Missing Value Imputation

Xinyu Zhang

2020/10/1

To begin with, after Qiang extracting the data from excel into a readable txt file, we take a look at this new dataset.

```
## tibble [68 x 56] (S3: spec_tbl_df/tbl_df/tbl/data.frame)
##  $ Q5_1         : num [1:68] 6 1 4 5 5 4 3 3 3 4 ...
##  $ Q5_2         : num [1:68] 6 5 4 5 2 1 1 4 4 3 ...
##  $ Q5_3         : num [1:68] 6 5 4 4 3 3 5 4 4 4 ...
##  $ Q5_4         : num [1:68] 6 5 5 5 5 5 5 5 4 4 ...
##  $ Q5_5         : num [1:68] 6 5 5 4 5 5 5 5 3 5 ...
##  $ Q5_6         : num [1:68] 6 5 5 5 5 5 5 5 5 5 ...
##  $ Q6box_1      : num [1:68] NA 1 NA 1 1 1 NA 1 1 1 ...
##  $ Q6box_2      : num [1:68] NA NA 2 2 NA NA NA NA 2 NA ...
##  $ Q6box_3      : num [1:68] NA NA NA NA NA 3 NA NA NA NA ...
##  $ Q6box_4      : num [1:68] 4 4 4 NA 4 NA NA NA NA NA ...
##  $ Q6_score     : num [1:68] NA 2 3 3 2 3 2 NA 1 2 ...
##  $ Q10_1        : num [1:68] 2 3 3 3 3 3 5 3 3 3 ...
##  $ Q10_2        : num [1:68] 2 3 3 3 3 3 4 3 3 3 ...
##  $ Q10_3        : num [1:68] 1 1 3 2 1 1 3 1 1 2 ...
##  $ Q10_4        : num [1:68] 1 1 3 2 1 1 3 1 1 2 ...
##  $ Q35_ERP      : num [1:68] 1 1 1 1 1 1 1 0 0 1 ...
##  $ Q40_TAA      : num [1:68] 0 1 1 1 1 1 1 0 0 0 ...
##  $ Q36_ITcount  : num [1:68] 3 2 3 3 NA 3 0 2 2 3 ...
##  $ Q37_ITassist : num [1:68] 1 0 3 0 1 1 2 1 1 3 ...
##  $ Q38_Taxcount : num [1:68] 23 23 40 12 100 200 55 19 15 45 ...
##  $ Q39_TAAexpert: num [1:68] 1 2.5 10 2 2 40 2 1 4 3 ...
##  $ Q22_box1     : num [1:68] 1 1 1 1 1 1 1 1 1 1 ...
##  $ Q22_box2     : num [1:68] NA 2 2 NA 2 2 NA NA NA 2 ...
##  $ Q22_box3     : num [1:68] 3 NA 3 NA 3 3 3 NA NA NA ...
##  $ Q22_box4     : num [1:68] 4 4 4 NA NA NA NA NA NA NA ...
##  $ Q22_box5     : num [1:68] NA NA 5 5 5 5 NA NA NA 5 ...
##  $ Q22_box6     : num [1:68] NA NA NA 6 NA NA NA NA NA NA ...
##  $ Q22_box7     : num [1:68] NA NA 7 NA NA NA 7 NA 7 7 ...
##  $ Q22_box8     : num [1:68] NA NA NA NA NA NA NA NA NA NA ...
##  $ Q22_box9     : logi [1:68] NA NA NA NA NA NA ...
##  $ Q22_box10    : num [1:68] NA NA NA NA 10 NA NA NA NA NA ...
##  $ Q23_1        : num [1:68] 4 3 3 1 2 3 4 2 3 2 ...
##  $ Q23_2        : num [1:68] 4 3 3 1 1 2 4 1 1 2 ...
##  $ Q26          : num [1:68] 6 6 6 6 6 6 6 6 6 6 ...
##  $ Q24_box1     : num [1:68] NA NA NA NA NA NA NA NA NA NA ...
##  $ Q24_box2     : num [1:68] NA NA 2 2 2 2 NA NA 2 NA ...
##  $ Q24_box3     : num [1:68] NA NA 3 3 3 3 NA NA 3 NA ...
##  $ Q24_box4     : num [1:68] 4 4 4 4 4 4 4 4 4 4 ...
##  $ Q27          : num [1:68] 2 2 3 3 3 3 3 2 3 3 ...
##  $ Q29          : num [1:68] 1 1 4 6 6 4 3 2 2 6 ...
##  $ Q41          : num [1:68] 1 1 3 6 NA 4 1 2 2 NA ...
##  $ Q30_1        : num [1:68] 5 1 3 4 4 3 1 4 4 3 ...
##  $ Q30_2        : num [1:68] 4 1 2 2 3 3 1 5 3 3 ...
##  $ Q30_3        : num [1:68] 4 3 2 4 5 4 1 3 5 3 ...
##  $ Q30_4        : num [1:68] NA NA NA NA NA 4 5 NA NA NA ...
##  $ Q31_1        : num [1:68] 3 5 3 4 1 3 5 1 2 3 ...
##  $ Q31_2        : num [1:68] 5 1 4 4 4 3 5 4 5 3 ...
##  $ Q31_3        : num [1:68] 1 5 3 4 4 3 5 4 5 3 ...
##  $ Q32_1        : num [1:68] 2 1 4 4 5 3 3 4 5 5 ...
##  $ Q32_2        : num [1:68] 6 1 4 4 5 2 1 4 5 3 ...
##  $ Q32_3        : num [1:68] 3 1 5 3 5 3 1 5 5 3 ...
##  $ Q32_4        : num [1:68] 6 1 5 3 5 1 3 4 3 6 ...
##  $ Q42          : num [1:68] 1 1 4 3 4 3 4 3 3 3 ...
##  $ Q45          : num [1:68] 30 15 0.33 1 2 0.92 13 1.25 7 2.5 ...
##  $ Q43          : chr [1:68] "CFO" "CFO" "CFO" "CFO" ...
##  $ Q44          : chr [1:68] NA "Vice President, Corporate Tax" "EVP of Tax" "SVP, Global Taxes" ...
##  - attr(*, "spec")=
##   .. cols(
##   ..   Q5_1 = col_double(),
##   ..   Q5_2 = col_double(),
##   ..   Q5_3 = col_double(),
##   ..   Q5_4 = col_double(),
##   ..   Q5_5 = col_double(),
##   ..   Q5_6 = col_double(),
##   ..   Q6box_1 = col_double(),
##   ..   Q6box_2 = col_double(),
##   ..   Q6box_3 = col_double(),
##   ..   Q6box_4 = col_double(),
##   ..   Q6_score = col_double(),
##   ..   Q10_1 = col_double(),
##   ..   Q10_2 = col_double(),
##   ..   Q10_3 = col_double(),
##   ..   Q10_4 = col_double(),
##   ..   Q35_ERP = col_double(),
##   ..   Q40_TAA = col_double(),
##   ..   Q36_ITcount = col_double(),
##   ..   Q37_ITassist = col_double(),
##   ..   Q38_Taxcount = col_double(),
##   ..   Q39_TAAexpert = col_double(),
##   ..   Q22_box1 = col_double(),
##   ..   Q22_box2 = col_double(),
##   ..   Q22_box3 = col_double(),
##   ..   Q22_box4 = col_double(),
##   ..   Q22_box5 = col_double(),
##   ..   Q22_box6 = col_double(),
##   ..   Q22_box7 = col_double(),
##   ..   Q22_box8 = col_double(),
##   ..   Q22_box9 = col_logical(),
##   ..   Q22_box10 = col_double(),
##   ..   Q23_1 = col_double(),
##   ..   Q23_2 = col_double(),
##   ..   Q26 = col_double(),
##   ..   Q24_box1 = col_double(),
##   ..   Q24_box2 = col_double(),
##   ..   Q24_box3 = col_double(),
##   ..   Q24_box4 = col_double(),
##   ..   Q27 = col_double(),
##   ..   Q29 = col_double(),
##   ..   Q41 = col_double(),
##   ..   Q30_1 = col_double(),
##   ..   Q30_2 = col_double(),
##   ..   Q30_3 = col_double(),
##   ..   Q30_4 = col_double(),
##   ..   Q31_1 = col_double(),
##   ..   Q31_2 = col_double(),
##   ..   Q31_3 = col_double(),
##   ..   Q32_1 = col_double(),
##   ..   Q32_2 = col_double(),
##   ..   Q32_3 = col_double(),
##   ..   Q32_4 = col_double(),
##   ..   Q42 = col_double(),
##   ..   Q45 = col_double(),
##   ..   Q43 = col_character(),
##   ..   Q44 = col_character()
##   .. )
```

```
## [1] 68 56
```

As we can see, there are 68 observations and 56 variables in the data.

However, our targeted analysis would be mainly based on the three dependent variables as introduced in Zhu, Kraemer, and Xu (2006):

- 1. initiation
- 2. adoption
- 3. routinization.

and five factors:

- 1. technological readiness
- 2. technology integration
- 3. managerial obstacles
- 4. competition intensity
- 5. regulartory environment

First Problem to Solve — Factor Analysis

Thus, our first question is how to combine the 56 variables into eight useful variables, and based on which conduct future analysis.

Second Problem to Solve — Variable Definition

We also noticed that the type of variable as "-attr(*, “spec”)=" shows still need further cleaning, that is, we need to define type of variables more carefully indicating whether it is numerical value, categorical variable, or ordered variable.

Third Problem to Solve — Missing Value Imputation

Next, we look at the proportion of missing values for the 56 variables in the dataset.

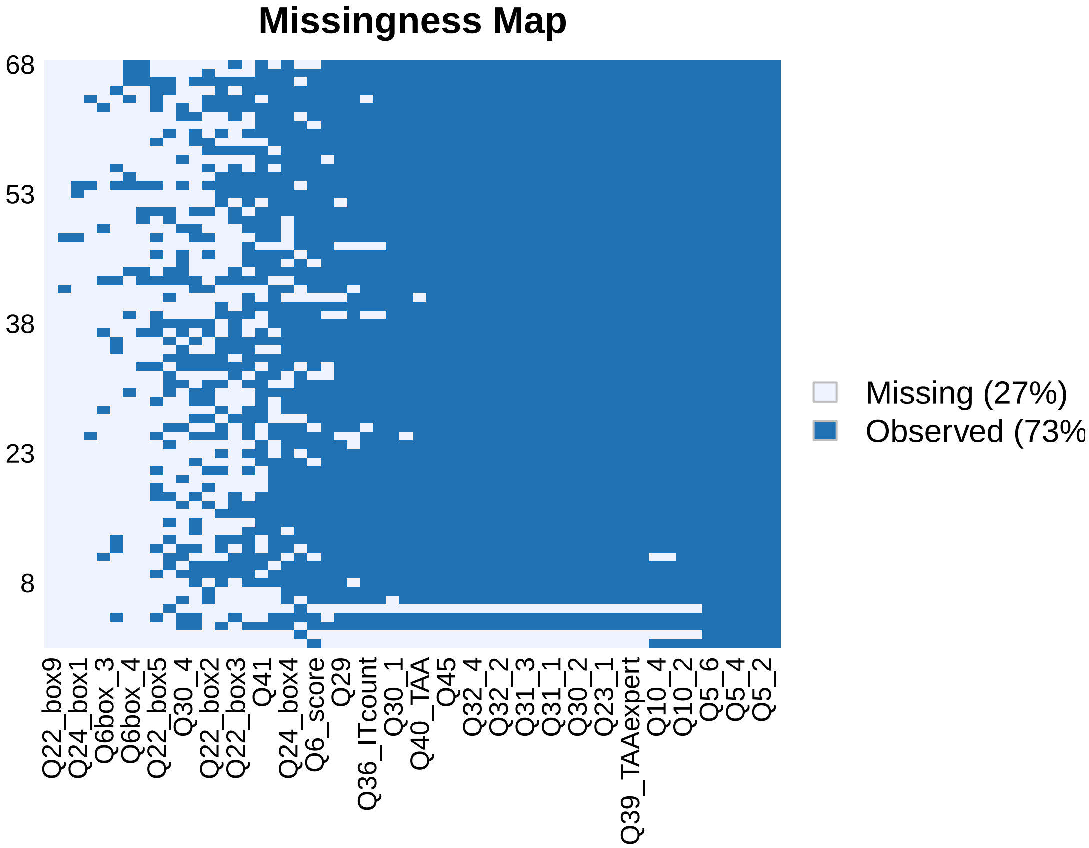
```r
pMiss <- function(x){sum(is.na(x))/length(x)*100}
psMiss <- round(apply(data,2,pMiss), 2)
psMiss <- psMiss[order(psMiss, decreasing = T)]
missing_percentage <- paste(psMiss,"%", sep="")
missMat <- cbind(c(1:56),names(psMiss), missing_percentage)
knitr::kable(missMat)
```

		missing_percentage
1	Q22_box9	100%
2	Q22_box8	97.06%
3	Q22_box10	95.59%
4	Q24_box1	95.59%
5	Q6box_3	91.18%
6	Q6box_4	86.76%
7	Q22_box6	86.76%
8	Q22_box4	85.29%
9	Q6box_2	67.65%
10	Q22_box5	67.65%
11	Q30_4	64.71%
12	Q22_box7	60.29%
13	Q22_box2	58.82%
14	Q22_box3	55.88%
15	Q24_box2	55.88%
16	Q24_box3	41.18%
17	Q41	35.29%
18	Q44	27.94%
19	Q24_box4	22.06%
20	Q6box_1	20.59%
21	Q6_score	14.71%
22	Q35_ERP	13.24%
23	Q26	11.76%
24	Q29	11.76%
25	Q36_ITcount	10.29%
26	Q37_ITassist	7.35%
27	Q40_TAA	5.88%
28	Q27	5.88%
29	Q30_1	5.88%
30	Q10_3	4.41%
31	Q10_4	4.41%
32	Q38_Taxcount	4.41%
33	Q39_TAAexpert	4.41%
34	Q22_box1	4.41%
35	Q23_1	4.41%
36	Q23_2	4.41%
37	Q30_2	4.41%
38	Q30_3	4.41%
39	Q31_1	4.41%
40	Q31_2	4.41%
41	Q31_3	4.41%
42	Q32_1	4.41%
43	Q32_2	4.41%

		missing_percentage
44	Q32_3	4.41%
45	Q32_4	4.41%
46	Q42	4.41%
47	Q45	4.41%
48	Q43	4.41%
49	Q10_1	2.94%
50	Q10_2	2.94%
51	Q5_1	0%
52	Q5_2	0%
53	Q5_3	0%
54	Q5_4	0%
55	Q5_5	0%
56	Q5_6	0%

We can see that there are many missing values in the survey, for which a missing value imputation might be useful.

```r
# plot missing map
missmap(data)
```



The above plot shows how the missing value is distributed among the data. The x-axis represent the variables that have missing value, and the y-axis shows the observation.

For some observation such as 1, 2, and 5, we could see there are too much missing values, thus it might need further consideration that if we should include those data with most variables missing.

For the variables from left to right, the missing proportion for each variable is decreasing, which need our further attention to select the proper variable for analysis.

Missing Value imputation for regular variables without considering Likrt scale

```
# add na as a new factor level for each categorical variable

# delete variables that have more than 50% missing value
mis50 <- c(1:56)[which(round(apply(data,2,pMiss), 2) > 50)]
data1 <- data[,-mis50] # reduce 15 variables
dim(data1)
```

```
## [1] 68 41
```

```
# use EM Algorithm to impute missing value, take column 7-11 as example
missP <- round(apply(data1,2,pMiss), 2)
index <- which((missP < 20 & missP > 10) )
names(data)[index]
```

```
## [1] "Q6box_2"  "Q10_2"    "Q10_4"    "Q22_box1" "Q22_box5"
```

```
dat_ame <- as.data.frame(data1[,index])
m=5
a.out <- amelia(x = dat_ame, m=m)
```

```
## Warning: There are observations in the data that are completely missing.
##          These observations will remain unimputed in the final datasets.
## -- Imputation 1 --
##
##   1  2  3  4  5  6  7  8
##
## -- Imputation 2 --
##
##   1  2  3  4  5  6
##
## -- Imputation 3 --
##
##   1  2  3  4  5  6  7  8
##
## -- Imputation 4 --
##
##   1  2  3  4  5  6  7  8
##
## -- Imputation 5 --
##
##   1  2  3  4  5  6
```

Notice that there are observations in the data that are completely missing, thus these observations will remain unimputed in the final datasets. This is because we only select several variable for illustration, in real application, as long as the obseration has one variable that's not missing, it can be imputed.

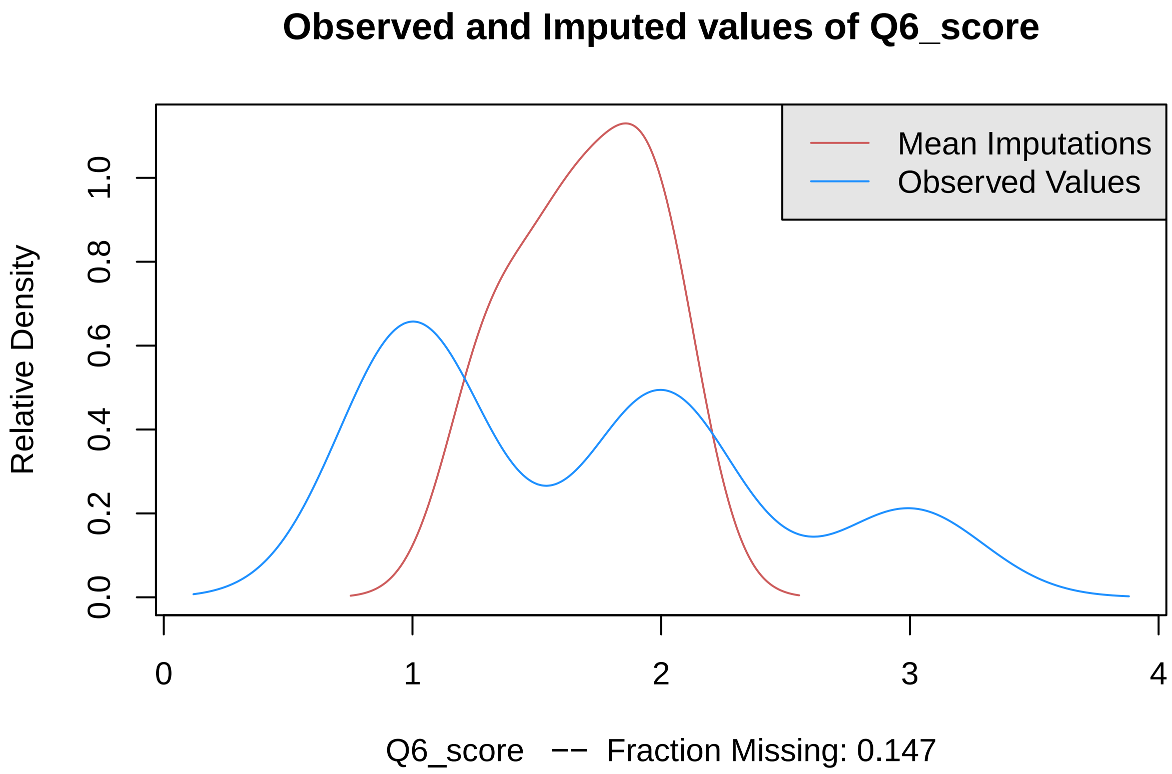
```
summary(a.out)
```

```
##
## Amelia output with 5 imputed datasets.
## Return code:  1
## Message:  Normal EM convergence.
```

```
##
## Chain Lengths:
## --------------
## Imputation 1:  8
## Imputation 2:  6
## Imputation 3:  8
## Imputation 4:  8
## Imputation 5:  6
##
## Rows after Listwise Deletion:  46
## Rows after Imputation:  66
## Patterns of missingness in the data:  14
##
## Fraction Missing for original variables:
## -----------------------------------------
##
##             Fraction Missing
## Q6_score           0.1470588
## Q35_ERP            0.1323529
## Q36_ITcount        0.1029412
## Q26                0.1176471
## Q29                0.1176471
```

This table output the iterations it takes to impute each variable as well the proportion of the missing value.

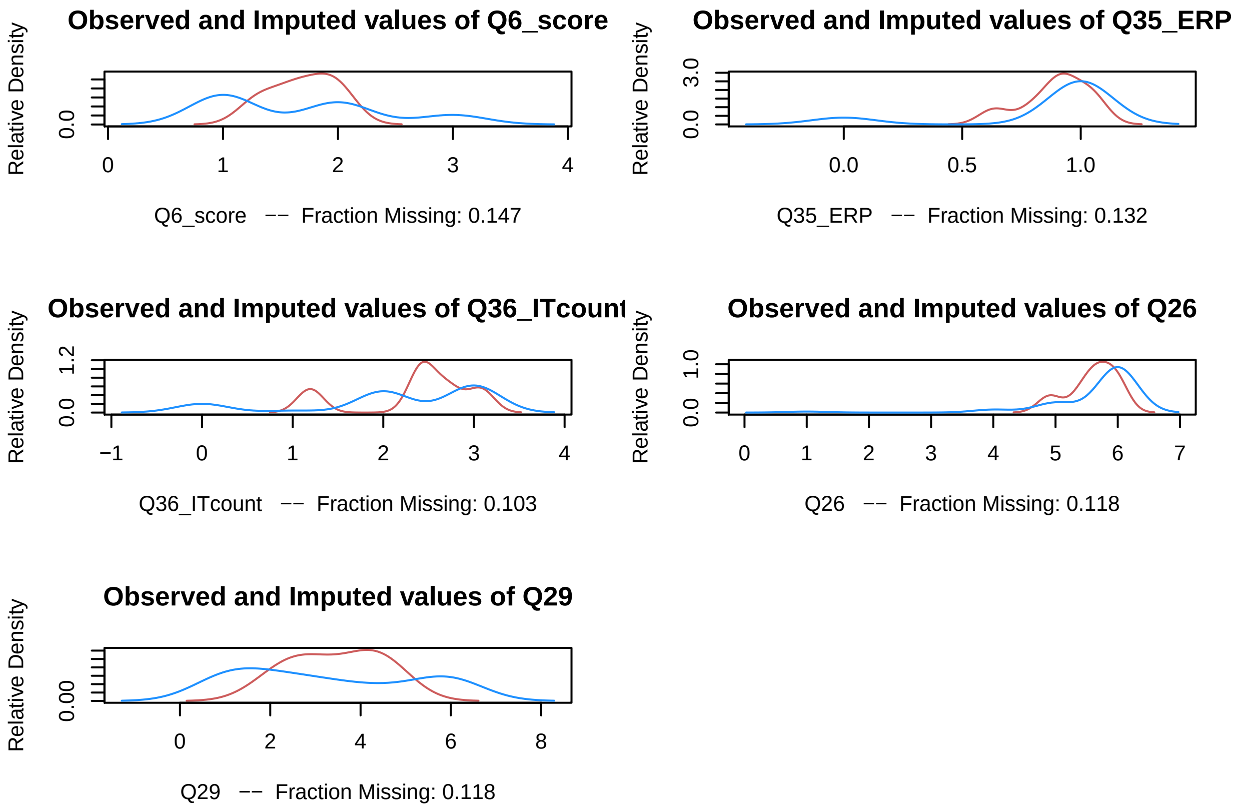
```
compare.density(a.out, var="Q6_score")
```



As the figure shows, the blue curve is the distribution curve for the original observed data without those missing records. The red curve is the distrbution curve of the “mean” imputed data.

Why here has a “mean”? Recall that m is the number of imputed datasets to create. When m is set to 1, the red curve is exactly the distrbution of the only imputed data. But usually, m is set to be larger such as 5, and the final imputed value would be the average of the five imputed values, which give us a more reliable result. Thus the “mean” here means the average of several parallel imputations.

```r
plot(a.out)
```



As we can see, the two curves vary a lot, and more subject knowledge can also be considered to correct the bias in imputation after knowing an ideal distribution.