432 Class 09 Slides

thomaselove.github.io/432

2022-02-08

Today's Agenda

- Data from the Heart and Estrogen/Progestin Study
- 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

Setup

```r
library(magrittr); library(janitor)
library(here); library(knitr)
library(naniar); library(simputation)

library(rms)
library(tidyverse)
```

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.

```
hers_raw <- read_csv(here("data/hersdata.csv")) %>%
    clean_names()

hers1 <- hers_raw %>%
    filter(diabetes == "no") %>%
    select(subject, ldl, ht, age, smoking, drinkany, sbp,
           physact, bmi, diabetes)
```

The 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
`smoking`	yes or no
`drinkany`	yes or no
`sbp`	systolic BP in mm Hg
`physact`	5-level factor, details next slide
`bmi`	body-mass index in kg/m^2

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

The `physact` variable

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

```
# A tibble: 5 x 2
  physact                    n
  <chr>                  <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 x 3
   variable n_miss pct_miss
   <chr>     <int>    <dbl>
 1 ldl           7   0.344
 2 drinkany      2   0.0984
 3 bmi           2   0.0984
 4 subject       0   0
 5 ht            0   0
 6 age           0   0
 7 smoking       0   0
 8 sbp           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 x 3
   variable   n_miss pct_miss
   <chr>       <int>    <dbl>
 1 subject         0        0
 2 ldl             0        0
 3 ht              0        0
 4 age             0        0
 5 smoking         0        0
 6 drinkany        0        0
 7 sbp             0        0
 8 physact         0        0
 9 bmi             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.

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

```
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 It
```

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 be used
- `data` specifies the source of the variables

`aregImpute` Imputation Results (1 of 4)

```
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, 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
```

`fit3` Imputation Results (2 of 4)

```
R-squares for Predicting Non-Missing Values for Each
Variable Using Last Imputations of Predictors
     ldl drinkany      bmi
   0.041    0.014    0.109
```

`fit3` Imputation Results (3 of 4)

Resampling results for determining the complexity of imputation models

Variable being imputed: ldl

Bootstrap bias-corrected summaries:

Statistic	nk = 0	nk = 3	nk = 4	nk = 5
R^2	0.0139	0.0149	0.00776	0.0124
mean absolute error	28.3594	42.9139	44.09937	39.8266
median abs. error	22.8301	35.5441	38.85302	32.6386

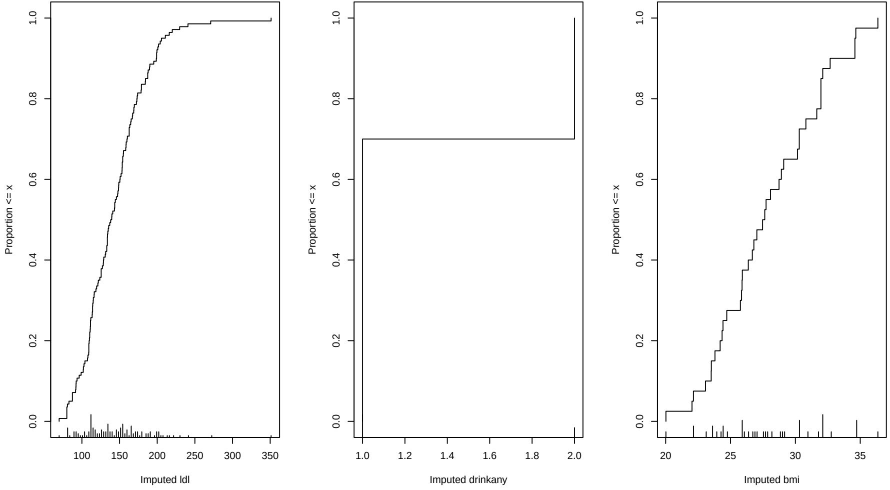
10-fold cross-validated:

Statistic	nk = 0	nk = 3	nk = 4	nk = 5
R^2	0.0214	0.0180	0.01517	0.0191
mean absolute error	145.7176	43.5007	45.02428	44.2456
median abs. error	141.4238	36.4102	38.88053	37.3141

fit3 Imputation Results (4 of 4)

```
Variable being imputed: drinkany
                              nk=0   nk=3   nk=4    nk=5
Bootstrap  R^2              0.0163 0.0113 0.0102 0.00986
10-fold cv R^2              0.0205 0.0249 0.0163 0.01358
Bootstrap  mean |error|     0.4470 0.4568 0.4558 0.46624
10-fold cv mean |error|     0.4450 0.4454 0.4476 0.44676
Bootstrap median |error|    0.0000 0.0000 0.0000 0.00000
10-fold cv median |error|   0.0000 0.0500 0.1000 0.00000

Variable being imputed: bmi
                               nk=0   nk=3   nk=4   nk=5
Bootstrap  R^2               0.0845 0.0932 0.0946 0.0847
10-fold cv R^2               0.0864 0.0903 0.0968 0.0899
Bootstrap  mean |error|      3.7829 4.8119 4.9226 5.1775
10-fold cv mean |error|     27.6776 4.8359 4.9390 5.1136
Bootstrap median |error|     2.9955 3.9704 3.9371 4.2634
10-fold cv median |error|   27.0143 3.9894 3.9431 4.1876
```

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



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

```
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.

Kitchen Sink Model (Main Effects only)

```r
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 ρ^2 Plot

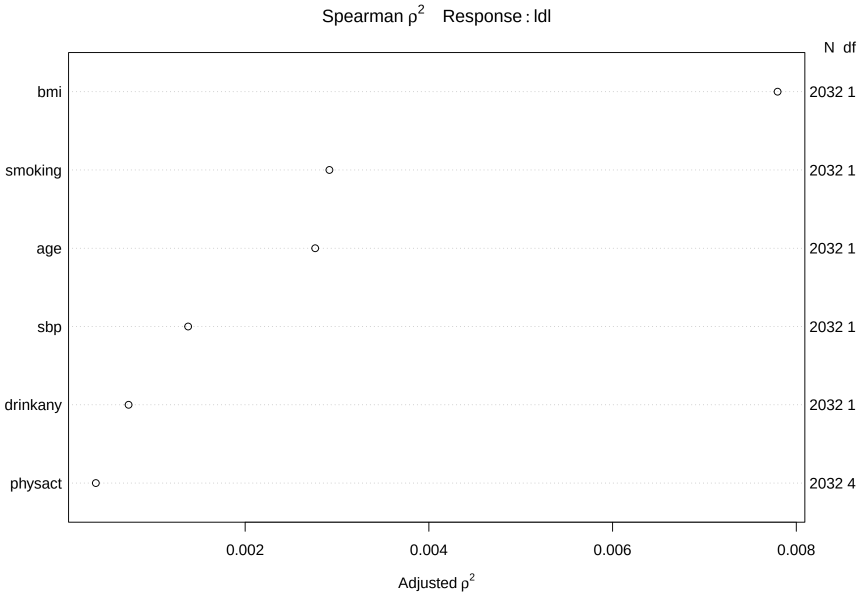
How should we prioritize the degrees of freedom we spend on non-linearity?

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

Plot's on the next page.

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

Spearman ρ^2 Plot Result



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 **14** degrees of freedom (i.e. a combined 5 more on interaction terms and other ways to capture non-linearity.)

What should we choose?

What did we see in the Spearman ρ^2 Plot?

Group 1 (largest adjusted ρ^2)

- `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

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 rcs with 4 knots uses 3 df, or 2 more than the main effect.
  - We restrict ourselves to considering splines with 3, 4, or 5 knots.
- 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 $\times$ df2 degrees of freedom.
  - An interaction of a binary and quantitative variable adds $1 \times 1 = 1$ additional degree of freedom 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.

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`

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 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.

m1 results (screen 1/2)

```
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
```

`m1` results (screen 2/2)

```r
m1
```

	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

Fit Model after Single Imputation

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

```
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 (screen 2/2)

```
m2
```

	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`

```r
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
```

Validation of summary statistics

```
set.seed(432); validate(m2)
```

	index.orig	training	test	optimism	index.corrected	n
R-square	0.0258	0.0307	0.0182	0.0125	0.0133	40
MSE	1333.3300	1323.5182	1343.7711	-20.2529	1353.5829	40
g	6.6306	7.1676	5.8338	1.3338	5.2968	40
Intercept	0.0000	0.0000	26.5316	-26.5316	26.5316	40
Slope	1.0000	1.0000	0.8174	0.1826	0.8174	40

summary(m2) results

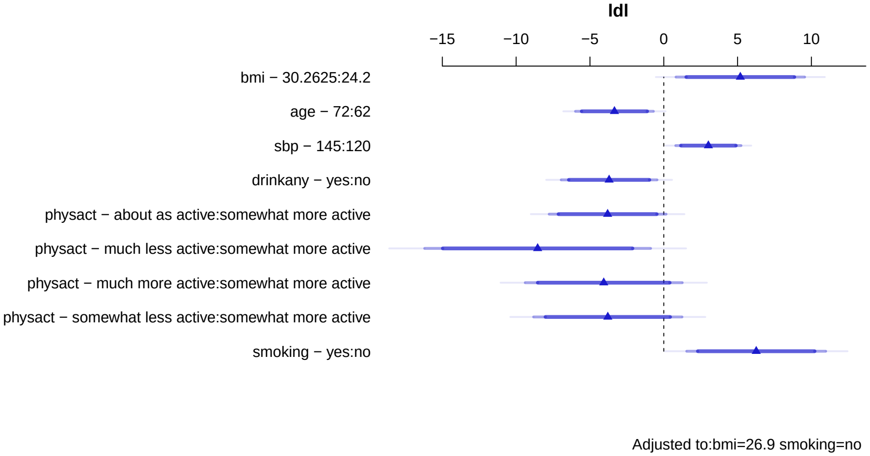
```
summary(m2)
```

```
             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
```

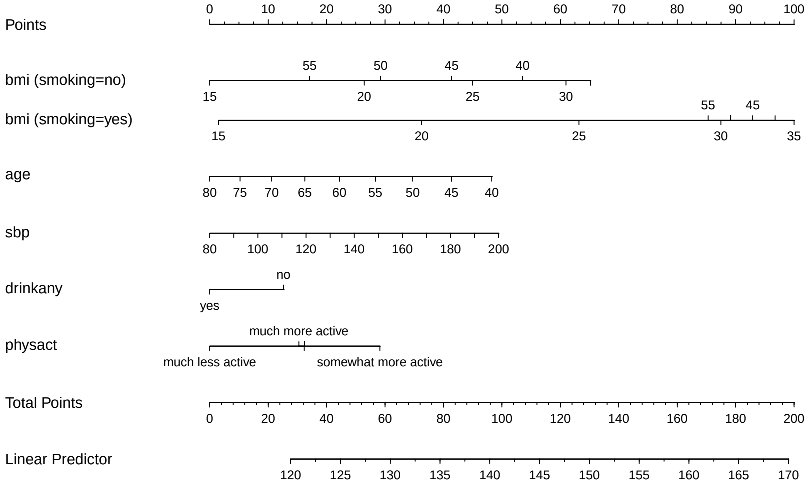
- Of course, these should really be plotted...

Effect Size Plot for `m2`

```
plot(summary(m2))
```



Nomogram for `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.

```
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.

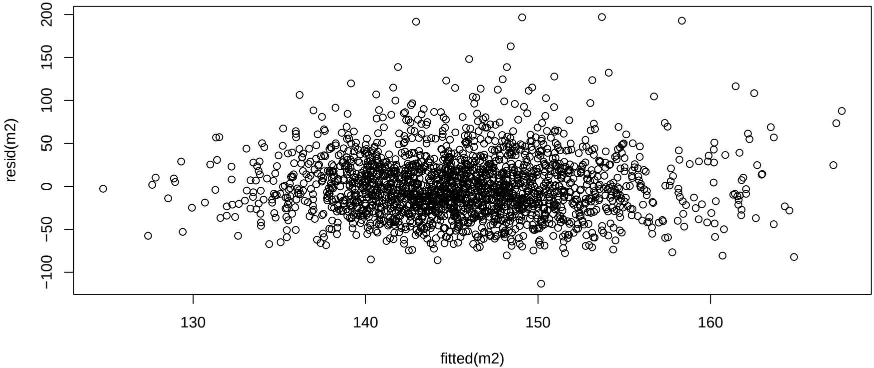
```
predict(m2, expand.grid(bmi = 30, age = 50, smoking = "yes",
                        physact = "about as active",
                        drinkany = "yes", sbp = 150),
        conf.int = 0.95, conf.type = "mean")
```

```
$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.

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 the Model using 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)
```

Now we put them together with the `fit.mult.impute` function...

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 (screen 1/2)

```
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, 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
```

m3imp results (screen 2/2)

```
m3imp
```

	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)
```

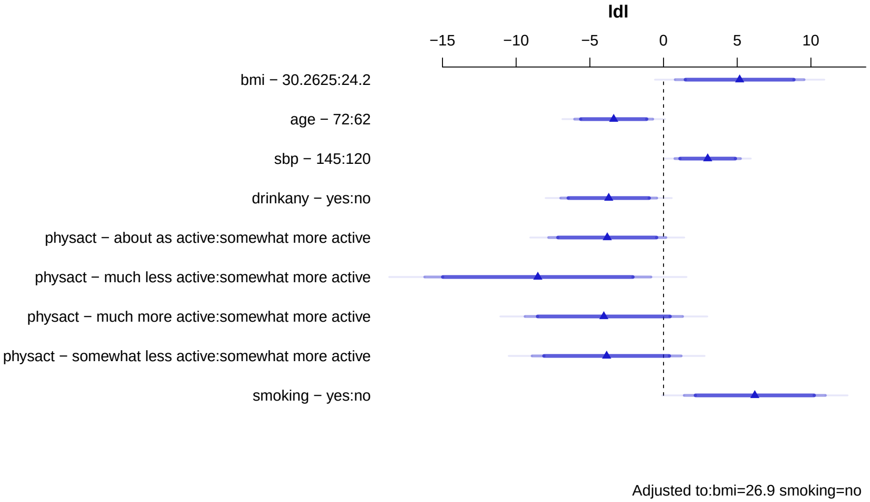
```
                Analysis of Variance          Response: ldl

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

Summary of Effect Estimates for m3imp

```
summary(m3imp)
```

```
             Effects              Response : ldl

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

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

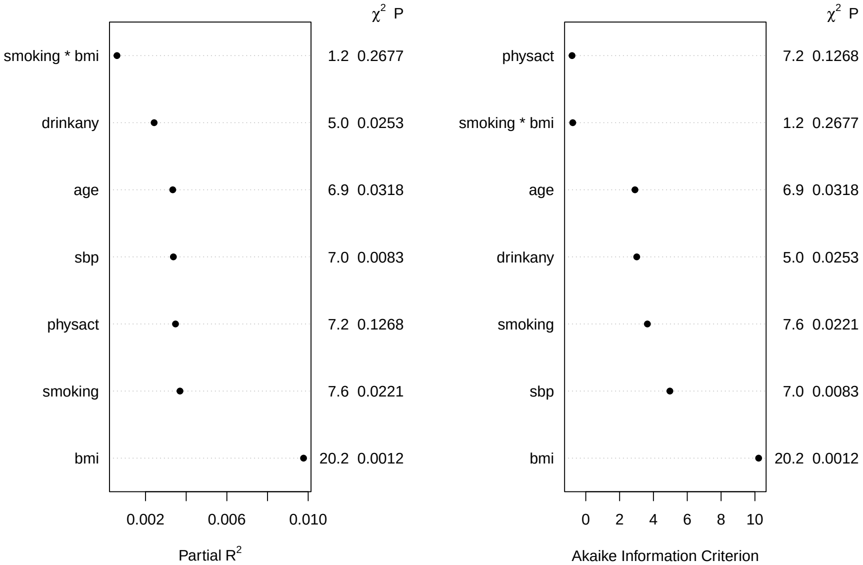
plot(summary(m3imp))



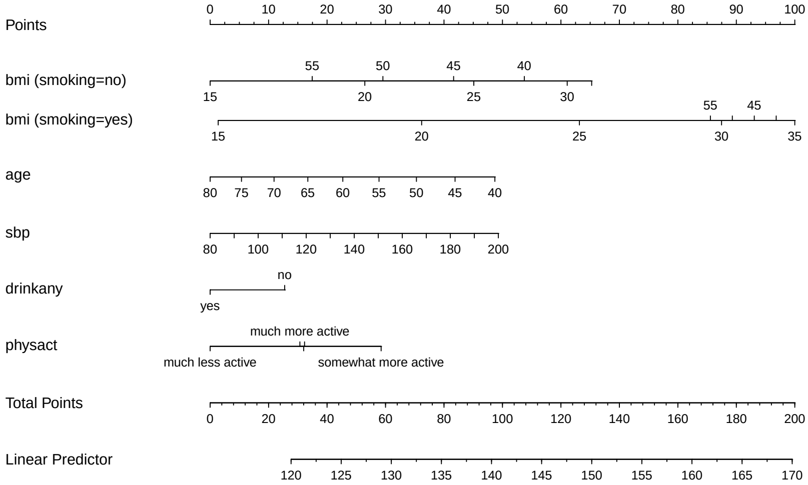
Evaluation via Partial R^2 and AIC (code)

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

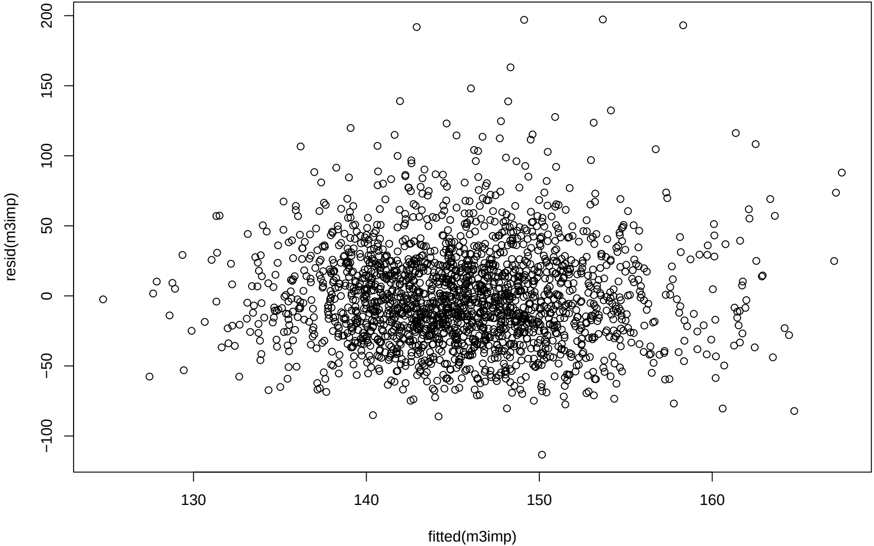
Evaluation via Partial R^2 and AIC (result)



plot(nomogram(m3imp))



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



Other Things I Might Need 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)
```

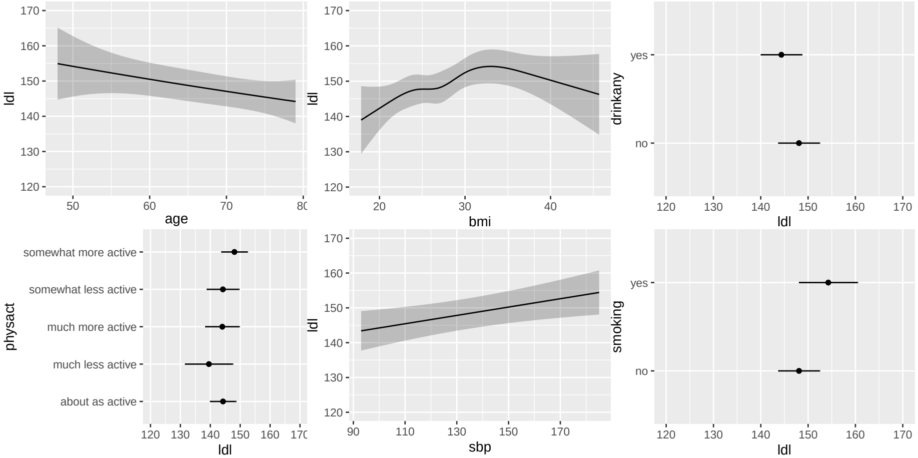
```
    d.f.
20425.29
```

```
BIC(m3imp)
```

```
    d.f.
20515.16
```

Can I run `ggplot(Predict())`?

```
ggplot(Predict(m3imp))
```



Pull out one imputation from aregImpute?

- How can I pull a single one (say, the fifth) of the imputations from aregImpute out?

Remember that fit3 was our imputation model here, build on the hers1 data, which keeps its subject identifiers in the subject column.

```
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) %>%
  type.convert() %>% tibble()
```

```
Warning in type.convert.default(x[[i]], ...): 'as.is' should
be specified by the caller; using TRUE
```

Our fifth_imp tibble

```
fifth_imp
```

```
# A tibble: 2,032 x 8
   subject   ldl   age smoking drinkany   sbp physact    bmi
     <int> <dbl> <int> <chr>      <int> <int> <chr>    <dbl>
 1       1  122.    70 no             1   138 much mo~  23.7
 2       2  242.    62 no             1   118 much le~  28.6
 3       4  116.    64 yes            2   152 much le~  24.4
 4       5  151.    65 no             1   175 somewha~  21.9
 5       6  138.    68 no             2   174 about a~  29.0
 6       8  121.    69 no             1   178 much mo~  23.2
 7       9  133     61 no             2   162 about a~  30.3
 8      10  220     62 yes            2   111 somewha~  45.7
 9      11  173.    72 no             1   122 about a~  22.2
10      12  124.    73 no             1   158 somewha~  25.3
# ... with 2,022 more rows
```

Create Residual Plots for this imputation?

```
model_for_resid_plots <-
  lm(ldl ~ rcs(bmi, 5) + pol(age, 2) + sbp +
                         drinkany + physact + smoking +
                         smoking %ia% bmi, data = fifth_imp)
```

We can look at this model with `glance` or `tidy` to see that it gives similar results to what we see across the multiple imputations.

```
broom::glance(model_for_resid_plots) %>%
  select(r.squared, AIC, BIC, nobs, df, df.residual) %>%
  kable(digits = 3)
```

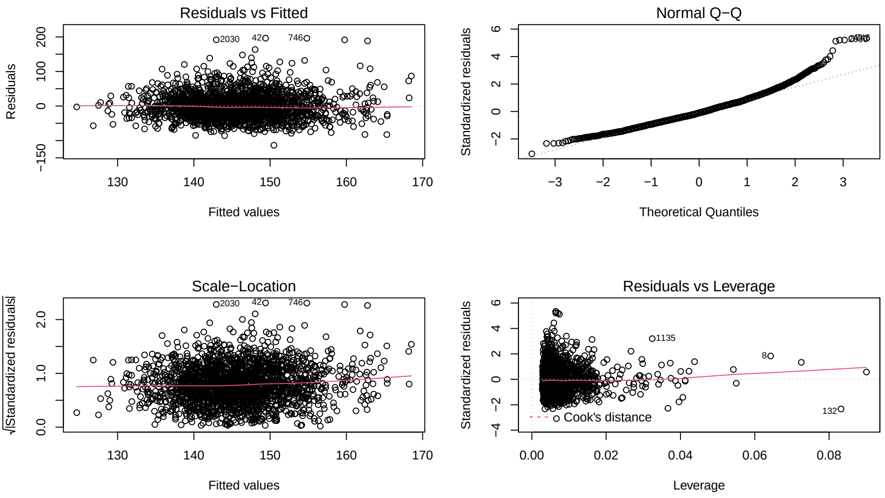
r.squared	AIC	BIC	nobs	df	df.residual
0.027	20451.36	20541.23	2032	14	2017

What else can we do?

We can plot residuals for the model fit to this single imputation, as shown on the next slide.

```
par(mfrow = c(2,2))
plot(model_for_resid_plots)
par(mfrow = c(1,1))
```

Residual Plots for Fifth Imputation



Next Step

Can we do all of this for a logistic regression model?