Incomplete Data Analysis

Assignment 3—**sketch** of the solutions

Contents

Question 1. (a)

```
require(mice)
nu_cc <- sum(complete.cases(nhanes))
nu_obs <- nrow(nhanes)
pc_cc <- nu_cc / nu_obs
(pc_incomplete <- 1 - pc_cc)
```

```
## [1] 0.48
```

Thus, 48% of all cases are incomplete.

(b) From `help(nhanes)` it follows that `age` and `hyp` are groups, and hence we treat them as factors:

```
nhanes$hyp <- as.factor(nhanes$hyp)
nhanes$age <- as.factor(nhanes$age)
```

Next, we run Steps~1–3:

```
imp <- mice(nhanes, seed = 1, printFlag = FALSE)
fits <- with(imp, lm(bmi ~ age + hyp + chl))
poolfits <- pool(fits)
```

Some diagnostics similar to those presented in the Lectures and Workshops are conducted (results not shown). Finally, the column of interest for finding the proportions of variance due to the missing data for each parameter is lambda, and here are its values:

```
##          term    estimate    lambda
## 1 (Intercept) 18.66788816 0.3225123
## 2        age2 -4.66637582 0.6107359
## 3        age3 -6.55049039 0.6027946
## 4        hyp2  2.27386418 0.2635386
## 5         chl  0.05312788 0.4519846
```

Since lambda is highest for the parameters related with age, these parameters seem to be the most affected by the nonresponses.

(c) We repeat the experiment for seeds 2–6:

```
for (i in 2:6) {
  imp <- mice(nhanes, seed = i, printFlag = FALSE)
  fits <- with(imp, lm(bmi ~ age + hyp + chl))
  poolfits <- pool(fits)
  cat("seed", i, "\n")
  print(poolfits[[3]][c(1, 3, 10)])
}
```

```
## seed 2
##          term    estimate    lambda
## 1 (Intercept) 18.26451299 0.4350074
## 2        age2 -6.04082855 0.3009555
## 3        age3 -7.71268039 0.2086637
## 4        hyp2  1.99079244 0.5160375
## 5         chl  0.05899711 0.5424351
## seed 3
##          term    estimate    lambda
## 1 (Intercept) 19.54151204 0.2132129
## 2        age2 -5.38528749 0.4400633
## 3        age3 -7.48648305 0.5147210
## 4        hyp2  3.01715917 0.4120117
## 5         chl  0.05182769 0.1164702
## seed 4
##          term    estimate     lambda
## 1 (Intercept) 18.50867683 0.22531441
## 2        age2 -5.38569153 0.39659660
## 3        age3 -6.82758119 0.55904230
## 4        hyp2  2.18863884 0.03683047
## 5         chl  0.05578638 0.30093907
## seed 5
##          term    estimate    lambda
## 1 (Intercept) 18.01788557 0.4852371
## 2        age2 -6.05597473 0.4827434
## 3        age3 -7.88827934 0.4029169
## 4        hyp2  2.68268400 0.5194407
## 5         chl  0.06186883 0.5294706
## seed 6
##          term    estimate    lambda
## 1 (Intercept) 19.74268076 0.2528861
## 2        age2 -4.91054386 0.3075109
## 3        age3 -7.18256707 0.2834432
## 4        hyp2  3.20696860 0.1919560
## 5         chl  0.04665664 0.1254126
```

Hence, the tentative finding from (a) does not hold for all seeds. Indeed, for example for seed 2 and 5, lambda is no longer highest for the parameters related with age.

(d) We now redo (c) but with $M = 100$, the code is exactly the same as above but we set `m = 100` in `mice`.

```
## seed 2
##          term    estimate    lambda
## 1 (Intercept) 18.45148801 0.1870491
## 2        age2 -5.50018197 0.3170976
## 3        age3 -7.47252924 0.4006068
## 4        hyp2  2.19072186 0.3775514
## 5         chl  0.05753776 0.2242792
## seed 3
##          term    estimate    lambda
## 1 (Intercept) 18.71582060 0.2595077
## 2        age2 -5.49701679 0.2875682
## 3        age3 -7.21033189 0.3456631
## 4        hyp2  2.23903630 0.3346525
## 5         chl  0.05584034 0.3065429
## seed 4
##          term    estimate    lambda
## 1 (Intercept) 19.24745768 0.1924386
## 2        age2 -5.39780457 0.3623307
## 3        age3 -7.20445319 0.3622102
## 4        hyp2  2.45142719 0.3212144
## 5         chl  0.05238749 0.2282279
## seed 5
##          term    estimate    lambda
## 1 (Intercept) 19.16421733 0.2462895
## 2        age2 -5.47083947 0.3084504
## 3        age3 -7.13601472 0.3027969
## 4        hyp2  2.12874382 0.2780733
## 5         chl  0.05348992 0.2869601
## seed 6
##          term    estimate    lambda
## 1 (Intercept) 19.15308076 0.2166248
## 2        age2 -5.32931519 0.2836349
## 3        age3 -6.92507648 0.3800772
## 4        hyp2  1.96156711 0.3602735
## 5         chl  0.05277679 0.2587395
```

With $M = 100$ the values of lambda between the seeds seem to vary less, and hence I would prefer running the analysis with higher M. The coefficient associated with `age3` is higher for the majority of the seeds.

Question 2. To calculate the empirical coverage probability of the 95% confidence intervals for β_1, the following code was used to *count* how many times over the 100 datasets the calculated intervals contain the true value of $\beta_1 = 3$.

```r
## COVERAGE FOR SRI
load('dataex2.Rdata')
n <- nrow(dataex2)
count <- 0
for (dataset in 1:n){
  ## run steps 1--3
  imp <- mice(dataex2[, ,dataset], m = 20, meth = 'norm.nob',
              seed = 1, printFlag = FALSE) #
  fits <- with(imp, lm(Y ~ X))
  pooledfits <- pool(fits)
  ## get lower and upper bound for CI of beta1
  sum_pool <- summary(pooledfits, conf.int = TRUE)
  lower_bound <- sum_pool$`2.5 %`[2]
  upper_bound <- sum_pool$`97.5 %`[2]
  # check if 3 is in CI
  if ((lower_bound <= 3) & (3 <= upper_bound))
    count <- count + 1
}
(coverage_sri <- count / n)
```

```
## [1] 0.88
```

Hence the empirical coverage of stochastic regression imputation (SRI) is 0.88. The code for the bootstrap-based version of SRI is the same as above but with `method = "norm.boot"`. The resulting empirical coverage is 0.95. The reason why SRI does not attain the nominal coverage is because it is a form of improper multiple imputation, and hence it ignores parameter uncertainty. The bootstrap-based version does not ignore parameter uncertainty, and hence it attains the nominal coverage of 0.95.

Question 3. *Strategy 1*: Recall that the predicted values in a linear regression model are $\widehat{y} = \mathbf{x}^{\mathrm{T}}\hat{\boldsymbol{\beta}} = \sum_{i=1}^{p} x_i\widehat{\beta}_i$, where $\hat{\boldsymbol{\beta}} = (\widehat{\beta}_1, \dots, \widehat{\beta}_p)^{\mathrm{T}}$ is the least squares estimator and $\mathbf{x} = (x_1, \dots, x_p)^{\mathrm{T}} \in \mathbb{R}^p$. Since we have M datasets, we have M predicted values given by $\widehat{y}^{(m)} = \sum_{i=1}^{p} x_i\widehat{\beta}_i^{(m)}$, where $\hat{\boldsymbol{\beta}}^{(m)} = (\widehat{\beta}_1^{(m)}, \dots, \widehat{\beta}_p^{(m)})^{\mathrm{T}}$ is the least squares estimator based on the mth dataset. The Rubin rule for Strategy 1 is hence,

$$\widehat{y}^{\mathrm{MI}} = \frac{1}{M}\sum_{m=1}^{M}\widehat{y}^{(m)} = \frac{1}{M}\sum_{m=1}^{M}\sum_{i=1}^{p} x_i\widehat{\beta}_i^{(m)}. \tag{1}$$

Strategy 2: The Rubin rule for Strategy 2 is $\hat{\boldsymbol{\beta}}^{\mathrm{MI}} = 1/M\sum_{m=1}\hat{\boldsymbol{\beta}}^{(m)}$, and hence the corresponding predicted values are

$$\tilde{y} = \mathbf{x}^{\mathrm{T}}\hat{\boldsymbol{\beta}}^{\mathrm{MI}} = \frac{1}{M}\sum_{m=1}^{M}\mathbf{x}^{\mathrm{T}}\hat{\boldsymbol{\beta}}^{(m)} = \frac{1}{M}\sum_{m=1}^{M}\sum_{i=1}^{p} x_i\widehat{\beta}_i^{(m)}, \tag{2}$$

and hence by comparing (1) and (2) it follows that both strategies coincide.

Question 4. (a)

```r
load('dataex4.Rdata')
imp_a <- mice(dataex4, m = 50, seed = 1, printFlag = FALSE) # check for problems
imp_a$loggedEvents
```

```
## NULL
```

```r
# check for convergence
# plot(imp_a, layout = c(1,2)) # looks ok
# densityplot(imp_a) # also looks good
fita <- with(imp_a, lm(y ~ x1 + x2 + x1*x2))
pool_fita <- pool(fita)
sum_poola <- summary(pool_fita, conf.int = TRUE)
sum_poola[, c(2, 3, 6, 7, 8)]
```

```
##    estimate  std.error p.value    2.5 %    97.5 %
## 1 1.5929831 0.09541331       0 1.404501 1.7814655
## 2 1.4112333 0.09732912       0 1.219397 1.6030697
## 3 1.9658191 0.05323220       0 1.860657 2.0709812
## 4 0.7550367 0.05701458       0 0.642302 0.8677715
```

```r
# all but beta 1 and 3 s CI's include their true values.
```

The table above reports estimates and 95% confidence intervals when only imputing the y and x_1 variables. This method, impute, then transform, leads to confidence intervals that do not include the parameters' true values for two of the variables ($\beta_1 = 1$ and $\beta_3 = 1$). The confidence interval for the interaction term x_1x_2 seems to be biased towards 0. One of the reasons for this is that we do not include the relationship for the interaction term in the imputation model, but only in the analysis model.

(b) Using passive imputation means that we include the interaction term in the imputation model. This gives the following results for parameter estimates and confidence intervals:

```r
# add column for x1*x2 to dataframe
dataex4b <- dataex4
dataex4b['x1x2'] <- dataex4$x1 * dataex4$x2
# dry run
impb_dry <- mice(dataex4b, printFlag = FALSE, maxit = 0)
methb <- impb_dry$method
methb['x1x2'] <- "~I(x1*x2)"
# we do not use x1x2 as predictor for imputation of x1 and x2
predb <- impb_dry$predictorMatrix
```

```
predb[c("x1", "x2"), "x1x2"] <- 0
# don't have to change anything for imputation of x2 since it is all observed
# full run
impb_proper_run <- mice(dataex4b, method = methb, predictorMatrix = predb,
                        printFlag = FALSE, seed = 1, m = 50)
impb_proper_run$loggedEvents
```

```
## NULL
```

```
fitb <- with(impb_proper_run, lm(y ~ x1 + x2 + x1x2))
pool_fitb <- pool(fitb)
sum_poolb <- summary(pool_fitb, conf.int = TRUE)
sum_poolb[, c(2, 3, 6, 7, 8)]
```

```
##    estimate  std.error p.value     2.5 %    97.5 %
## 1 1.5534782 0.08842211       0 1.3788626 1.7280939
## 2 1.1926170 0.09584345       0 1.0034980 1.3817360
## 3 1.9964402 0.04936582       0 1.8989468 2.0939336
## 4 0.8740573 0.05678521       0 0.7615712 0.9865434
```

As shown above, including passive imputation for the x_1x_2 term has improved the confidence intervals and estimates compared to a), i.e. the estimates are closer to the true values of β_i, $i = 1, 2, 3, 4$. Despite this improvement, the confidence intervals for β_1 and β_3 again do not include the their true values, suggesting that passive imputation, where we make use of the deterministic relationship that x_1x_2, can return biased confidence intervals.

(c) Imputing the interaction x_1x_2 as just another variable (JAV) leads to the best results in this question.

```
dataex4c <- dataex4b
impc <- mice(dataex4c, m = 50, seed = 1, printFlag = FALSE)
fitc <- with(impc, lm(y ~ x1 + x2 + x1x2))
pool_fitc <- pool(fitc)
sum_poolc <- summary(pool_fitc, conf.int = TRUE)
sum_poolc[, c(2, 3, 6, 7, 8)]
```

```
##   estimate  std.error p.value     2.5 %   97.5 %
## 1 1.499714 0.07821436       0 1.3452011 1.654227
## 2 1.003930 0.08228372       0 0.8414967 1.166363
## 3 2.026180 0.04371605       0 1.9398113 2.112548
## 4 1.017793 0.04428071       0 0.9303479 1.105238
```

Comparing the results for the estimates with the results before show that using JAV leads to estimates closest to their true value. Further, the confidence intervals finally include the true parameter values for all β_i, indicating that the model has performed well.

(d) When imputing an interaction term x_1x_2 under the just another variable approach, the imputed interaction term will not be equal to the product of the (partly) imputed x_1 and x_2, so that the imputed values in the interaction term are not equal to the originally specified deterministic relationship $x_1 \times x_2$.

Question 5

The model of interest for our analysis is:

$$\text{wgt} = \beta_0 + \beta_1\text{gender} + \beta_2\text{age} + \beta_3\text{hgt} + \beta_4\text{WC} + \varepsilon, \quad \varepsilon \sim \text{N}(0, \sigma^2). \tag{3}$$

The analysis is organized as follows. In §5.1 we will report the main findings of an exploratory analysis, §5.2 conducts a preliminary complete case analysis, and §5.3 conducts a multivariate imputation via a fully conditional specification. We close in §5.4 with some final remarks. Finally, Supporting Information is available as an appendix.

§5.1 Preliminary Exploratory Analysis. In Figure 1 we depict the missing data pattern for the `NHANES2` dataset. The chart shows that `wgt`, `gender`, `age`, which are part of the substantive model, do not have missing observations; yet, the other variables in the substantive model, `hgt` and `WC`, have missing observations. Note that this well-known dataset does not code the missing values in the most appropriate way, but this has no major consequences for the analysis.[1] Overall, it appears that we cannot write any of the variables as a function of the others, so I will not change the predictor matrix.

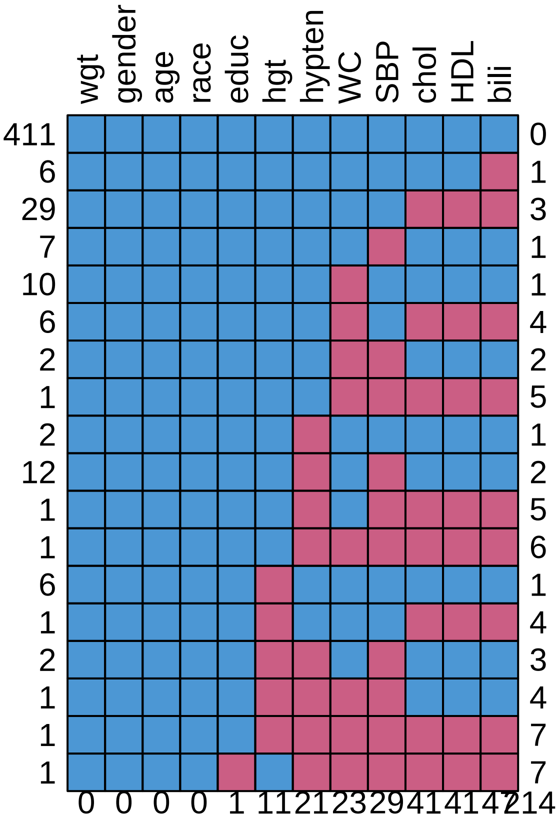

Figure 1: Missing-data pattern for `NHANES2` dataset.

Next, we shift our focus to the margins. Figure 2 in the Supporting Information displays the frequencies for each variable of interest. Parenthetically, we note the nice balance on the variable gender—which will be part of our substantive model. Finally, Figure 3 in the Supporting Information showcases the degree of linear association across all variables in the dataset. We would think that some of the variables, for example weight and height and waist circumference are correlated which can be confirmed from Figure 3.

§5.2 Complete Case Analysis. As a warm-up, we start with the complete case analysis. As can be seen from Figure 1 the complete case analysis will be based on 411 observations. Table 1 reports a summary of the fitted regression model by learning about the parameters in (3) using a complete case analysis. Table~1 suggests that a 1 year increase yields on average a weight reduction of about -0.16 kg, and that 10cm increase in height corresponds to corresponds to an average increase of about 5.16 kg, and so on. Table~1 suggests that—with the exception of gender—all covariates in (3) are significant. Standard regression diagnostics are available from the Supporting Information, and indicate an overall good fit of the model; see Figure 4. This is strengthened by the fact that the R^2 is relatively high ($R^2 = 0.86$).

[1] The best practice would be convert all `NaN` (“Not a Number”) into `NA` ("Not Available"). `NaN`s are typically due to numerical or definition issues (e.g. `0 / 0`), and hence not the same thing as a missing observation (`NA`). Yet, in most cases this has no impact in R as can be seen, for instance, by comparing `sum(is.nan(NHANES2$SBP))` with `sum(is.na(NHANES2$SBP))`

Table 1: Summary of fitted regression model in (3) under a complete case analysis.

Predictor	Estimate	SE	t value	p value
(Intercept)	-99.85	7.79	-12.83	0.00
genderfemale	-1.31	0.85	-1.55	0.12
age	-0.15	0.02	-6.95	0.00
hgt	51.60	4.45	11.58	0.00
WC	1.03	0.02	45.60	0.00

Table 2: Summary of fitted regression model in (3) under multiple imputation ($M = 30$); WI and BI respectively denote within and between variances.

Predictor	Estimate	SE	t value	p value	WI variance	BI variance	Proportion of variance
(Intercept)	-100.85	7.67	-13.14	0.0	55.90	2.88	0.05
genderfemale	-1.39	0.83	-1.66	0.1	0.67	0.02	0.03
age	-0.16	0.02	-7.36	0.0	0.00	0.00	0.05
hgt	52.43	4.40	11.93	0.0	18.27	1.03	0.05
WC	1.03	0.02	45.95	0.0	0.00	0.00	0.02

§5.3 Multiple Imputation by a Fully Conditional Specification. The complete case analysis from §5.2 is based on 411 observations, and now we aim to do a more effective use of the data by using multiple imputation. Using the `mice` package, we implement a fully conditional specification based on $M = 30$ complete datasets.

The results are shown in Table~2. Table~2 is in line with the naive predictions from the complete case analysis from Table~1. Hence, similar interpretations hold. The column from the proportion of variance (in the parameters due to missing values) suggests that the regression parameters associated with `age` and `hgt`, along with the intercept are the ones mostly affected by the missing values. Standard regression diagnostics are available from the Supporting Information, and indicate an overall good fit of the model; see Figure 5. This is strengthened by the fact that the R^2 is relatively high ($R^2 = 0.86$).

Some final comments on the obtained imputed datasets are in order. Predictive mean matching was used for the *imputation model* in the first step of the full conditional imputation. Visual inspection of the traceplots in Figure 6 in the Supporting Information suggests that all chains converged. In addition, the density and proportion plots available from the Supporting Information suggest a close agreement between the density the observed and imputed data, along with a `gender` effect in height and a less accurate though expected behavior for `educ`.\footnote{Note that `educ` only has one missing value. The full details on the specification used over `mice` (e.g. maximum number of iterations and seed) are available from the source *Rmd* file.

§5.4 Closing Remarks. This analysis fits the substantive model in (3) to the `NHANES2` dataset. A complete case analysis as well as a fully conditional specification were used to fit (3), and both lead to similar empirical findings. To supplement the analysis, we have also conducted a sensitivity analysis with a higher value of M, which is reported in the Supporting Information. The main findings remain unchanged and are once more similar to the ones reported above.

Some comments on future analyses are in order. It would have been interesting to compare our analysis which is based on a multiple imputation by chained equations with that of a *joint model imputation.* Re-doing the analysis with other variants of Step 1 would seem like another natural inquiry for future work.

Table 3: Summary of fitted regression model in (3) under multiple imputation ($M = 50$); WI and BI respectively denote within and between variances. This table should be compared with those in pp. ?? and 7

Predictor	Estimate	SE	t value	p value	WI variance	BI variance	Proportion of variance
(Intercept)	-101.28	7.64	-13.25	0.00	55.42	2.95	0.05
genderfemale	-1.34	0.83	-1.61	0.11	0.67	0.03	0.04
age	-0.16	0.02	-7.49	0.00	0.00	0.00	0.04
hgt	52.72	4.37	12.06	0.00	18.10	1.00	0.05
WC	1.03	0.02	46.09	0.00	0.00	0.00	0.02

Supporting Information

Statistical Packages

For the analyses above I have used the following R packages: `corrplot`, `devtools`, `dplyr`, `ggplot2`, `mice`, `reshape2`, `RColorBrewer`, `tidyr`, and `JointAI`.

Supplementary Reports

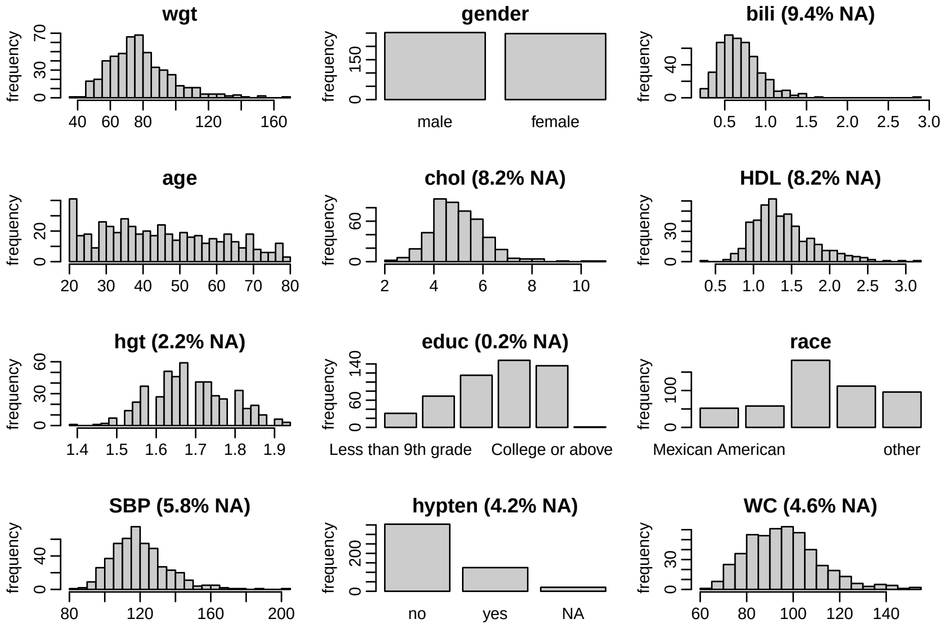

Figure 2: Marginal frequencies for all variables in the `NHANES2` along with percentages of missing data.

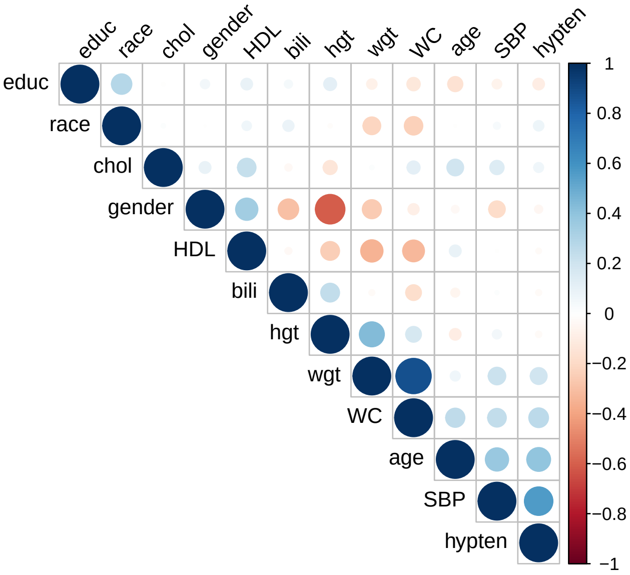

Figure 3: Visualization of the correlation matrix for the `NHANES2` dataset.

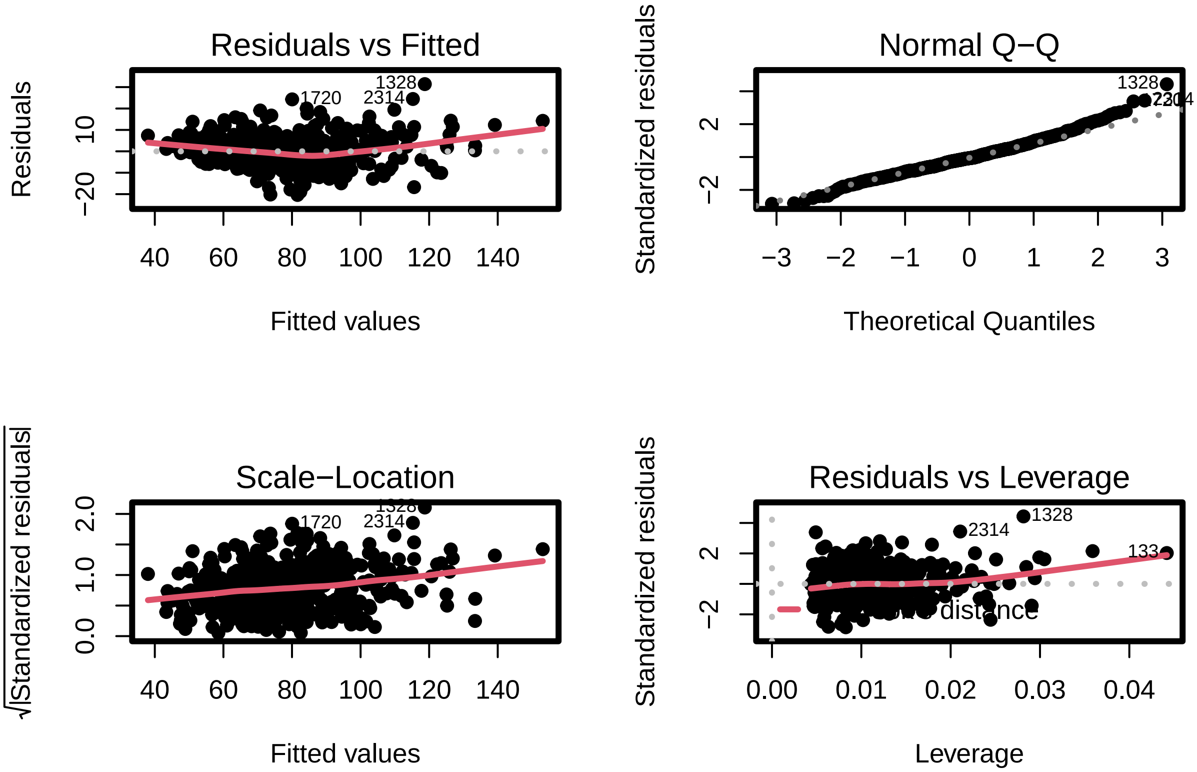

Figure 4: Regression diagnostics for fitted regression model in (3) for a complete case analysis of the `NHANES2` dataset.

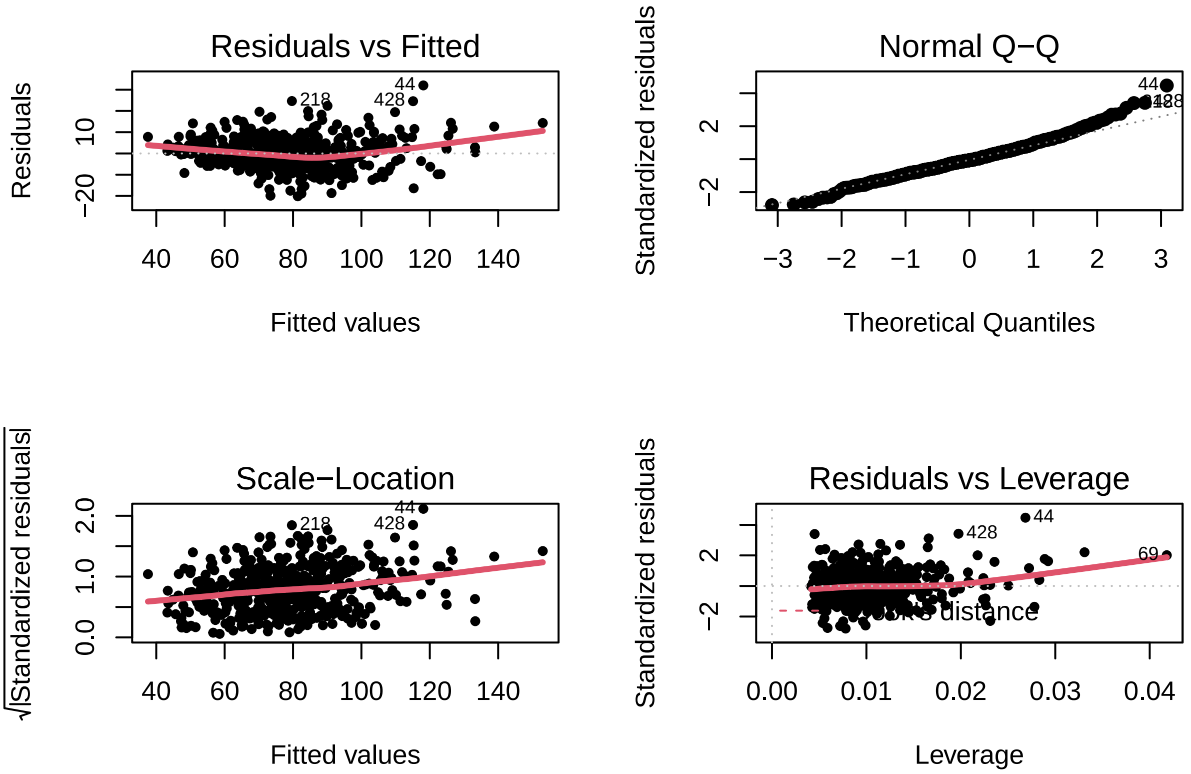

Figure 5: Regression diagnostics for fitted regression model in (3) for a multiple imputation analysis of the `NHANES2` dataset. Here we consider the fit from the 5th imputed dataset but similar findings hold for the remainder imputed datasets.

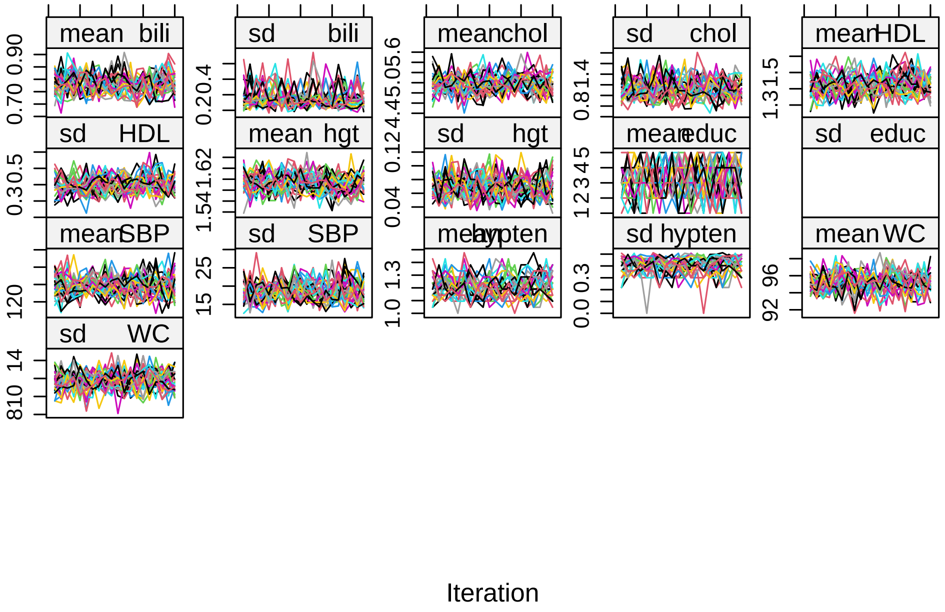

Figure 6: Convergence diagnostics. Note there is only 1 `NA` for `educ`.

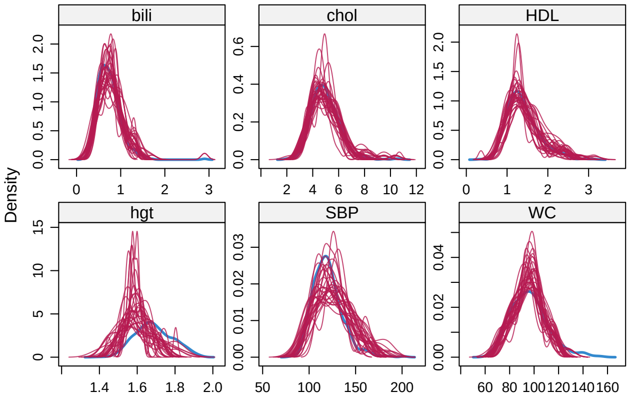

Figure 7: Density plots for observed (blue) and imputed (red) data. Note that there are only 11 `NA`s for `hgt`.

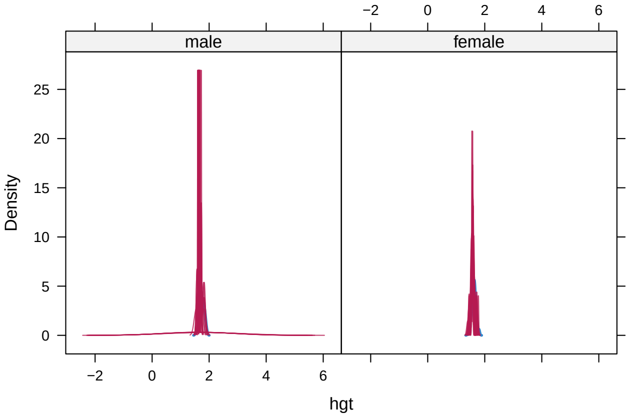

Figure 8: Density plots for `hgt` given `gender` for observed (blue) and imputed (red) data. Note that there are only 11 `NA`s for `hgt`.

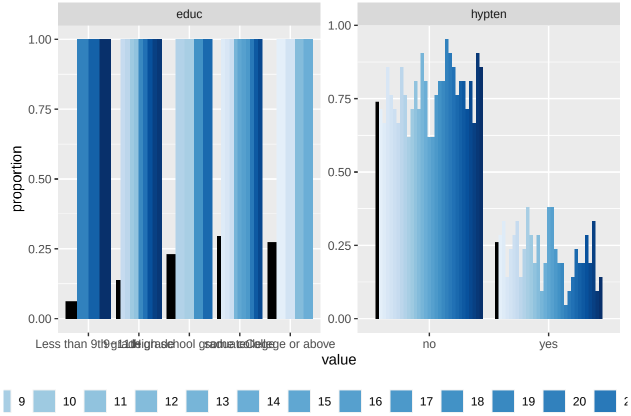

Figure 9: Proportion plots for `educ` and `hypten`. Note that there is only one NA for `educ`