COLUMBIA UNIVERSITY
IN THE CITY OF NEW YORK

STAT 4224/5224

Bayesian Statistics

Dobrin Marchev

Recall: Multivariate Normal Model

Assume that we have multivariate observations

$$\boldsymbol{X}_1, \ldots, \boldsymbol{X}_n \mid \boldsymbol{\theta}, \boldsymbol{\Sigma} \sim \mathrm{N}_p(\boldsymbol{\theta}, \boldsymbol{\Sigma})$$

If there are no missing values, then the likelihood is

$$f(\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n | \boldsymbol{\theta}, \boldsymbol{\Sigma})$$
$$= \prod_{i=1}^{n} (2\pi)^{-\frac{p}{2}} (\det \boldsymbol{\Sigma})^{-½} e^{-½(\boldsymbol{x}_i - \boldsymbol{\theta})' \boldsymbol{\Sigma}^{-1} (\boldsymbol{x}_i - \boldsymbol{\theta})}$$
$$= (2\pi)^{-\frac{np}{2}} (\det \boldsymbol{\Sigma})^{-\frac{n}{2}} e^{-½ \sum_{i=1}^{n} (\boldsymbol{x}_i - \boldsymbol{\theta})' \Sigma^{-1} (\boldsymbol{x}_i - \boldsymbol{\theta})}$$

Q: How do we compute $f(\boldsymbol{x}_i | \boldsymbol{\theta}, \boldsymbol{\Sigma}) =$
$(2\pi)^{-\frac{p}{2}} (\det \boldsymbol{\Sigma})^{-½} e^{-½(\boldsymbol{x}_i - \boldsymbol{\theta})' \boldsymbol{\Sigma}^{-1} (\boldsymbol{x}_i - \boldsymbol{\theta})}$ when the $\boldsymbol{x}_i$ is missing?

A: Impute a value.

Note: The worst idea is to impute the average!

Algorithm

- Let $\boldsymbol{X}$ be the $n \times p$ matrix of all data, both observed and missing.
- Let $\boldsymbol{R}$ be the $n \times p$ missing pattern matrix as defined before.
- $\boldsymbol{X}_{\text{obs}} = \{x_{ij}: r_{ij} = 1\}$ is the observed data.
- $\boldsymbol{X}_{\text{mis}} = \{x_{ij}: r_{ij} = 0\}$ is the unobserved or missing data.
- Note that $\boldsymbol{X}_{\text{mis}}$ has to be treated as unknown parameter!
- Goal: Obtain samples from the posterior distribution
$$f(\boldsymbol{\theta}, \boldsymbol{\Sigma}, \boldsymbol{X}_{mis} | \boldsymbol{X}_{obs})$$
- Solution: Run a three-stage Gibbs sample that iterates between
$$f(\boldsymbol{\theta} | \boldsymbol{X}_{obs}, \boldsymbol{\Sigma}, \boldsymbol{X}_{mis})$$
$$f(\boldsymbol{\Sigma} | \boldsymbol{X}_{obs}, \boldsymbol{\theta}, \boldsymbol{X}_{mis})$$
$$f(\boldsymbol{X}_{mis} | \boldsymbol{X}_{obs}, \boldsymbol{\theta}, \boldsymbol{\Sigma})$$
- Note that in steps 1 and 2, the fixed value of $\boldsymbol{X}_{\text{obs}}$ combines with the current value of $\boldsymbol{X}_{\text{mis}}$ to form a current version of a complete data matrix $\boldsymbol{X}$ having no missing values.

Sampling from $f(\boldsymbol{X}_{mis}|\boldsymbol{X}_{obs}, \boldsymbol{\theta}, \boldsymbol{\Sigma})$

Note that

$$f(\boldsymbol{X}_{mis}|\boldsymbol{X}_{obs}, \boldsymbol{\theta}, \boldsymbol{\Sigma}) \propto f(\boldsymbol{X}_{mis}, \boldsymbol{X}_{obs}|\boldsymbol{\theta}, \boldsymbol{\Sigma})$$
$$= \prod_{\boldsymbol{i}=\boldsymbol{1}}^{\boldsymbol{n}} f\left(\boldsymbol{x}_{i,mis}, \boldsymbol{x}_{i,obs}\middle|\boldsymbol{\theta}, \boldsymbol{\Sigma}\right) \propto \prod_{\boldsymbol{i}=\boldsymbol{1}}^{\boldsymbol{n}} f\left(\boldsymbol{x}_{i,mis}\middle|\boldsymbol{x}_{i,obs}, \boldsymbol{\theta}, \boldsymbol{\Sigma}\right)$$

so for each *i* we need to sample the missing elements of the data vector conditional on the observed elements.

Recall a result about multivariate normal distributions:

Let $\boldsymbol{x} \sim \mathrm{N}_p(\boldsymbol{\theta}, \boldsymbol{\Sigma})$, $\boldsymbol{a} \subset \{1, \dots, p\}, \boldsymbol{b} = \{1, \dots, p\} \backslash \boldsymbol{a}$. Then:

$\boldsymbol{x}_b \mid \boldsymbol{x}_a \sim \mathrm{N}(\boldsymbol{\theta}_{\boldsymbol{b}|\boldsymbol{a}}, \boldsymbol{\Sigma}_{\boldsymbol{b}|\boldsymbol{a}})$, where

$$\boldsymbol{\theta}_{\boldsymbol{b}|\boldsymbol{a}} = \boldsymbol{\theta}_{\boldsymbol{b}} + \boldsymbol{\Sigma}_{\boldsymbol{b},\boldsymbol{a}}\left(\boldsymbol{\Sigma}_{\boldsymbol{a},\boldsymbol{a}}\right)^{-1}(\boldsymbol{x}_{\boldsymbol{a}} - \boldsymbol{\theta}_{\boldsymbol{a}})$$
$$\boldsymbol{\Sigma}_{\boldsymbol{b}|\boldsymbol{a}} = \boldsymbol{\Sigma}_{\boldsymbol{b},\boldsymbol{b}} - \boldsymbol{\Sigma}_{\boldsymbol{b},\boldsymbol{a}}\left(\boldsymbol{\Sigma}_{\boldsymbol{a},\boldsymbol{a}}\right)^{-1}\boldsymbol{\Sigma}_{\boldsymbol{a},\boldsymbol{b}}$$

In the above $\boldsymbol{\Sigma}_{\boldsymbol{a},\boldsymbol{b}}$ refers to the submatrix made up of the elements that are in rows $\boldsymbol{a}$ and columns $\boldsymbol{b}$ of $\boldsymbol{\Sigma}$.

Example 1 (p. 115)

Four variables are taken from a dataset involving health-related measurements on 200 women of Pima Indian heritage living near Phoenix, Arizona (Smith et al, 1988). The four variables are glu (blood plasma glucose concentration), bp (diastolic blood pressure), skin (skin fold thickness) and bmi (body mass index).

The prior for the mean is $\boldsymbol{\mu}_0$ = (120, 64, 26, 26) and was obtained from national averages.

See R code.

Missing Data

Multiple Imputation
using the `mi` package in `R`

Example 2: mi

(1) Load the data

```
> data(nlsyV, package = "mi")
```

This extracts the nlsyV dataset from the mi package. This dataset pertains to children and their families in the United States. Variables are:

- ppvtr.36 -a numeric vector with data on the Peabody Picture Vocabulary Test administered at 36 months
- first - indicator for whether child was first-born
- b.marr - indicator if mother was married when child was born
- income - numeric data on family income in year after the child was born
- momage - a numeric vector with data on the age of the mother when the child was born
- momed - educational status of mother when child was born (1 = less than high school, 2 = high school graduate, 3 = some college, 4 = college graduate)
- romrace - race of mother (1 = black, 2 = Hispanic, 3 = white)

(2) Create a `missing_data` object, then look at the data and the missing data patterns

This class is similar to a `data.frame`, but is customized for the situation in which variables with missing data are being modeled for multiple imputation.

```
mdf = missing_data.frame(nlsyV)
summary(mdf)
image(mdf)
hist(mdf)
```

(3) Examine defaults to see if they make sense

```
> show(mdf)
```

```
                        type missing method  model
ppvtr.36          continuous      75    ppd linear
first                 binary       0   <NA>   <NA>
b.marr                binary      12    ppd  logit
income            continuous      82    ppd linear
momage            continuous       0   <NA>   <NA>
momed    ordered-categorical      40    ppd ologit
momrace  ordered-categorical     117    ppd ologit

              family     link transformation
ppvtr.36    gaussian identity    standardize
first           <NA>     <NA>           <NA>
b.marr      binomial    logit           <NA>
income      gaussian identity    standardize
momage          <NA>     <NA>    standardize
momed    multinomial    logit           <NA>
momrace  multinomial    logit           <NA>
```

Ordered and unordered categorical variables

Ordered and unordered categorical variables require special attention

- If such a variable has any missing data it should be included in your dataset as a single variable with multiple levels
- If these variables are coded as "factors" in R then the `mi` program will understand that they are categorical (you can convert using the as.factor() command)
- Otherwise, you can explicitly change the status using the change() command in the `mi` package
- unordered categoricals will be imputed using multinomial logit
- ordered categoricals will be imputed using ordered logit

(4) Make changes to imputation models

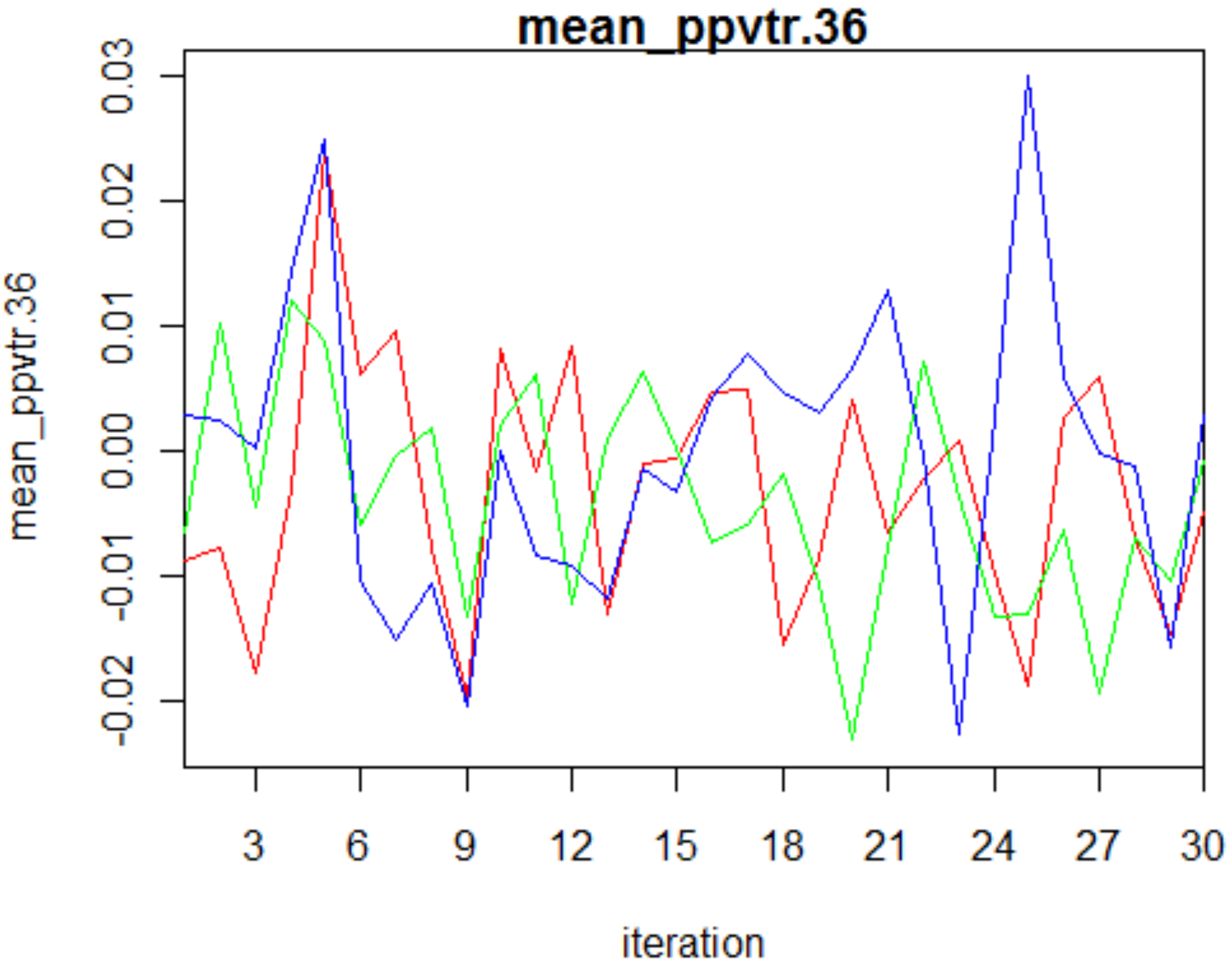
```
> mdf <- change(mdf, y = c"momrace"), what = "type",
  to = "un")
> show(mdf)
```

```
                          type missing method  model
ppvtr.36            continuous      75    ppd linear
first                   binary       0   <NA>   <NA>
b.marr                  binary      12    ppd  logit
income              continuous      82    ppd linear
momage              continuous       0   <NA>   <NA>
momed      ordered-categorical      40    ppd ologit
momrace  unordered-categorical     117    ppd mlogit
```

(5) Impute until converged

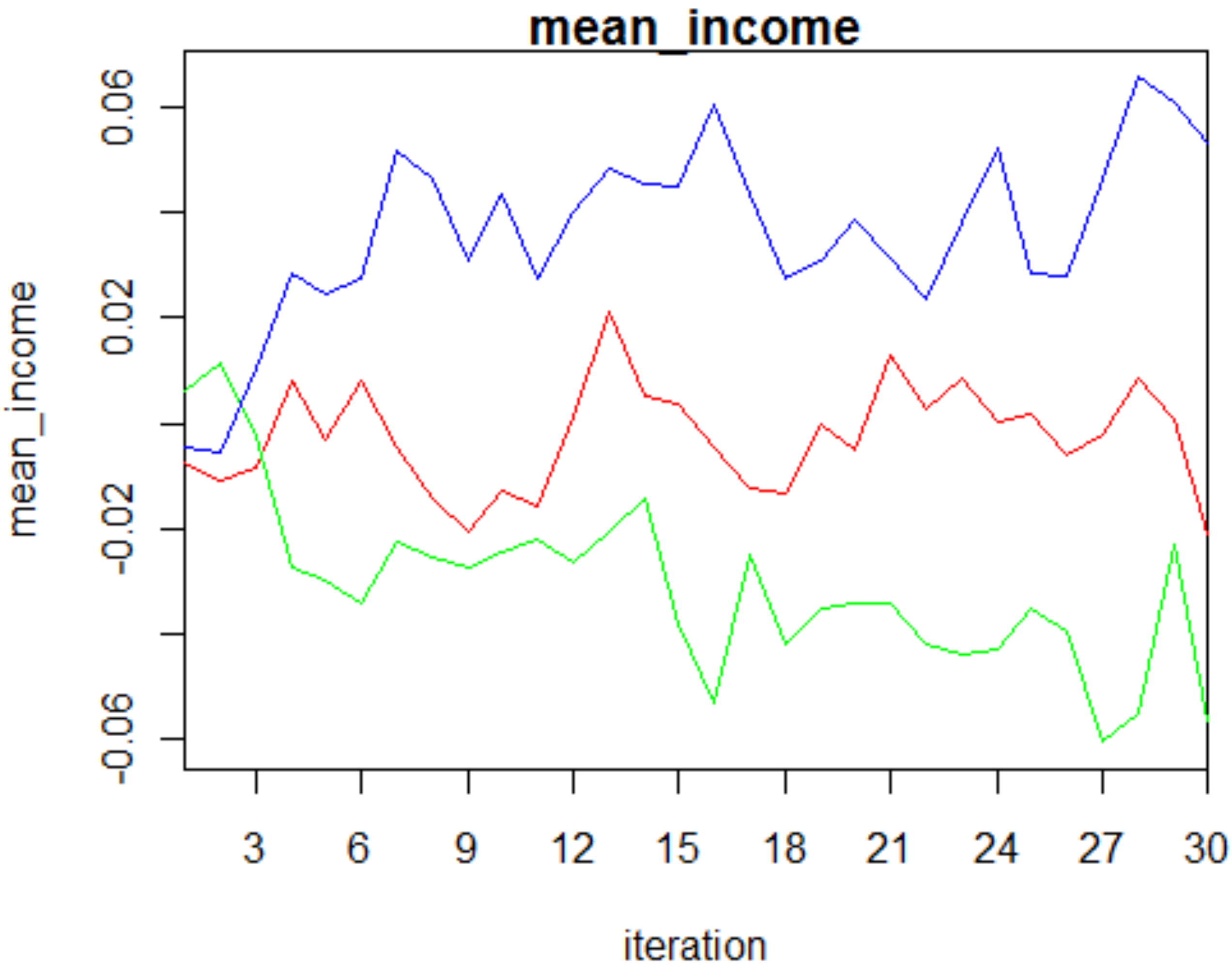
```
> imputations <- mi(mdf)
> converged <- mi2BUGS(imputations)
> print(converged)
> plot(converged)
```

Check convergence



now looking at convergence for the income imputations

```
> traceplot(converged)
```



not so good…

how can we quantify?

Rhat statistic (also called "estimated potential scale reduction") (Gelman and Rubin)

$$\hat{R} = \sqrt{\frac{\frac{N-1}{N}W + \frac{1}{N}B}{W}}$$

$$B = N\,\mathrm{var}(\bar{x}^m)$$

$$W = \frac{1}{m}\sum_m \mathrm{var}(x^m)$$

- *x* is a statistic of your choice; we look at the mean and sd of the *completed* data for each variable with missing data
- *N* is number iterations per chain

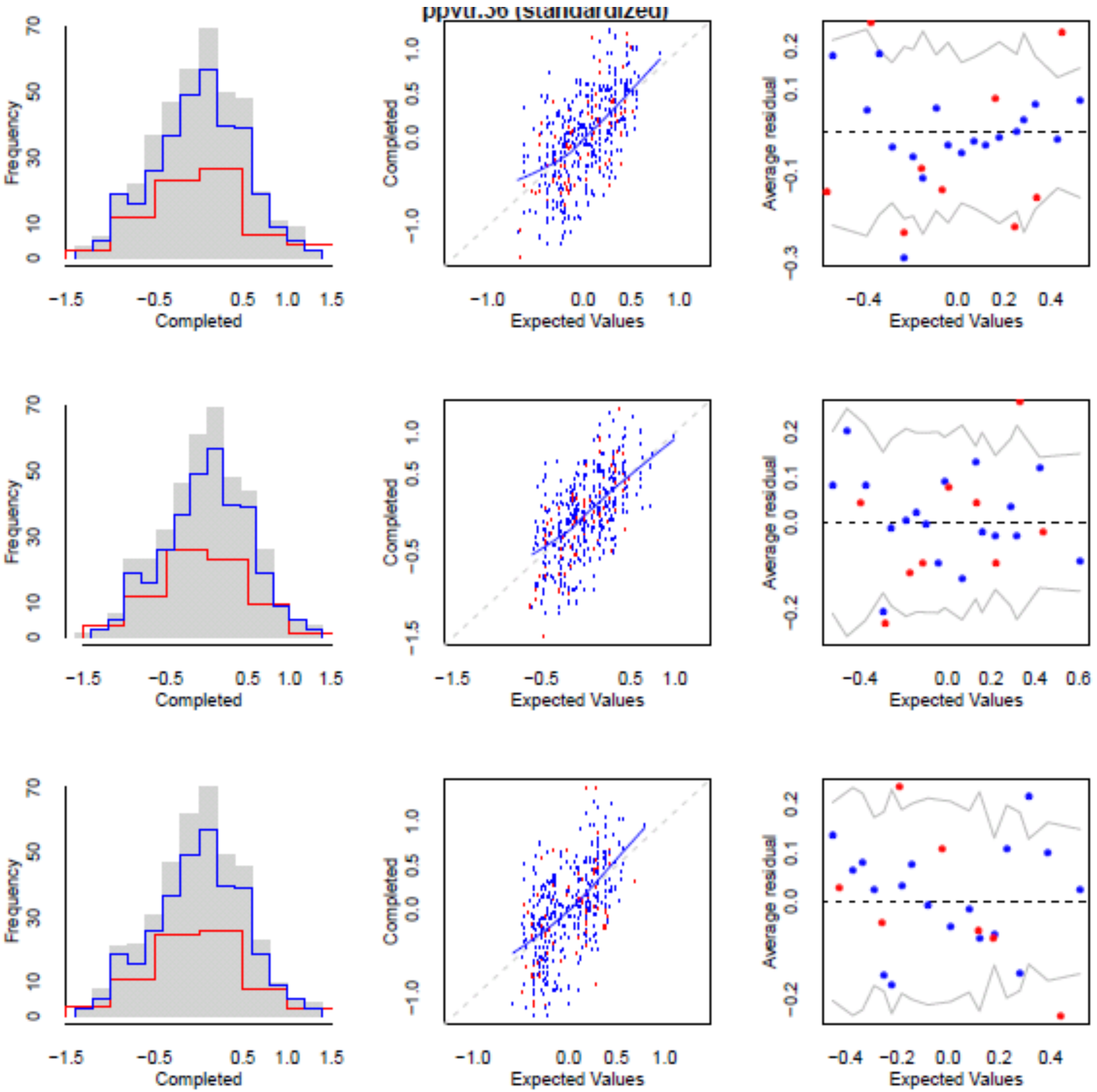
Convergence diagnostics: pay attention to “Rhat”

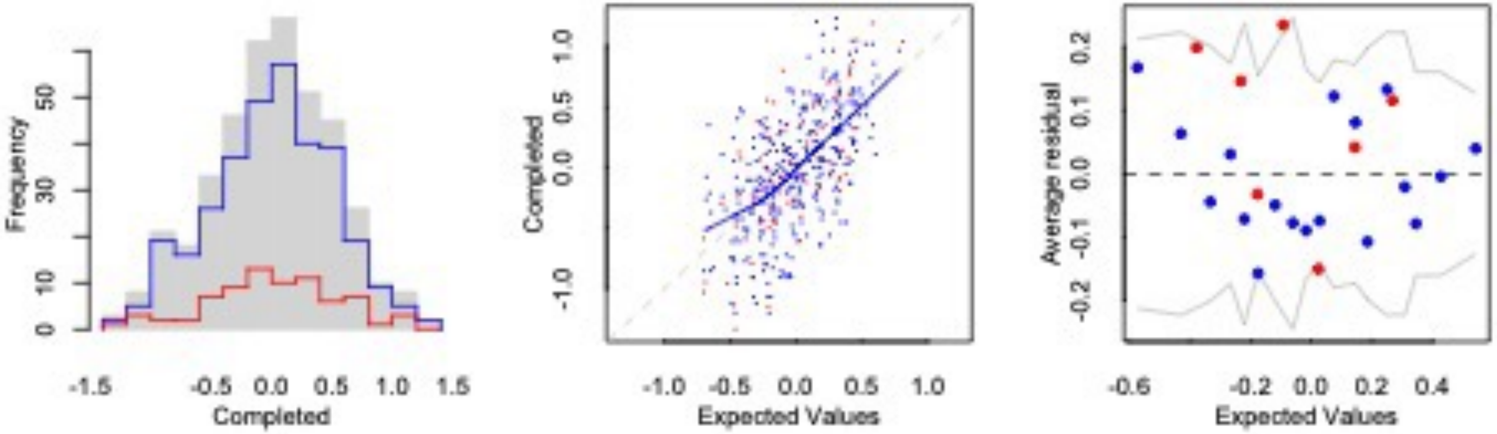
```
Rhats(imputations)
mean_ppvtr.36   mean_b.marr   mean_income
0.9998835       1.1728611     1.2706806

mean_momed  mean_momrace
1.0267368     1.0233137

sd_ppvtr.36     sd_b.marr     sd_income
0.9932876     1.1723309     1.0581659
sd_momed    sd_momrace
1.0291674     0.9840442
```

(6) Plot diagnostics

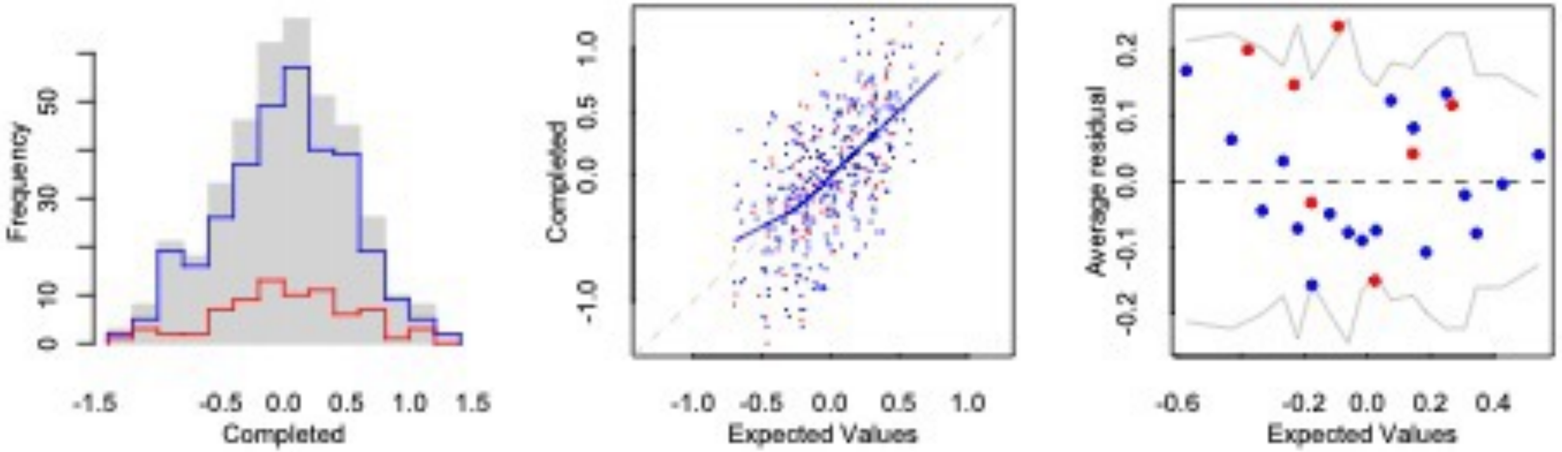
```
> plot(imputations)
> hist(imputations)
```

one set of
plots (a row)
for each
chain

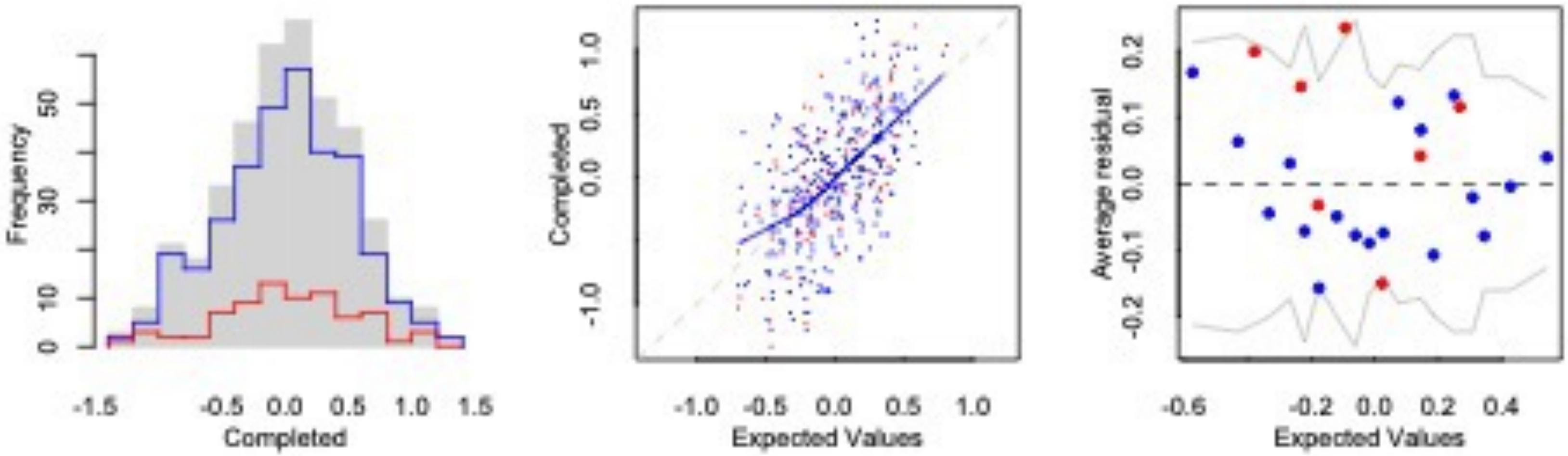




overlaid histograms: grey shaded for completed, blue outline for observed, red outline for imputed



expected values from fitted models plotted against observed (blue) and imputed (red) data points



Binned residual plot that plots the average of residuals in bins against the expected values with 95% error bounds. Each point in a binned residual plot is the average of the points that fall in each “bin” (interval of the variable on the *x*-axis) from a standard residual point.

Ref: Gelman,Goegebeur, Tuerlinckx, and Van Mechelen (2000)

Iterate steps (4)-(6) (if necessary)

Let’s treat income as “non-negative continuous,” a type that creates two new variables to replace the original

1) an indicator variable for whether the observation is 0 or not
2) the second forces a log transformation for the positive values and treats the 0 values as missing

```
mdf <- change(mdf, y = "income", what = "type", to = "nonn")
```

```
                           type missing method  model
ppvtr.36             continuous      75    ppd linear
first                    binary       0   <NA>   <NA>
b.marr                   binary      12    ppd  logit
income   nonnegative-continuous      82    ppd linear
momage               continuous       0   <NA>   <NA>
momed       ordered-categorical      40    ppd ologit
momrace   unordered-categorical     117    ppd mlogit
```

(7) Run pooled analysis

(Let’s use 5 imputed datasets)

```
> analysis <- pool(ppvtr.36 ~ first + b.marr + scale(income) +
   momage + momed + momrace, imputations, m=5)
> display(analysis)

glm(formula = ppvtr.36 ~ first + b.marr + scale(income) +
    momage + momed + momrace, data = imputations, m = 5)
              coef.est coef.se
(Intercept)   72.36     7.00
first1         3.59     1.63
b.marr1        4.74     1.97
scale(income)  0.66     0.80
momage        -0.06     0.28
momed2         4.03     1.89
momed3         9.00     2.28
momed4        14.36     3.51
momrace2      -5.41     2.45
momrace3      13.58     2.27
n = 400, k = 10
residual deviance = 87938.5, null deviance = 139952.0
   (difference = 52013.5)
overdispersion parameter = 219.8
residual sd is sqrt(overdispersion) = 14.83
```