Multiple Imputation of Missing Data

An Appendix to *An R Companion to Applied Regression*, third edition

John Fox & Sanford Weisberg

last revision: 2018-09-07

Abstract

After presenting some basic ideas concerning missing data, this appendix explains briefly how multiple imputation of missing data works, and then illustrates, using the **mice** package, how to apply this method to estimating a regression model in the presence of missing data.

In fitting statistical models in the *R Companion*, we deal with missing data by performing a *complete-case analysis*, removing all cases with missing values for any of the variables that appear in the model. When we fit more than one model to the same data set, we are careful to filter the data set to remove cases with missing values for variables that appear in *any* of the models, to insure that we fit all of the models to a consistent subset of cases, making it possible, for example, to compare nested models by likelihood-ratio tests.

Complete-case analysis is far from the worst strategy for dealing with missing data, and it can be a good strategy if only a small proportion of cases have missing values, but in many situations it can lead to seriously biased estimates and in others to inefficient estimates that don't make good use of the available data. This appendix introduces *multiple imputation*, an alternative, and more principled, strategy for dealing with missing data that provides more reasonable results under certain circumstances (explained below). We begin by making some key distinctions, proceed to explain how multiple imputation of missing data works, and conclude with an example using the **mice** package for R. There are other packages in R that implement various versions of multiple imputation for missing data, including the **norm** (Schafer, 2013), **cat** (Schafer, 2012), **mix** (Schafer, 2017), **mi** (Su et al., 2011), and **Amelia** packages (Honaker et al., 2011).

Multiple imputation of missing data, and, more generally, estimation in the presence of missing data, are large topics. We merely scratch the surface here. For more information, see the references given at the end of the appendix. In particular, our presentation here adapts some materials from Fox (2016, Chap. 20).

1 Basic Ideas

Let the matrix $\mathbf{X}_{n\times p}$ represent a complete data set with n cases and p variables, some of whose elements, in $\mathbf{X}_{\text{obs}}$, are observed, and others, in $\mathbf{X}_{\text{mis}}$, are missing.[1] We use the term *missing data* for values that exist but aren't observed, not for values that are undefined. For example, an employed individual's income that is unreported in a sample survey is missing, but the age of a childless respondent's oldest child is undefined. Both missing data and undefined data normally are coded as `NA` in an R data set.

In a seminal work on estimation with missing data, Rubin (1976) distinguished among three kinds of missing data: data that are *missing completely at random*, abbreviated *MCAR*; data that

[1] If you're unfamiliar with matrices, simply think of $\mathbf{X}$ as a data table with rows representing cases and columns representing variables. $\mathbf{X}_{\text{obs}}$ and $\mathbf{X}_{\text{mis}}$ aren't really matrices, but rather are subsets of the complete-data matrix.

are *missing at random*, *MAR*; and data that are *missing not at random*, *MNAR*. The meaning of the three terms—in particular, the distinction between MCAR and MAR—isn't immediately obvious, and, in retrospect, it probably would have been better had Rubin chosen different terminology. These terms, however, are now used nearly universally in the literature on statistical estimation with missing data, and so we're obliged to understand them.

- Data are MCAR if the missing data are effectively a simple random sample of the complete data, in which case the observed data are also a simple random sample of the complete data. Under these circumstances, the probability that a data value is missing, which Rubin terms *missingness*, is unrelated to the data value or to any other value, missing or observed, in the data set. When missing data are MCAR, complete-case analysis produces unbiased estimates of regression coefficients, although it may not use information in the sample efficiently.

- Data are MAR if missingness is unrelated to the missing data *conditional on* the observed data. MCAR, therefore, is a stronger condition than—and a special case of—MAR. Suppose, for example, that some individuals fail to provide their incomes in a sample survey, and further that individuals with higher incomes are more likely to decline to answer, but that conditional on other observed characteristics of the individuals, such as their occupation, education, age, and so on, refusal to answer the question is unrelated to income. Then the missing income data are MAR but not MCAR. Multiple imputation of missing data, described in this appendix, and some other strategies not described here, can provide unbiased and efficient estimates of regression coefficients when data are MAR.

- Data are MNAR when they aren't MAR—that is, when missingness is related to the missing values themselves even conditioning on the observed data. When data are MNAR, it's necessary explicitly to model the missingness mechanism in order to obtain unbiased estimates of regression coefficients, a much more difficult process than handling data that are MAR.[2] The missingness mechanism is therefore said to be *ignorable* when data are MCAR or MAR, and *nonignorable* when data are MNAR.

It's fair to say that unless missing data are generated by the design of the data-collection procedure (e.g., when some respondents are selected at random to answer one set of questions on a survey and other respondents another set of questions), missing data are almost always MNAR. Nevertheless, if missing data are close enough to MAR then methods like multiple imputation can produce much less biased estimates than complete-case analysis.

2 Outline of Multiple Imputation

Imputation of missing values—that is, filling in missing data with plausible values—is a long-standing general technique for dealing with missing data. Some common traditional imputation methods include *mean imputation*, replacing missing values for a variable with the mean of the observed values; *regression imputation*, or, more generally, *conditional mean imputation*, replacing missing values with predicted values, based, for example, on fitting a regression model to the observed data; and *hot-deck imputation*, replacing missing values with observed values for similar cases. Unconditional mean imputation is seriously flawed and is generally much worse than simply discarding missing data. Other single-imputation methods are also, but more subtly, flawed, in that they fail to capture the added uncertainty due to missing data, and as a consequence can bias not just coefficient standard errors but regression coefficient estimates themselves.

[2] A common example of such a model is Heckman's selection-regression model (Heckman, 1974, 1976), for which he won a Nobel Prize in economics; such models can be very sensitive to distributional assumptions and assumptions about the missing-data mechanism—see, e.g., Tukey's caustic comments following Heckman and Robb (1986).

Multiple imputation of missing data improves on regression imputation by sampling several times from the distribution of the missing data conditional on the observed data, producing several completed data sets. In doing so, it takes into account not only uncertainty due to residual variation—that is, the inability to predict missing values without error from the observed data (e.g., by sampling from the estimated error distribution for a continuous variable or sampling from the estimated conditional probability distribution of a factor)—but also uncertainty in the parameter estimates used to obtain the predictions (by sampling from the estimated distribution of the parameters of the imputation model). That said, multiple imputation isn't a single technique but rather a collection of methods.

One approach, taken, for example, by Schafer (1997), is to specify a multivariate model for the complete data, such as the multivariate-normal distribution, and to base imputations on that model. This approach is available in Schafer's **norm** package for R (Schafer, 2013). Although the multivariate-normal model is a very strong model for the complete data, there is evidence that multiple-imputation inferences based on it are robust, and the model can even be applied to highly non-normal data such as dummy variables generated from a factor (see, e.g., Allison, 2002).

Another, more flexible, approach is to build a *conditional* prediction model for each variable with missing data. For example, missing data in a continuous variable might, perhaps after suitable transformation, be predicted using a normal linear model, while missing data in a binary factor might be predicted using a logistic regression. Because the predictors in these conditional models are in general themselves subject to missing data, at each stage missing values in the predictors are filled in with current imputations, and the process is iterated, cycling through the imputation models until the aggregated predictions stabilize (see the example in Section 3). The whole iterated process is repeated to produce several completed data sets. This approach is implemented in the **mice** package (an acronym for **m**ultivariate **i**mputation by **c**hained **e**quations), which we use in Section 3.

Suppose now that we have M completed data sets, each with the missing values imputed, and that we fit a regression model to each completed data set. For concreteness, we'll suppose that this is a normal linear model to be fit by least squares,

$$(y|x_1, \ldots, x_k) \sim N(\beta_0 + \beta_1 x_1 + \cdots + \beta_k x_k, \sigma^2)$$

but it could be another regression model, such as a logistic-regression model for a binary response, or indeed any estimate of a population parameter derived from the data. For the mth data set we have estimates $\mathbf{b}_m = (b_{0m}, b_{1m}, \ldots, b_{km}) = \{b_{jm}\}$. Because the imputed missing values differ across the completed data sets so will the estimated regression coefficients. Using standard methods, for example, for least-squares estimates, we can also calculate standard errors for the coefficients in the model fit to the mth completed data set: $[\mathrm{SE}(b_{0m}), \mathrm{SE}(b_{1m}), \ldots, \mathrm{SE}(b_{km})] = \{\mathrm{SE}(b_{jm})\}$.

Rubin (1976) provides simple and very general rules for combining the multiple estimates $\{b_{jm}\}$ of β_j and standard errors $\{\mathrm{SE}(b_{jm})\}$ to produce an overall estimate $\widetilde{b}_j$ and its standard error.

- The overall estimate is particularly straightfoward, just the average of the M estimates:

$$\widetilde{b}_j = \frac{\sum_{m=1}^{M} b_{jm}}{M}$$

- The standard error of the overall estimate has two components, based respectively on within-imputation variation and between-imputation variation, the former reflecting sampling variation and the latter reflecting the additional uncertainty induced due to filling in the missing data:

$$\mathrm{SE}(\widetilde{b}_j) = \sqrt{V_j^W + V_j^B}$$

  where

$$
\begin{aligned}
V_j^W &= \frac{\sum_{m=1}^{M} \mathrm{SE}^2(b_{jm})}{M} \\
V_j^B &= \frac{\sum_{m=1}^{M} \left(b_{jm} - \widetilde{b}_j\right)^2}{M-1}
\end{aligned}
$$

Once $\widetilde{b}_j$ and $\mathrm{SE}(\widetilde{b}_j)$ are computed, standard statistical inference, that is, hypothesis tests and confidence intervals, can be based on the t-distribution, with large-sample degrees of freedom given by

$$
df_j = (M-1)\left(1 + \frac{1}{R_j}\right)
$$

where

$$
R_j = \frac{M+1}{M} \times \frac{V_j^B}{V_j^W}
$$

is the *the relative increase in variance* (or *riv*) of $\widetilde{b}_j$ due to missing data. The *estimated rate of missing information* is $\widetilde{\gamma}_j = R_j/(R_j+1)$. As mentioned, this formula for df is a large-sample result, which doesn't depend on the number of cases n in the data set. Adjustments to degrees of freedom are available for small data sets (see Barnard and Rubin, 1999).

The efficiency of the multiple-imputation estimator depends on the rate of missing information and increases rapidly with the number of multiple imputations M, asymptotically approaching the efficiency of the maximum-likelihood estimator; the relative efficiency of the multiple-imputation estimator is $\mathrm{RE}(\widetilde{b}_j) = M/(M+\gamma_j)$. Even for a high rate of missing information, such as $\gamma = 0.5$, for example, as few as five multiple imputations provide reasonably high relative efficiency, $\mathrm{RE}(\widetilde{b}_j) = 5/(5+0.5) = 0.91$; on the standard-error scale, this is $\sqrt{0.91} = 0.95$. In practice, more multiple imputations are typically used, say $M = 10$, 20, or even more.

2.1 * Multiple-Parameter Inference

The results in this section are based on Schafer (1997, Sec. 4.3.3). Suppose that we want to test the linear hypothesis H_0: $\mathbf{L}\boldsymbol{\beta} = \mathbf{c}$, where $\mathbf{L}$ is a $q \times (k+1)$ hypothesis matrix of rank q containing prespecified constants, and $\mathbf{c}$ is a prespecified $q \times 1$ vector, most often containing zeros (as in Section 5.3.5 of the *R Companion*). Then, for complete data and under H_0, the Wald test statistic

$$
Z_0^2 = (\mathbf{Lb} - \mathbf{c})' \left[\mathbf{L}\widehat{\mathbf{V}}\mathbf{L}'\right]^{-1} (\mathbf{Lb} - \mathbf{c})
$$

follows a large-sample χ^2 distribution with q degrees of freedom. Here, $\widehat{\mathbf{V}}$ is the estimated covariance matrix of the regression coefficients $\mathbf{b}$.

Now imagine that we don't have complete data but have estimates $\mathbf{b}_m$ and $\widehat{\mathbf{V}}_m$ for M data sets completed by multiple imputation of missing values. Let $\mathbf{h}_m = \mathbf{Lb}_m - \mathbf{c}$ be the value of the hypothesis and $\widehat{\mathbf{V}}_m^h = \mathbf{L}\widehat{\mathbf{V}}_m\mathbf{L}'$ be the estimated covariance matrix of the hypothesis for the mth completed data set. Then

$$
\begin{aligned}
\widetilde{\mathbf{h}} &= \frac{1}{M}\sum_{m=1}^{M} \mathbf{h}_m \\
\mathbf{V}_W^h &= \frac{1}{M}\sum_{m=1}^{M} \widehat{\mathbf{V}}_m^h \\
\mathbf{V}_B^h &= \frac{1}{M-1}\sum_{m=1}^{M} \mathbf{h}_m\mathbf{h}_m'
\end{aligned}
$$

Instead of pooling within- and between-imputation variation, $\mathbf{V}_W^h$ and $\mathbf{V}_B^h$, as in single-parameter parameter inference, it turns out to be simpler to base a Wald test statistic on just $\mathbf{V}_W^h$:

$$F_0 = \frac{\widetilde{\mathbf{h}}'\mathbf{V}_W^h\widetilde{\mathbf{h}}}{q(1+R)}$$

where in close analogy to the single-parameter case

$$R = \frac{M+1}{M} \times \frac{\text{trace}\left[\mathbf{V}_B^h\left(\mathbf{V}_W^h\right)^{-1}\right]}{q}$$

F_0 follows a large-sample F-distribution with q numerator degrees of freedom and denominator degrees of freedom equal to

$$4 + [q(M-1) - 4]\left[1 + \frac{1}{R} \times \frac{q(M-1)-2}{q(M-1)}\right] \text{ when } q(M-1) > 4$$

and

$$\tfrac{1}{2}(M-1)(q+1)\left(1 + \frac{1}{R}\right)^2 \text{ when } q(M-1) \leq 4$$

As in the one-parameter case, there are adjustments to the denominator degrees of freedom for small n(see Reiter, 2007).

3 An Example: Infant Mortality, GDP, and Women's Education

In this section, we adapt and elaborate an example from Fox (2016, Sec. 20.4.4) based on social-indicator data from the United Nations for 207 countries in 1998. The data are available in the `UN98` data set in the **carData** package:

```
library("carData")
summary(UN98)
```

```
      region        tfr       contraception  educationMale   educationFemale
  Africa :55   Min.   :1.19   Min.   : 2.0   Min.   : 3.30   Min.   : 2.00
  America:41   1st Qu.:1.95   1st Qu.:21.0   1st Qu.: 9.75   1st Qu.: 9.32
  Asia   :50   Median :3.07   Median :47.0   Median :11.25   Median :11.65
  Europe :44   Mean   :3.53   Mean   :43.4   Mean   :11.41   Mean   :11.28
  Oceania:17   3rd Qu.:4.98   3rd Qu.:64.0   3rd Qu.:13.90   3rd Qu.:13.65
               Max.   :8.00   Max.   :86.0   Max.   :17.20   Max.   :17.80
               NA's   :10     NA's   :63     NA's   :131     NA's   :131
    lifeMale      lifeFemale   infantMortality  GDPperCapita   economicActivityMale
  Min.   :36.0   Min.   :39.1   Min.   :  2.0   Min.   :   36   Min.   :51.2
  1st Qu.:57.4   1st Qu.:59.6   1st Qu.: 12.0   1st Qu.:  442   1st Qu.:72.3
  Median :66.5   Median :72.2   Median : 30.0   Median : 1779   Median :76.8
  Mean   :63.6   Mean   :68.4   Mean   : 43.5   Mean   : 6262   Mean   :76.5
  3rd Qu.:70.9   3rd Qu.:76.4   3rd Qu.: 66.0   3rd Qu.: 7272   3rd Qu.:81.2
  Max.   :77.4   Max.   :82.9   Max.   :169.0   Max.   :42416   Max.   :93.0
  NA's   :11     NA's   :11     NA's   :6       NA's   :10      NA's   :42
  economicActivityFemale illiteracyMale  illiteracyFemale
  Min.   : 1.9           Min.   : 0.20   Min.   : 0.20
  1st Qu.:37.0           1st Qu.: 2.95   1st Qu.: 4.85
  Median :48.4           Median :10.83   Median :20.10
```

```
   Mean   :46.8           Mean   :17.55   Mean   :27.91
   3rd Qu.:56.4           3rd Qu.:27.57   3rd Qu.:48.02
   Max.   :90.6           Max.   :79.10   Max.   :93.40
   NA's   :42             NA's   :47      NA's   :47
```

It's clear that there is a great deal of missing data, especially for the two education variables, `educationFemale` and `educationMale`, the average number of years of education respectively for women and men.

3.1 Preliminaries

We load the **carEx** and **mice** packages, which we'll use below. The **carEx** (**car Ex**tras or **Ex**perimental) package supplements the **car** package, which it loads, along with **carData**.[3]

```
library("carEx")
 Loading required package: car
library("mice")
 Loading required package: lattice

 Attaching package: 'mice'
 The following objects are masked from 'package:base':

     cbind, rbind
```

The `md.pattern()` function in the **mice** package reports the missing-data patterns that occur in the `UN98` data set:

```
md.pattern(UN98, plot=FALSE)
    region infantMortality tfr GDPperCapita lifeMale lifeFemale economicActivityMale
39       1               1   1            1        1          1                     1
58       1               1   1            1        1          1                     1
11       1               1   1            1        1          1                     1
19       1               1   1            1        1          1                     1
15       1               1   1            1        1          1                     1
3        1               1   1            1        1          1                     1
3        1               1   1            1        1          1                     1
5        1               1   1            1        1          1                     1
6        1               1   1            1        1          1                     0
15       1               1   1            1        1          1                     0
6        1               1   1            1        1          1                     0
2        1               1   1            1        1          1                     0
2        1               1   1            1        1          1                     0
4        1               1   1            1        1          1                     0
1        1               1   1            1        0          0                     1
2        1               1   1            0        1          1                     1
3        1               1   1            0        1          1                     0
1        1               1   1            0        0          0                     1
1        1               1   1            0        0          0                     1
2        1               1   0            1        0          0                     1
```

[3]The **carEx** package isn't available on CRAN but can be installed by the command `install.packages("carEx", repos="http://R-Forge.R-project.org")`. Eventually, the functions from the **carEx** package that we use in this appendix will likely migrate to the **car** package on CRAN.

```
2       1               1   0           1        0         0                  1
1       1               1   0           0        1         1                  1
1       1               0   1           1        1         1                  0
1       1               0   0           1        1         1                  1
1       1               0   0           1        0         0                  1
1       1               0   0           1        0         0                  0
2       1               0   0           0        0         0                  0
        0               6  10          10       11        11                 42
   economicActivityFemale illiteracyMale illiteracyFemale contraception
39                      1              1                1             1
58                      1              1                1             1
11                      1              1                1             0
19                      1              1                1             0
15                      1              0                0             1
3                       1              0                0             1
3                       1              0                0             0
5                       1              0                0             0
6                       0              1                1             1
15                      0              1                1             1
6                       0              1                1             0
2                       0              0                0             1
2                       0              0                0             1
4                       0              0                0             0
1                       1              0                0             1
2                       1              1                1             0
3                       0              0                0             0
1                       1              1                1             0
1                       1              0                0             0
2                       1              1                1             0
2                       1              0                0             1
1                       1              1                1             0
1                       0              0                0             0
1                       1              0                0             0
1                       1              0                0             1
1                       0              0                0             0
2                       0              0                0             0
                       42             47               47            63
   educationMale educationFemale
39             1               1   0
58             0               0   2
11             1               1   1
19             0               0   3
15             1               1   2
3              0               0   4
3              1               1   3
5              0               0   5
6              1               1   2
15             0               0   4
6              0               0   5
2              1               1   4
2              0               0   6
4              0               0   7
1              0               0   6
2              0               0   4
3              0               0   8
```

```
 1              0              0   6
 1              0              0   8
 2              0              0   6
 2              0              0   7
 1              0              0   5
 1              0              0   8
 1              0              0   7
 1              0              0   8
 1              0              0  11
 2              0              0  12
              131            131 551
```

Ones in the output represent observed data and zeroes missing data. Thus, for example, only 39 of the 207 cases are completely observed, and 58 are missing just `educationMale` and `educationFemale`. The last column counts the number of missing values in each pattern, and the last row the number of missing values in each variable. It's apparent that several paris of variables for males and females (e.g., `lifeFemale` and `lifeMale`) are always missing or observed together.

It's our object to regress `infantMortality` (infant deaths per 1000 live births) on `GDPperCapita` (in U. S. dollars), `educationFemale`, and `region`, but a complete-case analysis includes only $207 - 131 = 76$ of the 207 countries:

```
mod.un <- lm(log(infantMortality) ~ region + log(GDPperCapita) + educationFemale,
             data=UN98)
S(mod.un)
```

```
 Call: lm(formula = log(infantMortality) ~ region + log(GDPperCapita) +
          educationFemale, data = UN98)

 Coefficients:
                   Estimate Std. Error t value Pr(>|t|)
 (Intercept)         6.7078     0.2509   26.73  < 2e-16
 regionAmerica      -0.4032     0.1697   -2.38  0.02029
 regionAsia         -0.3464     0.1649   -2.10  0.03933
 regionEurope       -0.7507     0.1853   -4.05  0.00013
 regionOceania      -0.2866     0.2922   -0.98  0.33004
 log(GDPperCapita)  -0.2764     0.0539   -5.13  2.5e-06
 educationFemale    -0.0921     0.0281   -3.28  0.00163

 Residual standard deviation: 0.399 on 69 degrees of freedom
   (131 observations deleted due to missingness)
 Multiple R-squared: 0.86
 F-statistic: 70.5 on 6 and 69 DF,  p-value: <2e-16
    AIC    BIC
  84.86 103.51
```

This regression model reflects a preliminary examination of the data that motivated the log transformation of both the response `infantMortality` and the predictor `GDPperCapita`. Diagnostics applied to the fitted model suggest that the specification is adequate (although there is a suggestion of a nonlinear relationship of log infant mortality to women's education and perhaps to log GDP; see Figure 1) but that there is an influential case, Iraq (Figure 2):

```
crPlots(mod.un, smooth=list(span=0.9)) # large span for 76 cases
avPlots(mod.un)
outlierTest(mod.un)
```

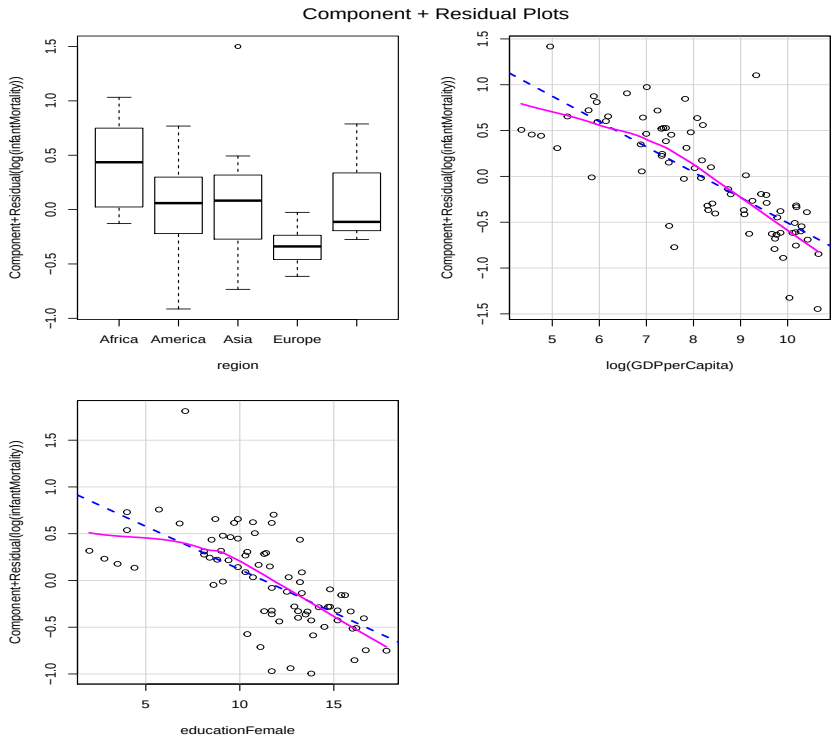

Figure 1: Component-plus-residual plots for the preliminary model fit to complete cases in the UN data.

```
      rstudent unadjusted p-value Bonferonni p
Iraq    4.3501          4.675e-05     0.003553
```

We refit the model dedicating a dummy regressor to Iraq (and invite the reader to redo the diagnostics for the updated model):

```
UN98$Iraq <- rownames(UN98) == "Iraq"
mod.un.2 <- update(mod.un, . ~ . + Iraq, data=UN98)
compareCoefs(mod.un, mod.un.2)
```

```
Calls:
1: lm(formula = log(infantMortality) ~ region + log(GDPperCapita) +
  educationFemale, data = UN98)
2: lm(formula = log(infantMortality) ~ region + log(GDPperCapita) +
  educationFemale + Iraq, data = UN98)

                  Model 1 Model 2
(Intercept)         6.708   6.830
SE                  0.251   0.225

regionAmerica      -0.403  -0.443
SE                  0.170   0.151

regionAsia         -0.346  -0.397
SE                  0.165   0.147
```

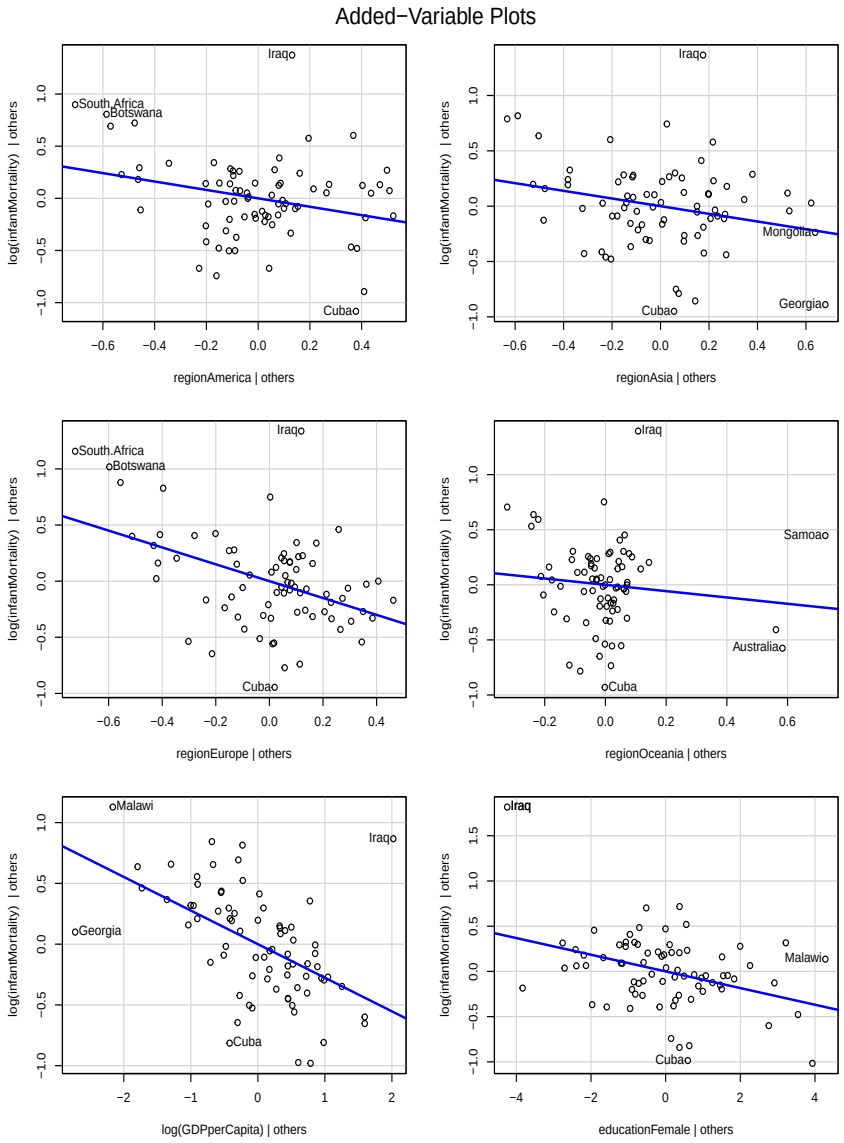

Figure 2: Added-variable plots for the preliminary model fit to complete cases in the UN data.

```
regionEurope       -0.751  -0.794
SE                  0.185   0.165

regionOceania      -0.287  -0.384
SE                  0.292   0.261

log(GDPperCapita) -0.2764 -0.3382
SE                 0.0539  0.0501

educationFemale   -0.0921 -0.0568
SE                 0.0281  0.0263

IraqTRUE                    1.681
SE                          0.386
```

It's generally desirable to take an "inclusive" approach to imputing missing data (see, e.g., Collins et al., 2001), using a richer imputation model than the eventual analytic model that we intend to fit to the multiply-imputed data. The object is to make the MAR assumption that underlies multiple imputation plausible by including variables in the imputation model that are likely related to the missing data and that should help to predict missingness. To this end, we include `tfr` (the total fertility rate, children per woman), `contraception` (percentage of married women using any method of contraception), `lifeFemale` (expectation of life at birth, in years), `economicActivityFemale` (percentage economically active), and `illiteracyFemale` (percentage 15 years and older who are illiterate):

```
UN2 <- UN98[, c(1, 2, 3, 5, 7, 8, 9, 11, 13)]
names(UN2)
```

```
 [1] "region"                "tfr"                   "contraception"
 [4] "educationFemale"       "lifeFemale"            "infantMortality"
 [7] "GDPperCapita"          "economicActivityFemale" "illiteracyFemale"
```

A scatterplot matrix (Figure 3) of the numeric variables in the reduced data set reveals that several have skewed distributions.

```
scatterplotMatrix(UN2[, -1], smooth=list(span=0.9)) # dropping region
```

Although the imputation method that we'll use doesn't require normal data or linear relationships among variables, we'll likely get more precise results if we can transform these variables towards normality; because the values of `lifeFemale` are far from zero, we use an arbitrary start of -30 for this variable:

```
UN2$lifeFemale <- UN2$lifeFemale - 30
summary(p<- powerTransform(UN2[, -1]))
```

```
 bcPower Transformations to Multinormality
                        Est Power Rounded Pwr Wald Lwr Bnd Wald Upr Bnd
 tfr                       0.6101        1.00       0.1335       1.0867
 contraception             1.0087        1.00       0.4279       1.5895
 educationFemale           1.4704        1.00       0.8749       2.0659
 lifeFemale                3.1525        3.15       2.0626       4.2423
 infantMortality           0.1624        0.00      -0.0681       0.3928
 GDPperCapita             -0.0504        0.00      -0.2645       0.1636
 economicActivityFemale    0.7235        1.00       0.2297       1.2174
 illiteracyFemale          0.3358        0.50       0.1342       0.5373
```

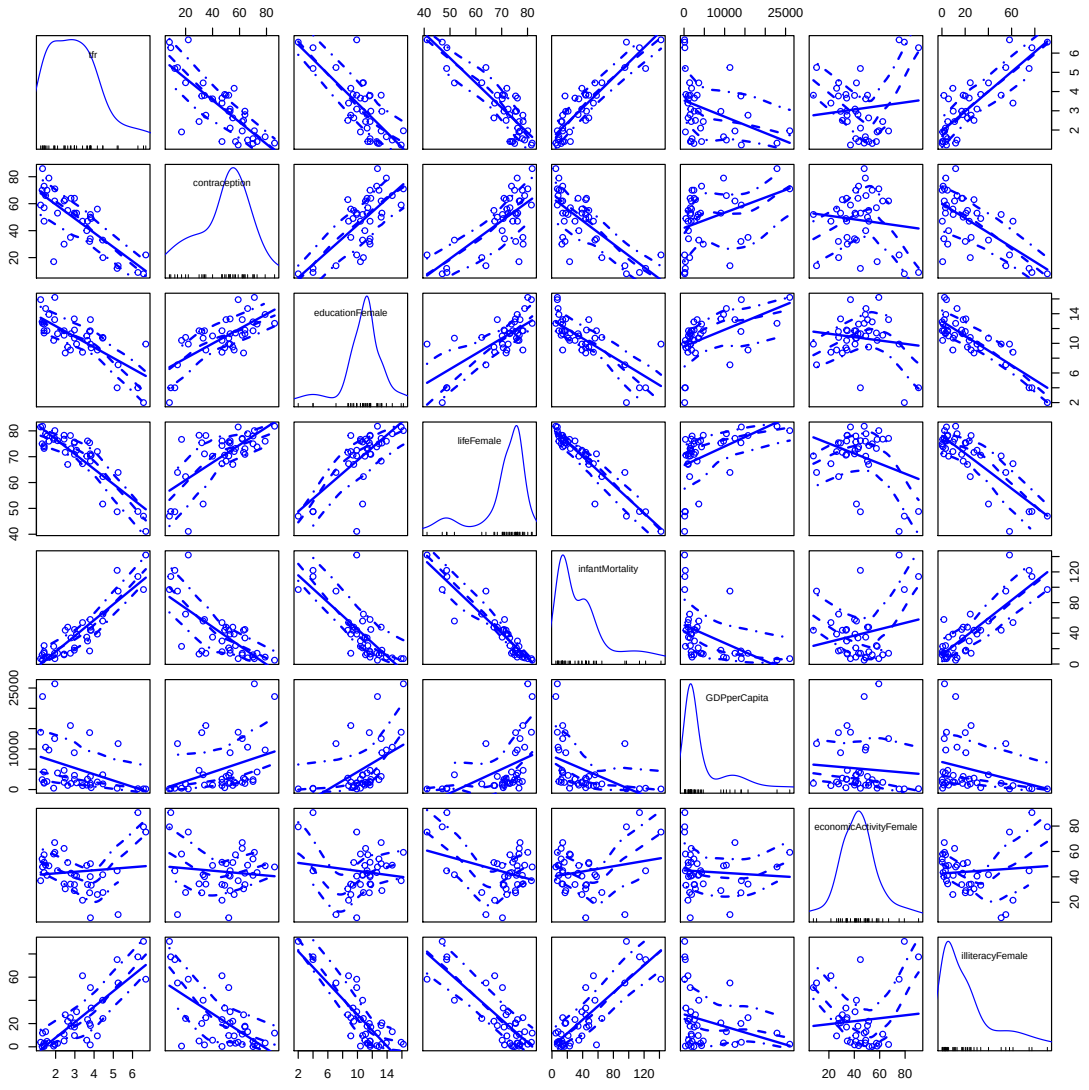

Figure 3: Scatterplot matrix for the retained numeric variables in the UN data.

```
 Likelihood ratio test that transformation parameters are equal to 0
  (all log transformations)
                                       LRT df   pval
 LR test, lambda = (0 0 0 0 0 0 0 0) 138.97  8 <2e-16

 Likelihood ratio test that no transformations are needed
                                       LRT df   pval
 LR test, lambda = (1 1 1 1 1 1 1 1) 144.97  8 <2e-16
```

These results suggest the log transformation of `infantMortality` and `GDPperCapita`, the square-root transformation of `illiteracyFemale`, essentially the cube of `lifeFemale - 30`, and leaving the other variables alone. Using the rounded transformations:

```
UN.t <- as.data.frame(mapply(basicPower, UN2[, -1], p$roundlam))
scatterplotMatrix(UN.t, smooth=list(span=0.9))
```

The transformed data, graphed in Figure 4, are better-behaved: The distributions of the variables are more nearly symmetric and most of the bivariate regressions are close to linear. The variable `economicActivityFemale` is clearly an exception, in that it has approximately quadratic relationships to many of the other variables in the data set.

3.2 Obtaining Multiple Imputations With mice

To further prepare the data for multiple imputation, we put back `region` and `Iraq`, and add a squared term for `educationFemale` because of the quadratic relationships we noted in the scatterplot matrix of the transformed numeric variables:

```
UN.t$region <- UN2$region
UN.t$Iraq <- rownames(UN2) == "Iraq"
UN.t$eaf2 <- UN.t$economicActivityFemale^2
```

As a general matter, it can be important to preserve features of the data that are reflected in the analytic model, such as nonlinear relationships and interactions, and it can be advantageous to make the imputation model more accurate.

Our next step is to use the `mice()` function in the **mice** package to generate 10 completed data sets:

```
> sample(1e6, 1)
 423249


system.time(UN.imps <- mice(UN.t, m=10, maxit=20, printFlag=FALSE, seed=423249))
    user  system elapsed
    7.21    0.00    7.22
```

The first argument to `mice()` is the observed data set; the argument `m=10` generates 10 multiple imputations of the missing values in the observed data; `maxit=20` specifies 20 iterations of the MICE algorithm for each set of imputations; `printFlag=FALSE` suppresses printing the iteration history; and `seed=423249` sets the seed for R's random-number generator, making the results reproducible. As is our habit, we randomly sample and then note the seed. The `mice()` function returns an object of class `"mids"` ("multiply-imputed data sets").

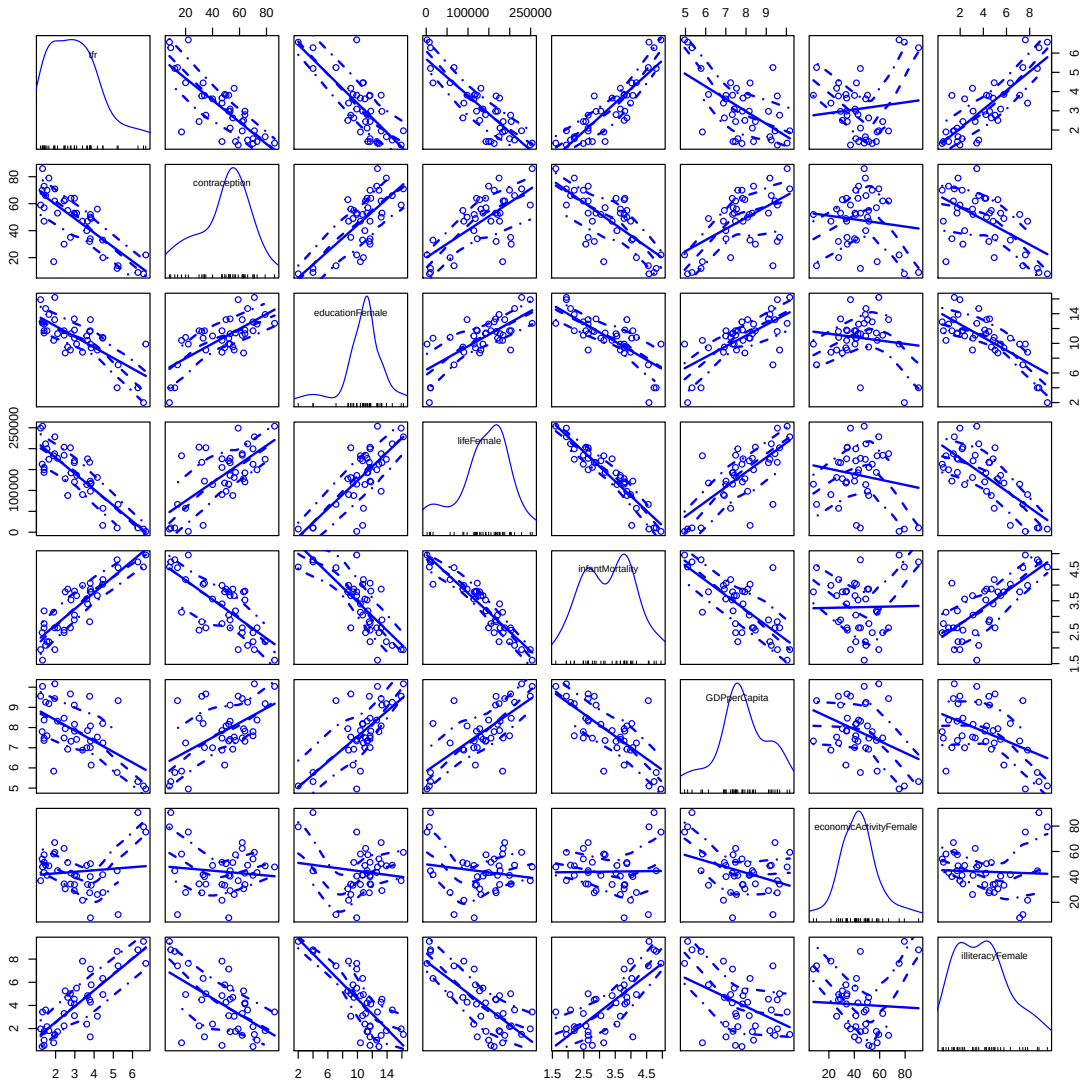

Figure 4: Scatterplot matrix for the transformed numeric variables in the UN data.

By default, `mice()` uses a method called *predictive mean matching*—essentially a cross between linear least-squares regression and hot-deck imputation—to fill in missing values for numeric variables, binomial logistic regression for dichotomous factors, and multinomial logistic regression for polytomous factors; there are no factors with missing data in the UN data set. These defaults can be changed via the `meth` argument to `mice()` (see `help("mice")`, van Buuren and Groothuis-Oudshoorn, 2011, and van Buuren, 2018).

It's generally a good idea to check the MICE imputations for convergence. One way to do this is to plot the average and standard deviation of the imputed data for each variable with missing data as a function of imputation and iteration. If the MICE imputations have converged, then these plots should level out, and the traces for different imputations should "mix," that is, cross one-another. Trace plots are generated by the `plot()` method for `"mids"` objects (see `help("plot.mids")`), and appear in Figure 5:

```
plot(UN.imps, layout=c(4, 5))
```

The results aren't terrible, but there's some indication that the imputations aren't mixing well, especially for the squared variable `eaf2` = `economicActivityFemale`^{2} and for `economicActivityFemale`. As we specified the imputation model, the imputed values of `eaf2` aren't constrained to be equal to the squares of `economicActivityFemale`.

This kind of inconsistency in the definition of derived variables doesn't generally invalidate their use in fitting an analytic model to multiply-imputed completed data sets, but it can lead to convergence problems in the MICE algorithm. We can address the issue by constraining derived variables, here `eaf2`, to be consistent with the variables from which they're defined. We also up the number of imputations to 20 and iterations to 50 (with the results shown in Figure 6), and (although it's not necessary to do so) we use the same seed as before for the random-number generator:

```
meth <- make.method(UN.t)
meth["eaf2"] <- '~ I(economicActivityFemale^2)'
system.time(UN.imps.2 <- mice(UN.t, m=20, maxit=50, method=meth,
                              printFlag=FALSE, seed=423249))
```

```
    user  system elapsed
   32.59    0.00   32.59
```

```
plot(UN.imps.2, layout=c(4, 5))
```

The imputation traces appear to mix better now and have all levelled out by 50 iterations.

3.3 Fitting the Analytic Model to the Multiply-Imputed Data

The **mice** package includes a `"mids"` method for the generic `with()` function, allowing us to conventiently fit a statistical model to each completed data set, producing an object of (primary) class `"mira"` ("multiply-imputed repeated analyses"):

```
un.mods <- with(UN.imps.2, {
    infantMortality <- exp(infantMortality)
    GDPperCapita <- exp(GDPperCapita)
    lm(log(infantMortality) ~ region + log(GDPperCapita) + educationFemale + Iraq)
})
```

As illustrated, we can also define variables in the completed data sets in the call to `with()`, in this case (although it isn't strictly necessary) undoing the log-transformations of `infantMortality` and `GDPperCapita`, allowing us to specify the analytic model exactly as we did for the complete cases (less the `data` argument).

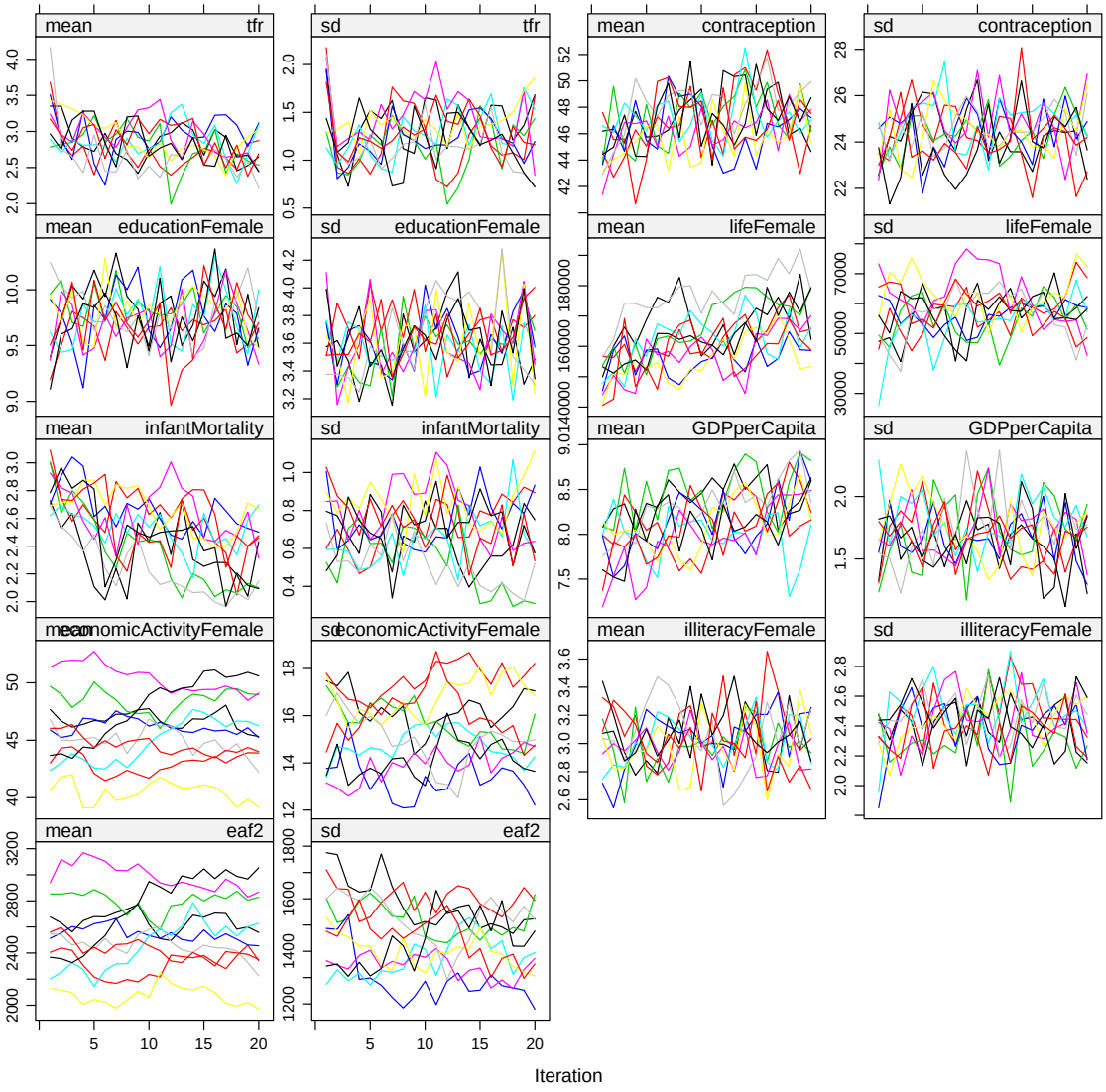

Figure 5: Trace plots for 10 multiple imputations of 20 iterations each applied to the transformed UN data.

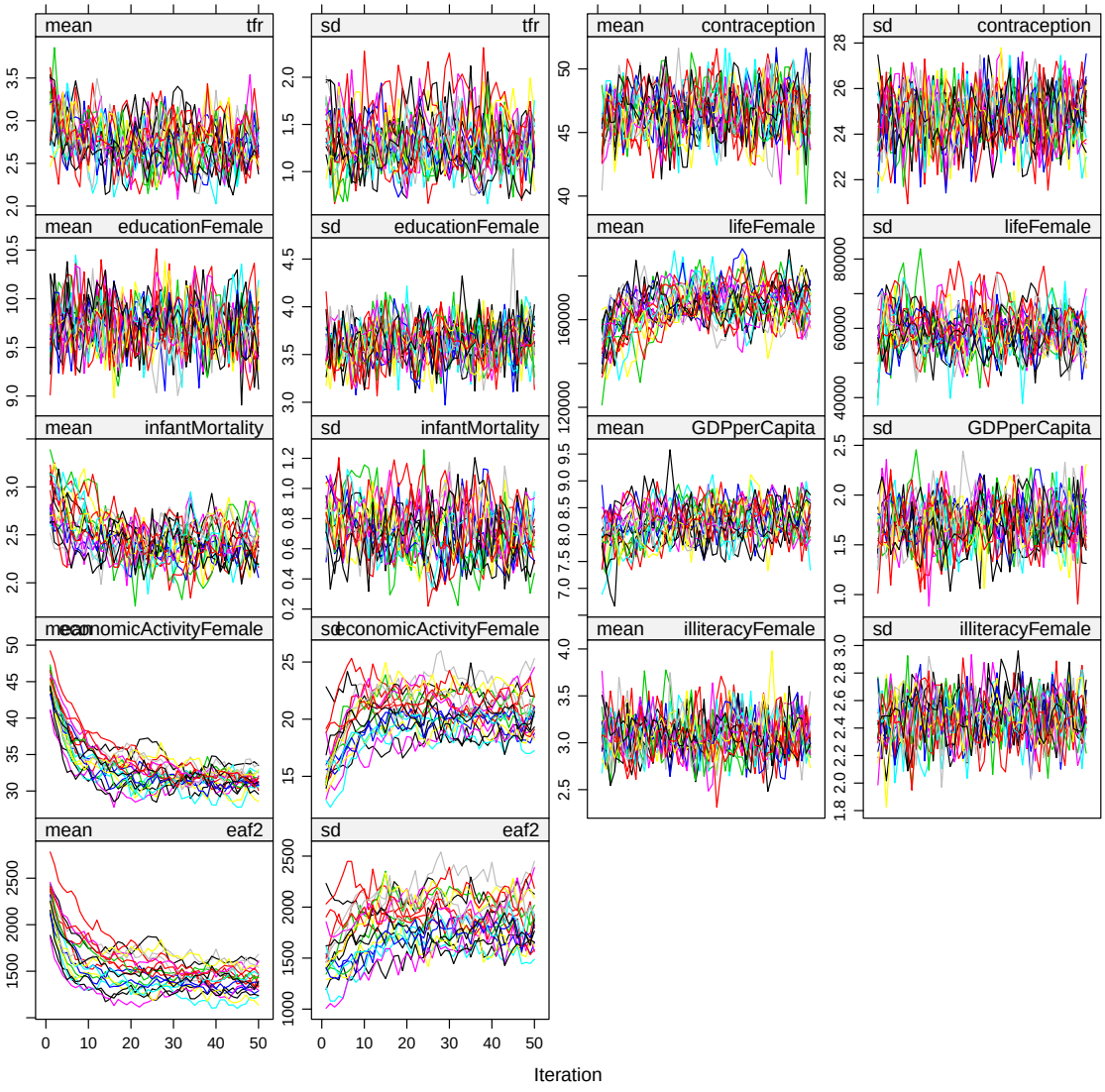

Figure 6: Trace plots for 20 multiple imputations of 50 iterations each applied to the transformed UN data, constraining `eaf2` to be consistent with `economicActivityFemale`.

The `pool()` function in **mice** applies Rubin's rules to the models fit in parallel to the $M = 20$ completed data sets:

```
(un.p <- pool(un.mods))
```

```
 Class: mipo    m = 20
                    estimate       ubar          b          t dfcom      df      riv
 (Intercept)        6.487314 0.02726510 0.0089767 0.0366906   199  97.057 0.345699
 regionAmerica     -0.439707 0.01232448 0.0030291 0.0155051   199 116.281 0.258070
 regionAsia        -0.450915 0.00940302 0.0014835 0.0109607   199 143.286 0.165653
 regionEurope      -0.898488 0.01422884 0.0047972 0.0192659   199  95.513 0.354006
 regionOceania     -0.628460 0.02113247 0.0140349 0.0358691   199  57.148 0.697345
 log(GDPperCapita) -0.252927 0.00121444 0.0018205 0.0031260   199  30.539 1.574018
 educationFemale   -0.082236 0.00029573 0.0007909 0.0011262   199  20.857 2.808082
 IraqTRUE           1.461989 0.23528218 0.0219610 0.2583413   199 166.886 0.098006
                     lambda     fmi
 (Intercept)       0.256892 0.27175
 regionAmerica     0.205132 0.21846
 regionAsia        0.142112 0.15384
 regionEurope      0.261451 0.27644
 regionOceania     0.410845 0.43044
 log(GDPperCapita) 0.611502 0.63467
 educationFemale   0.737401 0.75942
 IraqTRUE          0.089258 0.09998
```

```
summary(un.p)
```

```
                    estimate std.error statistic      df    p.value
 (Intercept)        6.487314  0.191548   33.8678  97.057 0.0000e+00
 regionAmerica     -0.439707  0.124519   -3.5312 116.281 5.3487e-04
 regionAsia        -0.450915  0.104693   -4.3070 143.286 2.8166e-05
 regionEurope      -0.898488  0.138802   -6.4732  95.513 1.0264e-09
 regionOceania     -0.628460  0.189391   -3.3183  57.148 1.1118e-03
 log(GDPperCapita) -0.252927  0.055911   -4.5238  30.539 1.1488e-05
 educationFemale   -0.082236  0.033559   -2.4505  20.857 1.5297e-02
 IraqTRUE           1.461989  0.508273    2.8764 166.886 4.5479e-03
```

The columns labelled `ubar`, `b`, and `t` are respectively the within-imputation, between-imputation, and pooled coefficient variance; `dfcom` are the degrees of freedom for the complete data and `df` the adjusted *df* for each coefficient; `riv` is the relative increase in variance due to missing data; `lambda` is the proportion of total coefficient variance due to missing data; and `fmi` is the proportion (fraction) of missing information.

The **carEx** package includes `Anova()`, `linearHypothesis()`, `deltaMethod()`, `coef()`, and `vcov()` methods for `"mira"` objects; for example:

```
Anova(un.mods)
```

```
 Analysis of Deviance Table (Type II tests)

 Response: log(infantMortality)
 Based on 20 multiple imputations

                       F num df den df missing info   riv  Pr(>F)
 region            11.48      4  172.1        0.224 0.289 2.8e-08
 log(GDPperCapita) 20.46      1   34.0        0.612 1.574 7.0e-05
 educationFemale    6.01      1   25.9        0.737 2.808  0.0213
 Iraq               8.27      1  175.0        0.089 0.098  0.0045
```

Because the hypotheses that it tests may have more than 1 numerator *df*, `Anova()` uses the approach in Reiter (2007) to compute denominator *df*. In contrast, `pool()` uses the approach in Barnard and Rubin (1999), which is appropriate only for *t*-tests of individual coefficients.

We can use the `compareCoefs()` function in the **car** package to compare the fits of the complete-data and multiple-imputation models:

```
compareCoefs(mod.un.2, un.mods)
```

```
Warning in compareCoefs(mod.un.2, un.mods): models to be compared are of different classes
```

```
Calls:
1: lm(formula = log(infantMortality) ~ region + log(GDPperCapita) +
  educationFemale + Iraq, data = UN98)
2: with.mids(data = UN.imps.2, expr = { infantMortality <- exp(infantMortality)
  GDPperCapita <- exp(GDPperCapita) lm(log(infantMortality) ~ region +
  log(GDPperCapita) + educationFemale + Iraq)})
```

	Model 1	Model 2
(Intercept)	6.830	6.487
SE	0.225	0.192
regionAmerica	-0.443	-0.440
SE	0.151	0.125
regionAsia	-0.397	-0.451
SE	0.147	0.105
regionEurope	-0.794	-0.898
SE	0.165	0.139
regionOceania	-0.384	-0.628
SE	0.261	0.189
log(GDPperCapita)	-0.3382	-0.2529
SE	0.0501	0.0559
educationFemale	-0.0568	-0.0822
SE	0.0263	0.0336
IraqTRUE	1.681	1.462
SE	0.386	0.508

The two sets of results aren't radically different, but they do differ in detail. In particular the magnitude of the estimated `log(GDPperCapita)` coefficient is about two standard errors smaller in the multiple-imputation analysis, and that of the `educationFemale` coefficient is about one standard error larger.

Finally, a word about regression diagnostics: In analyzing the complete data, we decided to treat Iraq as a special case, and we carried over this decision to our multiple-imputation analysis. As it turns out, the coefficient of the dummy regressor for Iraq is quite large in both analyses. But there are many more counties in the multiple-imputation analysis than in the complete-case analysis, and it's possible that some of these also require special treatment. It's not obvious how to perform unusual-case and other regression diagnostics for multiply-imputed data. One simple, if ad-hoc, approach is to use the standard diagnostics to examine a few of the models fit to the completed data sets. For our example, the model objects are stored in the list `un.mods$analyses`. We invite the reader to perform this analysis, starting, for example, with `avPlots(un.mods$analyses[[1]])`.

4 Complementary Reading and References

- Much of the material for this appendix is derived from Fox (2016, Chap. 20), which deals more generally with missing data in regression models and includes an explanation of multiple imputation.
- Little and Rubin (2002) is an accessible and wide-ranging treatment of methods for dealing with missing data by authors who contributed fundamentally to the field.
- Allison (2002) presents a brief, high-quality overview of statistical estimation in the presence of missing data, including a discussion of muliple imputation.
- Schafer (1997) is a more advanced treatment of the subject.
- van Buuren (2018) focuses on multiple imputation for missing data and is the definitive source for the **mice** package.

References

Allison, P. D. (2002). *Missing Data*. Thousand Oaks CA.

Barnard, J. and Rubin, D. B. (1999). Small-sample degrees of freedom with multiple imputation. *Biometrika*, 86:948–955.

Collins, L. M., Schafer, J. L., and Kam, C.-M. (2001). A comparison of inclusive and restrictive strategies in modern missing data procedures. *Psychological Methods*, 6:330–351.

Fox, J. (2016). *Applied Regression Analysis and Generalized Linear Models*. Sage, Thousand Oaks CA, third edition.

Heckman, J. J. (1974). Shadow prices, market wages, and labor supply. *Econometrica*, 42:679–693.

Heckman, J. J. (1976). The common structure of statistical models of truncation, sample selection and limited dependent variables and a simple estimator for such models. *Annals of Economic and Social Measurement*, 5:475–492.

Heckman, J. J. and Robb, R. (1986). Alternative methods for solving the problem of selection bias in evaluating the impact of treatments on outcomes. In Wainer, H., editor, *Drawing Inferences From Self-Selected Samples*, pages 63–107. Springer, New York.

Honaker, J., King, G., and Blackwell, M. (2011). Amelia II: A program for missing data. *Journal of Statistical Software*, 45(7):1–47.

Little, R. J. A. and Rubin, D. B. (2002). *Statistical Analysis with Missing Data*. Hoboken NJ.

Reiter, J. P. (2007). Small-sample degrees of freedom for multi-component significance tests with multiple imputation for missing data. *Biometrika*, 94:502–508.

Rubin, D. B. (1976). Inference and missing data (with discusison). *Biometrika*, 63:581–592.

Schafer, J. L. (1997). *Analysis of Incomplete Multivariate Data*. Chapman and Hall, London.

Schafer, J. L. (2012). *cat: Analysis of categorical-variable datasets with missing values*. R package version 0.0-6.5, ported to R by Ted Harding and Fernando Tusell.

Schafer, J. L. (2013). *norm: Analysis of multivariate normal datasets with missing values*. R package version 1.0-9.5; ported to R by Alvaro A. Novo.

Schafer, J. L. (2017). *mix: Estimation/Multiple Imputation for Mixed Categorical and Continuous Data*. R package version 1.0-10.

Su, Y.-S., Gelman, A., Hill, J., and Yajima, M. (2011). Multiple imputation with diagnostics (mi) in R: Opening windows into the black box. *Journal of Statistical Software*, 45(2):1–31.

van Buuren, S. (2018). *Flexible imputation of missing data*. CRC/Chapman and Hall, Boca Raton FL, second edition.

van Buuren, S. and Groothuis-Oudshoorn, K. (2011). mice: Multivariate imputation by chained equations in R. *Journal of Statistical Software*, 45(3):1–67.