Multiple Imputation in R

with the R package mice

Karin Groothuis-Oudshoorn (department HTSR)

Twente User R Group

7 March 2017

UNIVERSITY OF TWENTE.

Missing data

Missing data / Nonresponse

- *'The best solution to handle missing data is to have none'* (RA Fisher)
- *'Sooner or later (usually sooner), anyone who does statistics runs into problems with missing data'* (P Allison)
- Missing data problems are at the heart of statistics
  - Sample and population
  - Design of experiments
  - Data combination

Missing data patterns

MCAR

Missing Completely At Random

MAR

Missing At Random: no essential information associated with missing values is not known

MNAR

Missing Not At Random

BEWARE:

MCAR is testable, MAR not!

Causes missing data

Planned

- Sample from population
- Modulair survey, matrix sampling
- Routing questionnaire
- Censoring

Not planned

- Respondent has skipped an item
- Data transmissing / error in coding
- Drop out in longitudinal research
- Refused to cooperate

Consequences missing data

- Less information than planned before (enough statistical power?)
- Statistics are not defined (e.g. mean)
- Biases in data analysis
  - Systematic bias
  - Repressentativity
  - Confidence intervals, p-values?
- In general: missing data can make the interpretation and analysis of data more complicated.

Strategies

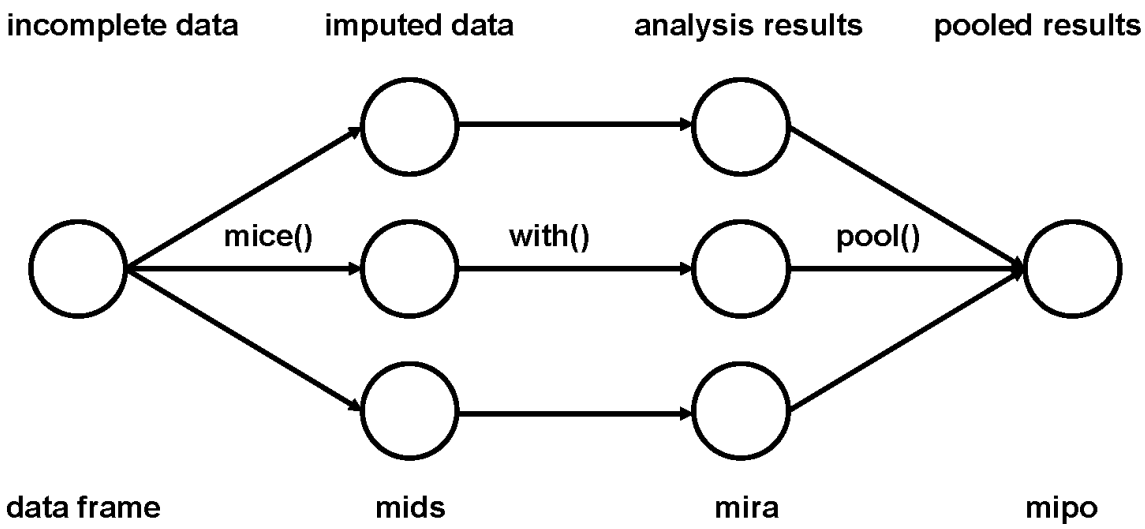
- Prevention
- Simple methods (LOCF, mean imputation)
- Weighting, GEE
- Likelihood methods (e.g. EM)
- **Multiple Imputation**

Package mice

- First version: 2000
- Second version: started in 2009 (workshop on R-User conference in Washington 2010)
- Developed by Stef van Buuren (UU, TNO) and myself with now a lot of contributions by others.

```
library(mice)
rm(list = ls())
```

Principle of multiple imputation



Principle of FCS (Fully Conditional Specification)

```
    age  bmi  hyp chl
1 20-39   NA <NA>  NA
2 40-59 22.7   no 187
3 20-39   NA   no 187
4 60-99   NA <NA>  NA
5 20-39 20.4   no 113
6 60-99   NA <NA> 184
```

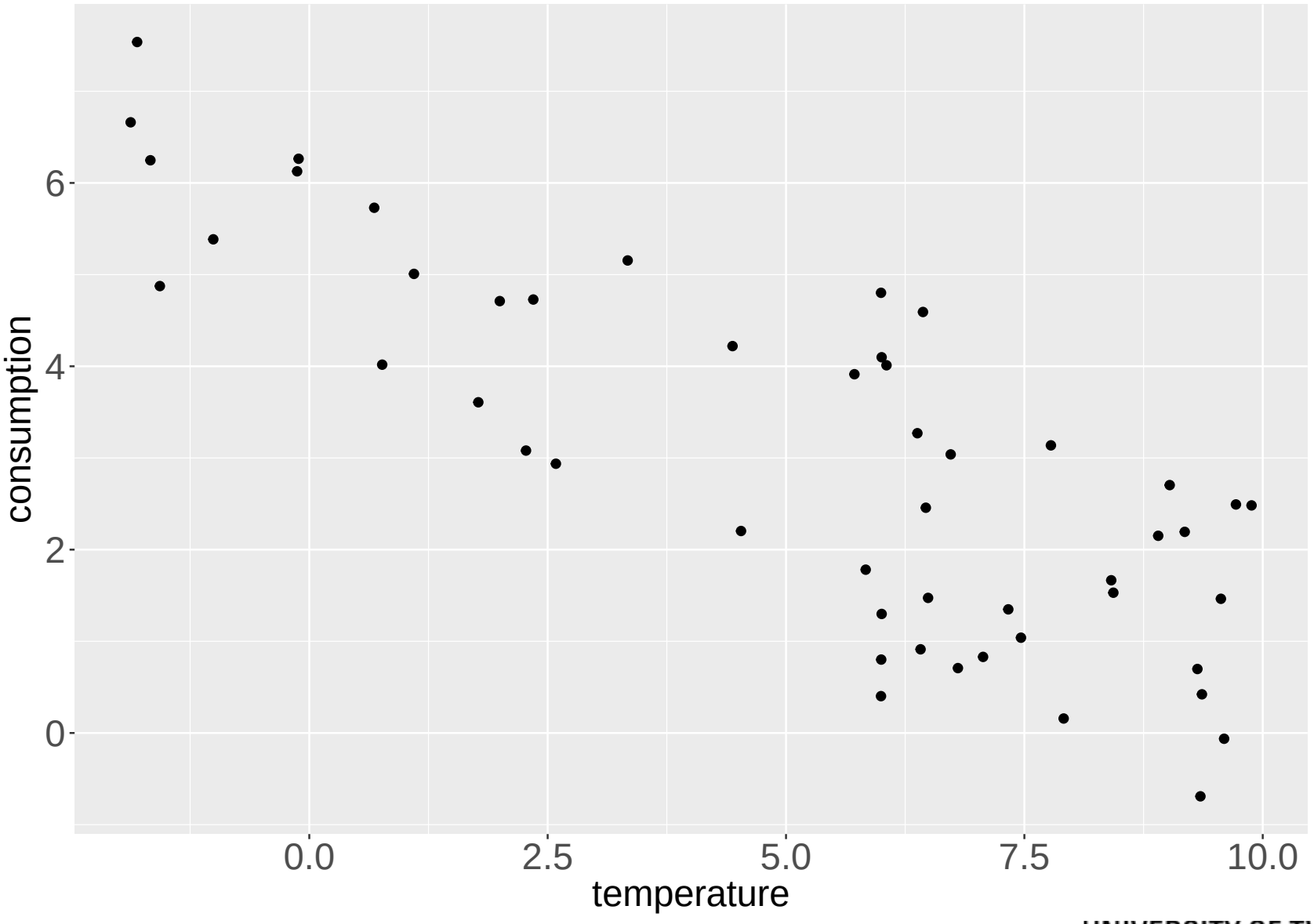
```
    age bmi hyp chl
age   0   0   0   0
bmi   1   0   1   1
hyp   1   1   0   1
chl   1   1   1   0
```

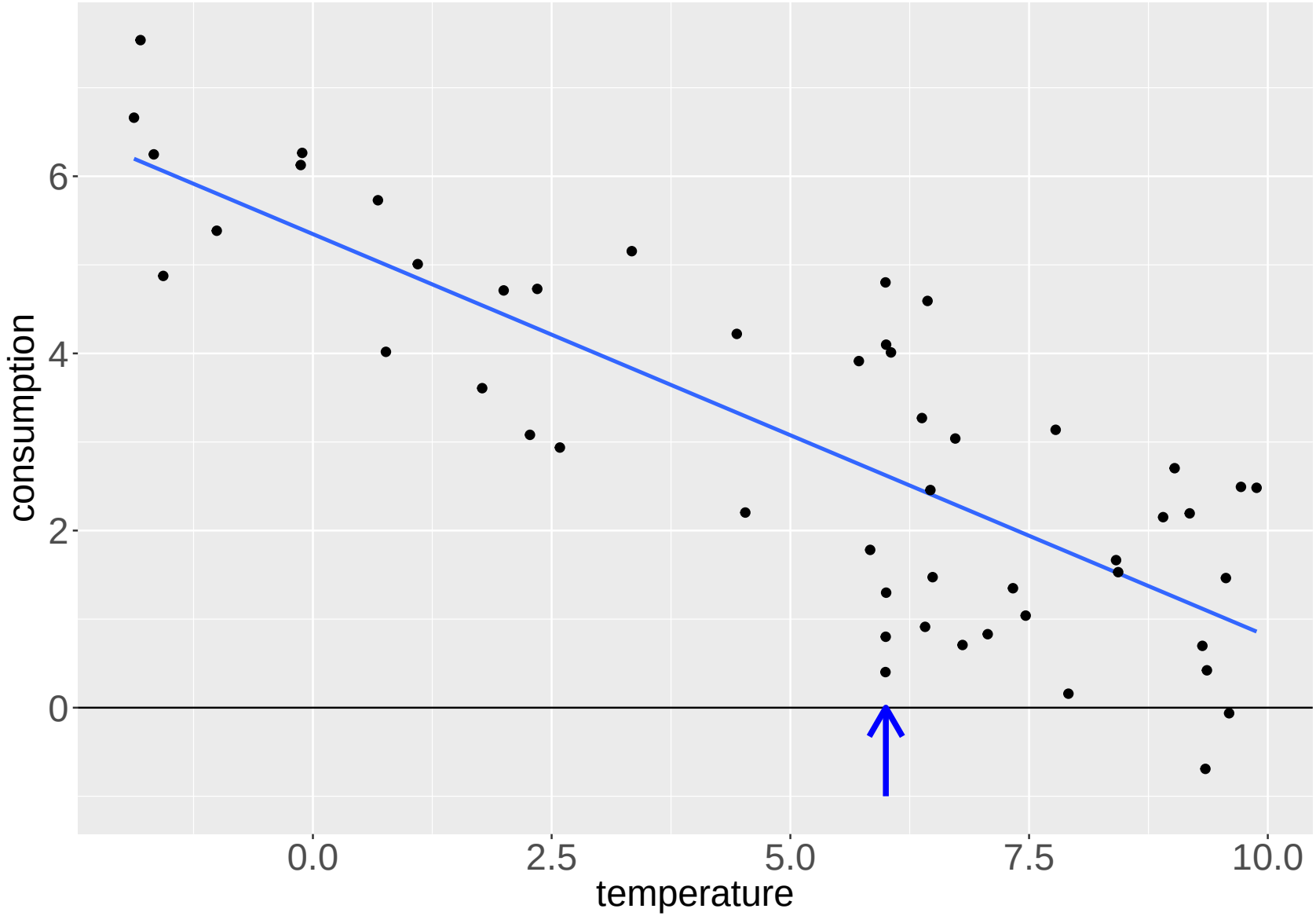
Simple imputation

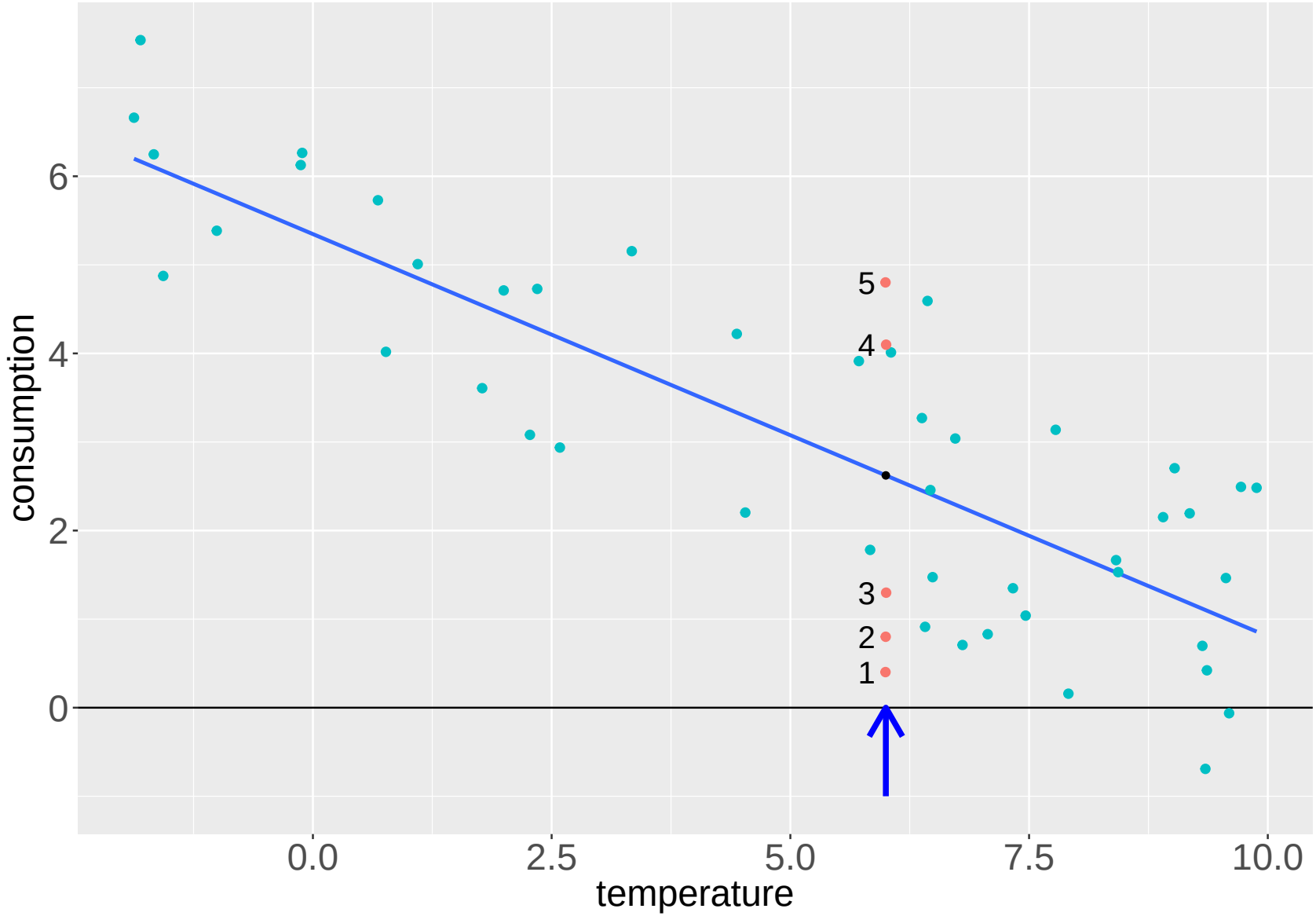
acronym	method	type of data
pmm	predictive mean matching	numeric
norm	bayesian linear regression	numeric
logreg	logistic regression	factor, 2 levels
polyreg	polytomous logistic regression	factor, $>$ 2 levels
polr	proportional odds model	ordered, $>$ 2 levels
sample	random sample observations	all

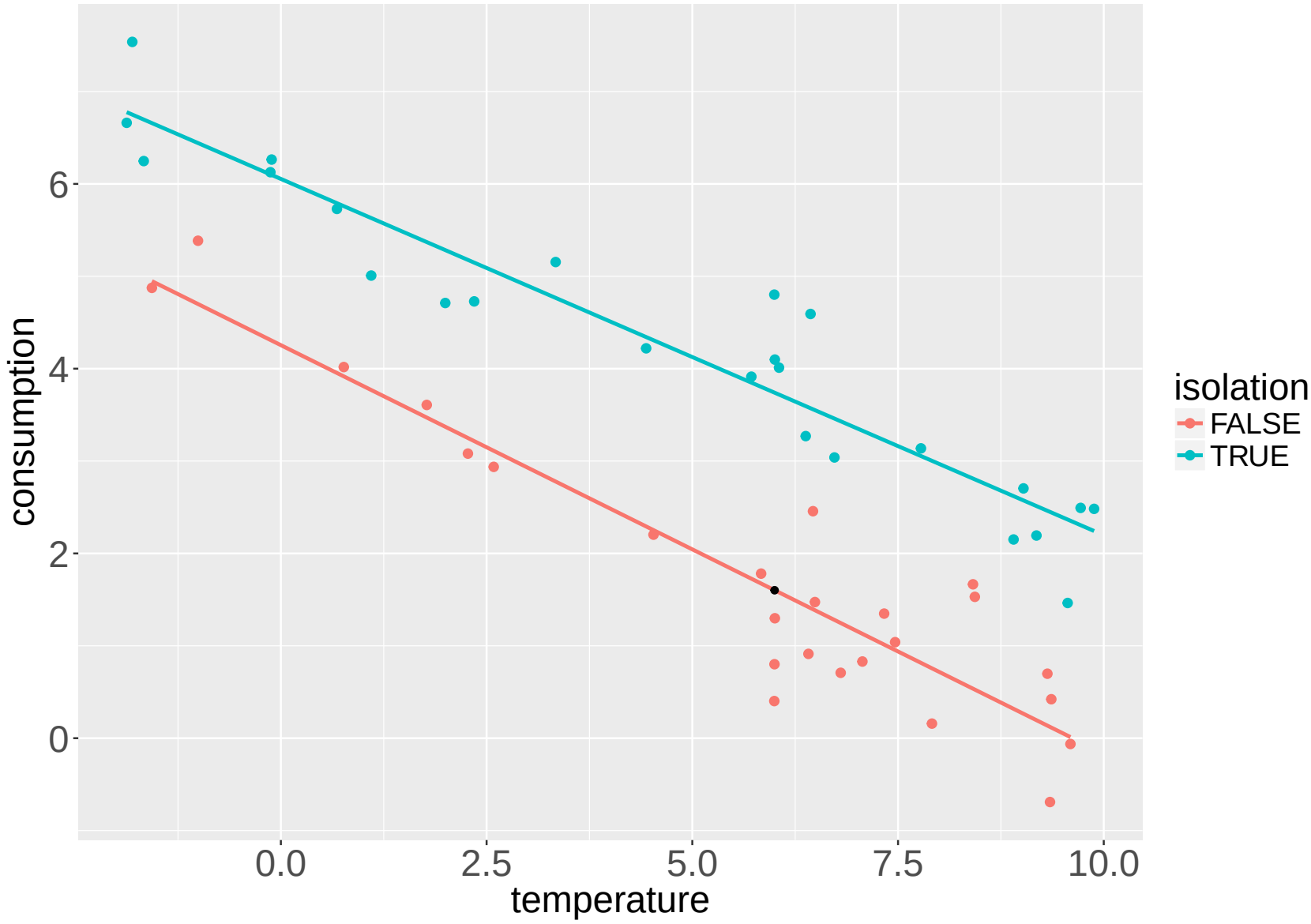
More methods exist based on cart, random forest, lda, for multilevel models etcetera.

How PMM works




consumption
6
4
2
0
0.0
2.5
5.0
7.5
10.0
temperature


consumption
6
4
2
0
0.0
2.5
5.0
7.5
10.0
temperature
5
4
3
2
1


consumption
6
4
2
0
0.0
2.5
5.0
7.5
10.0
temperature
isolation
FALSE
TRUE

Missing data NHANES

```
head(nhanes)
```

```
  age  bmi hyp chl
1   1   NA  NA  NA
2   2 22.7   1 187
3   1   NA   1 187
4   3   NA  NA  NA
5   1 20.4   1 113
6   3   NA  NA 184
```

Missing data NHANES: md.pattern()

```
md.pattern(nhanes)
```

```
   age hyp bmi chl
13   1   1   1   1  0
 1   1   1   0   1  1
 3   1   1   1   0  1
 1   1   0   0   1  2
 7   1   0   0   0  3
     0   8   9  10 27
```

Missing data pattern boys: md.pairs()

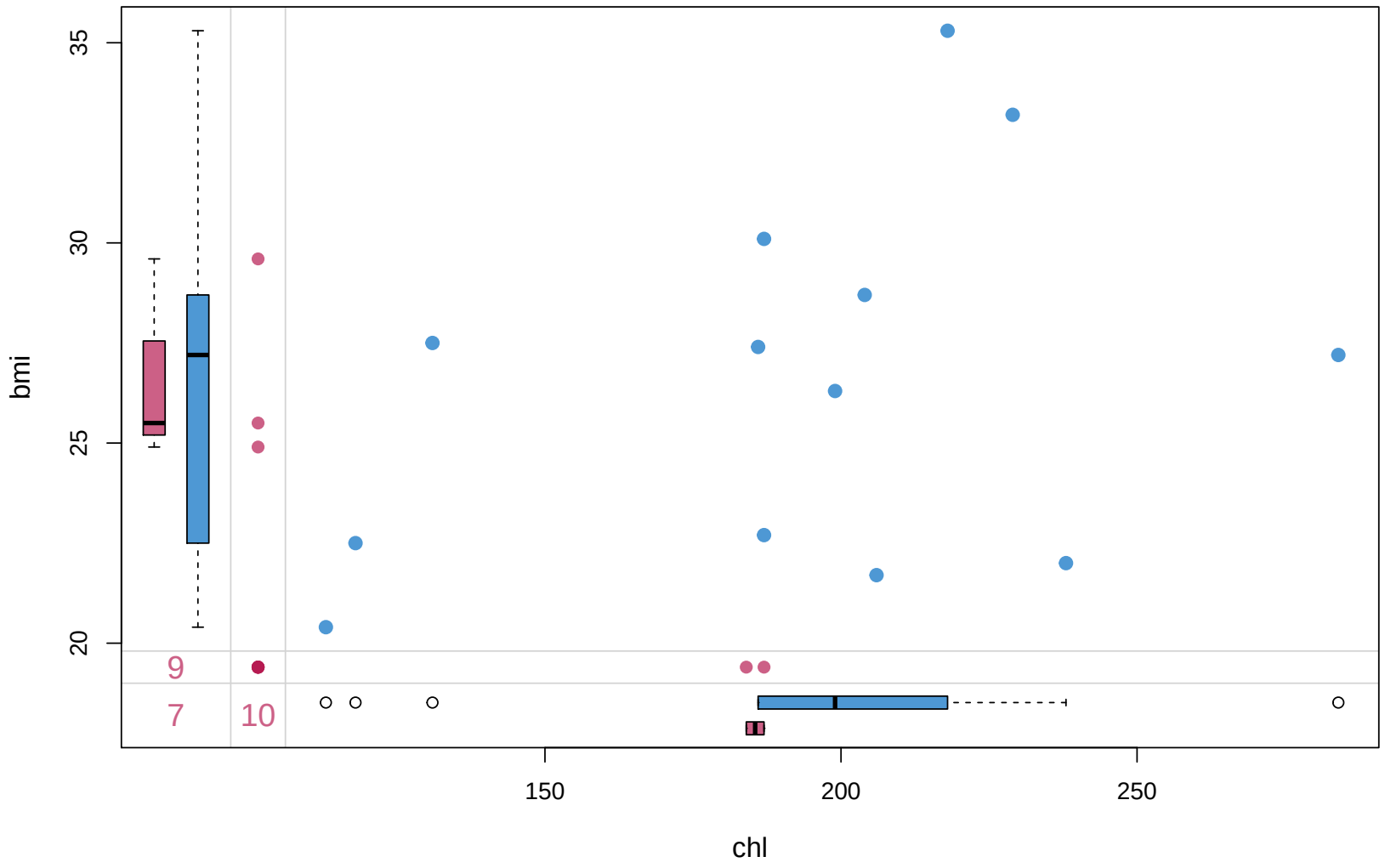
```
md.pairs(nhanes)
```

```
$rr
    age bmi hyp chl
age  25  16  17  15
bmi  16  16  16  13
hyp  17  16  17  14
chl  15  13  14  15

$rm
    age bmi hyp chl
age   0   9   8  10
bmi   0   0   0   3
hyp   0   1   0   3
chl   0   2   1   0

$mr
    age bmi hyp chl
age   0   0   0   0
bmi   9   0   1   2
hyp   8   0   0   1
chl  10   3   3   0
```

Marginplot ‘chl’ against ‘bmi’



Creating imputations

```
imp1 <- mice(nhanes, maxit = 20, m = 5, seed = 91168)
```

```
 iter imp variable
  1   1  bmi  hyp  chl
  1   2  bmi  hyp  chl
  1   3  bmi  hyp  chl
  1   4  bmi  hyp  chl
  1   5  bmi  hyp  chl
  2   1  bmi  hyp  chl
  2   2  bmi  hyp  chl
  2   3  bmi  hyp  chl
  2   4  bmi  hyp  chl
  2   5  bmi  hyp  chl
  3   1  bmi  hyp  chl
  3   2  bmi  hyp  chl
  3   3  bmi  hyp  chl
  3   4  bmi  hyp  chl
  3   5  bmi  hyp  chl
  4   1  bmi  hyp  chl
  4   2  bmi  hyp  chl
  4   3  bmi  hyp  chl
```

Result first imputation: print(imp1)

```
Multiply imputed data set
Call:
mice(data = nhanes, m = 5, maxit = 20, seed = 91168)
Number of multiple imputations:  5
Missing cells per column:
age bmi hyp chl
  0   9   8  10
Imputation methods:
  age   bmi   hyp   chl
   "" "pmm" "pmm" "pmm"
VisitSequence:
bmi hyp chl
  2   3   4
PredictorMatrix:
    age bmi hyp chl
age   0   0   0   0
bmi   1   0   1   1
hyp   1   1   0   1
chl   1   1   1   0
Random generator seed value:  91168
```

Imputated values for ‘chl’

```
imp1$imp$chl
```

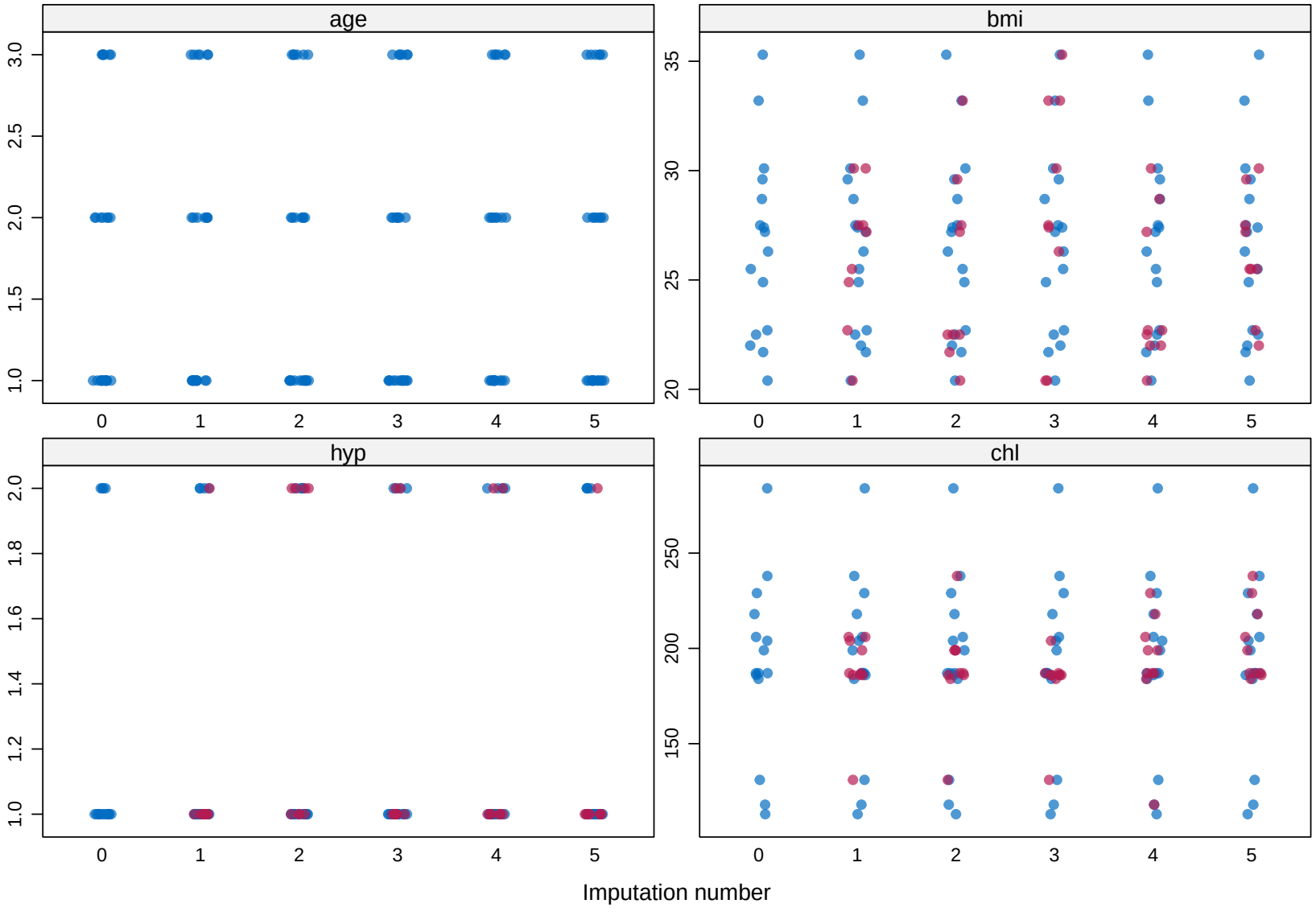
	1	2	3	4	5
1	131	238	184	199	187
4	186	184	186	184	184
10	206	187	204	199	187
11	187	131	186	118	187
12	204	199	186	229	206
15	206	199	187	187	229
16	186	187	131	187	199
20	186	186	186	206	218
21	187	199	187	187	238
24	199	186	186	218	186

Completed dataset

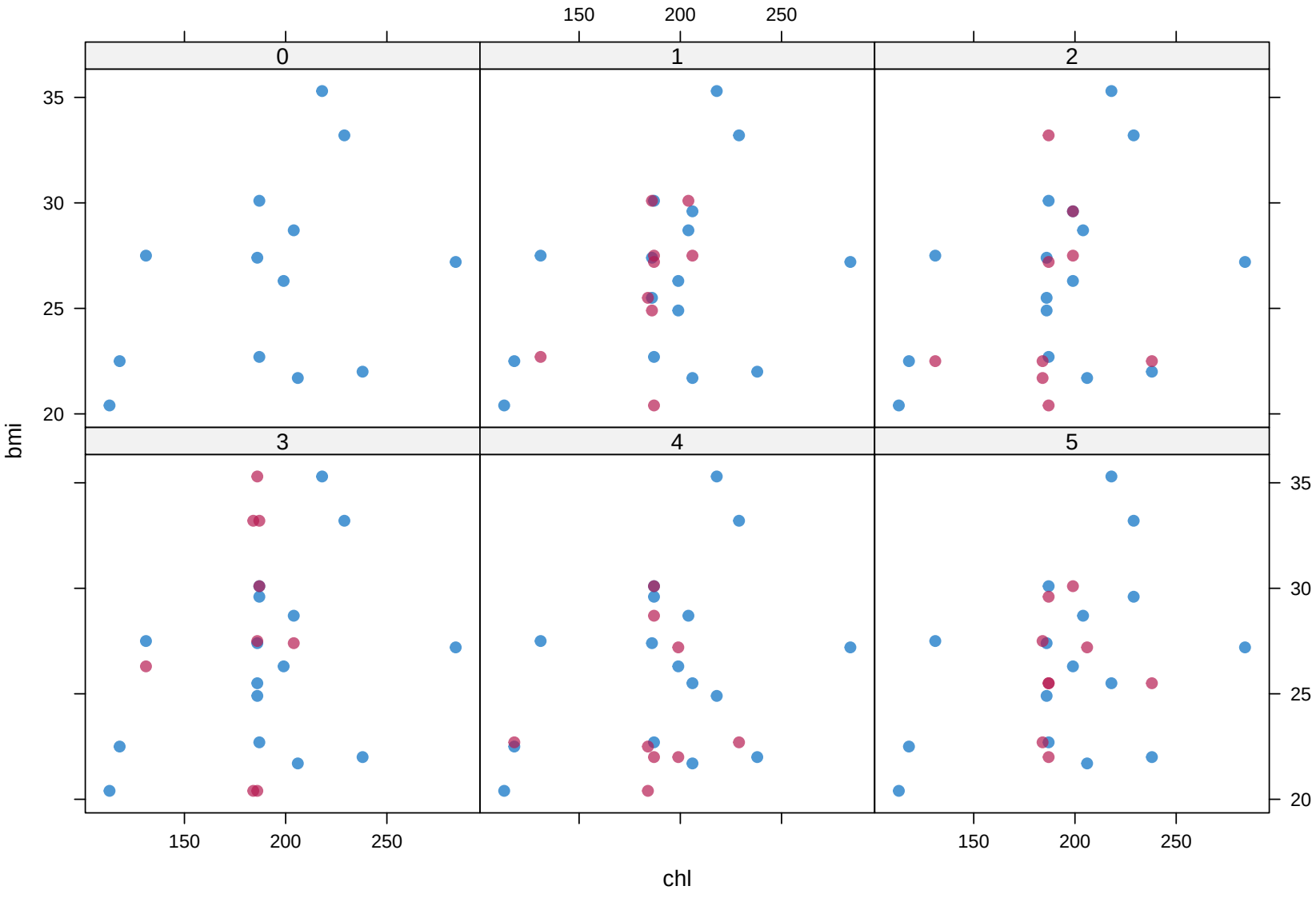
```
nhanes_imp <- mice:::complete(imp1, "long")
head(nhanes_imp)
```

```
  .imp .id age  bmi hyp chl
1    1   1   1 22.7   1 131
2    1   2   2 22.7   1 187
3    1   3   1 27.2   1 187
4    1   4   3 24.9   2 186
5    1   5   1 20.4   1 113
6    1   6   3 25.5   1 184
```

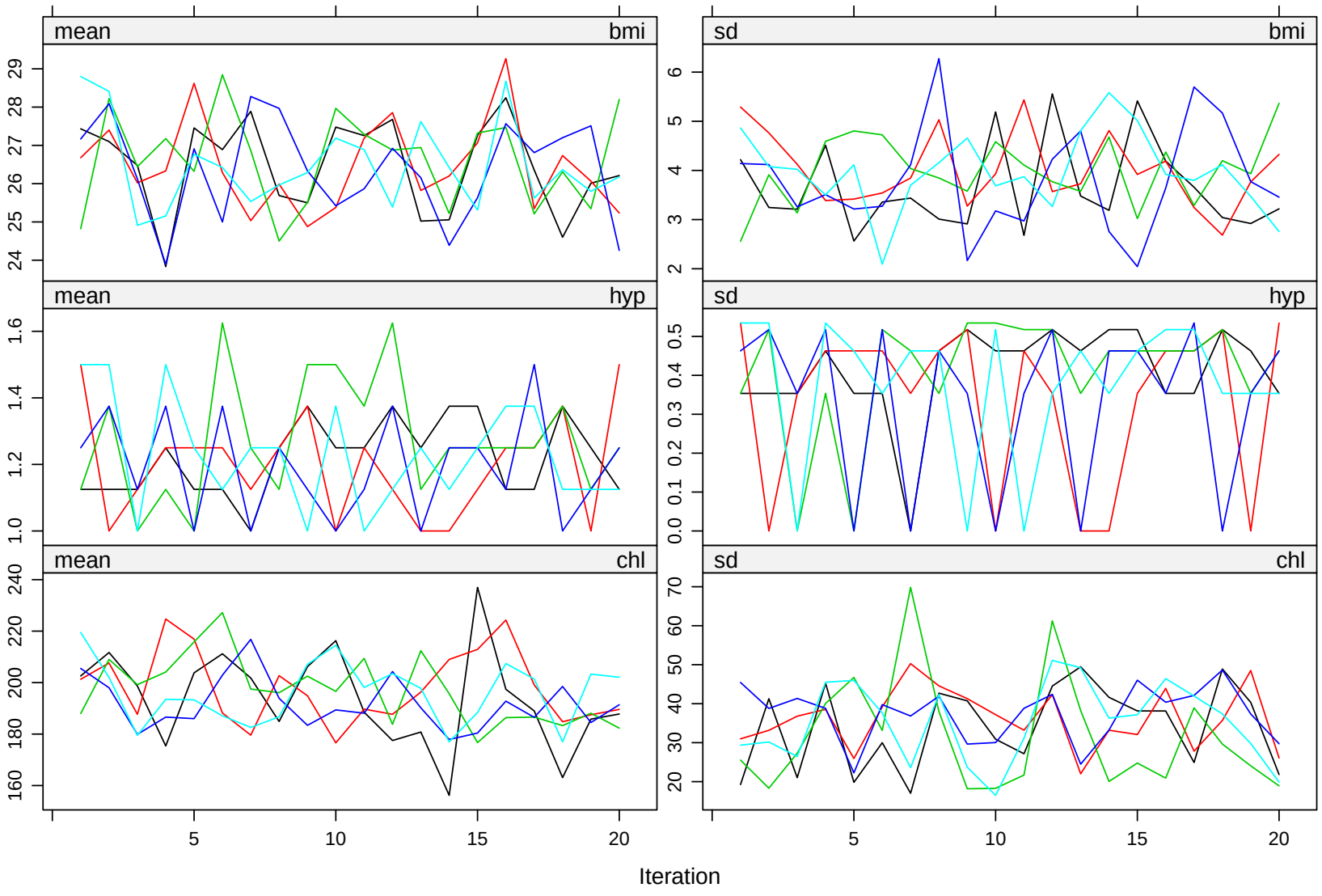
Diagnostic checking: stripplot()



Diagnostic checking: xyplot() between 'chl' and 'bmi'



Convergence of imputations: plot(imp)



Pooling

Pooling is combination of outcomes of seperate analyses

Rubin's principles of pooling:

- Point estimates are averaged over the different imputed datasets
- Variances: combination of **variance of** point estimates and **variance between** point estimates

mira: multiple imputed repeated analysis

```
imp10 <- mice(nhanes, m = 10, maxiter = 20, seed = 23109, printFlag = F
fit <- with(imp10, lm(chl ~ age + bmi))

fit$analyses[[1]]
```

```
Call:
lm(formula = chl ~ age + bmi)

Coefficients:
(Intercept)          age          bmi
     -6.903       35.863        5.199
```

mipo: multiple imputed pooled outcomes

```
print(pool(fit))
```

```
Call: pool(object = fit)

Pooled coefficients:
(Intercept)         age         bmi
 -23.707169   34.763243    5.808033

Fraction of information about the coefficients missing due to nonrespon
(Intercept)         age         bmi
  0.2822526   0.3862036   0.3607982
```

```
round(summary(pool(fit)), 2)
```

```
               est    se     t    df Pr(>|t|)   lo 95  hi 95 nmis  fmi
(Intercept) -23.71 62.55 -0.38 15.27     0.71 -156.82 109.41   NA 0.28
age          34.76 10.38  3.35 12.53     0.01   12.26  57.26    0 0.39
bmi           5.81  2.10  2.76 13.18     0.02    1.27  10.35    9 0.36
            lambda
(Intercept)   0.19
age           0.30
bmi           0.27
```

Example: Boys dataset

```
head(boys)
```

```
     age  hgt   wgt   bmi   hc  gen  phb tv   reg
3  0.035 50.1 3.650 14.54 33.7 <NA> <NA> NA south
4  0.038 53.5 3.370 11.77 35.0 <NA> <NA> NA south
18 0.057 50.0 3.140 12.56 35.2 <NA> <NA> NA south
23 0.060 54.5 4.270 14.37 36.7 <NA> <NA> NA south
28 0.062 57.5 5.030 15.21 37.3 <NA> <NA> NA south
36 0.068 55.5 4.655 15.11 37.0 <NA> <NA> NA south
```

Missing data pattern boys

```
md.pattern(boys)
```

```
     age reg wgt hgt bmi hc gen phb  tv
223    1   1   1   1   1  1   1   1   1    0
  1    1   1   1   1   1  1   1   0   1    1
 19    1   1   1   1   1  1   1   1   0    1
  1    1   1   1   1   1  1   0   1   0    2
  1    1   1   0   0   0  1   1   1   1    3
437    1   1   1   1   1  1   0   0   0    3
  1    1   1   0   0   0  0   1   1   1    4
 43    1   1   1   1   1  0   0   0   0    4
  3    1   0   1   1   1  1   0   0   0    4
 16    1   1   1   0   0  1   0   0   0    5
  1    1   1   0   1   0  1   0   0   0    5
  1    1   1   1   0   0  0   0   0   0    6
  1    1   1   0   0   0  0   0   0   0    7
       0   3   4  20  21 46 503 503 522 1622
```

Default imputation

```
ini <- mice(boys, maxit = 0, m = 5)
attributes(ini)
```

```
$names
 [1] "call"             "data"             "m"
 [4] "nmis"             "imp"              "method"
 [7] "predictorMatrix"  "visitSequence"    "form"
[10] "post"             "seed"             "iteration"
[13] "lastSeedValue"    "chainMean"        "chainVar"
[16] "loggedEvents"     "pad"

$class
[1] "mids"
```

Default method

```
ini$nmis
```

```
age hgt wgt bmi  hc gen phb  tv reg
  0  20   4  21  46 503 503 522   3
```

```
ini$method
```

```
      age       hgt       wgt       bmi        hc       gen       phb
       ""     "pmm"     "pmm"     "pmm"     "pmm"    "polr"    "polr"
       tv       reg
    "pmm" "polyreg"
```

Passive imputation: bmi function of hgt, wgt

```
md.pattern(boys[, c("hgt", "wgt", "bmi")])
```

```
     wgt hgt bmi
727    1   1   1  0
 17    1   0   0  2
  1    0   1   0  2
  3    0   0   0  3
       4  20  21 45
```

Passive imputation: synchronise bmi with hgt, wgt

```
meth <- ini$meth
meth["bmi"] <- "~I(wgt/(hgt/100)^2)"
pred <- ini$pred
pred[c("wgt", "hgt", "hc", "reg"), "bmi"] <- 0
pred[c("gen", "phb", "tv"), c("hgt", "wgt", "hc")] <- 0
pred
```

```
    age hgt wgt bmi hc gen phb tv reg
age   0   0   0   0  0   0   0  0   0
hgt   1   0   1   0  1   1   1  1   1
wgt   1   1   0   0  1   1   1  1   1
bmi   1   1   1   0  1   1   1  1   1
hc    1   1   1   0  0   1   1  1   1
gen   1   0   0   1  0   0   1  1   1
phb   1   0   0   1  0   1   0  1   1
tv    1   0   0   1  0   1   1  0   1
reg   1   1   1   0  1   1   1  1   0
```

Result of passive imputation

```
imp.idx <- mice(boys, pred = pred, meth = meth, maxit = 20, seed = 9212
head(mice:::complete(imp.idx)[is.na(boys$bmi), ], 3)
```

```
       age  hgt   wgt      bmi   hc gen phb tv   reg
103  0.087 61.8  4.54 11.88718 39.0  G1  P1  2  west
366  0.177 57.5  5.26 15.90926 40.4  G1  P1  2  west
1617 1.481 90.2 12.04 14.79835 47.5  G1  P1  2 north
```

Extensions

- Creating sumscores from items
- post calculations
- collinearity
- assessing convergence
- stepwise imputation
- sequence of variables
- creating own imputation function

More information

Article:

mice: Multivariate Imputation by Chained Equations in R

by Stef van Buuren and Karin Groothuis-Oudshoorn

(https://www.jstatsoft.org/article/view/v045i03)