432 Class 06

https://thomaselove.github.io/432-2024/

2024-02-01

## Today’s Agenda

* Data from the Heart and Estrogen/Progestin Study
* Using Spearman’s to guide decisions about spending degrees of freedom
* Using ols to fit linear regression models in the presence of missing values
* Using aregImpute to facilitate principled multiple imputation when fitting regressions
* Developing detailed regression results under a variety of imputation plans

## Today’s R Setup

knitr::opts\_chunk$set(comment = NA)  
  
library(janitor)  
library(broom)  
library(gt)  
library(naniar)  
library(simputation)  
library(rms)  
library(tidyverse)  
  
theme\_set(theme\_bw())

# The HERS Data

## Today’s Data

Heart and Estrogen/Progestin Study (HERS)

* Clinical trial of hormone therapy for the prevention of recurrent heart attacks and deaths among 2763 post-menopausal women with existing coronary heart disease (see Hulley et al 1998 and many subsequent references, including Vittinghoff, Chapter 4.)
* We’re excluding the women in the trial with a diabetes diagnosis.

## Ingesting the Data

hers\_raw <- read\_csv("c06/data/hersdata.csv", show\_col\_types = FALSE) |>   
 clean\_names() |> mutate\_if(is.character, as.factor)  
  
hers1 <- hers\_raw |>   
 filter(diabetes == "no") |>  
 mutate(subject = as.character(subject)) |>  
 select(subject, ldl, age, sbp, bmi, ht, smoking, drinkany,   
 physact, diabetes)  
  
dim(hers1)

[1] 2032 10

**Goal** Predict ldl using age, sbp, bmi, smoking, drinkany, and physact, across both HT levels but restricted to women without diabetes.

## Summary of data in hers1

summary(hers1)

subject ldl age sbp   
 Length:2032 Min. : 36.8 Min. :44.00 Min. : 83.0   
 Class :character 1st Qu.:120.6 1st Qu.:62.00 1st Qu.:120.0   
 Mode :character Median :141.4 Median :67.00 Median :132.0   
 Mean :145.6 Mean :66.89 Mean :133.4   
 3rd Qu.:166.0 3rd Qu.:72.00 3rd Qu.:145.0   
 Max. :351.2 Max. :79.00 Max. :197.0   
 NA's :7   
 bmi ht smoking drinkany   
 Min. :15.21 hormone therapy:1001 no :1733 no :1135   
 1st Qu.:24.20 placebo :1031 yes: 299 yes : 895   
 Median :26.89 NA's: 2   
 Mean :27.67   
 3rd Qu.:30.27   
 Max. :54.13   
 NA's :2   
 physact diabetes   
 about as active :674 no :2032   
 much less active :107 yes: 0   
 much more active :252   
 somewhat less active:322   
 somewhat more active:677

## hers1 Codebook (n = 2032)

| Variable | Description |
| --- | --- |
| subject | subject code |
| HT | factor: hormone therapy or placebo |
| diabetes | yes or no (all are no in our sample) |
| ldl | LDL cholesterol in mg/dl |
| age | age in years |
| sbp | systolic BP in mm Hg |
| bmi | body-mass index in kg/m2 |
| smoking | yes or no |
| drinkany | yes or no |
| physact | 5-level factor, details next slide |

## The physact variable

hers1 |> count(physact)

# A tibble: 5 × 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

Comparison is to activity levels for these women just before menopause.

## Any missing data?

miss\_var\_summary(hers1)

# A tibble: 10 × 3  
 variable n\_miss pct\_miss  
 <chr> <int> <dbl>  
 1 ldl 7 0.344   
 2 bmi 2 0.0984  
 3 drinkany 2 0.0984  
 4 subject 0 0   
 5 age 0 0   
 6 sbp 0 0   
 7 ht 0 0   
 8 smoking 0 0   
 9 physact 0 0   
10 diabetes 0 0

## Single 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…

miss\_var\_summary(hers2)

# A tibble: 11 × 3  
 variable n\_miss pct\_miss  
 <chr> <int> <dbl>  
 1 subject 0 0  
 2 ldl 0 0  
 3 age 0 0  
 4 sbp 0 0  
 5 bmi 0 0  
 6 ht 0 0  
 7 smoking 0 0  
 8 drinkany 0 0  
 9 physact 0 0  
10 diabetes 0 0  
11 drinkany\_n 0 0

## Multiple Imputation using aregImpute from Hmisc

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

There are four steps.

## Steps in aregImpute

1. aregImpute draws a sample with replacement from the observations where the target variable is 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

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, pr = FALSE,  
 data = hers1, B = 10, n.impute = 20)

## Multiple Imputation using aregImpute

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

## Multiple Imputation using aregImpute

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

* 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 be used
* data specifies the source of the variables
* pr = FALSE suppresses printing of iteration messages

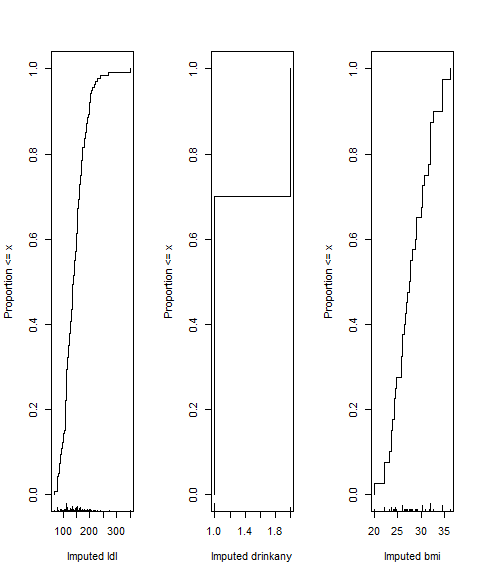
## aregImpute Imputation Results

fit3

Multiple Imputation using Bootstrap and PMM  
  
aregImpute(formula = ~ldl + age + smoking + drinkany + sbp +   
 physact + bmi, data = hers1, n.impute = 20, nk = c(0, 3:5),   
 tlinear = FALSE, pr = FALSE, B = 10)  
  
n: 2032 p: 7 Imputations: 20 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  
  
R-squares for Predicting Non-Missing Values for Each Variable  
Using Last Imputations of Predictors  
 ldl drinkany bmi   
 0.041 0.014 0.109   
  
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 0.0139 0.0149 0.00776 0.0124  
10-fold cross-validated R^2 0.0214 0.0180 0.01517 0.0191  
Bootstrap bias-corrected mean |error| 28.3594 42.9139 44.09937 39.8266  
10-fold cross-validated mean |error| 145.7176 43.5007 45.02428 44.2456  
Bootstrap bias-corrected median |error| 22.8301 35.5441 38.85302 32.6386  
10-fold cross-validated median |error| 141.4238 36.4102 38.88053 37.3141  
  
Variable being imputed: drinkany   
 nk=0 nk=3 nk=4 nk=5  
Bootstrap bias-corrected R^2 0.0163 0.0113 0.0102 0.00986  
10-fold cross-validated R^2 0.0205 0.0249 0.0163 0.01358  
Bootstrap bias-corrected mean |error| 0.4470 0.4568 0.4558 0.46624  
10-fold cross-validated mean |error| 0.4450 0.4454 0.4476 0.44676  
Bootstrap bias-corrected median |error| 0.0000 0.0000 0.0000 0.00000  
10-fold cross-validated median |error| 0.0000 0.0500 0.1000 0.00000  
  
Variable being imputed: bmi   
 nk=0 nk=3 nk=4 nk=5  
Bootstrap bias-corrected R^2 0.0845 0.0932 0.0946 0.0847  
10-fold cross-validated R^2 0.0864 0.0903 0.0968 0.0899  
Bootstrap bias-corrected mean |error| 3.7829 4.8119 4.9226 5.1775  
10-fold cross-validated mean |error| 27.6776 4.8359 4.9390 5.1136  
Bootstrap bias-corrected median |error| 2.9955 3.9704 3.9371 4.2634  
10-fold cross-validated median |error| 27.0143 3.9894 3.9431 4.1876

## Plot the imputed values…

par(mfrow = c(1,3)); plot(fit3); par(mfrow = c(1,1))



## Interpreting plot of imputations

* 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.

# Deciding Where to Try Non-Linear Terms

## Spending degrees of freedom wisely

* Suppose we have many possible predictors, and minimal theory or subject matter knowledge to guide us.
* We might want our final inferences to be as unbiased as possible. To accomplish this, we have to pay a penalty (in terms of degrees of freedom) for any “peeks” we make at the data in advance of fitting a model.
* So that rules out a lot of decision-making about non-linearity based on looking at the data, if our sample size isn’t incredibly large.

## The helpdat example from Class 5

helpdat <- read\_rds("c06/data/helpdat.Rds")  
dim(helpdat)

[1] 453 8

names(helpdat)

[1] "id" "cesd" "age" "sex" "subst" "mcs" "pcs" "pss\_fr"

* In this case, we are predicting cesd using *n* = 453 observations and 6 candidate predictors (age, sex, subst, mcs, pcs and pss\_fr.)
  + In addition, adding non-linearity to our model costs additional degrees of freedom, as we’ll see in the next two slides.

## Adding Non-Linear Terms Spends DF

What happens when we add a non-linear term?

* Adding a polynomial of degree D costs D degrees of freedom.
  + So a polynomial of degree 2 (quadratic) costs 2 df, or 1 more than the main effect alone.
* Adding a restricted cubic spline with K knots costs K-1 df.
  + So adding a spline with 4 knots uses 3 df, or 2 more than the main effect alone.
  + We’ll only consider splines with 3, 4, or 5 knots.

## Adding Non-Linear Terms Spends DF

Adding an interaction (product term) depends on the main effects of the predictors we are interacting

* If the product term’s predictors have df1 and df2 degrees of freedom, product term adds df1 df2 degrees of freedom.
  + An interaction of a binary and quantitative variable adds 1 1 = 1 more df to the main effects model.
* When we use a quantitative variable in a spline and interaction, we’ll do the interaction on the main effect, not the spline.

## Spearman’s plot: A smart first step?

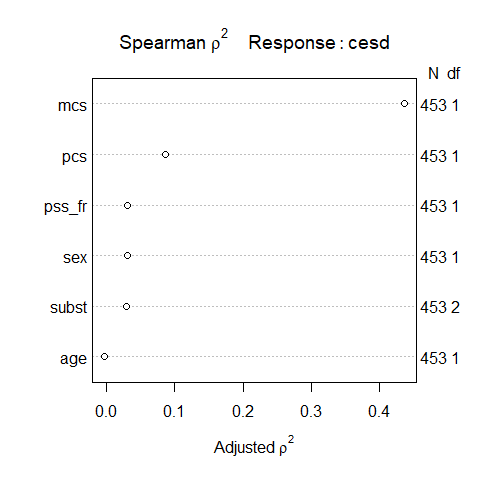
Spearman’s is an indicator (not a perfect one) of potential predictive punch, but doesn’t give away the game.

* Idea: Perhaps we should focus our efforts re: non-linearity on predictors that score better on this measure.

spear\_cesd <- spearman2(cesd ~ mcs + subst + pcs + age + sex + pss\_fr,   
 data = helpdat)

## Spearman’s Plot

plot(spear\_cesd)



## Conclusions from Spearman Plot

* mcs is the most attractive candidate for a non-linear term, as it packs the most potential predictive punch, so if it does turn out to need non-linear terms, our degrees of freedom will be well spent.
  + This **does not** mean that mcs actually needs a non-linear term, or will show meaningfully better results if a non-linear term is included. We’d have to fit a model with and without non-linearity in mcs to know that.
  + Non-linearity will often take the form of a product term, a polynomial term, or a restricted cubic spline.

## Conclusions from Spearman Plot

* pcs, also quantitative, has the next most potential predictive punch after mcs.
* These are followed, in order, by pss\_fr and sex.

## Grim Reality

With 453 observations (452 df) we should be thinking about models with modest numbers of regression inputs.

* Non-linear terms (polynomials, splines) just add to the problem, as they need additional df to be estimated.

In this case, we might choose to include non-linear terms in just two or three variables (and that’s it) and even that would be tough to justify with this modest sample size.

## Contents of spear\_cesd

spear\_cesd

Spearman rho^2 Response variable:cesd  
  
 rho2 F df1 df2 P Adjusted rho2 n  
mcs 0.438 350.89 1 451 0.0000 0.436 453  
subst 0.034 7.97 2 450 0.0004 0.030 453  
pcs 0.089 44.22 1 451 0.0000 0.087 453  
age 0.000 0.12 1 451 0.7286 -0.002 453  
sex 0.033 15.56 1 451 0.0001 0.031 453  
pss\_fr 0.033 15.57 1 451 0.0001 0.031 453

## Proposed New Model

Fit a model to predict cesd using:

* a 5-knot spline on mcs
* a 3-knot spline on pcs
* a linear term on pss\_fr
* a linear term on age
* an interaction of sex with the main effect of mcs (restricting our model so that terms that are non-linear in both sex and mcs are excluded), and
* a main effect of subst

## Our new model mod\_help2

Definitely more than we can reasonably do with 453 observations, but let’s see how it looks.

dd <- datadist(helpdat)  
options(datadist = "dd")  
  
mod\_help2 <- ols(cesd ~ rcs(mcs, 5) + rcs(pcs, 3) + sex +   
 mcs %ia% sex + pss\_fr + age + subst,   
 data = helpdat, x = TRUE, y = TRUE)

* %ia% tells R to fit an interaction term with sex and the main effect of mcs.
  + We have to include sex as a main effect for the interaction term (%ia%) to work. We already have the main effect of mcs in as part of the spline.

## Our fitted model mod\_help2

mod\_help2

Linear Regression Model  
  
ols(formula = cesd ~ rcs(mcs, 5) + rcs(pcs, 3) + sex + mcs %ia%   
 sex + pss\_fr + age + subst, data = helpdat, x = TRUE, y = TRUE)  
  
 Model Likelihood Discrimination   
 Ratio Test Indexes   
Obs 453 LR chi2 349.44 R2 0.538   
sigma8.6248 d.f. 12 R2 adj 0.525   
d.f. 440 Pr(> chi2) 0.0000 g 10.439   
  
Residuals  
  
 Min 1Q Median 3Q Max   
-26.7893 -5.9000 0.1545 5.5884 26.1304   
  
 Coef S.E. t Pr(>|t|)  
Intercept 76.3346 6.2540 12.21 <0.0001   
mcs -0.9306 0.2315 -4.02 <0.0001   
mcs' 1.6607 2.5040 0.66 0.5075   
mcs'' -2.8854 8.3945 -0.34 0.7312   
mcs''' 0.2942 7.9390 0.04 0.9705   
pcs -0.2341 0.0883 -2.65 0.0083   
pcs' -0.0151 0.1000 -0.15 0.8797   
sex=male -2.0330 2.5456 -0.80 0.4249   
mcs \* sex=male -0.0129 0.0783 -0.17 0.8690   
pss\_fr -0.2569 0.1046 -2.46 0.0144   
age -0.0466 0.0569 -0.82 0.4139   
subst=cocaine -2.6999 0.9965 -2.71 0.0070   
subst=heroin -2.1741 1.0677 -2.04 0.0423

## ANOVA for this model

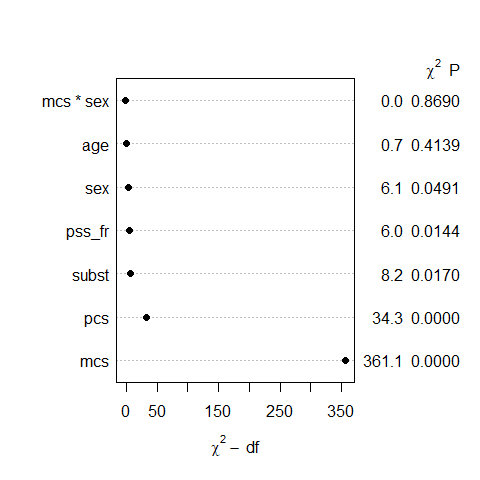
Remember this ANOVA testing is sequential, other than the TOTALS.

anova(mod\_help2)

Analysis of Variance Response: cesd   
  
 Factor d.f. Partial SS MS F   
 mcs (Factor+Higher Order Factors) 5 26857.364671 5371.472934 72.21  
 All Interactions 1 2.026255 2.026255 0.03  
 Nonlinear 3 293.502251 97.834084 1.32  
 pcs 2 2548.388579 1274.194290 17.13  
 Nonlinear 1 1.705031 1.705031 0.02  
 sex (Factor+Higher Order Factors) 2 451.578352 225.789176 3.04  
 All Interactions 1 2.026255 2.026255 0.03  
 mcs \* sex (Factor+Higher Order Factors) 1 2.026255 2.026255 0.03  
 pss\_fr 1 448.812293 448.812293 6.03  
 age 1 49.758786 49.758786 0.67  
 subst 2 611.625952 305.812976 4.11  
 TOTAL NONLINEAR 4 293.512204 73.378051 0.99  
 TOTAL NONLINEAR + INTERACTION 5 294.601803 58.920361 0.79  
 REGRESSION 12 38058.315322 3171.526277 42.64  
 ERROR 440 32730.174744 74.386761   
 P   
 <.0001  
 0.8690  
 0.2688  
 <.0001  
 0.8797  
 0.0491  
 0.8690  
 0.8690  
 0.0144  
 0.4139  
 0.0170  
 0.4146  
 0.5558  
 <.0001

## Plotting ANOVA results for mod\_help2

plot(anova(mod\_help2))



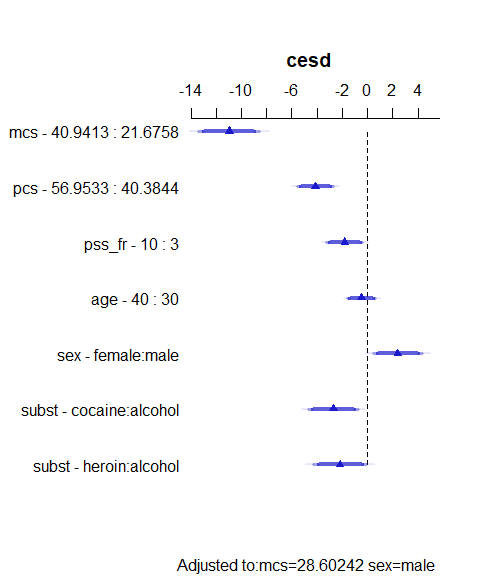
## Validation of Summary Statistics

set.seed(432); validate(mod\_help2)

index.orig training test optimism index.corrected n  
R-square 0.5376 0.5513 0.5233 0.0280 0.5096 40  
MSE 72.2520 69.8358 74.4984 -4.6627 76.9147 40  
g 10.4392 10.5053 10.2718 0.2335 10.2056 40  
Intercept 0.0000 0.0000 0.7893 -0.7893 0.7893 40  
Slope 1.0000 1.0000 0.9751 0.0249 0.9751 40

## summary results for mod\_help2

plot(summary(mod\_help2))



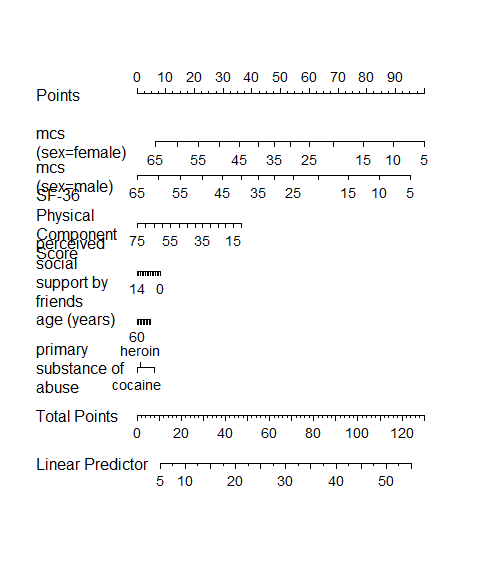
## summary results for mod\_help2

summary(mod\_help2)

Effects Response : cesd   
  
 Factor Low High Diff. Effect S.E. Lower 0.95  
 mcs 21.676 40.941 19.266 -10.96400 1.23340 -13.38800   
 pcs 40.384 56.953 16.569 -4.10790 0.73381 -5.55010   
 pss\_fr 3.000 10.000 7.000 -1.79860 0.73225 -3.23780   
 age 30.000 40.000 10.000 -0.46552 0.56918 -1.58420   
 sex - female:male 2.000 1.000 NA 2.40260 0.99054 0.45577   
 subst - cocaine:alcohol 1.000 2.000 NA -2.69990 0.99647 -4.65830   
 subst - heroin:alcohol 1.000 3.000 NA -2.17410 1.06770 -4.27250   
 Upper 0.95  
 -8.539800   
 -2.665700   
 -0.359500   
 0.653130   
 4.349300   
 -0.741430   
 -0.075632   
  
Adjusted to: mcs=28.60242 sex=male

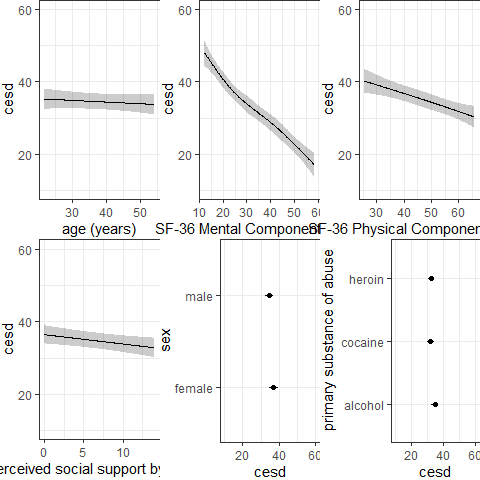
## Nomogram for mod\_help2

plot(nomogram(mod\_help2))



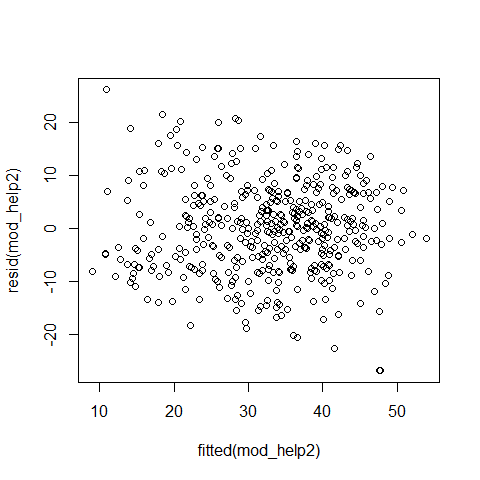
## Impact of non-linearity?

ggplot(Predict(mod\_help2))



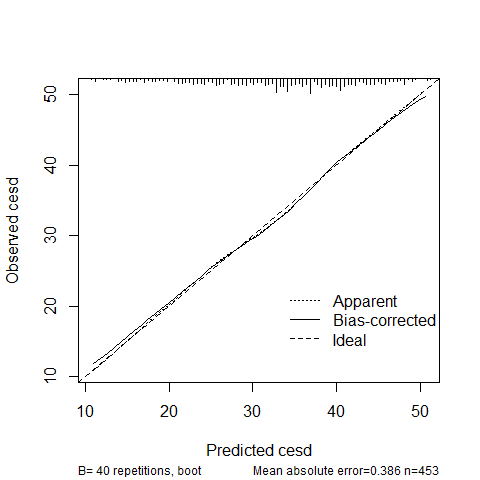
## Residuals vs. Fitted Values?

plot(resid(mod\_help2) ~ fitted(mod\_help2))



## Checking the model’s calibration

set.seed(432); plot(calibrate(mod\_help2))



n=453 Mean absolute error=0.386 Mean squared error=0.19775  
0.9 Quantile of absolute error=0.704

## Limitations of lm for fitting complex linear models

We can certainly assess this big, complex model using lm, too:

* with in-sample summary statistics like adjusted , AIC and BIC,
* we can assess its assumptions with residual plots, and
* we can also compare out-of-sample predictive quality through cross-validation,

## We can/will use both lm and ols

But to really delve into the details of how well this complex model works, and to help plot what is actually being fit, we’ll probably want to fit the model using ols.

* In Project A, we expect some results that are most easily obtained using lm and others that are most easily obtained using ols.

# Back to the hers2 data

## Kitchen Sink Model (Main Effects only)

mod\_ks <- ols(ldl ~ age + smoking + drinkany + sbp +   
 physact + bmi, data = hers2)  
anova(mod\_ks)

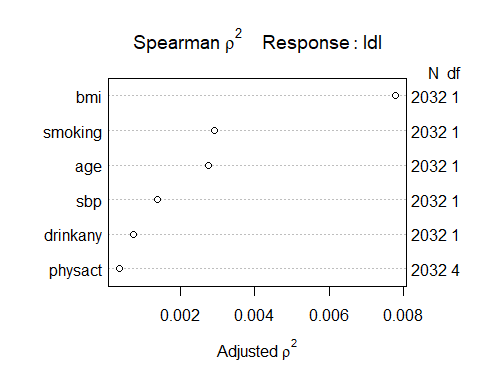
Analysis of Variance Response: ldl   
  
 Factor d.f. Partial SS MS F P   
 age 1 9330.911 9330.911 6.93 0.0085  
 smoking 1 8199.755 8199.755 6.09 0.0137  
 drinkany 1 6444.424 6444.424 4.79 0.0288  
 sbp 1 9274.287 9274.287 6.89 0.0087  
 physact 4 10874.528 2718.632 2.02 0.0891  
 bmi 1 15876.957 15876.957 11.80 0.0006  
 REGRESSION 9 60077.708 6675.301 4.96 <.0001  
 ERROR 2022 2721037.890 1345.716

## Spearman Plot

How should we prioritize the degrees of freedom we spend on non-linearity?

* Note the use of the simple imputation hers2 data here. Why?

plot(spearman2(ldl ~ age + smoking + drinkany + sbp +   
 physact + bmi, data = hers2))



## Spending Degrees of Freedom

We’re spending 9 degrees of freedom in our kitchen sink model. (We can verify this with anova or the plot.)

* Each quantitative main effect costs 1 df to estimate
* Each binary categorical variable also costs 1 df
* Multi-categorical variables with L levels cost L-1 df to estimate

Suppose we’re willing to spend up to a total of **16** degrees of freedom (i.e. a combined 7 more on interaction terms and other ways to capture non-linearity.)

## What did Spearman Plot show?

Group 1 (largest adjusted )

* bmi, a quantitative predictor, is furthest to the right

Group 2 (next largest)

* smoking, a binary predictor, is next, followed closely by
* age, a quantitative predictor

Other predictors (rest of the group)

* sbp, quantitative
* drinkany, binary
* physact, multi-categorical (5 levels)

## Impact of Adding Non-Linear Terms on Spent DF

* Adding a polynomial of degree D costs D degrees of freedom.
* Adding a restricted cubic spline with K knots costs K-1 df.
* Adding an interaction (product term) where the predictors have df1 and df2 degrees of freedom, product term adds df1 df2 degrees of freedom.
  + When we use a quantitative variable in a spline *and* interaction, we’ll do the interaction on the main effect, not the spline.

## Model we’ll fit with ols

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 the main effect of bmi

## Dealing with missing data?

We can fit this to the data:

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

# Using only the Complete Cases

## Fitting 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.

## m1 results

m1

Frequencies of Missing Values Due to Each Variable  
 ldl bmi age sbp drinkany physact smoking   
 7 2 0 0 2 0 0   
  
Linear Regression Model  
  
ols(formula = ldl ~ rcs(bmi, 5) + pol(age, 2) + sbp + drinkany +   
 physact + smoking + smoking %ia% bmi, data = hers1, x = TRUE,   
 y = TRUE)  
  
 Model Likelihood Discrimination   
 Ratio Test Indexes   
Obs 2021 LR chi2 52.61 R2 0.026   
sigma36.7430 d.f. 14 R2 adj 0.019   
d.f. 2006 Pr(> chi2) 0.0000 g 6.629   
  
Residuals  
  
 Min 1Q Median 3Q Max   
-113.440 -24.519 -3.778 20.940 197.087   
  
 Coef S.E. t Pr(>|t|)  
Intercept 121.6057 68.2000 1.78 0.0747   
bmi 1.5687 1.0107 1.55 0.1208   
bmi' -8.6685 9.1577 -0.95 0.3440   
bmi'' 40.5712 37.4468 1.08 0.2787   
bmi''' -55.8872 44.5946 -1.25 0.2103   
age -0.5791 1.9657 -0.29 0.7683   
age^2 0.0018 0.0149 0.12 0.9024   
sbp 0.1221 0.0453 2.69 0.0072   
drinkany=yes -3.7427 1.6629 -2.25 0.0245   
physact=much less active -4.5660 3.8904 -1.17 0.2407   
physact=much more active -0.3291 2.7521 -0.12 0.9048   
physact=somewhat less active -0.0160 2.5270 -0.01 0.9950   
physact=somewhat more active 3.7731 2.0293 1.86 0.0631   
smoking=yes -7.0832 12.0586 -0.59 0.5570   
smoking=yes \* bmi 0.4961 0.4391 1.13 0.2587

# Using Single Imputation

## Fitting 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, again, %ia% identifies the linear interaction alone.

## m2 results

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   
  
 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

## ANOVA results for m2 from ols

anova(m2)

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

## Validation of summary statistics

Complete cases only…

set.seed(432001); validate(m1)

index.orig training test optimism index.corrected n  
R-square 0.0257 0.0345 0.0184 0.0161 0.0096 40  
MSE 1340.0254 1324.8222 1350.0695 -25.2473 1365.2727 40  
g 6.6287 7.5809 5.9177 1.6632 4.9655 40  
Intercept 0.0000 0.0000 31.2738 -31.2738 31.2738 40  
Slope 1.0000 1.0000 0.7863 0.2137 0.7863 40

After single imputation…

set.seed(432002); validate(m2)

index.orig training test optimism index.corrected n  
R-square 0.0258 0.0337 0.0188 0.0150 0.0108 40  
MSE 1333.3300 1336.3384 1342.9706 -6.6322 1339.9622 40  
g 6.6306 7.5648 5.9723 1.5924 5.0382 40  
Intercept 0.0000 0.0000 30.1440 -30.1440 30.1440 40  
Slope 1.0000 1.0000 0.7932 0.2068 0.7932 40

## summary(m2) results

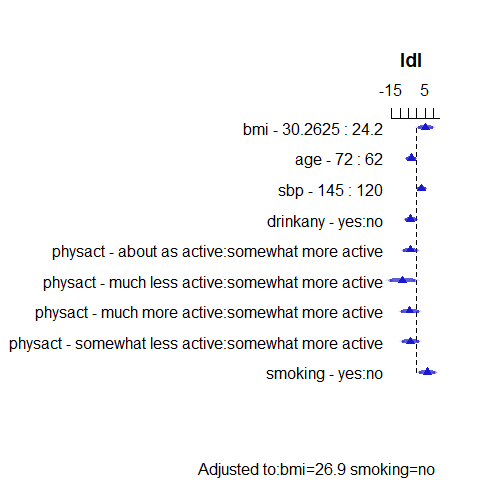
summary(m2)

Effects Response : ldl   
  
 Factor Low High Diff.   
 bmi 24.2 30.263 6.0625  
 age 62.0 72.000 10.0000  
 sbp 120.0 145.000 25.0000  
 drinkany - yes:no 1.0 2.000 NA  
 physact - about as active:somewhat more active 5.0 1.000 NA  
 physact - much less active:somewhat more active 5.0 2.000 NA  
 physact - much more active:somewhat more active 5.0 3.000 NA  
 physact - somewhat less active:somewhat more active 5.0 4.000 NA  
 smoking - yes:no 1.0 2.000 NA  
 Effect S.E. Lower 0.95 Upper 0.95  
 5.1862 2.2217 0.82921 9.54330   
 -3.3412 1.3450 -5.97890 -0.70357   
 3.0218 1.1270 0.81165 5.23190   
 -3.7023 1.6544 -6.94690 -0.45779   
 -3.8031 2.0193 -7.76310 0.15695   
 -8.5439 3.9035 -16.19900 -0.88862   
 -4.0666 2.7125 -9.38630 1.25310   
 -3.7901 2.5633 -8.81720 1.23690   
 6.2635 2.4009 1.55500 10.97200   
  
Adjusted to: bmi=26.9 smoking=no

* Of course, these should really be plotted…

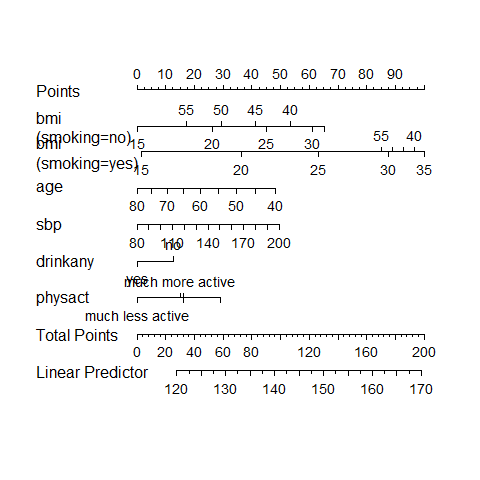
## Effect Size Plot for m2

plot(summary(m2))



## Nomogram for m2

plot(nomogram(m2))



## Making Predictions for an Individual

Suppose we want a prediction for a new individual subject with bmi = 30, age = 50, smoking = yes, physact = about as active, drinkany= yes and sbp of 150.

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

$linear.predictors  
 1   
160.9399   
  
$lower  
 1   
88.48615   
  
$upper  
 1   
233.3936

This is called a *prediction interval*.

## Predictions for a Long-Run Mean

The other prediction we might make is for the *mean* of a series of subjects with the same predictor values…

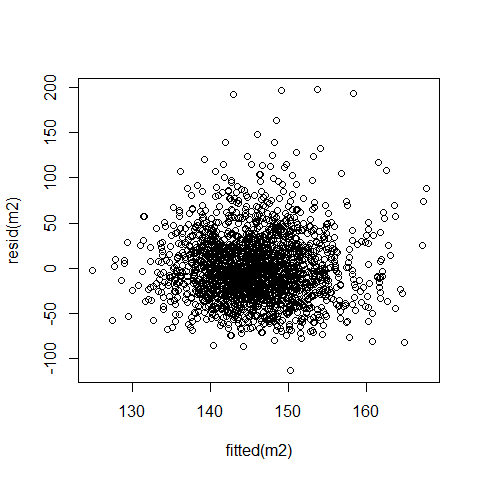
predict(m2, expand.grid(bmi = 30, age = 50, sbp = 150, smoking = "yes",  
 physact = "about as active", drinkany = "yes"),  
 conf.int = 0.95, conf.type = "mean")

$linear.predictors  
 1   
160.9399   
  
$lower  
 1   
151.8119   
  
$upper  
 1   
170.0679

Note that the confidence interval will always be narrower than the prediction interval given the same predictor values.

## Residuals vs. Fitted Values?

plot(resid(m2) ~ fitted(m2))



## Influential Points?

which.influence(m2, cutoff = 0.4)

$Intercept  
[1] "1135"  
  
$age  
[1] "1135"  
  
$smoking  
[1] "132"  
  
$`smoking \* bmi`  
[1] "132"

# Using Multiple Imputation

## Fitting after Multiple Imputation

What do we have now?

* 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 (from m1 or m2)

ols(ldl ~ rcs(bmi, 5) + pol(age, 2) + sbp +  
 drinkany + physact + smoking + smoking %ia% bmi,  
 x = TRUE, y = TRUE)

Put them together with fit.mult.impute()…

## Linear Regression & Imputation Model

m3imp <-   
 fit.mult.impute(ldl ~ rcs(bmi, 5) + pol(age, 2) + sbp +  
 drinkany + physact + smoking +   
 smoking %ia% bmi,  
 fitter = ols, xtrans = fit3,   
 data = hers1, pr = FALSE)

* When you run this without the pr = FALSE it generates considerable output related to the imputations, which we won’t use today.
* Let’s look at the rest of the output this yields…

## m3imp results

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, pr = FALSE)  
  
 Model Likelihood Discrimination   
 Ratio Test Indexes   
Obs 2032 LR chi2 52.74 R2 0.026   
sigma36.7331 d.f. 14 R2 adj 0.019   
d.f. 2017 Pr(> chi2) 0.0000 g 6.621   
  
Residuals  
  
 Min 1Q Median 3Q Max   
-113.345 -24.510 -3.803 20.777 197.295   
  
 Coef S.E. t Pr(>|t|)  
Intercept 119.8951 67.8409 1.77 0.0773   
bmi 1.5436 1.0097 1.53 0.1265   
bmi' -8.3664 9.1409 -0.92 0.3602   
bmi'' 39.2149 37.3458 1.05 0.2938   
bmi''' -54.2873 44.5323 -1.22 0.2230   
age -0.5002 1.9555 -0.26 0.7981   
age^2 0.0012 0.0148 0.08 0.9351   
sbp 0.1198 0.0454 2.64 0.0083   
drinkany=yes -3.7196 1.6613 -2.24 0.0253   
physact=much less active -4.7109 3.8716 -1.22 0.2238   
physact=much more active -0.2328 2.7512 -0.08 0.9326   
physact=somewhat less active -0.0417 2.5246 -0.02 0.9868   
physact=somewhat more active 3.8197 2.0286 1.88 0.0599   
smoking=yes -6.8967 12.0503 -0.57 0.5672   
smoking=yes \* bmi 0.4866 0.4389 1.11 0.2677

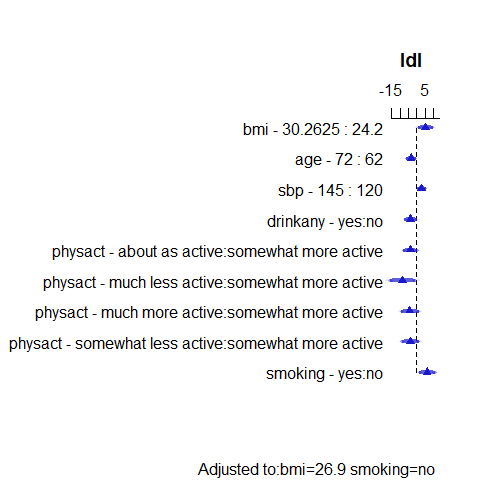
## ANOVA results for m3imp

anova(m3imp)

Analysis of Variance Response: ldl   
  
 Factor d.f. Partial SS MS   
 bmi (Factor+Higher Order Factors) 5 2.728300e+04 5456.600791  
 All Interactions 1 1.658459e+03 1658.458931  
 Nonlinear 3 9.585703e+03 3195.234412  
 age 2 9.320445e+03 4660.222299  
 Nonlinear 1 8.950493e+00 8.950493  
 sbp 1 9.407603e+03 9407.602954  
 drinkany 1 6.763854e+03 6763.853503  
 physact 4 9.698175e+03 2424.543639  
 smoking (Factor+Higher Order Factors) 2 1.031090e+04 5155.452328  
 All Interactions 1 1.658459e+03 1658.458931  
 smoking \* bmi (Factor+Higher Order Factors) 1 1.658459e+03 1658.458931  
 TOTAL NONLINEAR 4 9.587178e+03 2396.794504  
 TOTAL NONLINEAR + INTERACTION 5 1.152744e+04 2305.487432  
 REGRESSION 14 7.030149e+04 5021.535034  
 ERROR 2017 2.721574e+06 1349.317884  
 F P   
 4.04 0.0012  
 1.23 0.2677  
 2.37 0.0690  
 3.45 0.0318  
 0.01 0.9351  
 6.97 0.0083  
 5.01 0.0253  
 1.80 0.1268  
 3.82 0.0221  
 1.23 0.2677  
 1.23 0.2677  
 1.78 0.1309  
 1.71 0.1293  
 3.72 <.0001

## Effect Estimates for m3imp

plot(summary(m3imp))



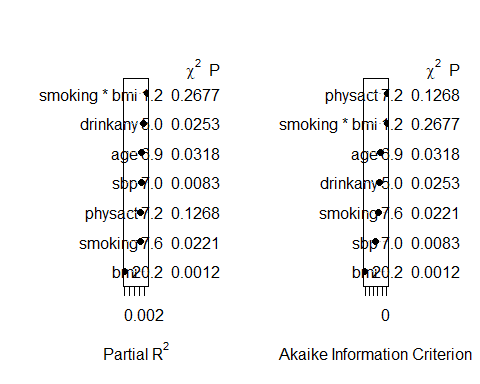
## Effect Estimates for m3imp

summary(m3imp)

Effects Response : ldl   
  
 Factor Low High Diff.   
 bmi 24.2 30.263 6.0625  
 age 62.0 72.000 10.0000  
 sbp 120.0 145.000 25.0000  
 drinkany - yes:no 1.0 2.000 NA  
 physact - about as active:somewhat more active 5.0 1.000 NA  
 physact - much less active:somewhat more active 5.0 2.000 NA  
 physact - much more active:somewhat more active 5.0 3.000 NA  
 physact - somewhat less active:somewhat more active 5.0 4.000 NA  
 smoking - yes:no 1.0 2.000 NA  
 Effect S.E. Lower 0.95 Upper 0.95  
 5.1643 2.2300 0.79099 9.53750   
 -3.3824 1.3518 -6.03340 -0.73144   
 2.9955 1.1345 0.77068 5.22040   
 -3.7196 1.6613 -6.97780 -0.46150   
 -3.8197 2.0286 -7.79800 0.15861   
 -8.5306 3.9152 -16.20900 -0.85228   
 -4.0525 2.7260 -9.39850 1.29350   
 -3.8614 2.5796 -8.92030 1.19760   
 6.1923 2.4427 1.40190 10.98300   
  
Adjusted to: bmi=26.9 smoking=no

## Evaluation via Partial and AIC

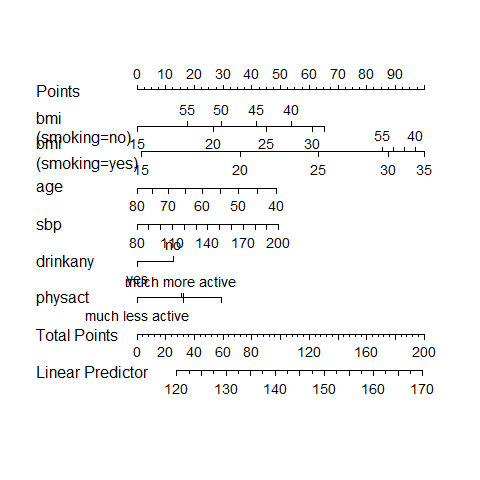
par(mfrow = c(1,2))  
plot(anova(m3imp), what="partial R2")  
plot(anova(m3imp), what="aic")



par(mfrow = c(1,1))

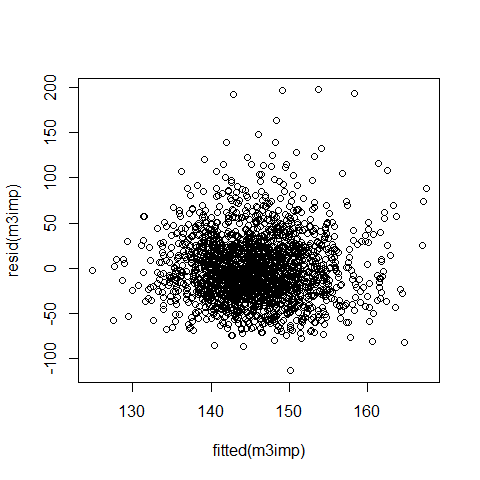
## Nomogram for m3imp

plot(nomogram(m3imp))



## Residuals vs. Fitted Values

plot(resid(m3imp) ~ fitted(m3imp))



## More after aregImpute?

* How can I estimate the AIC (and BIC) of a model fit with fit.mult.impute?

glance won’t work with an ols fit, but we can just use…

AIC(m3imp)

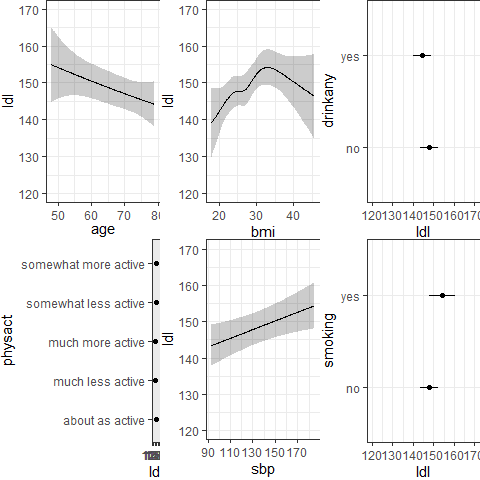
d.f.   
20425.29

BIC(m3imp)

d.f.   
20515.16

## Viewing the m3imp effects?

ggplot(Predict(m3imp))



## Pull out one imputation?

* How can I pull (say, the fifth) imputation from aregImpute?

fit3 was our imputation model here, built on the hers1 data, with subject identifiers in subject…

imputed\_5 <- impute.transcan(fit3, data = hers1, imputation = 5,   
 list.out = T, pr = F, check = F)  
  
imputed\_df5 <- as.data.frame(do.call(cbind, imputed\_5))  
  
fifth\_imp <- bind\_cols(subject = hers1$subject, imputed\_df5) |>  
 tibble() |> mutate\_if(is.character, as.factor) |>  
 mutate(subject = as.character(subject))

## Our fifth\_imp tibble

fifth\_imp

# A tibble: 2,032 × 8  
 subject ldl age smoking drinkany sbp physact bmi  
 <chr> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl>  
 1 1 122. 70 1 1 138 3 23.7  
 2 2 242. 62 1 1 118 2 28.6  
 3 4 116. 64 2 2 152 2 24.4  
 4 5 151. 65 1 1 175 4 21.9  
 5 6 138. 68 1 2 174 1 29.0  
 6 8 121. 69 1 1 178 3 23.2  
 7 9 133 61 1 2 162 1 30.3  
 8 10 220 62 2 2 111 4 45.7  
 9 11 173. 72 1 1 122 1 22.2  
10 12 124. 73 1 1 158 5 25.3  
# ℹ 2,022 more rows

n\_miss(fifth\_imp)

[1] 0

## Model with lm for 5th imputation?

model\_for\_imp5 <-  
 lm(ldl ~ rcs(bmi, 5) + pol(age, 2) + sbp +  
 drinkany + physact + smoking +   
 smoking %ia% bmi, data = fifth\_imp)  
  
model\_for\_imp5

Call:  
lm(formula = ldl ~ rcs(bmi, 5) + pol(age, 2) + sbp + drinkany +   
 physact + smoking + smoking %ia% bmi, data = fifth\_imp)  
  
Coefficients:  
 (Intercept) rcs(bmi, 5)bmi rcs(bmi, 5)bmi' rcs(bmi, 5)bmi''   
 111.901199 1.032315 -7.262365 33.206787   
rcs(bmi, 5)bmi''' pol(age, 2)age pol(age, 2)age^2 sbp   
 -46.082107 -0.024238 -0.002339 0.125763   
 drinkany physact smoking smoking %ia% bmi   
 -3.700570 1.017248 -6.928039 0.500132

## Checking m3imp in imputation 5

glance(model\_for\_imp5) |>  
 gt() |> fmt\_number(columns = r.squared:p.value, decimals = 3)

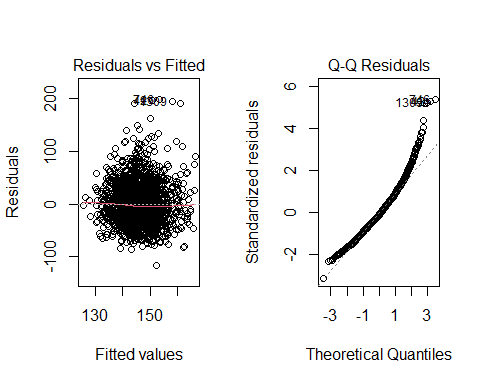
| r.squared | adj.r.squared | sigma | statistic | p.value | df | logLik | AIC | BIC | deviance | df.residual | nobs |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 0.025 | 0.020 | 36.946 | 4.794 | 0.000 | 11 | -10211.66 | 20449.32 | 20522.34 | 2757257 | 2020 | 2032 |

anova(model\_for\_imp5)

Analysis of Variance Table  
  
Response: ldl  
 Df Sum Sq Mean Sq F value Pr(>F)   
rcs(bmi, 5) 4 27169 6792.2 4.9761 0.0005428 \*\*\*  
pol(age, 2) 2 8836 4417.8 3.2366 0.0395027 \*   
sbp 1 12237 12237.3 8.9652 0.0027852 \*\*   
drinkany 1 6544 6543.8 4.7941 0.0286721 \*   
physact 1 5343 5343.4 3.9146 0.0480038 \*   
smoking 1 10090 10090.5 7.3924 0.0066060 \*\*   
smoking %ia% bmi 1 1761 1761.3 1.2903 0.2561224   
Residuals 2020 2757257 1365.0   
---  
Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

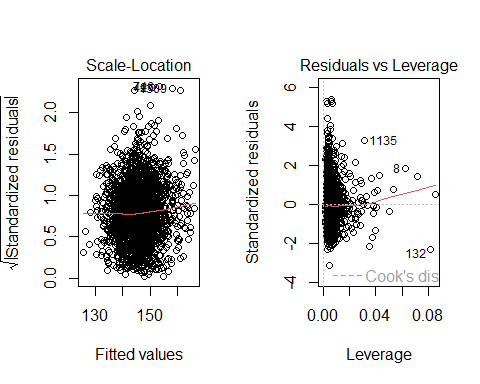
## Plot residuals for 5th imputation?

par(mfrow = c(1,2)); plot(model\_for\_imp5, which = c(1:2))



## Plot residuals for 5th imputation?

par(mfrow = c(1,2)); plot(model\_for\_imp5, which = c(3,5))



## Next Week

Logistic Regression: Predicting a Binary Outcome