432 Class 11 Slides

github.com/THOMASELOVE/432-2018

2018-02-20

Setup

```r
library(skimr)
library(pROC)
library(ROCR)
library(rms) # note: also loads Hmisc
library(simputation)
library(broom)
library(tidyverse)
```

Today's Materials

- Logistic Regression and the Framingham Study (part 2)
- Performing Linear Regression with `ols`
- Hormone Therapy and Baseline LDL in the `HERS` trial

The HERS trial is described in Vittinghoff et al., especially Chapter 4.

Logistic Regression and Framingham

Data Ingest, Cleanup (from Class 10)

```r
fram <- read.csv("data/fram_new.csv") %>% tbl_df
set.seed(432001)
fram1 <- fram %>%
    impute_pmm(educ + cigs_day + heart_r ~ age + smoker) %>%
    impute_rlm(bmi + tot_chol ~ sex + age + sbp + heart_r) %>%
    impute_pmm(bp_meds ~ hx_htn + bmi + tot_chol) %>%
    impute_rlm(glucose ~ hx_dm + bmi + tot_chol + age) %>%
    mutate(ed_f = fct_recode(factor(educ),
                   "1_Some_HS" = "1", "2_HS_grad" = "2",
                   "3_Some_Col" = "3", "4_Col_grad" = "4"))
fram2 <- fram1 %>%
    select(subj, sex, age, smoker, cigs_day, bp_meds,
           hx_stroke, hx_htn, hx_dm, ed_f, tot_chol,
           sbp, dbp, bmi, heart_r, glucose, CHD_10)
```

The Models We've Fit (predicting CHD_10)

```r
m_01 <- glm(CHD_10 ~ hx_htn, data = fram2,
            family = binomial)

d <- datadist(fram2)
options(datadist = "d")
m_01_lrm <- lrm(CHD_10 ~ hx_htn, data = fram2, x = T, y = T)

m_02 <- glm(CHD_10 ~ hx_htn + tot_chol,
            data = fram2, family = binomial)
m_02_lrm <- lrm(CHD_10 ~ hx_htn + tot_chol, data = fram2,
                x = TRUE, y = TRUE)
```

Assessing Predictive Quality: Discrimination

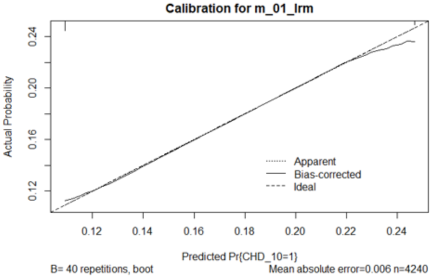
Key measures: C statistic, Nagelkerke R^2

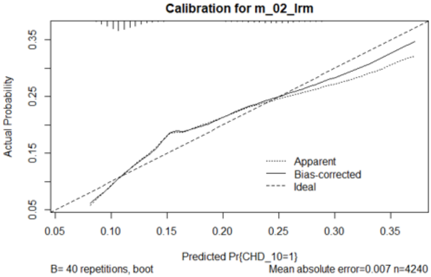
Model	C statistic	Nagelkerke R^2
`m_01_lrm`	0.614	0.051
`m_02_lrm`	0.640	0.055

and we could use `validate(model)` to address how well these results might hold up in new data.

Assessing Predictive Quality: Calibration Curves

```r
plot(calibrate(m_01_lrm), main = "Calibration for m_01_lrm")
plot(calibrate(m_02_lrm), main = "Calibration for m_02_lrm")
```





Assessing Predictive Quality: Goodness of Fit Test

This uses the le Cessie-van Houwelingen-Copas-Hosmer unweighted sum of squares test statistic. to produce (using up just one degree of freedom) a global goodness of fit test. It's available through `residuals` applied to a `lrm` fit, with `type = "gof"`).

The essential components of a logistic regression fit are:

1. The logit transformation is the correct function linking the covariates with the conditional mean,
2. The linear predictor is correct (we don't need to include additional variables, transformations of predictors or interaction terms), and
3. The variance follows a Bernoulli distribution.

See Hosmer et al. 1997

The Omnibus Goodness of Fit Test

As in any omnibus test, a significant result here is difficult to interpret, but it means that something somewhere in the model is probably wrong.

- Harrell: I focus on directed tests such as allowing all continuous variables to have nonlinear effects or allowing selected interactions, and finding out how important the complex model terms are.

```
round(residuals(m_01_lrm, type = "gof"),3)
round(residuals(m_02_lrm, type = "gof"),3)
```

```
       Sum of squared   Expected
           errors       value|H0      SD           Z     P
Model 1   528.985        528.985   0.000   -2268.981 0.000
Model 2   527.948        527.291   0.331       1.986 0.047
```

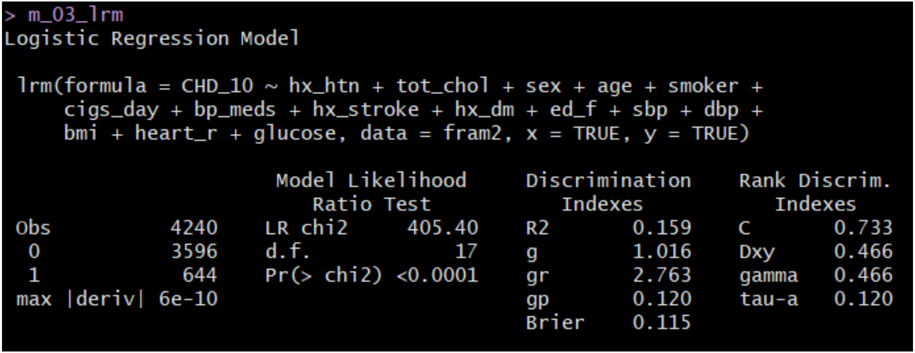
Looking better in `m_02_lrm` but still some work to do.

Goal 3. Kitchen Sink Model for `CHD_10`

Focus on model with `lrm` first!

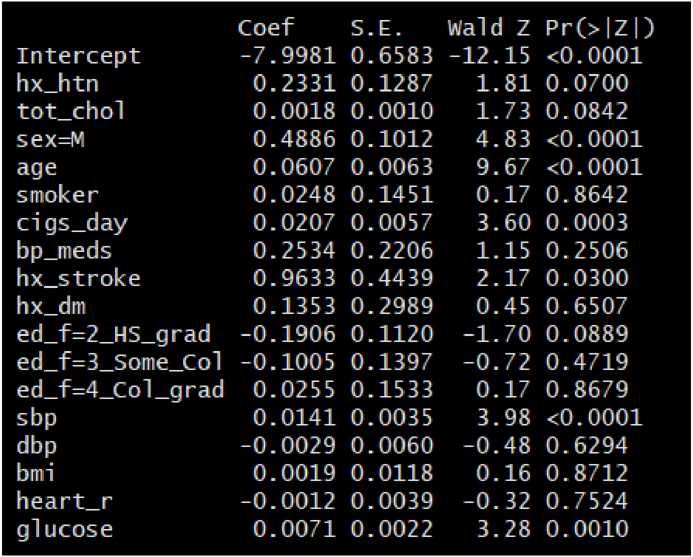
```r
m_03 <- glm(CHD_10 ~ hx_htn + tot_chol + sex + age +
                    smoker + cigs_day + bp_meds +
                    hx_stroke + hx_dm + ed_f + sbp + dbp +
                    bmi + heart_r + glucose,
                data = fram2, family = binomial)

d <- datadist(fram2)
options(datadist = "d")
m_03_lrm <- lrm(CHD_10 ~ hx_htn + tot_chol + sex + age +
                    smoker + cigs_day + bp_meds +
                    hx_stroke + hx_dm + ed_f + sbp + dbp +
                    bmi + heart_r + glucose,
                data = fram2, x = TRUE, y = TRUE)
```

m_03_lrm (first section of output)



```
> m_03_lrm
Logistic Regression Model

 lrm(formula = CHD_10 ~ hx_htn + tot_chol + sex + age + smoker + 
     cigs_day + bp_meds + hx_stroke + hx_dm + ed_f + sbp + dbp + 
     bmi + heart_r + glucose, data = fram2, x = TRUE, y = TRUE)

                      Model Likelihood     Discrimination    Rank Discrim.    
                         Ratio Test           Indexes           Indexes       
 Obs          4240    LR chi2     405.40    R2       0.159    C       0.733    
  0           3596    d.f.            17    g        1.016    Dxy     0.466    
  1            644    Pr(> chi2) <0.0001    gr       2.763    gamma   0.466    
 max |deriv| 6e-10                          gp       0.120    tau-a   0.120    
                                            Brier    0.115                     
```

m_03_lrm (second section of output)



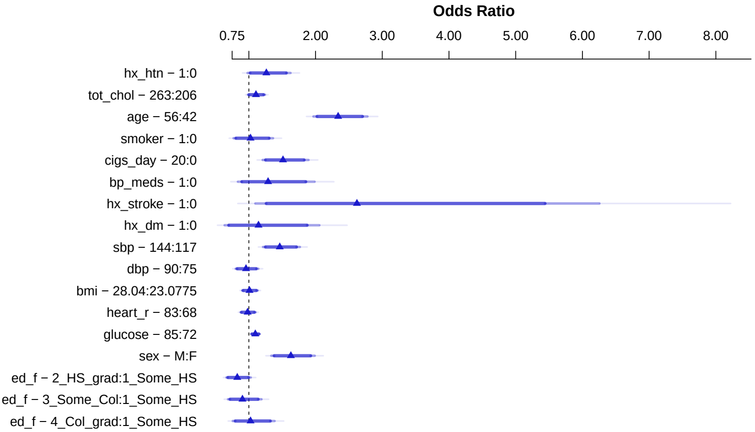
```
                 Coef    S.E.   Wald Z Pr(>|Z|)
Intercept        -7.9981 0.6583 -12.15 <0.0001
hx_htn            0.2331 0.1287   1.81 0.0700
tot_chol          0.0018 0.0010   1.73 0.0842
sex=M             0.4886 0.1012   4.83 <0.0001
age               0.0607 0.0063   9.67 <0.0001
smoker            0.0248 0.1451   0.17 0.8642
cigs_day          0.0207 0.0057   3.60 0.0003
bp_meds           0.2534 0.2206   1.15 0.2506
hx_stroke         0.9633 0.4439   2.17 0.0300
hx_dm             0.1353 0.2989   0.45 0.6507
ed_f=2_HS_grad   -0.1906 0.1120  -1.70 0.0889
ed_f=3_Some_Col  -0.1005 0.1397  -0.72 0.4719
ed_f=4_Col_grad   0.0255 0.1533   0.17 0.8679
sbp               0.0141 0.0035   3.98 <0.0001
dbp              -0.0029 0.0060  -0.48 0.6294
bmi               0.0019 0.0118   0.16 0.8712
heart_r          -0.0012 0.0039  -0.32 0.7524
glucose           0.0071 0.0022   3.28 0.0010
```

Validating our Summary Statistics

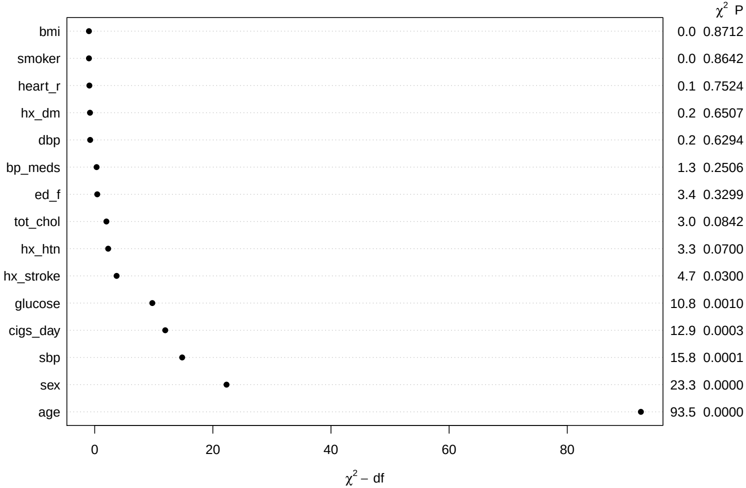
```r
set.seed(432020) # probably better to set a seed
validate(m_03_lrm)[1:4,] # to fit things in the slide
```

```
          index.orig  training       test    optimism
Dxy        0.4658670 0.4690634  0.4575484 0.011515041
R2         0.1590194 0.1624443  0.1526612 0.009783165
Intercept  0.0000000 0.0000000 -0.0466690 0.046668999
Slope      1.0000000 1.0000000  0.9635322 0.036467806
          index.corrected  n
Dxy             0.4543520 40
R2              0.1492362 40
Intercept      -0.0466690 40
Slope           0.9635322 40
```

plot(summary(m_03_lrm))

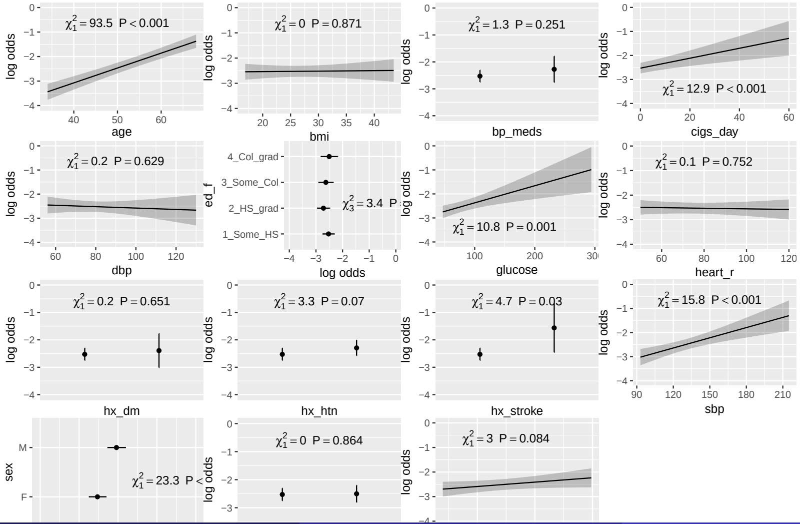


plot(anova(m_03_lrm))



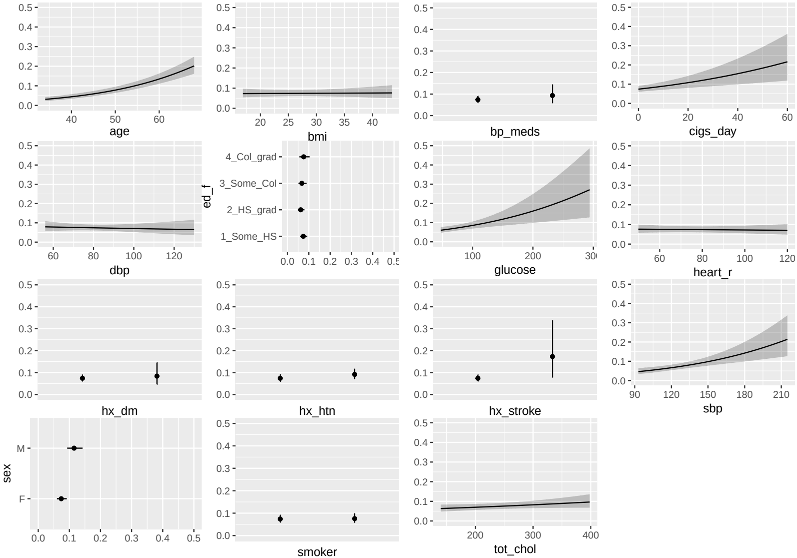
Can we see the prediction results?

```
ggplot(Predict(m_03_lrm),
       anova = anova(m_03_lrm), pval = TRUE)
```



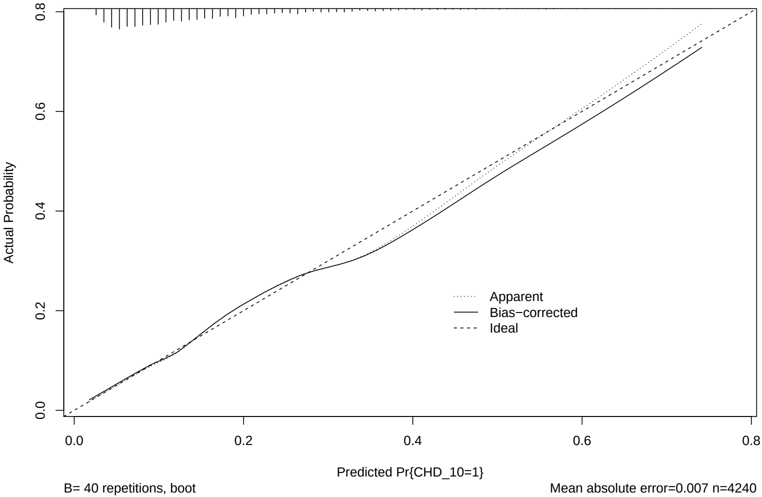
What about on a better scale?

```
ggplot(Predict(m_03_lrm, fun = plogis))
```



Calibration of `mod_03_lrm`

```r
set.seed(432029); plot(calibrate(m_03_lrm))
```



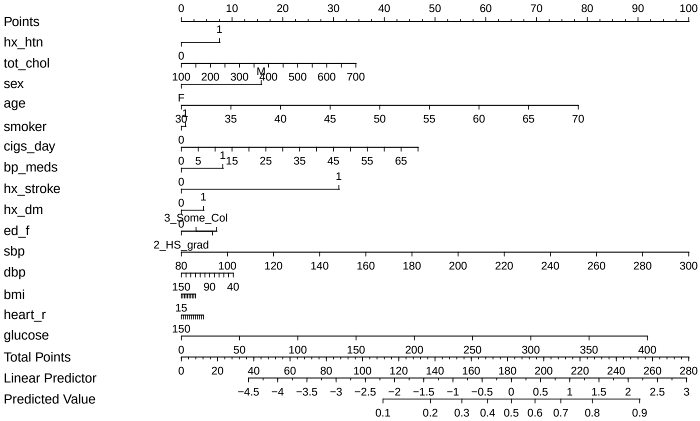
Goodness of fit test?

```r
round(residuals(m_03_lrm, type = "gof"),3)
```

```
Sum of squared errors     Expected value|H0
              488.329               488.026
                   SD                     Z
                1.407                 0.215
                    P
                0.830
```

Nomogram of `mod_03_lrm`

```r
plot(nomogram(m_03_lrm, fun = plogis))
```



Comparing our Three Nested Models

```
anova(m_01, m_02, m_03)
```

```
Analysis of Deviance Table

Model 1: CHD_10 ~ hx_htn
Model 2: CHD_10 ~ hx_htn + tot_chol
Model 3: CHD_10 ~ hx_htn + tot_chol + sex + age + smoker + cig
    bp_meds + hx_stroke + hx_dm + ed_f + sbp + dbp + bmi + hea
    glucose
  Resid. Df Resid. Dev Df Deviance
1      4238     3486.9
2      4237     3475.5  1   11.411
3      4222     3206.8 15  268.682
```

Model 2 vs. Model 3 at a glance

```r
glance(m_02)
```

```
  null.deviance df.null    logLik      AIC      BIC
1      3612.209    4239 -1737.748 3481.495 3500.552
  deviance df.residual
1 3475.495        4237
```

```r
glance(m_03)
```

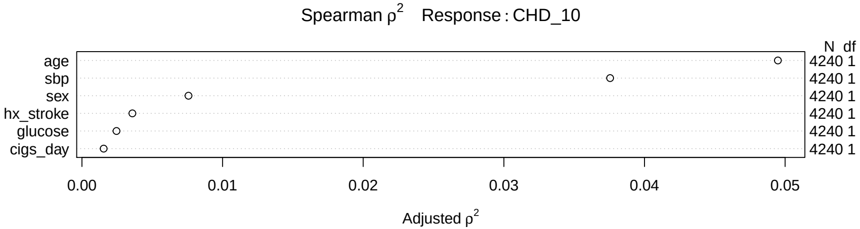
```
  null.deviance df.null    logLik      AIC      BIC
1      3612.209    4239 -1603.407 3242.813 3357.155
  deviance df.residual
1 3206.813        4222
```

Fitting a 6-predictor, but still useful model

What looks useful?

By ANOVA on `m_03_lrm` it looks like `age`, `sex`, `sbp`, `cigs_day`, `glucose`, `hx_stroke` for sure.

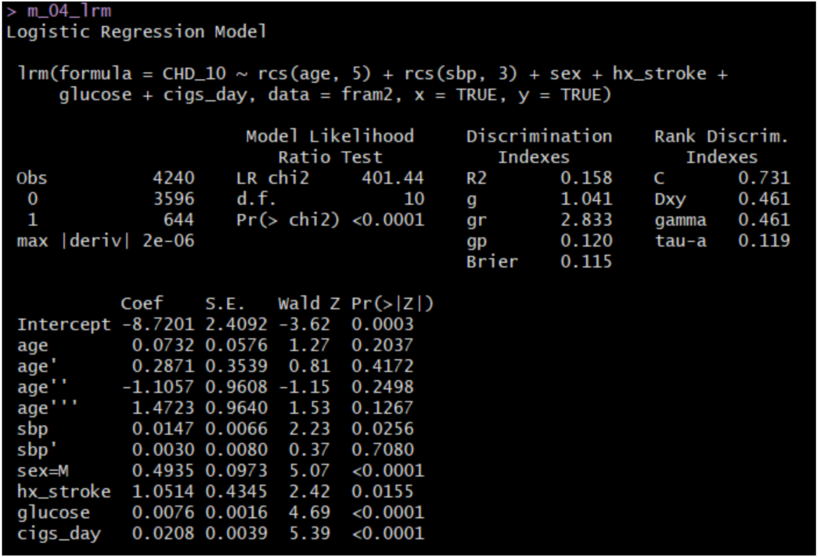
```
plot(spearman2(CHD_10 ~ age + sex + sbp + cigs_day +
                   glucose + hx_stroke, data = fram2))
```



New Model 4

```
m_04 <- glm(CHD_10 ~ rcs(age, 5) + rcs(sbp, 3) + sex +
                    hx_stroke + glucose + cigs_day,
                data = fram2, family = binomial)



dd <- datadist(fram2)
options(datadist = "dd")

m_04_lrm <- lrm(CHD_10 ~ rcs(age, 5) + rcs(sbp, 3) + sex +
                    hx_stroke + glucose + cigs_day,
                data = fram2, x = TRUE, y = TRUE)
```

m_04_lrm



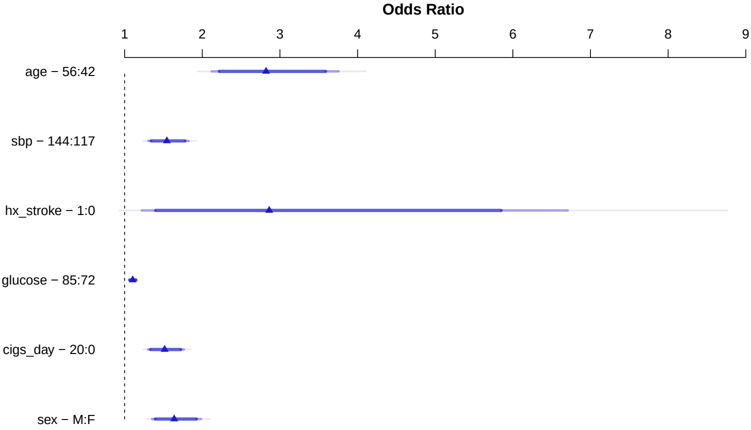
```
> m_04_lrm
Logistic Regression Model

 lrm(formula = CHD_10 ~ rcs(age, 5) + rcs(sbp, 3) + sex + hx_stroke +
     glucose + cigs_day, data = fram2, x = TRUE, y = TRUE)

                      Model Likelihood     Discrimination    Rank Discrim.
                         Ratio Test           Indexes           Indexes
 Obs          4240    LR chi2     401.44    R2       0.158    C       0.731
  0           3596    d.f.            10    g        1.041    Dxy     0.461
  1            644    Pr(> chi2) <0.0001    gr       2.833    gamma   0.461
 max |deriv| 2e-06                          gp       0.120    tau-a   0.119
                                            Brier    0.115

           Coef    S.E.   Wald Z Pr(>|Z|)
 Intercept -8.7201 2.4092 -3.62  0.0003
 age        0.0732 0.0576  1.27  0.2037
 age'       0.2871 0.3539  0.81  0.4172
 age''     -1.1057 0.9608 -1.15  0.2498
 age'''     1.4723 0.9640  1.53  0.1267
 sbp        0.0147 0.0066  2.23  0.0256
 sbp'       0.0030 0.0080  0.37  0.7080
 sex=M      0.4935 0.0973  5.07  <0.0001
 hx_stroke  1.0514 0.4345  2.42  0.0155
 glucose    0.0076 0.0016  4.69  <0.0001
 cigs_day   0.0208 0.0039  5.39  <0.0001
```

Validating our Summary Statistics

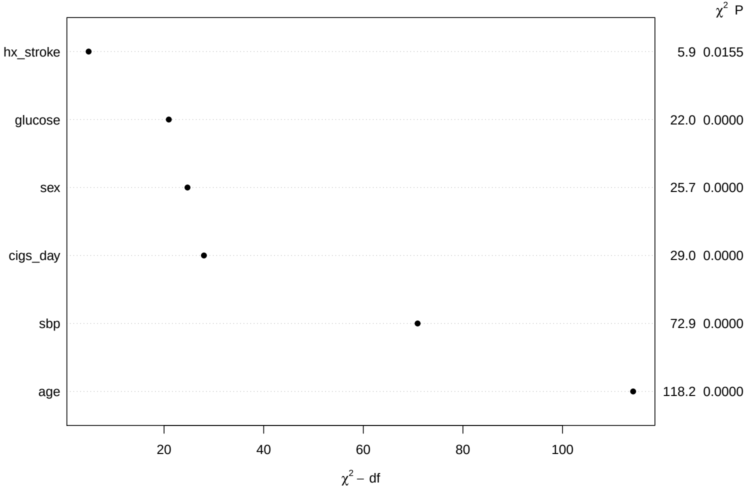
```
set.seed(432329); validate(m_04_lrm)[1:4,]
```

```
          index.orig  training        test    optimism
Dxy        0.4611110 0.4672261  0.45678154 0.010444568
R2         0.1575417 0.1629811  0.15375377 0.009227324
Intercept  0.0000000 0.0000000 -0.04796015 0.047960154
Slope      1.0000000 1.0000000  0.96803833 0.031961665
          index.corrected  n
Dxy            0.45066647 40
R2             0.14831438 40
Intercept     -0.04796015 40
Slope          0.96803833 40
```

plot(summary(m_04_lrm))

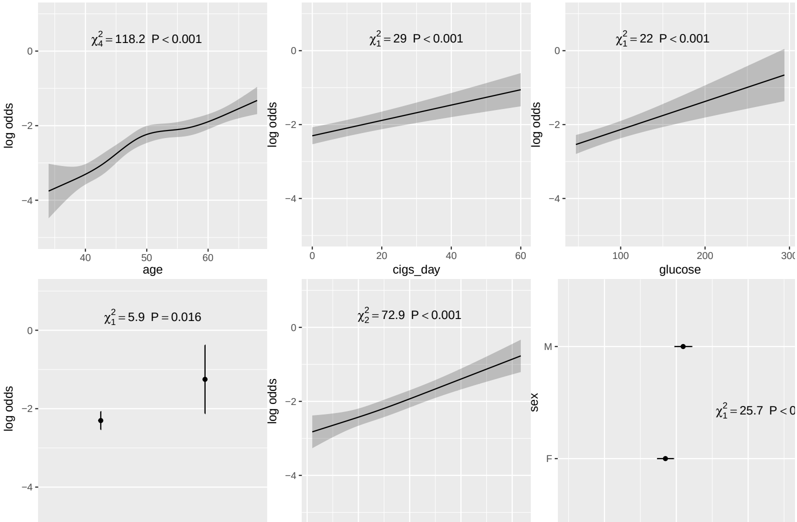


plot(anova(m_04_lrm))



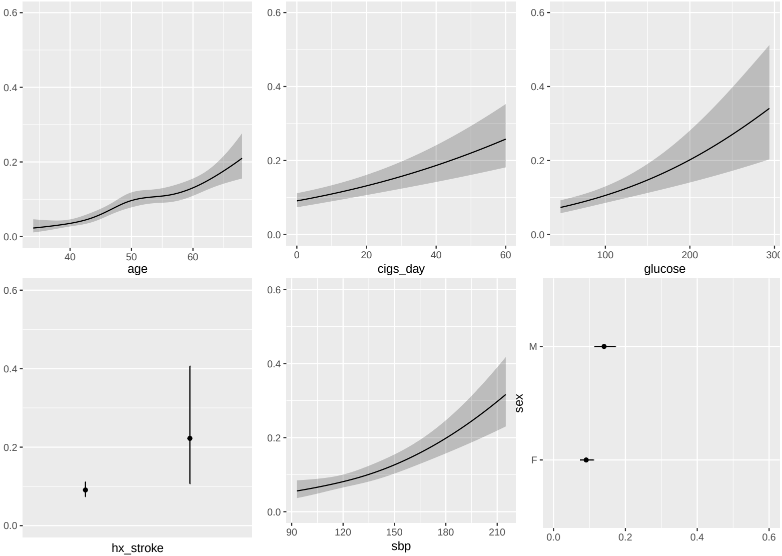
Can we see the prediction results?

```
ggplot(Predict(m_04_lrm),
       anova = anova(m_04_lrm), pval = TRUE)
```



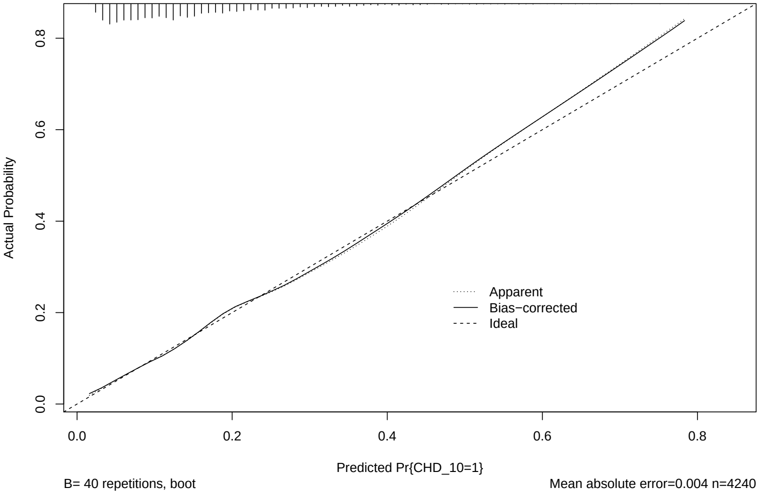
What about on a better scale?

```
ggplot(Predict(m_04_lrm, fun = plogis))
```



Calibration of `mod_04_lrm`

```
set.seed(432612); plot(calibrate(m_04_lrm))
```



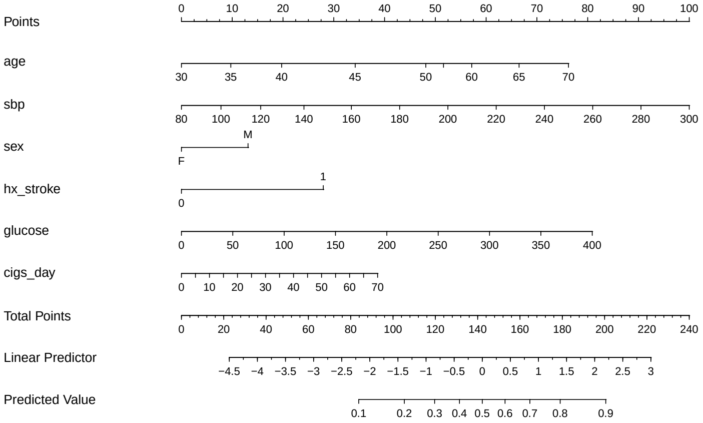
Goodness of fit test?

```
round(residuals(m_04_lrm, type = "gof"),3)
```

```
Sum of squared errors     Expected value|H0
              488.554                489.423
                   SD                      Z
                1.183                 -0.735
                    P
                0.462
```

Nomogram of `mod_04_lrm`

```
plot(nomogram(m_04_lrm, fun = plogis))
```



Comparing Models 3 and 4 (which aren't nested)

```r
glance(m_03) # kitchen sink but no non-linear terms
```

```
  null.deviance df.null    logLik      AIC      BIC
1      3612.209    4239 -1603.407 3242.813 3357.155
  deviance df.residual
1 3206.813        4222
```

```r
glance(m_04) # six predictors but with non-linear terms
```

```
  null.deviance df.null    logLik      AIC     BIC deviance
1      3612.209    4239 -1605.382 3232.764 3302.64 3210.764
  df.residual
1        4229
```

Checking Residuals?

- Yes/No outcomes contain less information than quantitative outcomes
- Residuals cannot be observed - predicted
  - There are several different types of residuals defined
- Assumptions of logistic regression are different
  - Model is deliberately non-linear
  - Error variance is a function of the mean, so it isn't constant
  - Errors aren't assumed to follow a Normal distribution
  - Only thing that's the same: leverage and influence

So, plot 5 (residuals/leverage/influence) can be a little useful, but that's it.

- We'll need better diagnostic tools for generalized linear models.

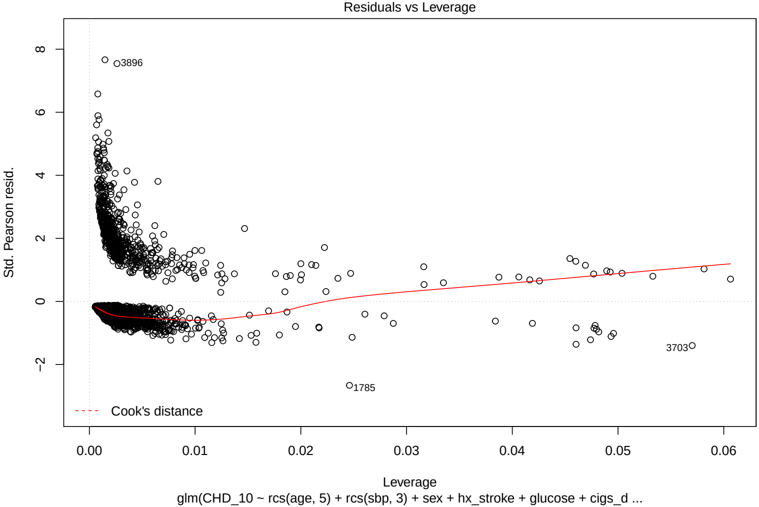
Any observations particularly influential on Model 4?

```r
which.influence(m_04_lrm, cutoff = 0.3)
```

```
$Intercept
[1] "3896"

$glucose
[1] "1785" "3703"
```

Influence and Model 4?

```
plot(m_04, which = 5)
```



The HERS Trial

Hormone Therapy and Baseline LDL in the HERS Trial

HERS clinical trial of hormone therapy (HT). We're excluding the women with diabetes here.

```
hers <- read.csv("data/hersdata.csv") %>% tbl_df

hers1 <- hers %>%
    filter(diabetes == "no") %>%
    select(subject, LDL, HT, age, smoking, drinkany, SBP,
           physact, BMI, diabetes)
```

The Data

```r
head(hers1)
```

```
# A tibble: 6 x 10
  subject   LDL HT        age smoking drinkany   SBP physact
    <int> <dbl> <fct>   <int> <fct>   <fct>    <int> <fct>  
1       1   122 place~     70 no      no         138 much m~
2       2   242 place~     62 no      no         118 much l~
3       4   116 place~     64 yes     yes        152 much l~
4       5   151 place~     65 no      no         175 somewh~
5       6   138 hormo~     68 no      yes        174 about ~
6       8   121 hormo~     69 no      no         178 much m~
# ... with 2 more variables: BMI <dbl>, diabetes <fct>
```

The Codebook (n = 2,032 women without diabetes)

Variable	Description	Missing?
`subject`	subject code	0
`LDL`	LDL cholesterol in mg/dl	7
`HT`	factor: hormone therapy or placebo	0
`age`	age in years	0
`smoking`	yes or no	0
`drinkany`	yes or no	2
`SBP`	systolic BP in mm Hg	0
`physact`	5-level factor	0
`BMI`	body-mass index in kg/m^2	2
`diabetes`	yes or no (all of these are no)	0

Our Modeling Goal

Predict LDL using

- age
- smoking
- drinkany
- SBP
- physact
- BMI
- the interaction of smoking and BMI

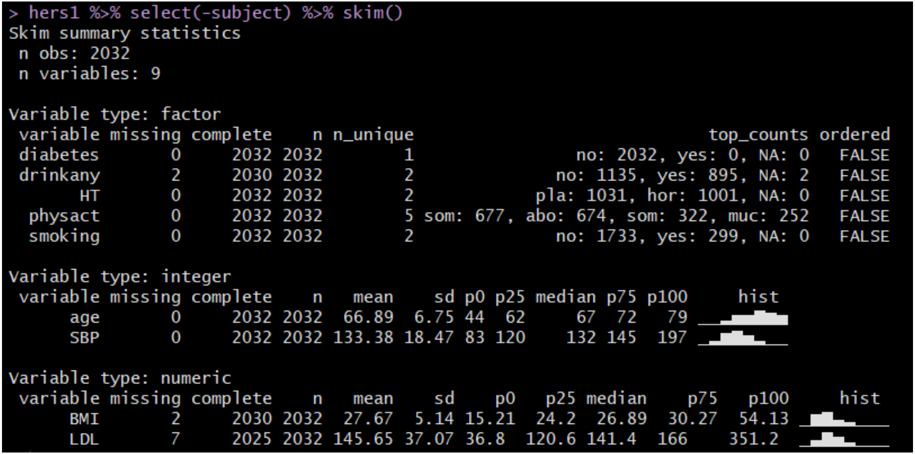
Details on physact variable

```
hers1 %>% count(physact)
```

```
# A tibble: 5 x 2
  physact                  n
  <fct>                <int>
1 about as active        674
2 much less active       107
3 much more active       252
4 somewhat less active   322
5 somewhat more active   677
```

Skim?

```r
hers1 %>% select(-subject) %>% skim()
```



```
> hers1 %>% select(-subject) %>% skim()
Skim summary statistics
 n obs: 2032
 n variables: 9

Variable type: factor
 variable missing complete    n n_unique                             top_counts ordered
 diabetes       0     2032 2032        1           no: 2032, yes: 0, NA: 0   FALSE
 drinkany       2     2030 2032        2         no: 1135, yes: 895, NA: 2   FALSE
       HT       0     2032 2032        2       pla: 1031, hor: 1001, NA: 0   FALSE
  physact       0     2032 2032        5 som: 677, abo: 674, som: 322, muc: 252   FALSE
  smoking       0     2032 2032        2         no: 1733, yes: 299, NA: 0   FALSE

Variable type: integer
 variable missing complete    n   mean    sd p0 p25 median p75 p100     hist
      age       0     2032 2032  66.89  6.75 44  62     67  72   79
      SBP       0     2032 2032 133.38 18.47 83 120    132 145  197

Variable type: numeric
 variable missing complete    n   mean    sd    p0   p25 median   p75   p100     hist
      BMI       2     2030 2032  27.67  5.14 15.21  24.2  26.89 30.27  54.13
      LDL       7     2025 2032 145.65 37.07 36.8  120.6 141.4  166    351.2
```

Missingness pattern?

```r
na.pattern(hers1) # from Hmisc
```

```
pattern
0000000000 0000000010 0000010000 0100000000 
      2021          2          2          7 
```

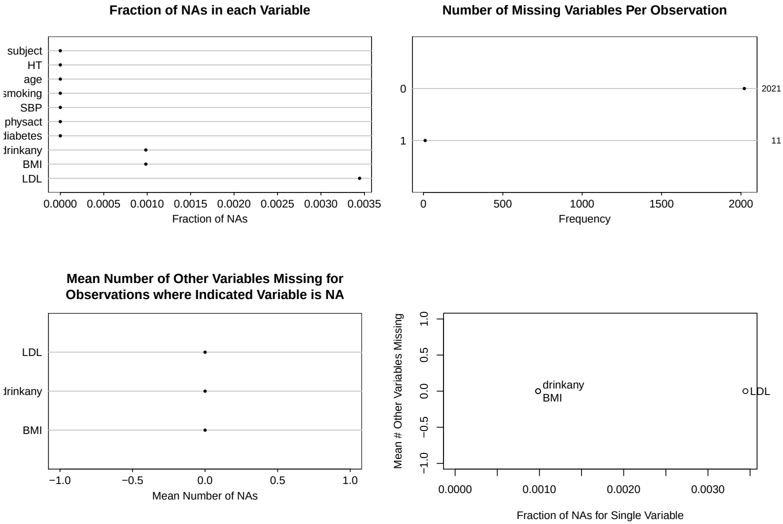
```r
names(hers1)
```

```
 [1] "subject"  "LDL"      "HT"       "age"      "smoking" 
 [6] "drinkany" "SBP"      "physact"  "BMI"      "diabetes"
```

Next slide

```r
par(mfrow = c(2,2))
naplot(naclus(hers1))
par(mfrow = c(1,1))
```

naplot(naclus(hers1))



Simple Imputation into `hers2`

Simple Imputation for drinkany, BMI and LDL

Since drinkany is a factor, we have to do some extra work to impute.

```
set.seed(432092)

hers2 <- hers1 %>%
    mutate(drinkany_n =
               ifelse(drinkany == "yes", 1, 0)) %>%
    impute_pmm(drinkany_n ~ age + smoking) %>%
    mutate(drinkany =
               ifelse(drinkany_n == 1, "yes", "no")) %>%
    impute_rlm(BMI ~ age + smoking + SBP) %>%
    impute_rlm(LDL ~ age + smoking + SBP + BMI)
```

Now, check missingness. . .

```r
na.pattern(hers2)
```

```
pattern
00000000000 
       2032 
```

```r
names(hers2)
```

```
 [1] "subject"    "LDL"        "HT"         "age"       
 [5] "smoking"    "drinkany"   "SBP"        "physact"   
 [9] "BMI"        "diabetes"   "drinkany_n"
```

Multiple Imputation with `aregImpute`

Multiple Imputation using `aregImpute` from `Hmisc`

Model to predict all missing values of any variables, using additive regression bootstrapping and predictive mean matching.

Steps are:

1. `aregImpute` draws a sample with replacement from the observations where the target variable is observed, not missing.
2. It then fits a flexible additive model to predict this target variable while finding the optimum transformation of it.
3. It then uses this fitted flexible model to predict the target variable in all of the original observations.
4. Finally, it imputes each missing value of the target variable with the observed value whose predicted transformed value is closest to the predicted transformed value of the missing value.

Fitting a Multiple Imputation Model

```r
set.seed(4320132)
dd <- datadist(hers1)
options(datadist = "dd")
fit3 <- aregImpute(~ LDL + age + smoking + drinkany +
                        SBP + physact + BMI,
                     nk = c(0, 3:5), tlinear = FALSE,
                     data = hers1, B = 10, n.impute = 20)
```

```
Iteration 1
Iteration 2
Iteration 3
Iteration 4
Iteration 5
Iteration 6
Iteration 7
Iteration 8
Iteration 9
```

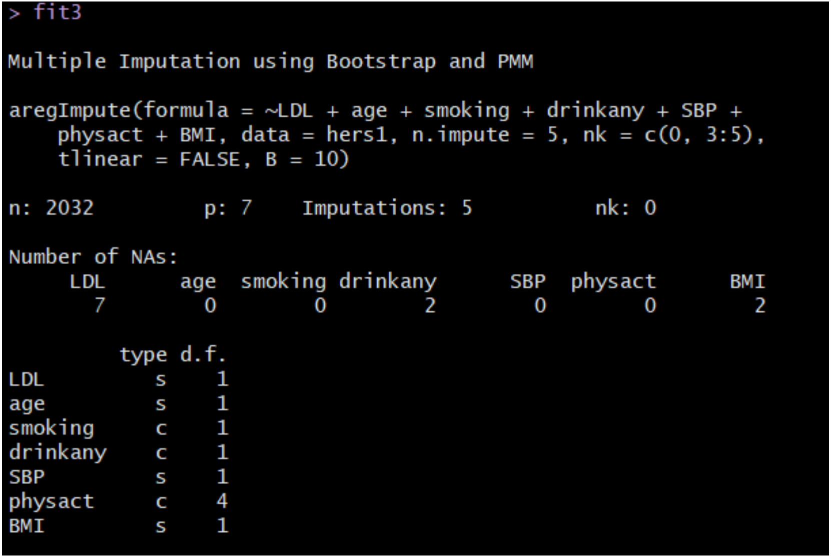
Multiple Imputation using `aregImpute` from `Hmisc`

`aregImpute` requires specifications of all variables, and several other details:

- `n.impute` = number of imputations, we'll run 20
- `nk` = number of knots to describe level of complexity, with our choice `nk = c(0, 3:5)` we'll fit both linear models and models with restricted cubic splines with 3, 4, and 5 knots
- `tlinear = FALSE` allows the target variable to have a non-linear transformation when `nk` is 3 or more
- `B = 10` specifies 10 bootstrap samples will bs used
- `data` specifies the source of the variables

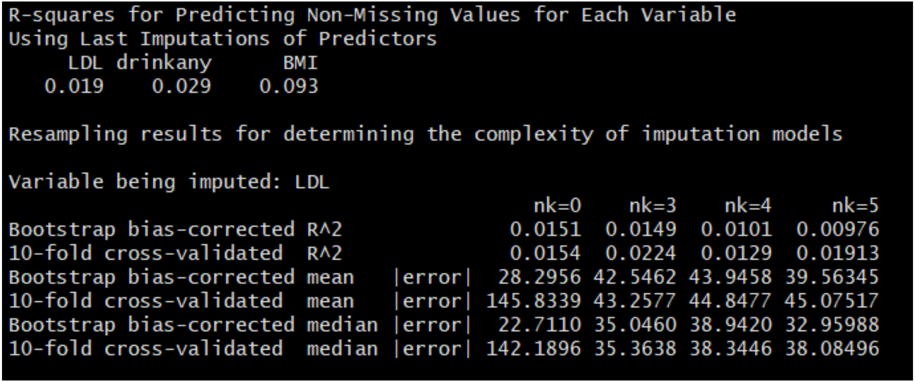
aregImpute Imputation Results (1 of 3)

```
fit3
```



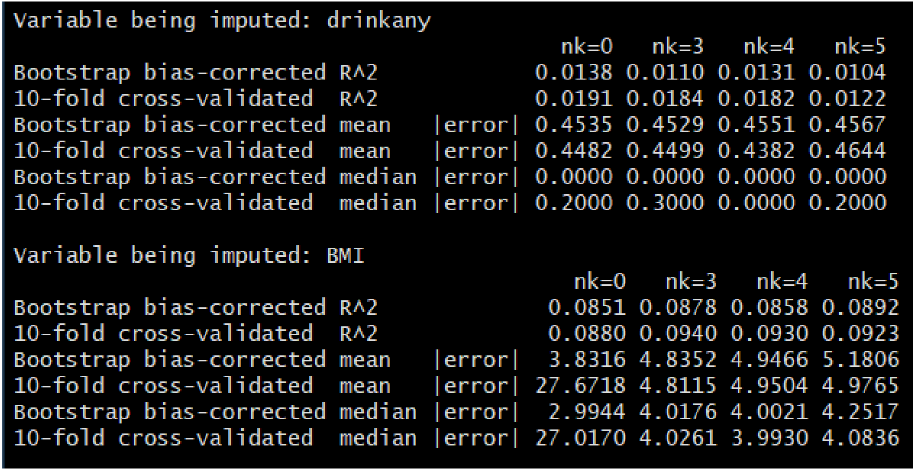
```
> fit3

Multiple Imputation using Bootstrap and PMM

aregImpute(formula = ~LDL + age + smoking + drinkany + SBP +
    physact + BMI, data = hers1, n.impute = 5, nk = c(0, 3:5),
    tlinear = FALSE, B = 10)

n: 2032         p: 7    Imputations: 5          nk: 0

Number of NAs:
     LDL      age  smoking drinkany      SBP  physact      BMI
       7        0        0        2        0        0        2

         type d.f.
LDL         s    1
age         s    1
smoking     c    1
drinkany    c    1
SBP         s    1
physact     c    4
BMI         s    1
```

aregImpute Imputation Results (2 of 3)



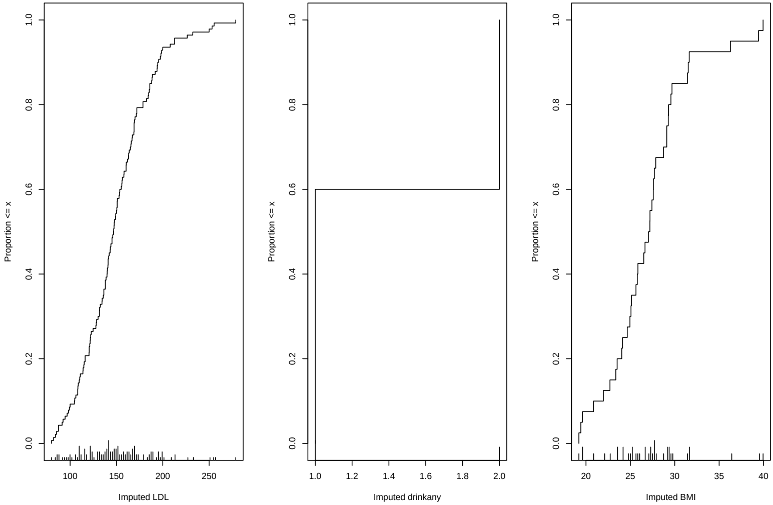
```
R-squares for Predicting Non-Missing Values for Each Variable
Using Last Imputations of Predictors
     LDL drinkany      BMI
   0.019    0.029    0.093

Resampling results for determining the complexity of imputation models

Variable being imputed: LDL
                                             nk=0    nk=3    nk=4     nk=5
Bootstrap bias-corrected R^2               0.0151  0.0149  0.0101  0.00976
10-fold cross-validated  R^2               0.0154  0.0224  0.0129  0.01913
Bootstrap bias-corrected mean   |error|   28.2956 42.5462 43.9458 39.56345
10-fold cross-validated  mean   |error|  145.8339 43.2577 44.8477 45.07517
Bootstrap bias-corrected median |error|   22.7110 35.0460 38.9420 32.95988
10-fold cross-validated  median |error|  142.1896 35.3638 38.3446 38.08496
```

aregImpute Imputation Results (3 of 3)



```
Variable being imputed: drinkany
                                          nk=0   nk=3   nk=4   nk=5
Bootstrap bias-corrected R^2            0.0138 0.0110 0.0131 0.0104
10-fold cross-validated  R^2            0.0191 0.0184 0.0182 0.0122
Bootstrap bias-corrected mean   |error| 0.4535 0.4529 0.4551 0.4567
10-fold cross-validated  mean   |error| 0.4482 0.4499 0.4382 0.4644
Bootstrap bias-corrected median |error| 0.0000 0.0000 0.0000 0.0000
10-fold cross-validated  median |error| 0.2000 0.3000 0.0000 0.2000

Variable being imputed: BMI
                                           nk=0   nk=3   nk=4   nk=5
Bootstrap bias-corrected R^2             0.0851 0.0878 0.0858 0.0892
10-fold cross-validated  R^2             0.0880 0.0940 0.0930 0.0923
Bootstrap bias-corrected mean   |error|  3.8316 4.8352 4.9466 5.1806
10-fold cross-validated  mean   |error| 27.6718 4.8115 4.9504 4.9765
Bootstrap bias-corrected median |error|  2.9944 4.0176 4.0021 4.2517
10-fold cross-validated  median |error| 27.0170 4.0261 3.9930 4.0836
```

A plot of the imputed values. . . (results)



A plot of the imputed values. . . (code)

```r
par(mfrow = c(1,3))
plot(fit3)
par(mfrow = c(1,1))
```

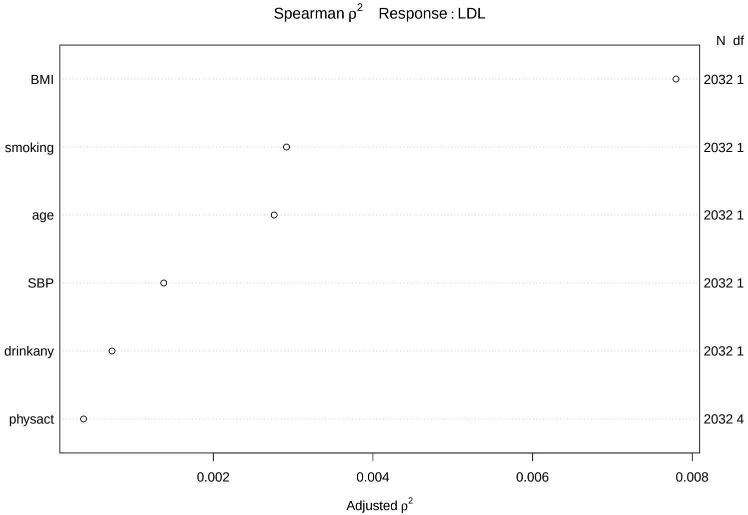
- For LDL, we imputed most of the 7 missing subjects in most of the 20 imputation runs to values within a range of around 120 through 200, but occasionally, we imputed values that were substantially lower than 100.
- For drinkany we imputed about 70% no and 30% yes.
- For BMI, we imputed values ranging from about 23 to 27 in many cases, and up near 40 in other cases.
- This method never imputes a value for a variable that doesn't already exist in the data.

Spearman ρ^2 Plot

We've already decided to include a BMI*smoking product term, but how should we prioritize the degrees of freedom we spend on non-linearity otherwise?

```
plot(spearman2(LDL ~ age + smoking + drinkany + SBP +
                  physact + BMI, data = hers2))
```

Spearman ρ^2 Plot Result



Fitting a Linear Regression with `ols`

Model we'll fit

Fitting a model to predict `LDL` using

- `BMI` with a restricted cubic spline, 5 knots
- `age` with a quadratic polynomial
- `SBP` as a linear term
- `drinkany` indicator
- `physact` factor
- `smoking` indicator and its interaction with `BMI`

We could fit this to the data

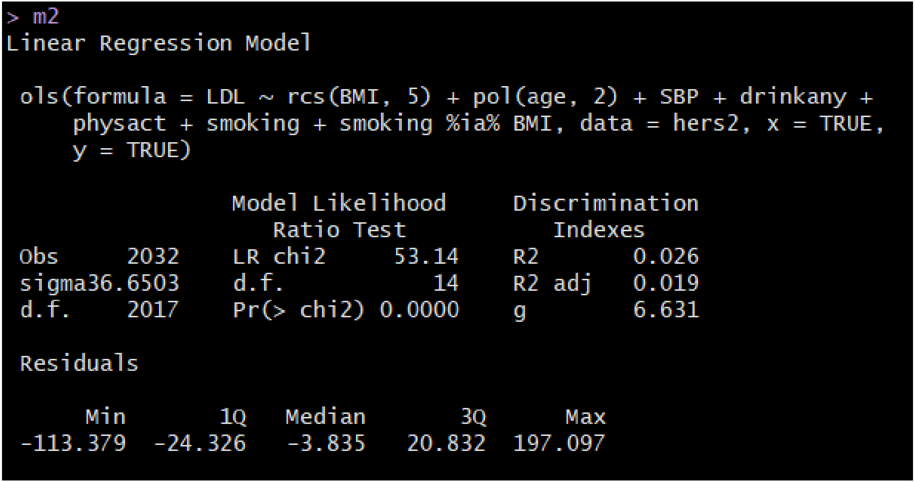
- restricted to complete cases (hers1, effectively)
- after simple imputation (hers2)
- after our multiple imputation (fit3)

Fitting the model after simple imputation

```
dd <- datadist(hers2)
options(datadist = "dd")

m2 <- ols(LDL ~ rcs(BMI, 5) + pol(age, 2) + SBP +
              drinkany + physact + smoking +
              smoking %ia% BMI, data = hers2,
          x = TRUE, y = TRUE)
```

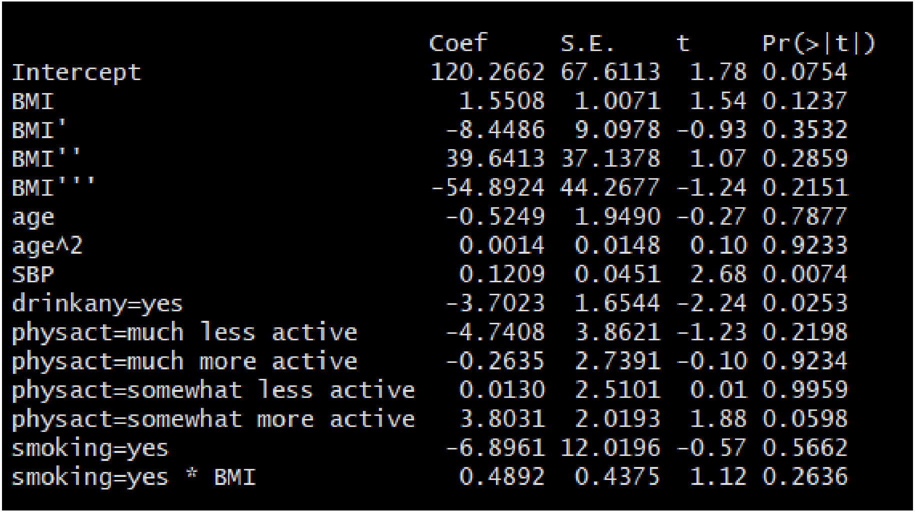
where %ia% identifies the linear interaction alone.

`m2` results (slide 1 of 2)



```
> m2
Linear Regression Model

 ols(formula = LDL ~ rcs(BMI, 5) + pol(age, 2) + SBP + drinkany +
     physact + smoking + smoking %ia% BMI, data = hers2, x = TRUE,
     y = TRUE)

                Model Likelihood     Discrimination
                   Ratio Test           Indexes
 Obs      2032    LR chi2     53.14    R2       0.026
 sigma36.6503    d.f.           14    R2 adj   0.019
 d.f.     2017    Pr(> chi2) 0.0000    g        6.631

 Residuals

      Min       1Q   Median      3Q     Max
 -113.379  -24.326   -3.835  20.832 197.097
```

m2 results (slide 2 of 2)



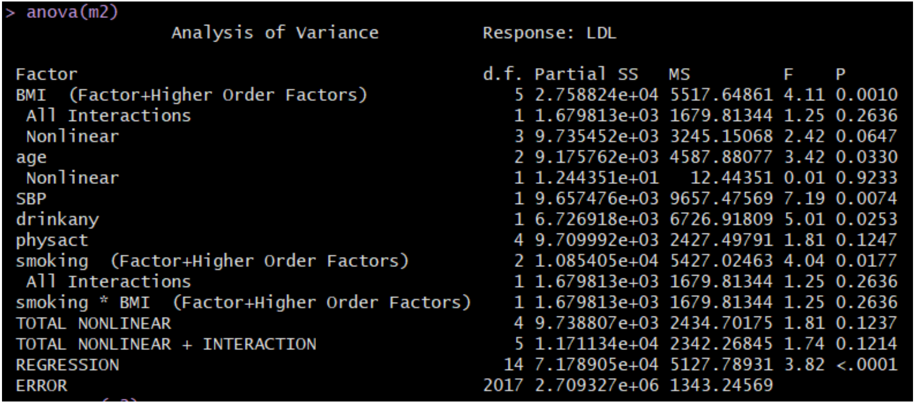
```
                               Coef     S.E.    t     Pr(>|t|)
Intercept                      120.2662 67.6113  1.78 0.0754
BMI                              1.5508  1.0071  1.54 0.1237
BMI'                            -8.4486  9.0978 -0.93 0.3532
BMI''                           39.6413 37.1378  1.07 0.2859
BMI'''                         -54.8924 44.2677 -1.24 0.2151
age                             -0.5249  1.9490 -0.27 0.7877
age^2                            0.0014  0.0148  0.10 0.9233
SBP                              0.1209  0.0451  2.68 0.0074
drinkany=yes                    -3.7023  1.6544 -2.24 0.0253
physact=much less active        -4.7408  3.8621 -1.23 0.2198
physact=much more active        -0.2635  2.7391 -0.10 0.9234
physact=somewhat less active     0.0130  2.5101  0.01 0.9959
physact=somewhat more active     3.8031  2.0193  1.88 0.0598
smoking=yes                     -6.8961 12.0196 -0.57 0.5662
smoking=yes * BMI                0.4892  0.4375  1.12 0.2636
```

Validation of summary statistics

```r
validate(m2)
```

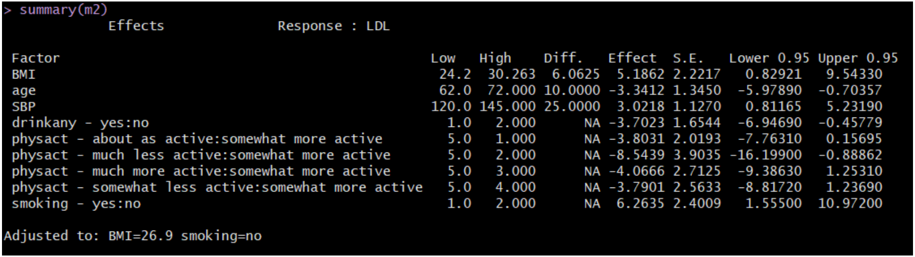
```
          index.orig  training      test optimism
R-square      0.0258    0.0307    0.0188   0.0119
MSE        1333.3300 1320.0677 1342.9027 -22.8350
g             6.6306    7.1548    5.8726   1.2821
Intercept     0.0000    0.0000   26.2153 -26.2153
Slope         1.0000    1.0000    0.8208   0.1792
          index.corrected  n
R-square           0.0139 40
MSE             1356.1650 40
g                  5.3485 40
Intercept         26.2153 40
Slope              0.8208 40
```

anova(m2) results



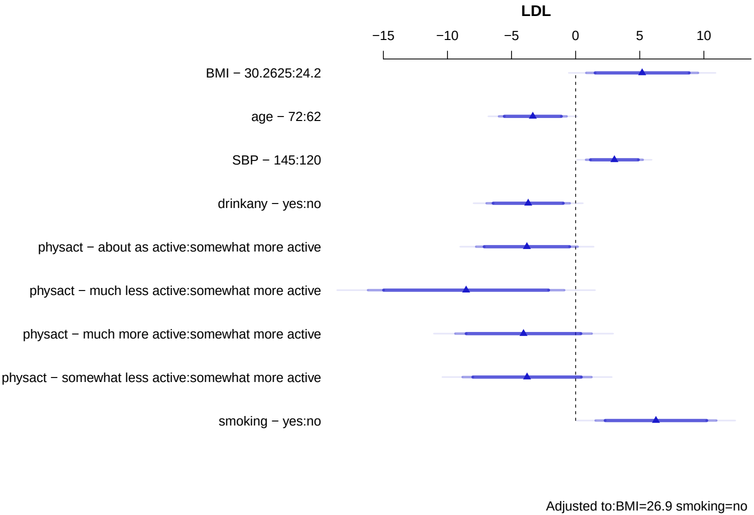
```
> anova(m2)
                Analysis of Variance          Response: LDL

 Factor                                           d.f. Partial SS   MS          F    P
 BMI  (Factor+Higher Order Factors)                  5 2.758824e+04 5517.64861 4.11 0.0010
  All Interactions                                   1 1.679813e+03 1679.81344 1.25 0.2636
  Nonlinear                                          3 9.735452e+03 3245.15068 2.42 0.0647
 age                                                 2 9.175762e+03 4587.88077 3.42 0.0330
  Nonlinear                                          1 1.244351e+01   12.44351 0.01 0.9233
 SBP                                                 1 9.657476e+03 9657.47569 7.19 0.0074
 drinkany                                            1 6.726918e+03 6726.91809 5.01 0.0253
 physact                                             4 9.709992e+03 2427.49791 1.81 0.1247
 smoking  (Factor+Higher Order Factors)              2 1.085405e+04 5427.02463 4.04 0.0177
  All Interactions                                   1 1.679813e+03 1679.81344 1.25 0.2636
 smoking * BMI  (Factor+Higher Order Factors)        1 1.679813e+03 1679.81344 1.25 0.2636
 TOTAL NONLINEAR                                     4 9.738807e+03 2434.70175 1.81 0.1237
 TOTAL NONLINEAR + INTERACTION                       5 1.171134e+04 2342.26845 1.74 0.1214
 REGRESSION                                         14 7.178905e+04 5127.78931 3.82 <.0001
 ERROR                                            2017 2.709327e+06 1343.24569
```

`summary(m2)` results

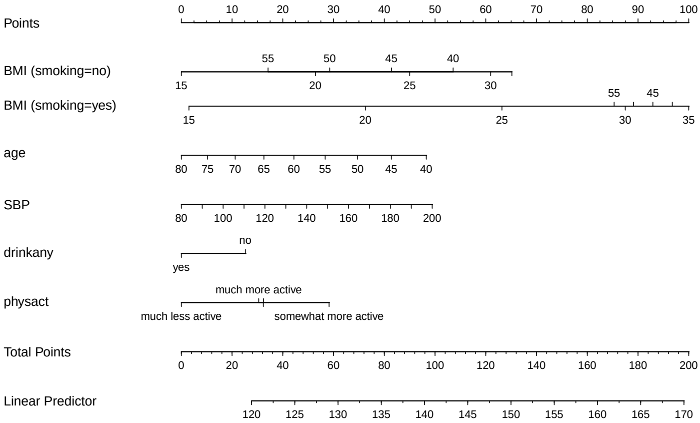


```
> summary(m2)
             Effects              Response : LDL

 Factor                                              Low   High    Diff.   Effect  S.E.   Lower 0.95 Upper 0.95
 BMI                                                  24.2  30.263  6.0625  5.1862 2.2217   0.82921    9.54330
 age                                                  62.0  72.000 10.0000 -3.3412 1.3450  -5.97890   -0.70357
 SBP                                                 120.0 145.000 25.0000  3.0218 1.1270   0.81165    5.23190
 drinkany - yes:no                                     1.0   2.000      NA -3.7023 1.6544  -6.94690   -0.45779
 physact - about as active:somewhat more active        5.0   1.000      NA -3.8031 2.0193  -7.76310    0.15695
 physact - much less active:somewhat more active       5.0   2.000      NA -8.5439 3.9035 -16.19900   -0.88862
 physact - much more active:somewhat more active       5.0   3.000      NA -4.0666 2.7125  -9.38630    1.25310
 physact - somewhat less active:somewhat more active   5.0   4.000      NA -3.7901 2.5633  -8.81720    1.23690
 smoking - yes:no                                      1.0   2.000      NA  6.2635 2.4009   1.55500   10.97200

Adjusted to: BMI=26.9 smoking=no
```

plot(summary(m2)) results



plot(nomogram(m2))



Making Predictions for an Individual

Suppose now that we want to use R to get a prediction for a new individual subject with `BMI` = 30, age = 50, `smoking` = yes and `physact` = about as active, `drinkany`= yes and SBP of 150.

```r
predict(m2, expand.grid(BMI = 30, age = 50, smoking = "yes",
                        physact = "about as active",
                        drinkany = "yes", SBP = 150),
        conf.int = 0.95, conf.type = "individual")
```

```
$linear.predictors           $lower      $upper
          160.9399         88.48615    233.3936
```

Making Predictions for a Long-Run Mean

The other kind of prediction we might wish to make is for the mean of a series of subjects whose `BMI` = 30, age = 50, `smoking` = yes and `physact` = about as active, `drinkany`= yes and `SBP` of 150.

```
predict(m2, expand.grid(BMI = 30, age = 50, smoking = "yes",
                        physact = "about as active",
                        drinkany = "yes", SBP = 150),
        conf.int = 0.95, conf.type = "mean")
```

```
$linear.predictors          $lower      $upper
          160.9399        151.8119    170.0679
```

Of course, the confidence interval will always be narrower than the prediction interval given the same predictor values.

Influential Points?

```
which.influence(m2, cutoff = 0.4)
```

```
$Intercept
[1] 1135

$age
[1] 1135

$smoking
[1] 132

$`smoking * BMI`
[1] 132
```

Fitting the model to the complete cases

```
d <- datadist(hers1)
options(datadist = "d")

m1 <- ols(LDL ~ rcs(BMI, 5) + pol(age, 2) + SBP +
              drinkany + physact + smoking +
              smoking %ia% BMI, data = hers1,
          x = TRUE, y = TRUE)
```

where %ia% identifies the linear interaction alone.

Putting it Together

What have we got?

- An imputation model `fit3`

```
fit3 <- aregImpute(~ LDL + age + smoking + drinkany + SBP +
           physact + BMI, nk = c(0, 3:5), tlinear = FALSE,
           data = hers1, B = 10, n.impute = 20, x = TRUE)
```

- A prediction model

```
m1 <- ols(LDL ~ rcs(BMI, 5) + pol(age, 2) + SBP +
            drinkany + physact + smoking + smoking %ia% BMI,
            x = TRUE, y = TRUE)
```

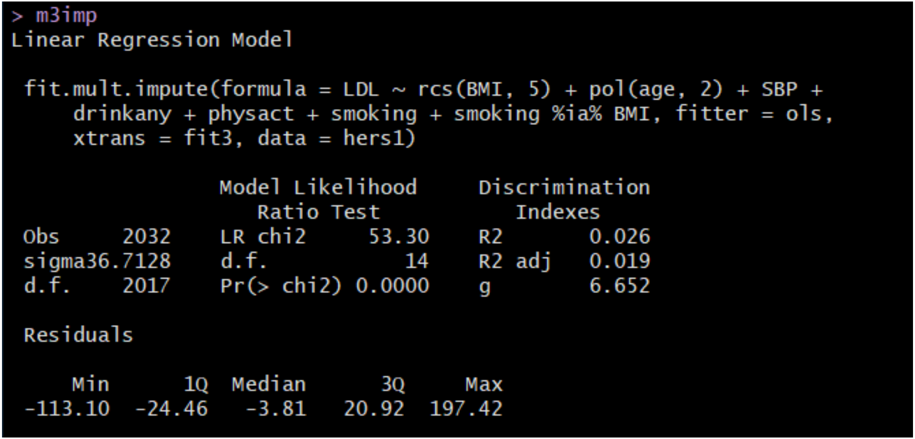
Now we put them together

Linear Regression & Imputation Model

```r
m3imp <-
    fit.mult.impute(LDL ~ rcs(BMI, 5) + pol(age, 2) + SBP +
                        drinkany + physact + smoking +
                        smoking %ia% BMI,
                    fitter = ols, xtrans = fit3,
                    data = hers1)
```

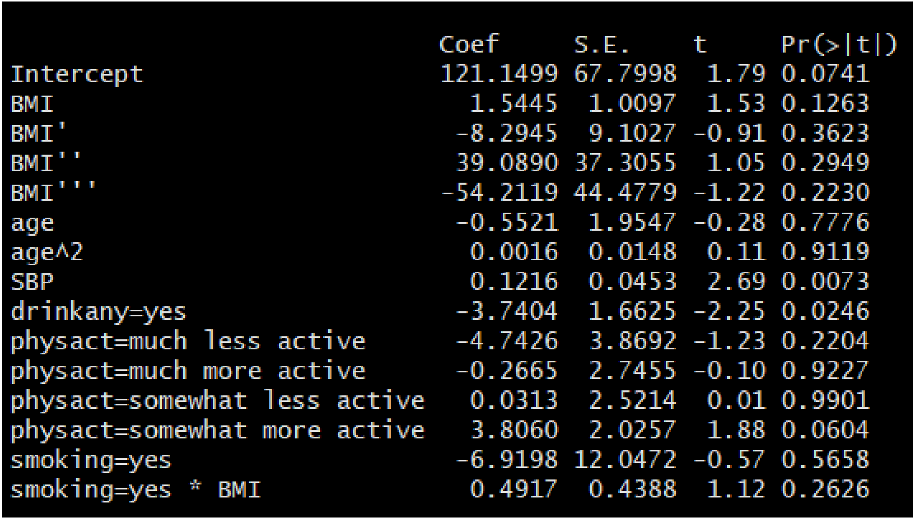
```
Variance Inflation Factors Due to Imputation:

                   Intercept                          BMI
                        1.00                         1.00
                        BMI'                        BMI''
                        1.00                         1.00
                      BMI'''                          age
                        1.00                         1.00
                       age^2                          SBP
```

`m3imp` results (1 of 2)



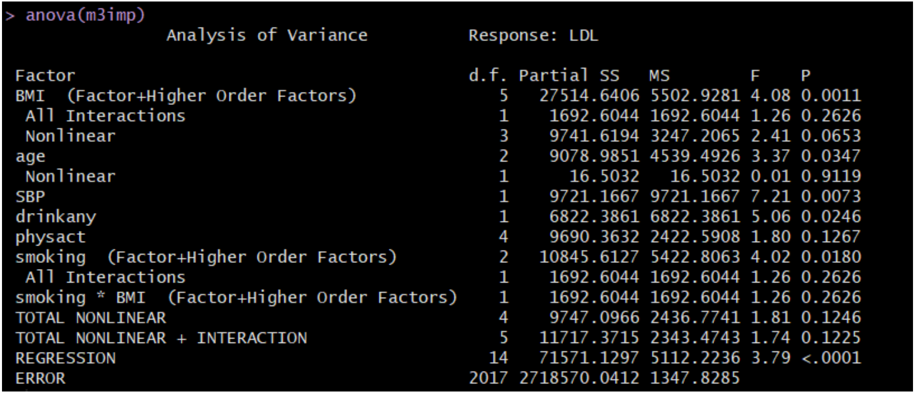
```
> m3imp
Linear Regression Model

 fit.mult.impute(formula = LDL ~ rcs(BMI, 5) + pol(age, 2) + SBP +
     drinkany + physact + smoking + smoking %ia% BMI, fitter = ols,
     xtrans = fit3, data = hers1)

                 Model Likelihood     Discrimination
                    Ratio Test           Indexes
 Obs      2032    LR chi2     53.30     R2       0.026
 sigma36.7128    d.f.           14     R2 adj   0.019
 d.f.     2017    Pr(> chi2) 0.0000     g        6.652

 Residuals

     Min      1Q  Median      3Q     Max
 -113.10  -24.46   -3.81   20.92  197.42
```

`m3imp` results (2 of 2)



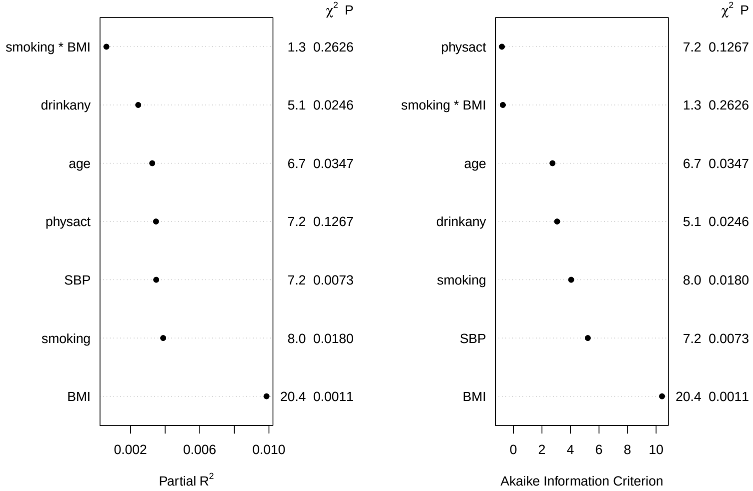
```
                                    Coef     S.E.    t     Pr(>|t|)
Intercept                           121.1499 67.7998  1.79 0.0741
BMI                                   1.5445  1.0097  1.53 0.1263
BMI'                                 -8.2945  9.1027 -0.91 0.3623
BMI''                                39.0890 37.3055  1.05 0.2949
BMI'''                              -54.2119 44.4779 -1.22 0.2230
age                                  -0.5521  1.9547 -0.28 0.7776
age^2                                 0.0016  0.0148  0.11 0.9119
SBP                                   0.1216  0.0453  2.69 0.0073
drinkany=yes                         -3.7404  1.6625 -2.25 0.0246
physact=much less active             -4.7426  3.8692 -1.23 0.2204
physact=much more active             -0.2665  2.7455 -0.10 0.9227
physact=somewhat less active          0.0313  2.5214  0.01 0.9901
physact=somewhat more active          3.8060  2.0257  1.88 0.0604
smoking=yes                          -6.9198 12.0472 -0.57 0.5658
smoking=yes * BMI                     0.4917  0.4388  1.12 0.2626
```

anova(m3imp)



```
> anova(m3imp)
                Analysis of Variance          Response: LDL

 Factor                                       d.f. Partial SS   MS          F    P
 BMI  (Factor+Higher Order Factors)              5   27514.6406 5502.9281 4.08 0.0011
  All Interactions                               1    1692.6044 1692.6044 1.26 0.2626
  Nonlinear                                      3    9741.6194 3247.2065 2.41 0.0653
 age                                             2    9078.9851 4539.4926 3.37 0.0347
  Nonlinear                                      1      16.5032   16.5032 0.01 0.9119
 SBP                                             1    9721.1667 9721.1667 7.21 0.0073
 drinkany                                        1    6822.3861 6822.3861 5.06 0.0246
 physact                                         4    9690.3632 2422.5908 1.80 0.1267
 smoking  (Factor+Higher Order Factors)          2   10845.6127 5422.8063 4.02 0.0180
  All Interactions                               1    1692.6044 1692.6044 1.26 0.2626
 smoking * BMI  (Factor+Higher Order Factors)    1    1692.6044 1692.6044 1.26 0.2626
 TOTAL NONLINEAR                                 4    9747.0966 2436.7741 1.81 0.1246
 TOTAL NONLINEAR + INTERACTION                   5   11717.3715 2343.4743 1.74 0.1225
 REGRESSION                                     14   71571.1297 5112.2236 3.79 <.0001
 ERROR                                        2017 2718570.0412 1347.8285
```

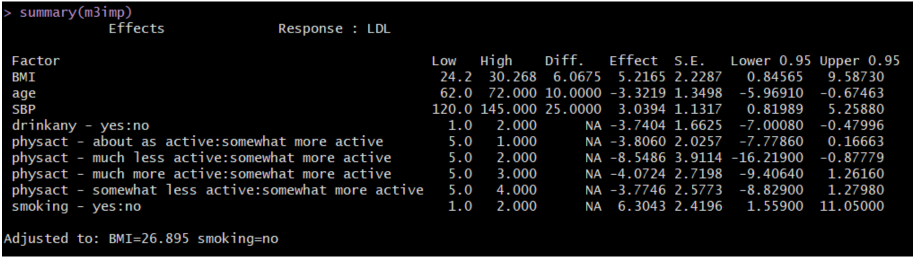
Evaluation via Partial R^2 and AIC (result)



Evaluation via Partial R^2 and AIC (code)

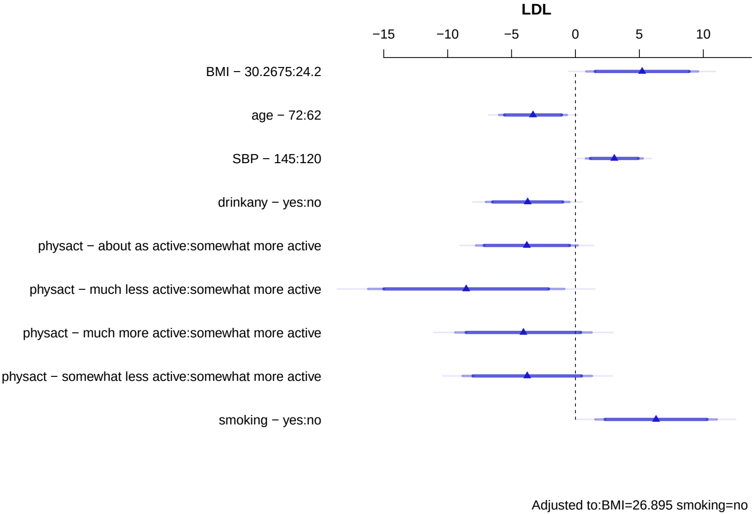
```
par(mfrow = c(1,2))
plot(anova(m3imp), what="partial R2")
plot(anova(m3imp), what="aic")
par(mfrow = c(1,1))
```

summary(m3imp)

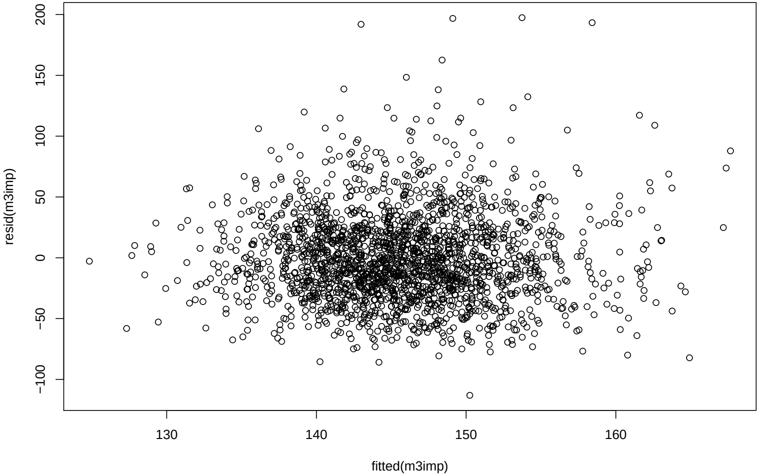


```
> summary(m3imp)
             Effects              Response : LDL

 Factor                                            Low   High    Diff.   Effect  S.E.   Lower 0.95 Upper 0.95
 BMI                                                24.2  30.268  6.0675  5.2165 2.2287   0.84565    9.58730
 age                                                62.0  72.000 10.0000 -3.3219 1.3498  -5.96910   -0.67463
 SBP                                               120.0 145.000 25.0000  3.0394 1.1317   0.81989    5.25880
 drinkany - yes:no                                   1.0   2.000      NA -3.7404 1.6625  -7.00080   -0.47996
 physact - about as active:somewhat more active      5.0   1.000      NA -3.8060 2.0257  -7.77860    0.16663
 physact - much less active:somewhat more active     5.0   2.000      NA -8.5486 3.9114 -16.21900   -0.87779
 physact - much more active:somewhat more active     5.0   3.000      NA -4.0724 2.7198  -9.40640    1.26160
 physact - somewhat less active:somewhat more active 5.0   4.000      NA -3.7746 2.5773  -8.82900    1.27980
 smoking - yes:no                                    1.0   2.000      NA  6.3043 2.4196   1.55900   11.05000

Adjusted to: BMI=26.895 smoking=no
```

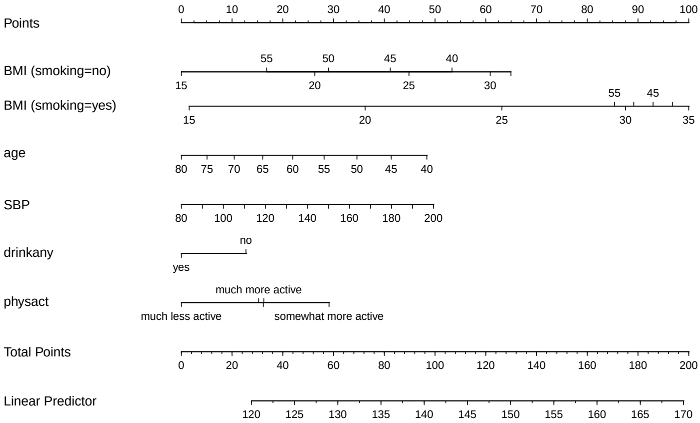
`plot(summary(m3imp))`



plot(resid(m1imp) ~ fitted(m1imp))



plot(nomogram(m3imp))


Points
BMI (smoking=no)
BMI (smoking=yes)
age
SBP
drinkany
no
yes
physact
much more active
much less active
somewhat more active
Total Points
Linear Predictor