ------------------------------------------------------------------
## age : Age
##        n missing  unique    Info    Mean     .05     .10     .25
##     1000       0     970       1   62.47   33.76   38.91   51.81
##      .50     .75     .90     .95
##    64.90   74.50   81.87   86.00
##
## lowest :  18.04  18.41  19.76  20.30  20.31
## highest:  95.51  96.02  96.71 100.13 101.85
## ------------------------------------------------------------------
## death : Death at any time up to NDI date:31DEC94
##        n missing  unique    Info     Sum    Mean
##     1000       0       2    0.67     668   0.668
## ------------------------------------------------------------------
## sex
##        n missing  unique
##     1000       0       2
##
## female (438, 44%), male (562, 56%)
## ------------------------------------------------------------------
## hospdead : Death in Hospital
##        n missing  unique    Info     Sum    Mean
##     1000       0       2    0.57     253   0.253
## ------------------------------------------------------------------
## slos : Days from Study Entry to Discharge
##        n missing  unique    Info    Mean     .05     .10     .25
##     1000       0      88       1   17.86       4       4       6
##      .50     .75     .90     .95
##       11      20      37      53
##
## lowest :   3   4   5   6   7, highest: 145 164 202 236 241
## ------------------------------------------------------------------
## d.time : Days of Follow-Up
##        n missing  unique    Info    Mean     .05     .10     .25
##     1000       0     582       1   475.7     5.0     8.0    27.0
##      .50     .75     .90     .95
##    256.5   725.0  1464.3  1757.1
##
## lowest :    3    4    5    6    7
## highest: 2006 2011 2022 2024 2029
## ------------------------------------------------------------------
## dzgroup
##        n missing  unique
##    1000       0       8
##
##           ARF/MOSF w/Sepsis COPD CHF Cirrhosis Coma Colon Cancer
## Frequency               391  116 143         55   60            49
## %                        39   12  14          6    6             5
##           Lung Cancer MOSF w/Malig
## Frequency         100           86
## %                  10            9
## -----------------------------------------------------------------
## dzclass
##       n missing  unique
##    1000       0       4
##
## ARF/MOSF (477, 48%)
## COPD/CHF/Cirrhosis (314, 31%)
## Coma (60, 6%), Cancer (149, 15%)
## -----------------------------------------------------------------
## num.co : number of comorbidities
##       n missing  unique    Info    Mean
##    1000       0       8    0.94   1.886
##
##             0   1   2   3  4  5  6 7
## Frequency 120 337 269 151 76 29 17 1
## %          12  34  27  15  8  3  2 0
## -----------------------------------------------------------------
## edu : Years of Education
##       n missing  unique    Info    Mean     .05     .10     .25
##     798     202      25    0.97   11.78       6       8      10
##     .50     .75     .90     .95
##      12      14      16      18
##
## lowest :  0  1  2  3  4, highest: 20 21 22 24 30
## -----------------------------------------------------------------
## income
##       n missing  unique
##     651     349       4
##
## under $11k (309, 47%), $11-$25k (161, 25%)
## $25-$50k (106, 16%), >$50k (75, 12%)
## -----------------------------------------------------------------
## scoma : SUPPORT Coma Score based on Glasgow D3
##       n missing  unique    Info    Mean     .05     .10     .25
##    1000       0      11    0.65   11.74     0.0     0.0     0.0
##     .50     .75     .90     .95
##     0.0     9.0    44.0    62.4
##
```

```
##             0  9 26 37 41 44 55 61 89 94 100
## Frequency 704 83 58 24 19 45 11  6  8  7  35
## %          70  8  6  2  2  4  1  1  1  1   4
## ------------------------------------------------------------------
## charges : Hospital Charges
##        n missing  unique    Info    Mean     .05     .10     .25
##      975      25     967       1   56271    3757    4688   10029
##      .50     .75     .90     .95
##    26499   63622  147109  223582
##
## lowest :   1636   1680   1830   2045   2082
## highest: 504660 538323 543761 706577 740010
## ------------------------------------------------------------------
## totcst : Total RCC cost
##        n missing  unique    Info    Mean     .05     .10     .25
##      895     105     895       1   30490    2484    3081    5899
##      .50     .75     .90     .95
##    15110   37598   72906  114932
##
## lowest :      0   1162   1201   1285   1396
## highest: 269131 299966 338955 357919 390461
## ------------------------------------------------------------------
## totmcst : Total micro-cost
##        n missing  unique    Info    Mean     .05     .10     .25
##      628     372     617       1   26168    1653    2548    5297
##      .50     .75     .90     .95
##    13828   33691   66229   96753
##
## lowest :      0.0    829.6    834.6    914.4   1016.4
## highest: 154709.0 198047.0 234875.9 246904.0 271467.2
## ------------------------------------------------------------------
## avtisst : Average TISS, Days 3-25
##        n missing  unique    Info    Mean     .05     .10     .25
##      994       6     241       1   22.64    6.00    8.00   12.00
##      .50     .75     .90     .95
##    19.00   31.75   43.33   48.00
##
## lowest :  1.667  2.500  3.000  3.500  4.000
## highest: 58.500 59.000 60.000 61.000 64.000
## ------------------------------------------------------------------
## race
##        n missing  unique
##      995       5       5
##
##           white black asian other hispanic
## Frequency   781   157     9    12       36
## %            78    16     1     1        4
## -----------------------------------------------------------------
## meanbp : Mean Arterial Blood Pressure Day 3
##        n missing  unique    Info    Mean     .05     .10     .25
##     1000       0     122       1   84.98   47.00   55.00   64.75
##      .50     .75     .90     .95
##    78.00  107.00  120.00  128.05
##
## lowest :   0  20  27  30  32, highest: 155 158 161 162 180
## -----------------------------------------------------------------
## wblc : White Blood Cell Count Day 3
##        n missing  unique    Info    Mean     .05     .10     .25
##      976      24     282       1    12.4   2.475   4.800   6.899
##      .50     .75     .90     .95
##   10.449  15.500  22.248  27.524
##
## lowest :   0.05000   0.06999   0.09999   0.14999   0.19998
## highest:  60.00000  61.19531  67.09375  79.39062 100.00000
## -----------------------------------------------------------------
## hrt : Heart Rate Day 3
##        n missing  unique    Info    Mean     .05     .10     .25
##     1000       0     124       1   97.87    54.0    60.0    72.0
##      .50     .75     .90     .95
##    100.0   120.0   135.0   146.1
##
## lowest :   0  11  30  35  36, highest: 189 193 199 232 300
## -----------------------------------------------------------------
## resp : Respiration Rate Day 3
##        n missing  unique    Info    Mean     .05     .10     .25
##     1000       0      45    0.99   23.49       9      10      18
##      .50     .75     .90     .95
##       24      29      36      40
##
## lowest :  0  4  6  7  8, highest: 48 49 52 60 64
## -----------------------------------------------------------------
## temp : Temperature (celcius) Day 3
##        n missing  unique    Info    Mean     .05     .10     .25
##     1000       0      64       1   37.08   35.50   35.80   36.20
##      .50     .75     .90     .95
##    36.70   38.09   38.80   39.20
##
## lowest : 32.50 33.70 34.00 34.09 34.40
## highest: 40.20 40.59 40.90 41.00 41.20
## -----------------------------------------------------------------
## pafi : PaO2/(.01*FiO2) Day 3
##        n missing  unique    Info    Mean     .05     .10     .25
##      747     253     463       1   244.2   92.61  115.00  156.33
##      .50     .75     .90     .95
##   226.66  310.00  400.00  442.81
##
## lowest :  34.00  39.00  45.00  48.00  53.33
## highest: 600.00 623.75 640.00 680.00 869.38
## ---------------------------------------------------------------------------
## alb : Serum Albumin Day 3
##        n missing  unique    Info    Mean     .05     .10     .25
##      622     378      38       1   2.917   1.800   2.000   2.400
##      .50     .75     .90     .95
##    2.800   3.400   4.000   4.199
##
## lowest : 1.100 1.200 1.300 1.400 1.500
## highest: 4.500 4.600 4.699 4.800 4.899
## ---------------------------------------------------------------------------
## bili : Bilirubin Day 3
##        n missing  unique    Info    Mean     .05     .10     .25
##      703     297     115       1   2.527  0.3000  0.3000  0.5000
##      .50     .75     .90     .95
##   0.7999  1.7998  5.8594 12.5896
##
## lowest :  0.09999  0.19998  0.29999  0.39996  0.50000
## highest: 32.39844 33.00000 35.00000 39.29688 50.09375
## ---------------------------------------------------------------------------
## crea : Serum creatinine Day 3
##        n missing  unique    Info    Mean     .05     .10     .25
##      997       3      87       1   1.808  0.6000  0.7000  0.8999
##      .50     .75     .90     .95
##   1.2000  1.8999  3.6396  5.5996
##
## lowest :  0.3  0.4  0.5  0.6  0.7
## highest: 10.4 10.6 11.2 11.6 11.8
## ---------------------------------------------------------------------------
## sod : Serum sodium Day 3
##        n missing  unique    Info    Mean     .05     .10     .25
##     1000       0      42       1   137.7     129     131     134
##      .50     .75     .90     .95
##      137     141     145     148
##
## lowest : 118 120 121 122 123, highest: 156 157 158 168 175
## ---------------------------------------------------------------------------
## ph : Serum pH (arterial) Day 3
##        n missing  unique    Info    Mean     .05     .10     .25
##      750     250      53       1   7.416   7.289   7.319   7.380
##      .50     .75     .90     .95
##   7.420   7.470   7.500   7.520
##
## lowest : 6.960 6.989 7.069 7.119 7.130
## highest: 7.590 7.600 7.609 7.659 7.670
## -----------------------------------------------------------------
## glucose : Glucose Day 3
##       n missing  unique    Info    Mean     .05     .10     .25
##     530     470     226       1   156.4    74.0    82.0   100.0
##     .50     .75     .90     .95
##   128.0   185.0   269.3   327.5
##
## lowest :  11  25  30  42  51, highest: 492 512 528 576 598
## -----------------------------------------------------------------
## bun : BUN Day 3
##       n missing  unique    Info    Mean     .05     .10     .25
##     545     455     106       1   32.61     7.0     9.0    14.0
##     .50     .75     .90     .95
##    23.0    43.0    68.6    88.8
##
## lowest :   1   2   3   4   5, highest: 124 125 127 128 146
## -----------------------------------------------------------------
## urine : Urine Output Day 3
##       n missing  unique    Info    Mean     .05     .10     .25
##     483     517     359       1    2194   141.7   600.0  1208.5
##     .50     .75     .90     .95
##  1925.0  2900.0  4087.6  4822.5
##
## lowest :    0    1    5    8   15
## highest: 7275 7360 7455 7560 7750
## -----------------------------------------------------------------
## adlp : ADL Patient Day 3
##       n missing  unique    Info    Mean
##     366     634       8    0.84   1.246
##
##             0   1   2   3   4   5   6 7
## Frequency 194 73 28 20 12 19 16 4
## %          53 20  8  5  3  5  4 1
## -----------------------------------------------------------------
## adls : ADL Surrogate Day 3
##       n missing  unique    Info    Mean
##     690     310       8     0.9   1.755
##
##             0   1  2  3  4  5  6  7
## Frequency 313 126 54 39 42 34 52 30
## %          45  18  8  6  6  5  8  4
## -----------------------------------------------------------------
## sfdm2
##        n missing  unique
##      841     159       5
##
## no(M2 and SIP pres) (326, 39%)
## adl>=4 (>=5 if sur) (111, 13%)
## SIP>=30 (59, 7%), Coma or Intub (7, 1%)
## <2 mo. follow-up (338, 40%)
## ----------------------------------------------------------------
## adlsc : Imputed ADL Calibrated to Surrogate
##        n missing  unique    Info    Mean     .05     .10     .25
##     1000       0     251    0.97    1.98   0.000   0.000   0.000
##      .50     .75     .90     .95
##    1.670   3.042   5.000   6.000
##
## lowest : 0.0000 0.4948 0.4948 1.0000 1.1667
## highest: 5.9932 6.0000 6.3398 6.4658 7.0000
## ----------------------------------------------------------------
```

To make a plot, we first drop the categorical variables from the data set, then we compute a correlation matrix, reorganize it with a factor analysis and then plot it using the facilities of `levelplot` in lattice.

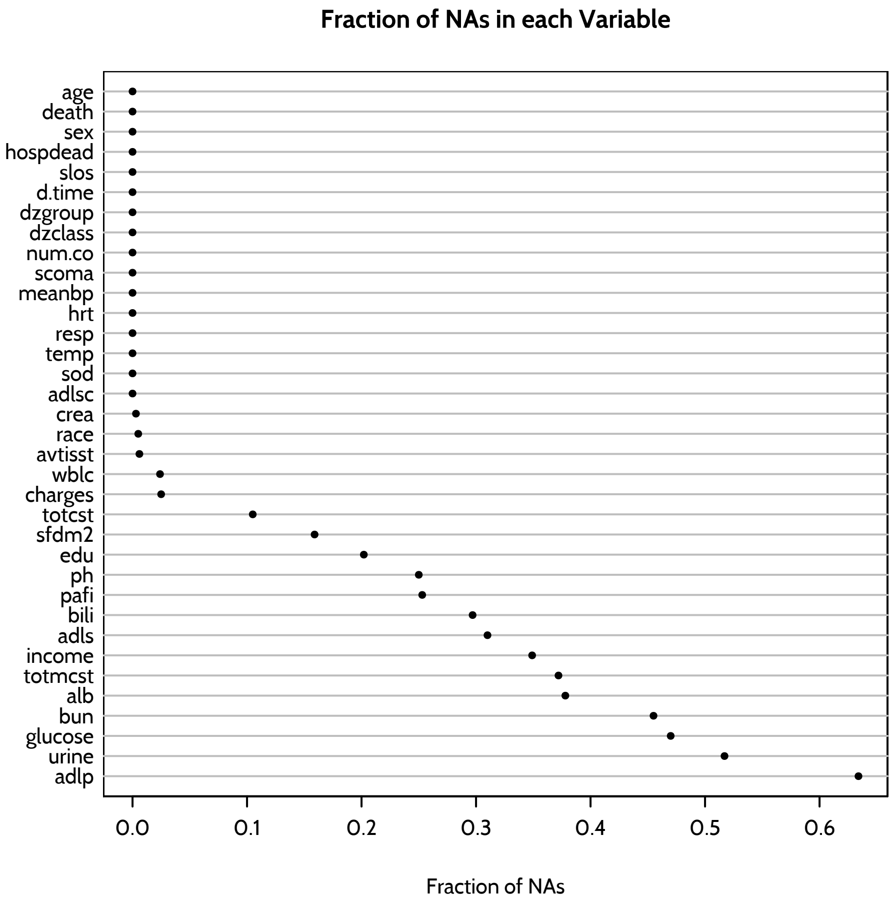
```
# Subset the support data set to remove categorical variables
support.cont <- support[, sapply(support, is.numeric)]
support.cont <- support.cont[, -(2:3)]

# Print a scatterplot matrix
library(psych)
library(lattice)
library(latticeExtra)

m <- mat.sort(as.matrix(cor(support.cont,
                            use = "pairwise.complete.obs")))

levelplot(m,
          scales = list(x = list(rot = 90)),
          at = seq(-1, 1, length.out = 100))
```

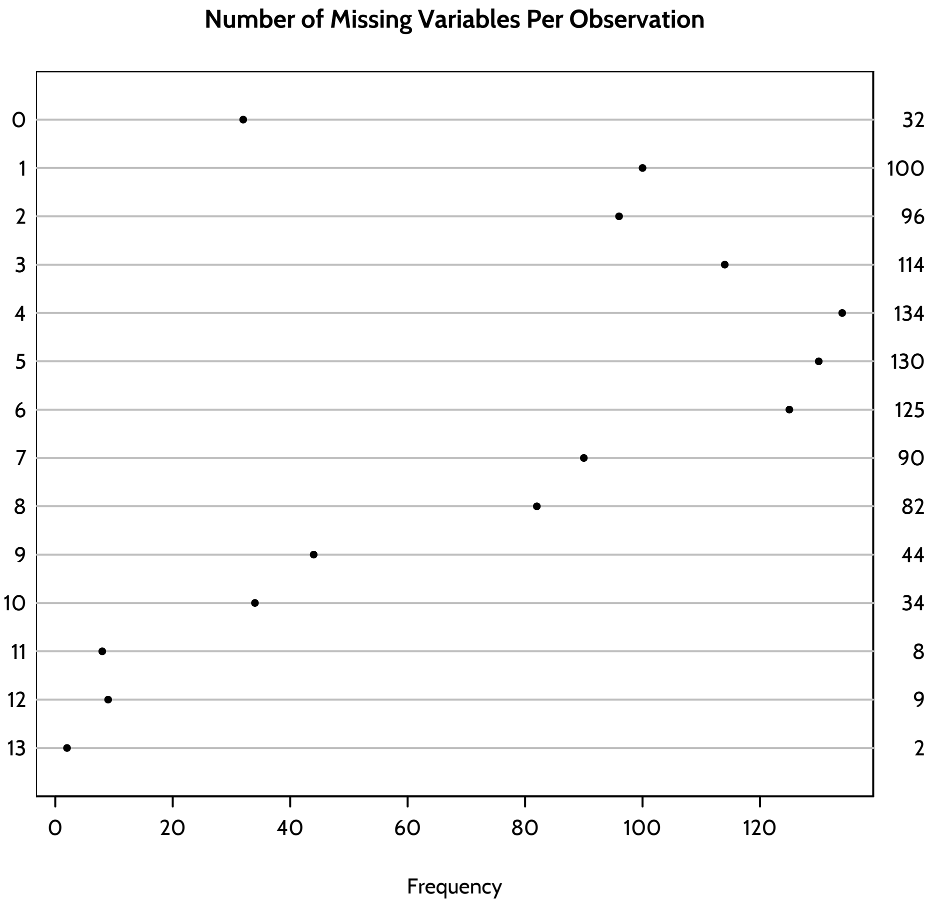


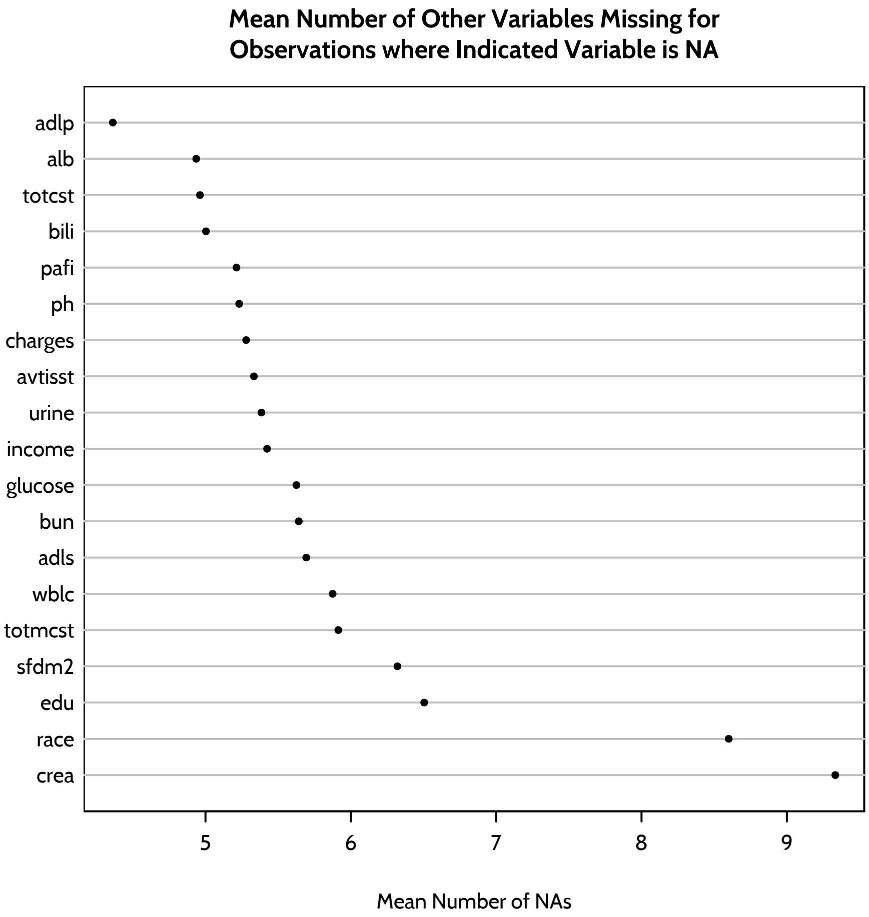
8.2 Make a plot showing the extent of missing data and tendencies for some variables to be missing on the same patients.

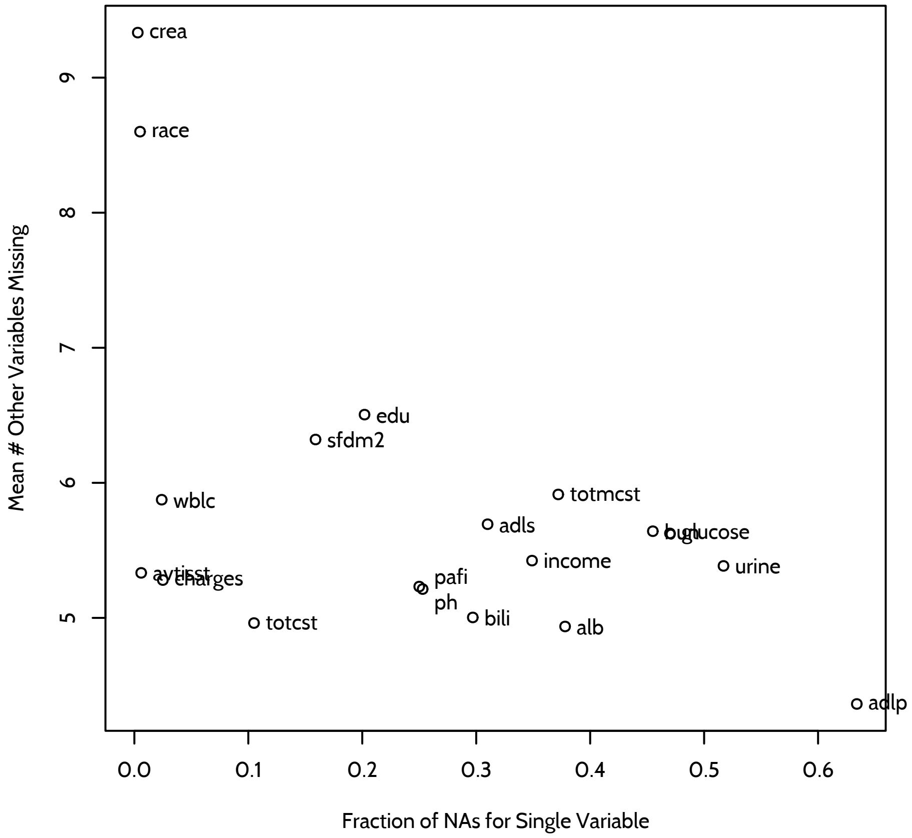
We use Harrel's `Hmisc` package to plot the relationships of missingness between variables.

```
library(Hmisc)
naplot(naclus(support), which = "all")
```


Fraction of NAs in each Variable
age
death
sex
hospdead
slos
d.time
dzgroup
dzclass
num.co
scoma
meanbp
hrt
resp
temp
sod
adlsc
crea
race
avtisst
wblc
charges
totcst
sfdm2
edu
ph
pafi
bili
adls
income
totmcst
alb
bun
glucose
urine
adlp
0.0
0.1
0.2
0.3
0.4
0.5
0.6
Fraction of NAs


Number of Missing Variables Per Observation
0
1
2
3
4
5
6
7
8
9
10
11
12
13
32
100
96
114
134
130
125
90
82
44
34
8
9
2
0
20
40
60
80
100
120
Frequency

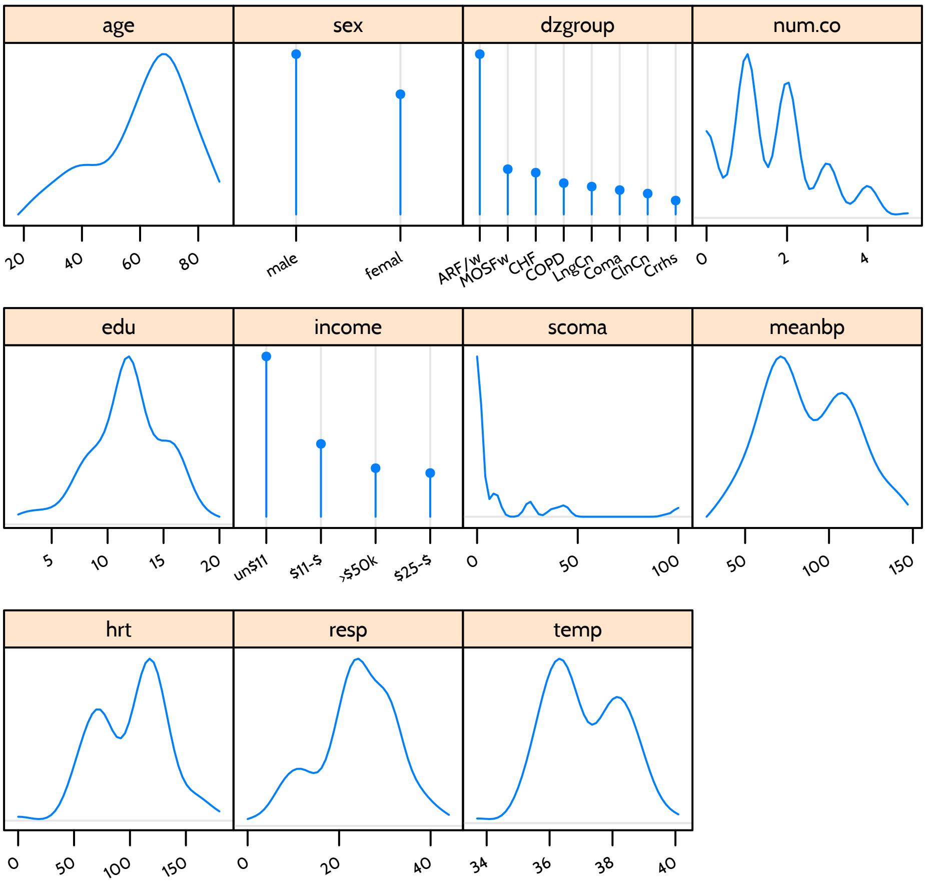

Mean Number of Other Variables Missing for
Observations where Indicated Variable is NA
adlp
alb
totcst
bili
pafi
ph
charges
avtisst
urine
income
glucose
bun
adls
wblc
totmcst
sfdm2
edu
race
crea
5
6
7
8
9
Mean Number of NAs



8.3 Total hospital costs (variable totcst) were estimated from hospitalspecific Medicare cost-to-charge ratios. Characterize what kind of patients have missing totcst. For this characterization use the following patient descriptors: age, sex, dzgroup, num.co, edu, income, scoma, meanbp, hrt, resp, temp.

We can solve this by plotting distributions of the various variables involved.

```
library(dplyr) # For data wrangling
support.na <- support[is.na(support$totcst), ] %>%
 select(age, sex, dzgroup, num.co, edu,
        income, scoma, meanbp, hrt, resp, temp)
trellis.par.set(fontsize = list(text = 8, points = 6))
marginal.plot(support.na)
```



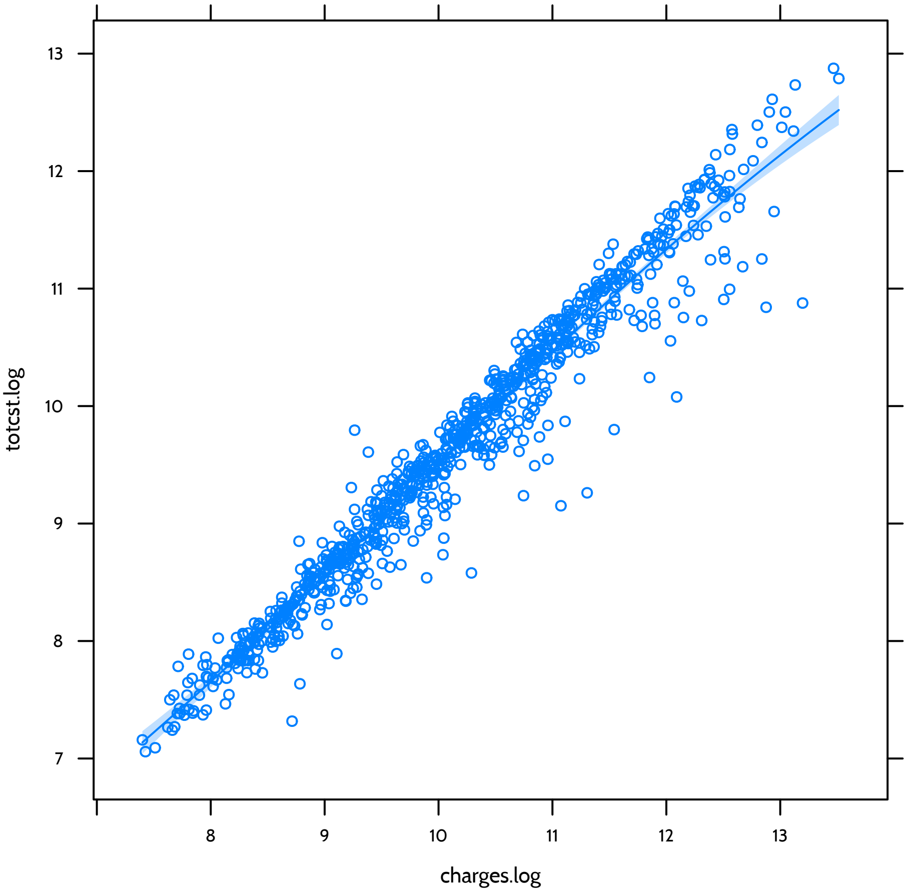
8.4 Prepare for later development of a model to predict costs by developing reliable imputations for missing costs. Remove the observation having zero totcst.

We start by removing values for which the outcome (`totcst`) is missing or zero.

```
support.complete <- subset(support, !is.na(totcst) & totcst > 0)
```

8.4.1 The cost estimates are not available on 105 patients. Total hospital charges (bills) are available on all but 25 patients. Relate these two variables to each other with an eye toward using charges to predict totcst when totcst is missing. Make graphs that will tell whether linear regression or linear regression after taking logs of both variables is better.

```
# As suggested, we use a log transformation
support.complete$totcst.log <- log(support.complete$totcst)
support.complete$charges.log <- log(support.complete$charges)

xplot(totcst.log ~ charges.log, support.complete,
       panel = function(...) {
         panel.xyplot(...)
         panel.smoother(..., span = 0.9)
})
```



8.4.2 Impute missing total hospital costs in SUPPORT based on a regression model relating charges to costs, when charges are available.

We need to specify an imputation model, which means we need to find the slope and intercept of the regression model

```
support.lm <- lm(totcst.log ~ charges.log, support.complete)
i <- coefficients(support.lm)[1] # intercept
b <- coefficients(support.lm)[2] # slope

# Impute values
support.tf <- transform(support,
                        totcst = ifelse(is.na(totcst),
                                  (exp(i + b * log(charges))),
                                  totcst))
```

8.4.3 Compute the likely error in approximating total cost using charges by computing the median absolute difference between predicted and observed total costs in the patients having both variables available. If you used a log transformation, also compute the median absolute percent error in imputing total costs by anti-logging the absolute difference in predicted logs.

We start in reverse with the mmedian absolute percent error:

```
a <- support.complete[complete.cases(support.complete$charges),
                      "totcst.log"]
b <- predict(support.lm)
exp(mad(a - b)) # The exponential of the natural logarithm

## Total RCC cost
## [1] 1.179605
```

Next, the median absolute deviation:

```
a <- support.complete[complete.cases(support.complete$charges),
                      "totcst"]
b <- exp(predict(support.lm))
mad(a - b) # In absolutes

## Total RCC cost
## [1] 2652.82
```

8.5 State briefly why single conditional median imputation is OK here.

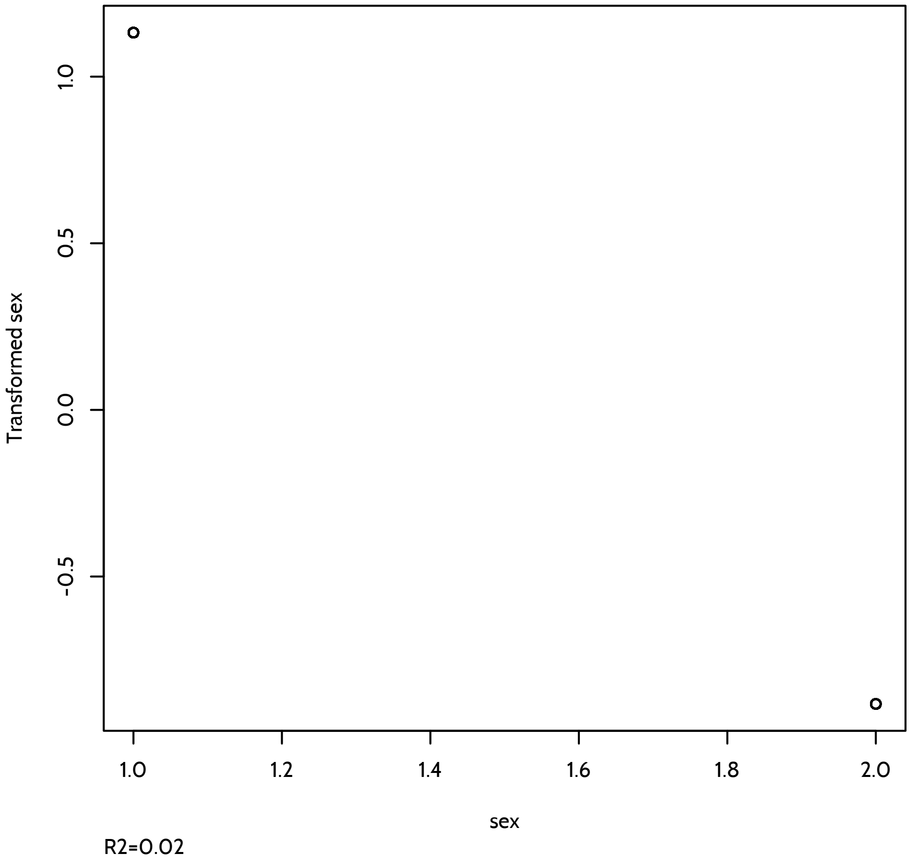
```
lm.cc <- lm(totcst ~ charges, support.complete)
lm.im <- lm(totcst ~ charges, support.tf)
```

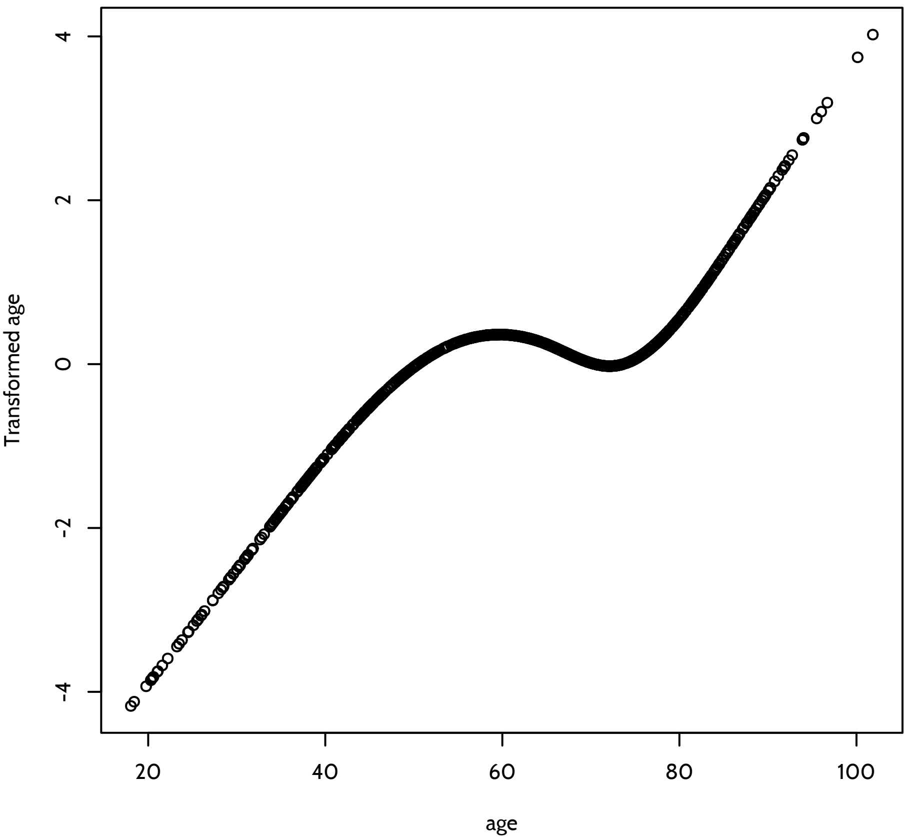
Although we have quite a large proportion of missing values (14%), we are introducing very little bias since our median absolute percent error is only 1.18%. Simply put: because all observed outcomes lie close to the predicted ones, the bias we get from assuming that our missing values are on the slope is neglible.

8.6 Use transcan to develop single imputations for total cost, commenting on the strength of the model fitted by transcan as well as how strongly each variable can be predicted from all the others.

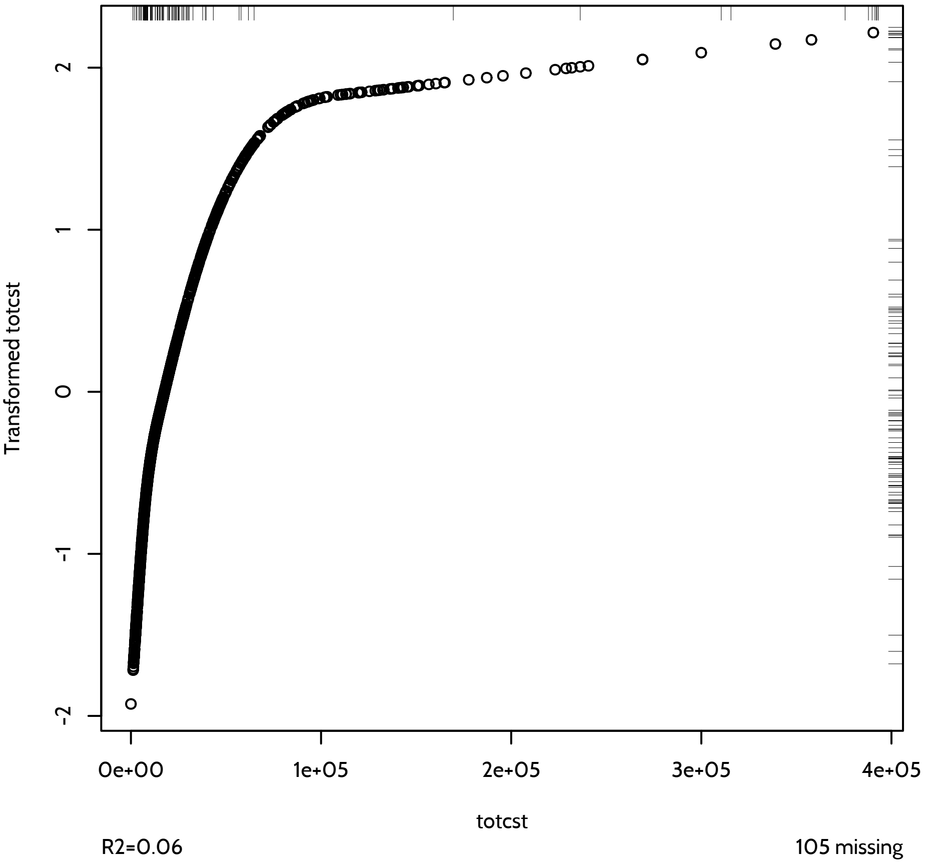
I decided to pick only some of the variables.

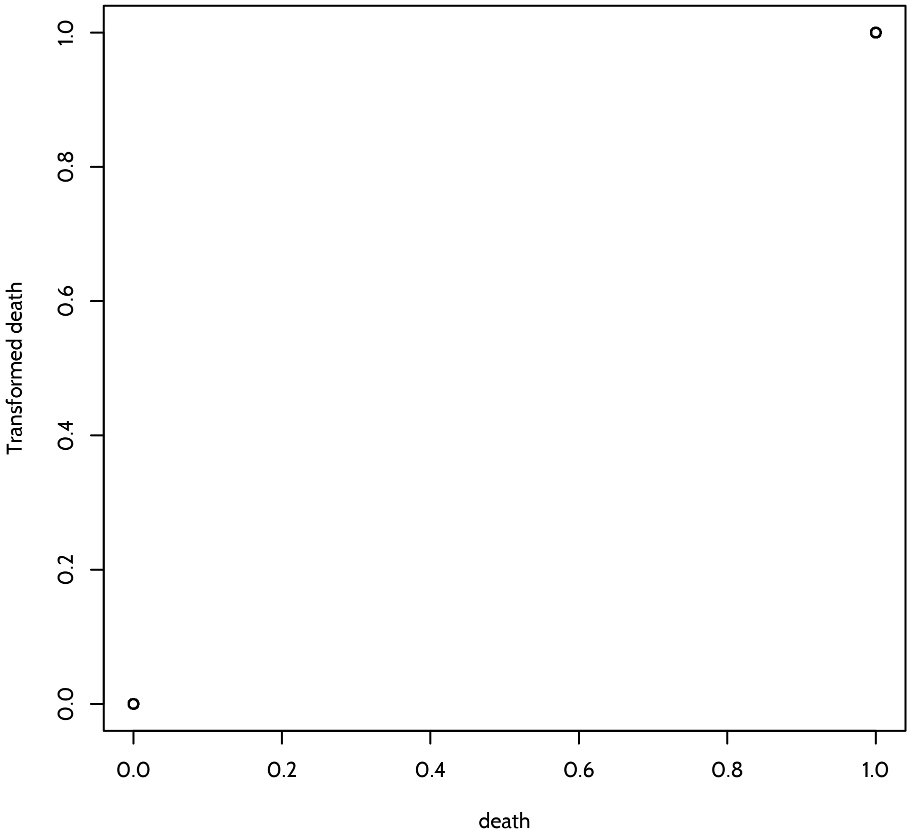
```
support.trans <- transcan(~ sex + age + totcst + death, data = support,
                          imputed = T, transformed=TRUE)
```


Transformed sex
-0.5
0.0
0.5
1.0
1.0
1.2
1.4
1.6
1.8
2.0
sex
R2=0.02



R2=0.11


Transformed totcst
2
1
0
-1
-2
0e+00
1e+05
2e+05
3e+05
4e+05
totcst
R2=0.06
105 missing



R2=0.02

```
summary(support.trans)

## transcan(x = ~sex + age + totcst + death, imputed = T, transformed = TRUE,
##     data = support)
##
## Iterations: 5
##
## R-squared achieved in predicting each variable:
##
##    sex    age totcst  death
##  0.019  0.113  0.058  0.022
##
## Adjusted R-squared:
##
##    sex    age totcst  death
##  0.014  0.106  0.050  0.017
##
## Coefficients of canonical variates for predicting each (row) variable
##
```

```
##         sex   age   totcst death
## sex           0.70  0.83  -0.95
## age     0.26        -0.87   0.84
## totcst  0.37 -1.04          0.35
## death  -0.03  0.07  0.02
##
## Summary of imputed values
##
## totcst
##        n missing  unique    Info    Mean     .05     .10     .25
##      105       0      99       1   55123    5853    7291    8965
##      .50     .75     .90     .95
##    14743   28658  210999  390460
##
## lowest :   1528   1719   2213   4376   4595
## highest: 238595 313517 313661 375291 390460
##
## Starting estimates for imputed values:
##
##         sex         age      totcst       death
##      2.0000     64.8965 15110.1000      1.0000
```

totcst can be imputed beneficially by sex and age but not by death.

8.7 Use predictive mean matching to multiply impute cost 10 times per missing observation. Describe graphically the distributions of imputed values and briefly compare these to distributions of non-imputed values. State in a simple way what the sample variance of multiple imputations for a single observation of a continuous predictor is approximating.

We use the aregImpute function here.

```
support.aregImp <- aregImpute(~ age + sex + slos +
                                num.co + totcst + charges,
                              data = support,
                              type = "pmm",
                              n.impute = 10)

## Iteration 1
Iteration 2
Iteration 3
Iteration 4
Iteration 5
Iteration 6
Iteration 7
Iteration 8
Iteration 9
Iteration 10
Iteration 11
Iteration 12
Iteration 13


support.aregImp

##
## Multiple Imputation using Bootstrap and PMM
##
## aregImpute(formula = ~age + sex + slos + num.co + totcst + charges,
##     data = support, n.impute = 10, type = "pmm")
##
## n: 1000  p: 6  Imputations: 10   nk: 3
##
## Number of NAs:
##     age     sex    slos  num.co  totcst charges
##       0       0       0       0     105      25
##
##         type d.f.
## age        s    2
## sex        c    1
## slos       s    2
## num.co     s    2
## totcst     s    2
## charges    s    1
##
## Transformation of Target Variables Forced to be Linear
##
## R-squares for Predicting Non-Missing Values for Each Variable
## Using Last Imputations of Predictors
##  totcst charges
##   0.908   0.910
```

The sample variance of multiple imputations is approximating the measurement error that comes from the imputation itself. Since we are fitting an imputation model to the data, our imputed values are also following the random error of those fits; with multiple imputations, we can estimate that bias, which in turn will yield estimates that are less biased that those from single imputations.

References

[1] Kuhn M, Johnson K. Applied Predictive Modeling. Springer Science mathplus Business Media; 2013. Available from: http://dx.doi.org/10.1007/978-1-4614-6849-3.

[2] Knol MJ, Janssen KJM, Donders ART, Egberts ACG, Heerdink ER, Grobbee DE, et al. Unpredictable bias when using the missing indicator method or complete case analysis for missing confounder values: an empirical example. Journal of Clinical Epidemiology. 2010 Jul;63(7):728–736.

[3] Carpenter, James, Kenward, Michael. Multiple Imputation and its Application. 1st ed. Statistics in Practice. West Sussex, UK: Wiley; 2013. Available from: http://eu.wiley.com/WileyCDA/WileyTitle/productCd-0470740523.html.

[4] Steyerberg EW. Clinical Prediction Models. Statistics for Biology and Health. New York, NY: Springer New York; 2009. Available from: http://link.springer.com/10.1007/978-0-387-77244-8.

[5] Frank E Harrell. Regression Modeling Strategies. 2nd ed. Springer International Publishing; 2015. Available from: http://dx.doi.org/10.1007/978-3-319-19425-7.

[6] Rubin DB. Multiple Imputation for Nonresponse in Surveys. 1st ed. Wiley series in probability and mathematical statistics. US: John Wiley & Sons; 1987. Available from: http://onlinelibrary.wiley.com/book/10.1002/9780470316696.

[7] von Hippel PT. 4. Regression with Missing Ys: An Improved Strategy for Analyzing Multiply Imputed Data. Sociological Methodology. 2007 Aug;37(1):83–117.

[8] Graham, John W . Missing data: Analysis and Design. 1st ed. Statistics for Social and Behavioral Sciences. New York, US: Springer; 2012. Available from: http://www.springer.com/us/book/9781461440178.