432 Class 12 Slides

thomaselove.github.io/432

2022-02-17

Today's Agenda

Some reminders and loose ends

- for linear regression models
- for logistic regression models

We'll return to `tidymodels` next time.

Setup

```
library(here); library(knitr)
library(magrittr); library(janitor)
library(naniar); library(equatiomatic)
library(GGally); library(broom)
library(rms)

library(tidyverse)

theme_set(theme_bw())
```

Linear Regression

The day12 Data Set

These data are simulated.

```
dat12 <- readRDS(here("data/dat12.Rds"))

names(dat12)
```

```
[1] "subj"   "result" "sur_s"  "typeA"  "sbp"    "sroh"
```

```
miss_case_table(dat12)
```

```
# A tibble: 1 x 3
  n_miss_in_case n_cases pct_cases
           <int>   <int>     <dbl>
1              0     400       100
```

The dat12 codebook

Variable	Description	Type
result	Our outcome (0-500 scale)	quant.
sur_s	Survey sur_s (0-200 scale)	quant.
typeA	Type A (No or Yes)	binary
sbp	Systolic Blood Pressure	quant.
sroh	Self-Reported Health (E/VG/G/F)	4 cats.

Summary of dat12

```
summary(dat12 %>% select(-subj))
```

```
     result          sur_s          typeA           sbp
 Min.   : 46.0   Min.   : 39.0   No :203   Min.   : 85.0
 1st Qu.:160.0   1st Qu.: 87.0   Yes:197   1st Qu.:132.0
 Median :168.0   Median :101.0             Median :147.0
 Mean   :168.8   Mean   :100.2             Mean   :148.1
 3rd Qu.:177.0   3rd Qu.:114.0             3rd Qu.:165.0
 Max.   :483.0   Max.   :185.0             Max.   :215.0
 sroh
 E : 68
 VG:147
 G :139
 F : 46
```

OLS Model for ‘result“ without Non-Linear Terms

Variable	Description
`result`	Our outcome (0-500 scale)
`sur_s`	Survey sur_s (0-200 scale)
`typeA`	Type A (No or Yes)
`sbp`	Systolic Blood Pressure
`sroh`	Self-Reported Health (E/VG/G/F)

```
d <- datadist(dat12)
options(datadist = "d")

modA <- ols(result ~ sur_s + typeA + sbp + sroh,
                data = dat12, x = TRUE, y = TRUE)
```

How many degrees of freedom does the model `modA` use?

Model modA

```
modA
```

```
> modA
Linear Regression Model

 ols(formula = result ~ sur_s + typeA + sbp + sroh, data = dat12,
     x = TRUE, y = TRUE)

                  Model Likelihood    Discrimination
                        Ratio Test           Indexes
 Obs       400     LR chi2     296.82    R2       0.524
 sigma16.8657     d.f.            6    R2 adj   0.517
 d.f.      393     Pr(> chi2) 0.0000    g       19.692

 Residuals

      Min       1Q   Median       3Q      Max
 -69.1043  -7.1600  -0.7081   6.0485 250.0511


           Coef     S.E.   t     Pr(>|t|)
 Intercept 134.0500 7.1558 18.73 <0.0001
 sur_s       0.7180 0.0411 17.48 <0.0001
 typeA=Yes  10.1832 1.6977  6.00 <0.0001
 sbp        -0.2292 0.0365 -6.28 <0.0001
 sroh=VG    -7.7635 2.4803 -3.13 0.0019
 sroh=G    -11.1855 2.5028 -4.47 <0.0001
 sroh=F    -12.9429 3.2348 -4.00 <0.0001
```

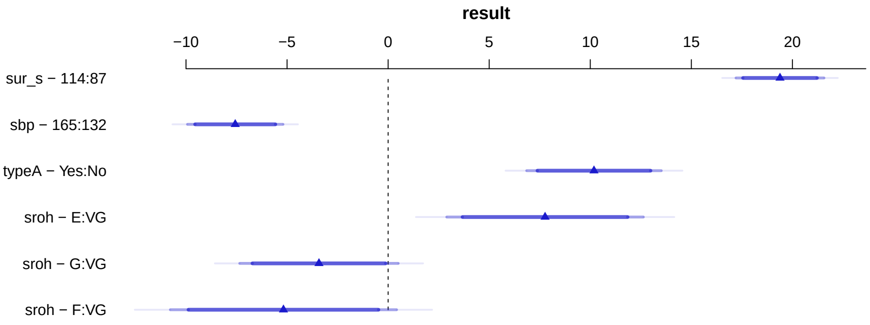
ANOVA results for `modA`

```
anova(modA)
```

```
                Analysis of Variance          Response: result

 Factor     d.f. Partial SS MS          F      P
 sur_s        1   86894.325 86894.3250 305.48 <.0001
 typeA        1   10233.702 10233.7022  35.98 <.0001
 sbp          1   11222.442 11222.4417  39.45 <.0001
 sroh         3    6825.387  2275.1291   8.00 <.0001
 REGRESSION   6  122994.902 20499.1504  72.07 <.0001
 ERROR      393  111789.458   284.4515
```

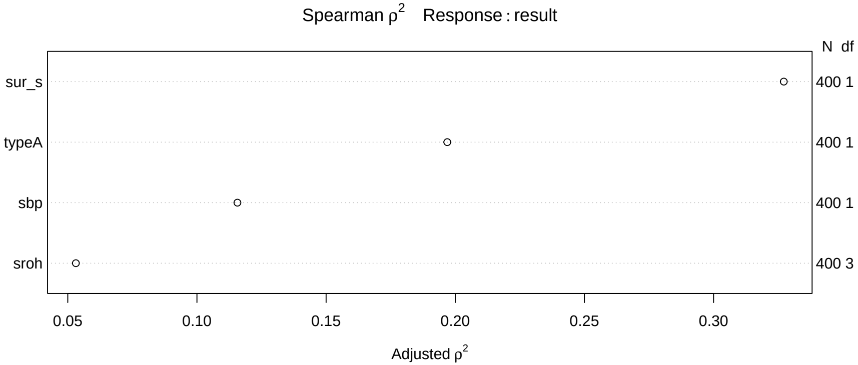
Plot Effect Sizes

```
plot(summary(modA))
```



Consider Potential Non-Linear Terms

```
plot(spearman2(result ~ sur_s + typeA + sbp + sroh,
               data = dat12))
```



Using the Spearman plot as a guide...

Variable	Description	Adj. Spearman ρ^2
`sur_s`	Survey sur_s (0-200 scale)	Highest
`typeA`	Type A (No or Yes)	2nd Highest
`sbp`	Systolic Blood Pressure	3rd Highest
`sroh`	Self-Reported Health (E/VG/G/F)	Lowest

Using Polynomials or Splines

- Can we build a (polynomial or spline) non-linear term that will add one more degree of freedom to our original main-effects model?
- What if we can afford 2 additional df? Or 3?

Using Interaction terms

- How many df does the best categorical-categorical interaction use?
- How many df does the best categorical-quantitative interaction use?

Adding Polynomial Terms in `sur_s`

We'll look at a quadratic, then a cubic polynomial...

```
modP2 <- ols(result ~ pol(sur_s,2) + typeA + sbp + sroh,
                data = dat12, x = TRUE, y = TRUE)
modP3 <- ols(result ~ pol(sur_s,3) + typeA + sbp + sroh,
                data = dat12, x = TRUE, y = TRUE)
```

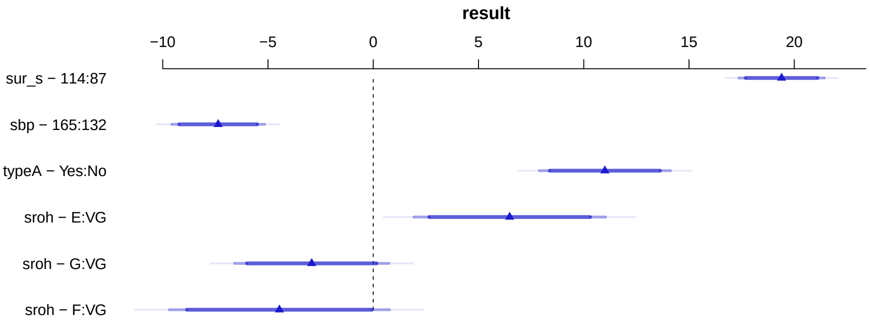
Quadratic Polynomial adds 1 df to `modA`'s 6

```
modP2
```

```
> modP2
Linear Regression Model

 ols(formula = result ~ pol(sur_s, 2) + typeA + sbp + sroh, data = dat12,
     x = TRUE, y = TRUE)

                  Model Likelihood    Discrimination
                        Ratio Test           Indexes
 Obs       400    LR chi2    353.07    R2       0.586
 sigma15.7405    d.f.            7    R2 adj   0.579
 d.f.      392    Pr(> chi2) 0.0000    g       19.680

 Residuals

       Min        1Q    Median        3Q       Max
 -100.7397   -7.3692    0.6981    7.5392  188.5970


           Coef     S.E.    t     Pr(>|t|)
 Intercept 222.3626 13.2799 16.74 <0.0001
 sur_s      -1.1718  0.2486 -4.71 <0.0001
 sur_s^2     0.0094  0.0012  7.69 <0.0001
 typeA=Yes  11.0031  1.5881  6.93 <0.0001
 sbp        -0.2232  0.0341 -6.55 <0.0001
 sroh=VG    -6.4802  2.3209 -2.79 0.0055
 sroh=G     -9.4029  2.3473 -4.01 <0.0001
 sroh=F    -10.9390  3.0302 -3.61 0.0003
```

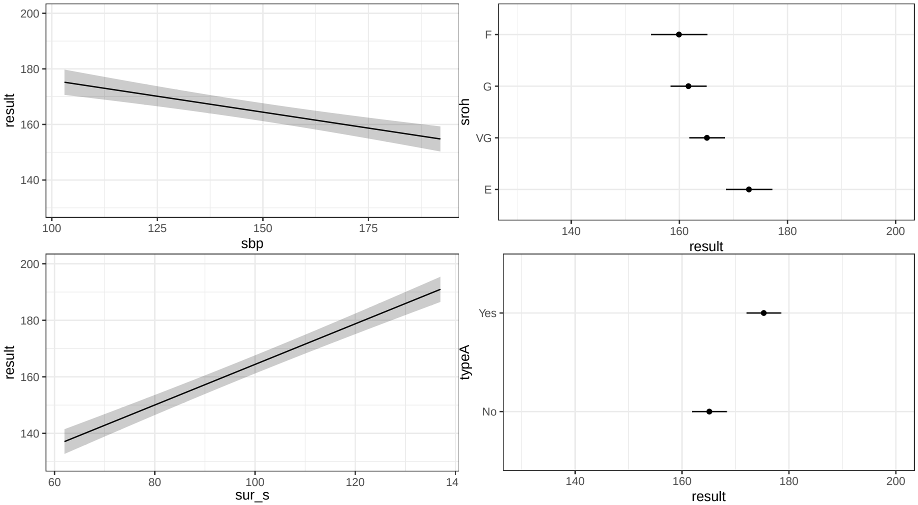
Plot Effect Sizes

```
plot(summary(modP2))
```



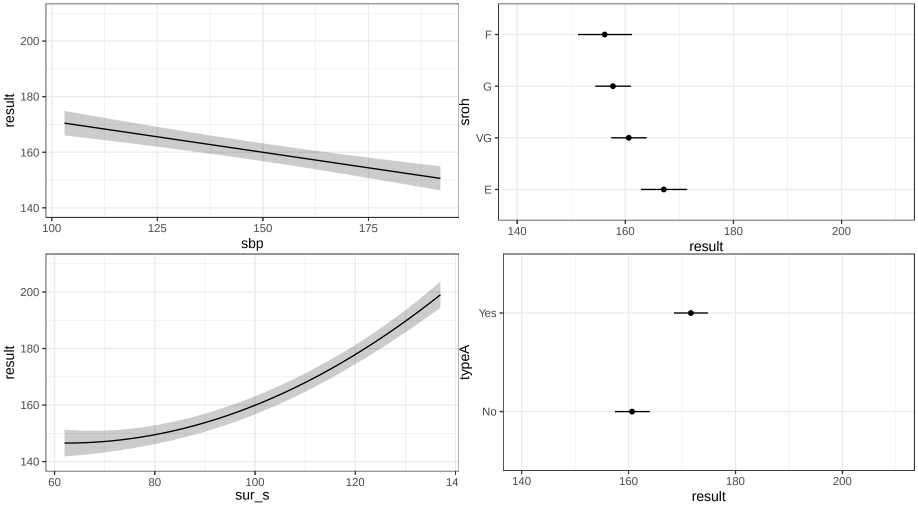
What does model `modA` look like?

```
ggplot(Predict(modA))
```



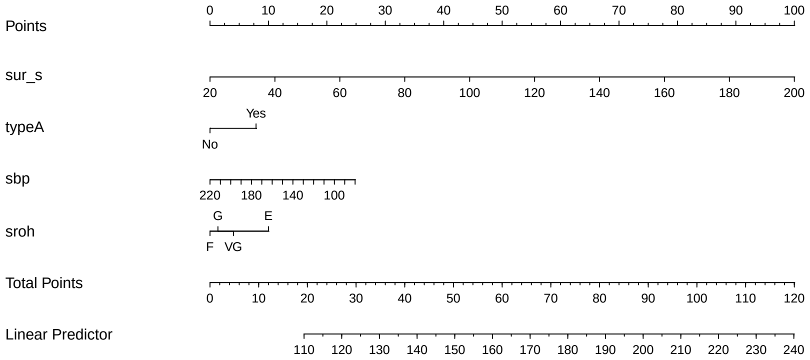
What does model `modP2` look like?

```
ggplot(Predict(modP2))
```



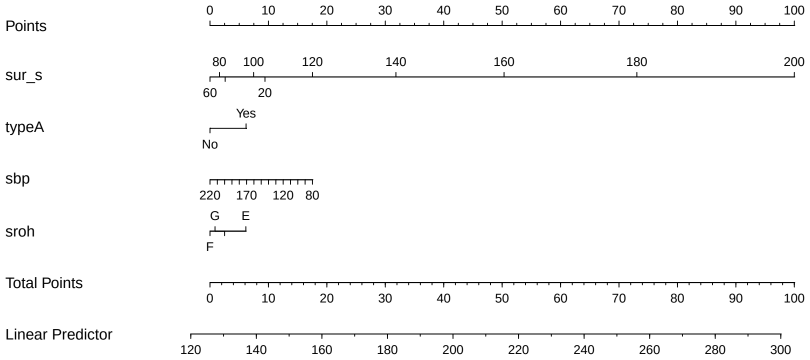
Nomogram for model `modA`

```
plot(nomogram(modA))
```



Nomogram for model `modP2`

```
plot(nomogram(modP2))
```



Do the non-linear terms in modP2 do much?

```
anova(modP2)
```

```
                Analysis of Variance          Response: result

 Factor      d.f. Partial SS MS         F      P
 sur_s         2  101560.602 50780.3008 204.95 <.0001
  Nonlinear    1   14666.277 14666.2767  59.19 <.0001
 typeA         1   11894.005 11894.0047  48.01 <.0001
 sbp           1   10636.075 10636.0753  42.93 <.0001
 sroh          3    4795.273  1598.4244   6.45 3e-04
 REGRESSION    7  137661.179 19665.8827  79.37 <.0001
 ERROR       392   97123.181   247.7632
```

Do the non-linear terms in modP2 help much?

```
AIC(modA); BIC(modA)
```

```
    d.f.
3404.314

    d.f.
3436.246
```

```
AIC(modP2); BIC(modP2)
```

```
    d.f.
3350.059

    d.f.
3385.982
```

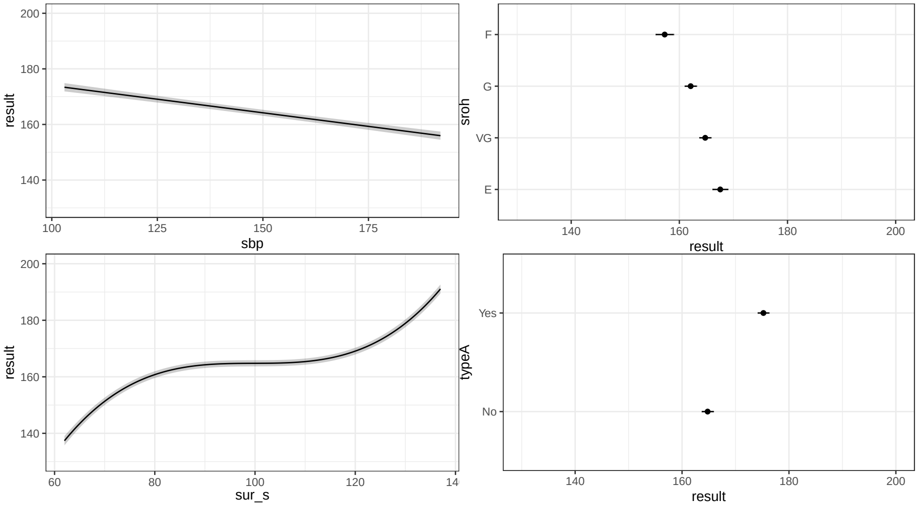
Cubic (degree 3) polynomial adds 2 df to `modA`'s 6

```
modP3
```

```
> modP3
Linear Regression Model

 ols(formula = result ~ pol(sur_s, 3) + typeA + sbp + sroh, data = dat12,
     x = TRUE, y = TRUE)

                    Model Likelihood    Discrimination
                          Ratio Test           Indexes
 Obs      400     LR chi2     1227.39    R2       0.954
 sigma5.2836     d.f.            8    R2 adj   0.953
 d.f.     391     Pr(> chi2)  0.0000    g       17.754

 Residuals

      Min       1Q   Median       3Q      Max
 -20.9404  -3.5022   0.1353   3.2907  14.6336


           Coef      S.E.    t      Pr(>|t|)
 Intercept -304.4469 10.4759 -29.06 <0.0001
 sur_s       15.0487  0.3036  49.57 <0.0001
 sur_s^2     -0.1508  0.0029 -51.79 <0.0001
 sur_s^3      0.0005  0.0000  55.57 <0.0001
 typeA=Yes   10.4406  0.5332  19.58 <0.0001
 sbp         -0.1955  0.0114 -17.08 <0.0001
 sroh=VG     -2.7809  0.7819  -3.56 0.0004
 sroh=G      -5.4697  0.7911  -6.91 <0.0001
 sroh=F     -10.2759  1.0172 -10.10 <0.0001
```

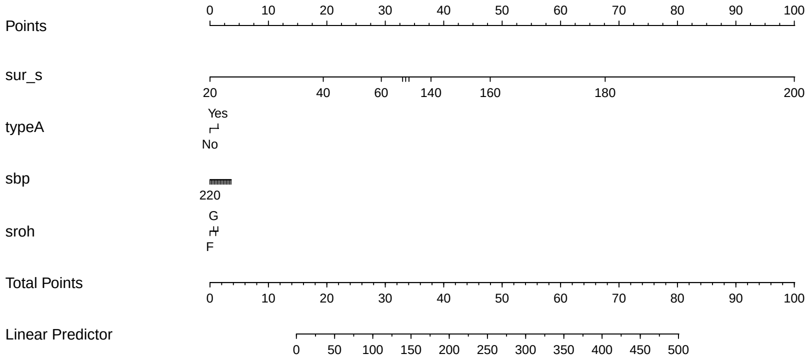
What does model modP3 look like?

```
ggplot(Predict(modP3))
```



Nomogram for model `modP3`

```
plot(nomogram(modP3))
```



How about a restricted cubic spline in cigs?

```
modC3 <- ols(result ~ rcs(sur_s,3) + typeA + sbp + sroh,
               data = dat12, x = TRUE, y = TRUE)
modC4 <- ols(result ~ rcs(sur_s,4) + typeA + sbp + sroh,
               data = dat12, x = TRUE, y = TRUE)
modC5 <- ols(result ~ rcs(sur_s,5) + typeA + sbp + sroh,
               data = dat12, x = TRUE, y = TRUE)
```

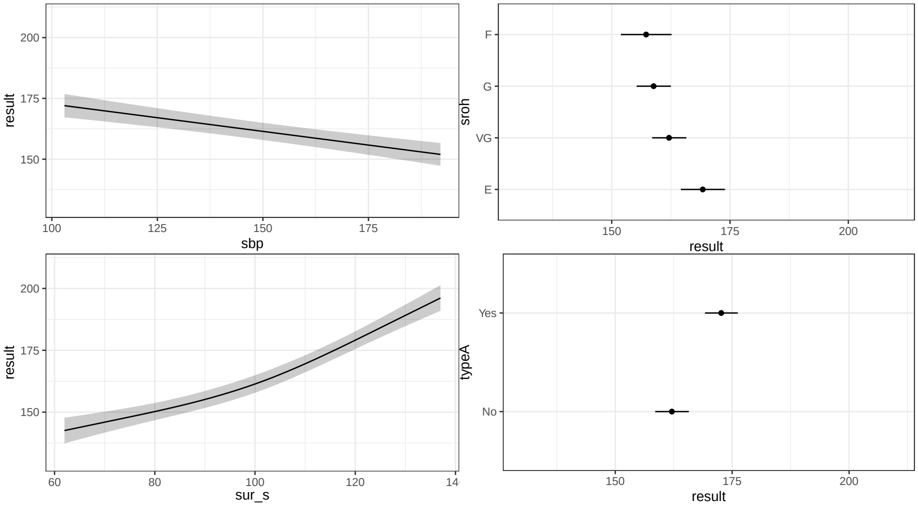
RCS with 3 knots adds 1 df to `modA`'s 6

```
modC3
```

```
> modC3
Linear Regression Model

 ols(formula = result ~ rcs(sur_s, 3) + typeA + sbp + sroh, data = dat12,
     x = TRUE, y = TRUE)

                  Model Likelihood    Discrimination
                     Ratio Test           Indexes
 Obs       400    LR chi2     310.56    R2       0.540
 sigma16.5997    d.f.             7    R2 adj   0.532
 d.f.      392    Pr(> chi2) 0.0000    g       19.734

 Residuals

       Min         1Q     Median         3Q       Max
 -81.57619  -7.35312  -0.04281   6.90506 231.78407


           Coef     S.E.   t     Pr(>|t|)
 Intercept 156.6095 9.3149 16.81 <0.0001
 sur_s       0.4221 0.0896  4.71 <0.0001
 sur_s'      0.3544 0.0958  3.70 0.0002
 typeA=Yes  10.5500 1.6739  6.30 <0.0001
 sbp        -0.2251 0.0359 -6.26 <0.0001
 sroh=VG    -7.1198 2.4474 -2.91 0.0038
 sroh=G    -10.3836 2.4729 -4.20 <0.0001
 sroh=F    -11.9694 3.1947 -3.75 0.0002
```

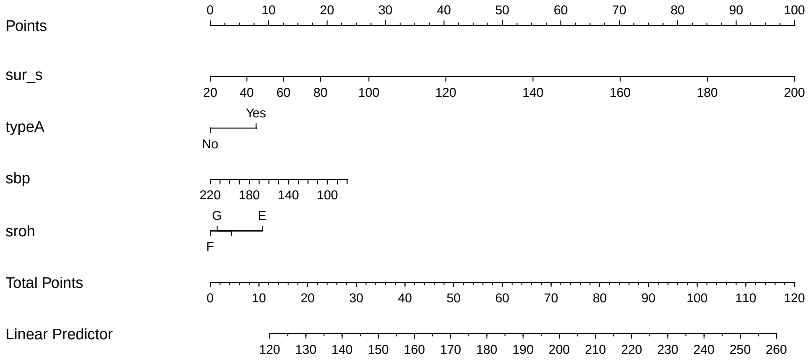
What does model `modC3` look like?

```
ggplot(Predict(modC3))
```



What does the nomogram for `modC3` look like?

```
plot(nomogram(modC3))
```



Do the non-linear terms help much in `modC3`?

```r
AIC(modC3); BIC(modC3)
```

```
    d.f.
3392.579
```

```
    d.f.
3428.502
```

```r
AIC(modA); BIC(modA)
```

```
    d.f.
3404.314
```

```
    d.f.
3436.246
```

ANOVA table for `modC3`?

```
anova(modC3)
```

```
                Analysis of Variance          Response: result

 Factor     d.f. Partial SS MS         F      P
 sur_s        2   90667.755 45333.8775 164.52 <.0001
  Nonlinear   1    3773.430  3773.4300  13.69 2e-04
 typeA        1   10945.709 10945.7087  39.72 <.0001
 sbp          1   10806.525 10806.5247  39.22 <.0001
 sroh         3    5820.777  1940.2589   7.04 1e-04
 REGRESSION   7  126768.332 18109.7617  65.72 <.0001
 ERROR      392  108016.028   275.5511
```

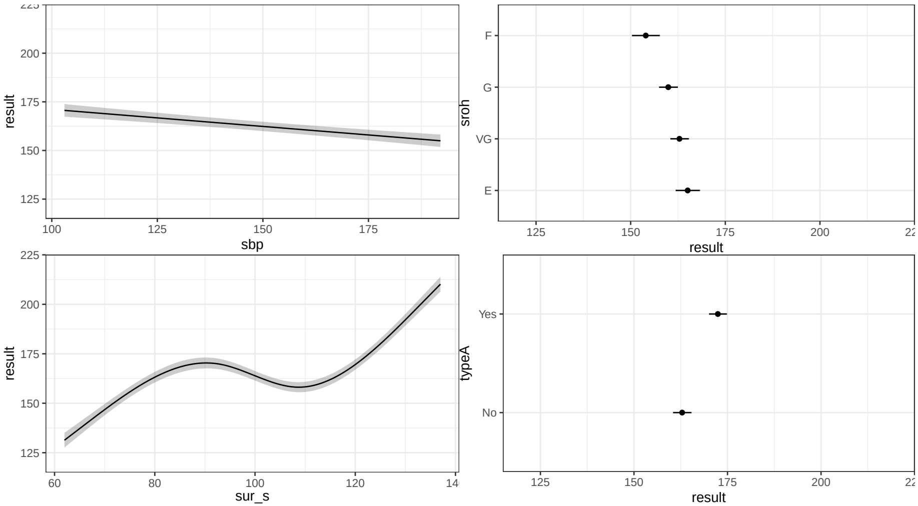
RCS with 4 knots adds 2 df to `modA`'s 6

```
modC4
```

```
> modC4
Linear Regression Model

 ols(formula = result ~ rcs(sur_s, 4) + typeA + sbp + sroh, data = dat12,
     x = TRUE, y = TRUE)

                  Model Likelihood    Discrimination
                        Ratio Test           Indexes
 Obs       400    LR chi2     617.42    R2       0.786
 sigma11.3258    d.f.             8    R2 adj   0.782
 d.f.      391    Pr(> chi2) 0.0000    g       20.628

 Residuals

      Min       1Q   Median       3Q      Max
 -36.3707  -4.5731   0.2394   4.8794 147.0369


           Coef     S.E.   t      Pr(>|t|)
 Intercept  39.4204 8.7154   4.52 <0.0001
 sur_s       1.9340 0.0989  19.55 <0.0001
 sur_s'     -4.9038 0.2683 -18.27 <0.0001
 sur_s''    23.5323 1.1447  20.56 <0.0001
 typeA=Yes   9.5443 1.1433   8.35 <0.0001
 sbp        -0.1753 0.0246  -7.12 <0.0001
 sroh=VG    -2.1722 1.6858  -1.29 0.1983
 sroh=G     -5.1198 1.7053  -3.00 0.0029
 sroh=F    -11.0668 2.1802  -5.08 <0.0001
```

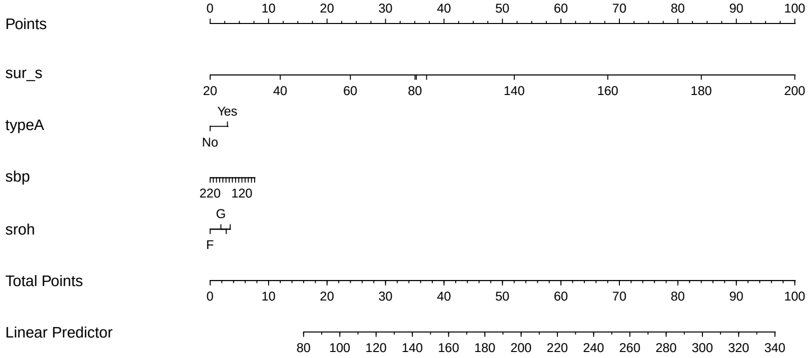
What does model modC4 look like?

```
ggplot(Predict(modC4))
```



What does the nomogram for `modC4` look like?

```
plot(nomogram(modC4))
```



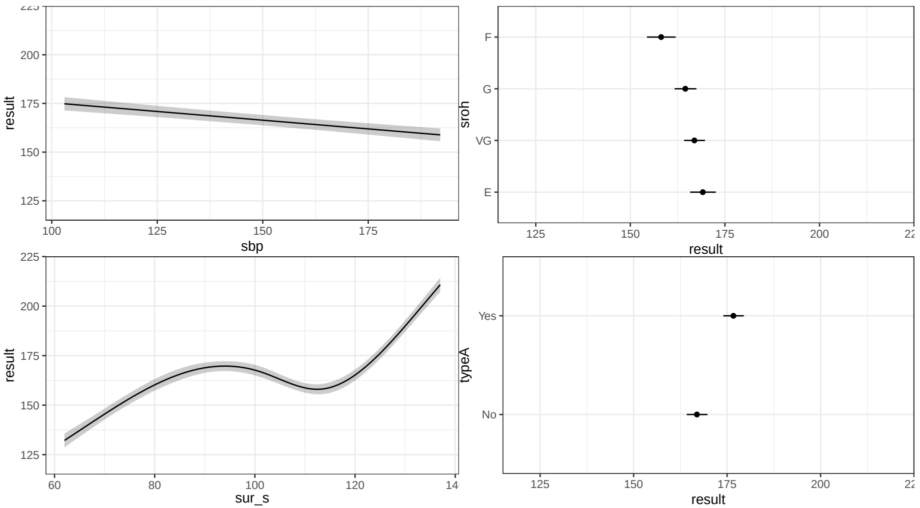
RCS with 5 knots adds 3 df to modA's 6

```
modC5
```

```
> modC5
Linear Regression Model

 ols(formula = result ~ rcs(sur_s, 5) + typeA + sbp + sroh, data = dat12,
     x = TRUE, y = TRUE)

                  Model Likelihood    Discrimination
                        Ratio Test           Indexes
 Obs      400    LR chi2    665.58    R2       0.811
 sigma10.6778    d.f.            9    R2 adj   0.806
 d.f.     390    Pr(> chi2) 0.0000    g       20.398

 Residuals

      Min       1Q   Median       3Q      Max
 -43.8254  -4.1905   0.2477   4.9296 126.8751


           Coef     S.E.   t     Pr(>|t|)
 Intercept  57.3276 9.6251  5.96 <0.0001
 sur_s       1.6682 0.1186 14.06 <0.0001
 sur_s'     -3.2491 0.5682 -5.72 <0.0001
 sur_s''     1.6160 3.0831  0.52 0.6005
 sur_s'''   27.0995 5.2221  5.19 <0.0001
 typeA=Yes   9.7539 1.0783  9.05 <0.0001
 sbp        -0.1791 0.0233 -7.70 <0.0001
 sroh=VG    -2.2273 1.5891 -1.40 0.1618
 sroh=G     -4.6241 1.6095 -2.87 0.0043
 sroh=F    -11.0161 2.0555 -5.36 <0.0001
```

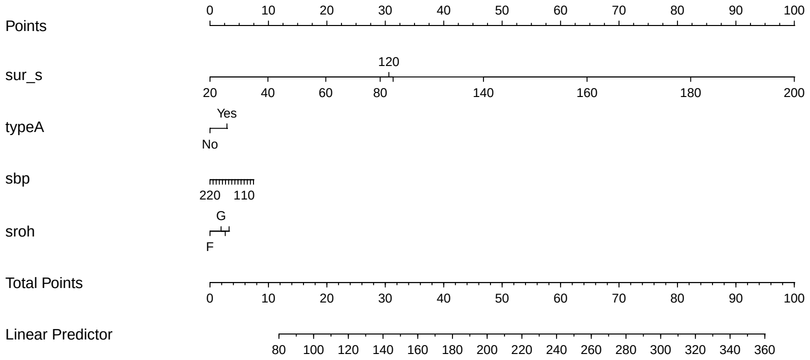
What does model modC5 look like?

```
ggplot(Predict(modC5))
```



What does the nomogram for `modC5` look like?

```
plot(nomogram(modC5))
```



Splines and Polynomials with `ols` (or `lrm`)

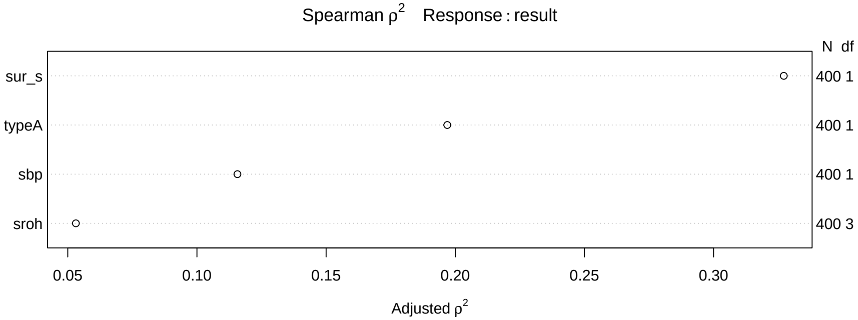
Model	Coeffs.	"Bends"	DF added
Main Effects (`modA`)	None	None	–
Polynomial, degree 2 (`P2`)	`^2`	1	1
Polynomial, degree 3 (`P3`)	`^2, ^3`	2	2
RCS, 3 knots (`C3`)	`'`	2	1
RCS, 4 knots (`C4`)	`', ''`	3	2
RCS, 5 knots (`C5`)	`', '', '''`	4	3

- RCS = Restricted Cubic Spline

What about an interaction term instead?

1. How many df does the best categorical-categorical interaction use?
2. How many df does the best categorical-quantitative interaction use?

```
plot(spearman2(result ~ sur_s + typeA + sbp + sroh,
               data = dat12))
```



Models with Interaction Terms

```
# already set up datadist for dat12

modI1 <- ols(result ~ sur_s * typeA + sbp + sroh,
             data = dat12, x = TRUE, y = TRUE)
modI2 <- ols(result ~ rcs(sur_s, 4) + typeA +
                 sur_s %ia% typeA + sbp + sroh,
             data = dat12, x = TRUE, y = TRUE)
```

Model `modI1` adds how many df to `modA`?

```
modI1
```

```
> modI1
Linear Regression Model

 ols(formula = result ~ sur_s * typeA + sbp + sroh, data = dat12)

                  Model Likelihood    Discrimination
                        Ratio Test           Indexes
 Obs       400     LR chi2     312.57    R2        0.542
 sigma16.5580     d.f.            7      R2 adj    0.534
 d.f.      392     Pr(> chi2) 0.0000    g        19.753

 Residuals

      Min       1Q   Median       3Q      Max
 -61.0473  -6.8744  -0.7916   6.0512 238.3297


                   Coef     S.E.   t     Pr(>|t|)
 Intercept         118.8604 8.0008 14.86 <0.0001
 sur_s               0.8596 0.0539 15.96 <0.0001
 typeA=Yes          42.5882 8.3362  5.11 <0.0001
 sbp                -0.2245 0.0359 -6.26 <0.0001
 sroh=VG            -7.1991 2.4392 -2.95 0.0034
 sroh=G            -10.3201 2.4668 -4.18 <0.0001
 sroh=F            -12.7668 3.1761 -4.02 <0.0001
 sur_s * typeA=Yes  -0.3233 0.0815 -3.97 <0.0001
```

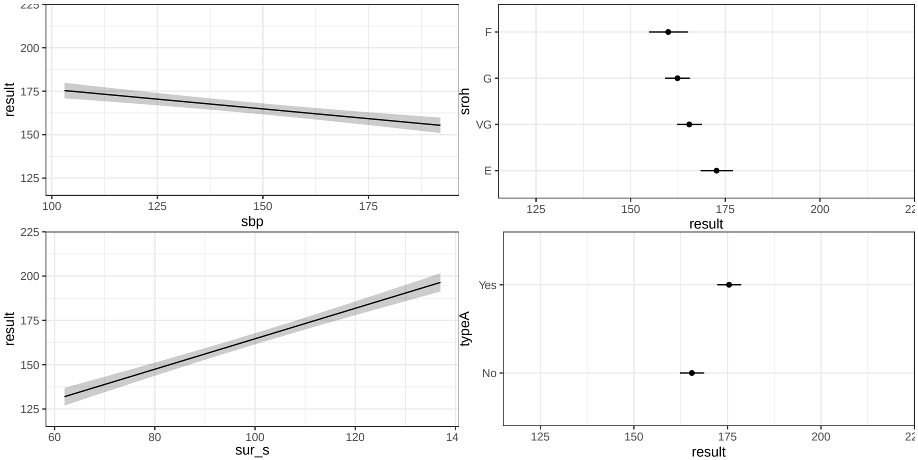
ANOVA for modI1

```
anova(modI1)
```

```
> anova(modI1)
                Analysis of Variance          Response: result

 Factor                                        d.f. Partial SS MS         F      P
 sur_s  (Factor+Higher Order Factors)            2   91209.748 45604.8740 166.34 <.0001
  All Interactions                               1    4315.423  4315.4231  15.74 1e-04
 typeA  (Factor+Higher Order Factors)            2   14549.125  7274.5626  26.53 <.0001
  All Interactions                               1    4315.423  4315.4231  15.74 1e-04
 sbp                                             1   10749.174 10749.1735  39.21 <.0001
 sroh                                            3    6136.426  2045.4754   7.46 1e-04
 sur_s * typeA  (Factor+Higher Order Factors)    1    4315.423  4315.4231  15.74 1e-04
 REGRESSION                                      7  127310.325 18187.1893  66.34 <.0001
 ERROR                                         392  107474.035   274.1685
```

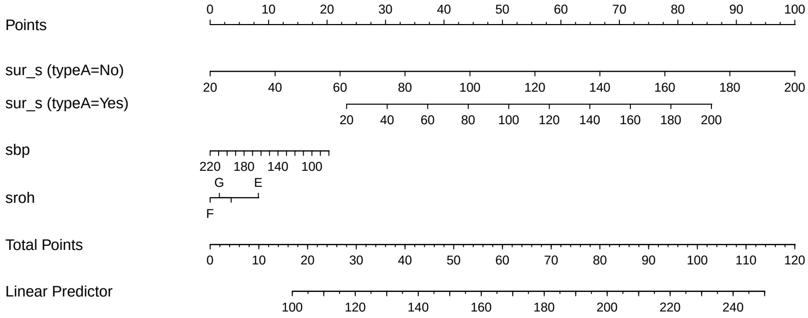
What does `modI1` look like?

```
ggplot(Predict(modI1))
```



Nomogram for `modI1`

```
plot(nomogram(modI1))
```



Model `modI2` adds how many df to `modC4`?

```
modI2
```

```
> modI2
Linear Regression Model

 ols(formula = result ~ rcs(sur_s, 4) + typeA + sur_s %ia% typeA +
     sbp + sroh, data = dat12)

                  Model Likelihood    Discrimination
                        Ratio Test           Indexes
 Obs       400    LR chi2     634.39    R2         0.795
 sigma11.1022     d.f.             9    R2 adj     0.791
 d.f.      390    Pr(> chi2) 0.0000    g         20.660

 Residuals

     Min       1Q  Median       3Q     Max
 -32.937  -4.914  -0.253   5.024 138.939


                   Coef     S.E.    t      Pr(>|t|)
 Intercept          33.6458 8.6580   3.89 0.0001
 sur_s               1.9693 0.0973  20.23 <0.0001
 sur_s'             -4.7400 0.2660 -17.82 <0.0001
 sur_s''            22.9316 1.1316  20.26 <0.0001
 typeA=Yes          32.4408 5.6801   5.71 <0.0001
 sur_s * typeA=Yes  -0.2278 0.0554  -4.11 <0.0001
 sbp                -0.1728 0.0242  -7.15 <0.0001
 sroh=VG            -1.8306 1.6546  -1.11 0.2693
 sroh=G             -4.5556 1.6773  -2.72 0.0069
 sroh=F            -10.8748 2.1376  -5.09 <0.0001
```

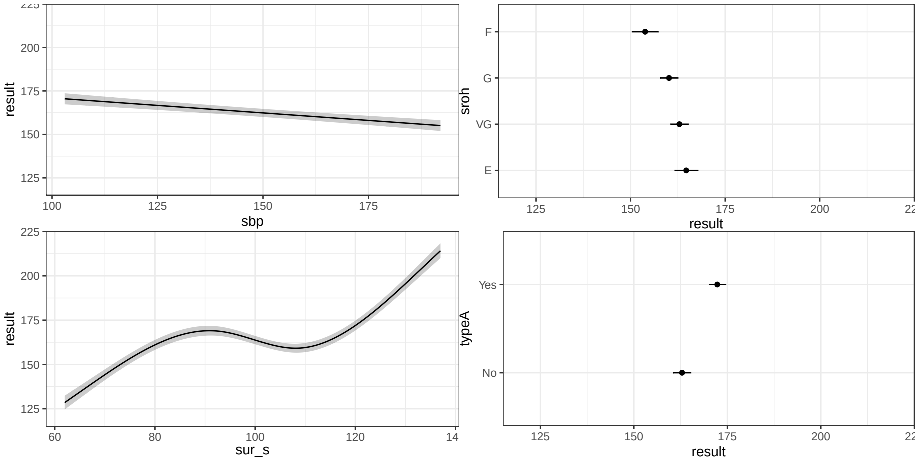
ANOVA for `modI2`

```
anova(modI2)
```

```
> anova(modI2)
                Analysis of Variance          Response: result

 Factor                                        d.f. Partial SS MS          F      P
 sur_s  (Factor+Higher Order Factors)            4  150612.707 37653.1768 305.48 <.0001
  All Interactions                               1    2083.922  2083.9220  16.91 <.0001
  Nonlinear                                      2   59402.959 29701.4795 240.97 <.0001
 typeA  (Factor+Higher Order Factors)            2   11023.724  5511.8620  44.72 <.0001
  All Interactions                               1    2083.922  2083.9220  16.91 <.0001
 sur_s * typeA  (Factor+Higher Order Factors)    1    2083.922  2083.9220  16.91 <.0001
 sbp                                             1    6308.212  6308.2124  51.18 <.0001
 sroh                                            3    3849.970  1283.3234  10.41 <.0001
 TOTAL NONLINEAR + INTERACTION                   3   63718.382 21239.4607 172.32 <.0001
 REGRESSION                                      9  186713.284 20745.9205 168.31 <.0001
 ERROR                                         390   48071.076   123.2592
```

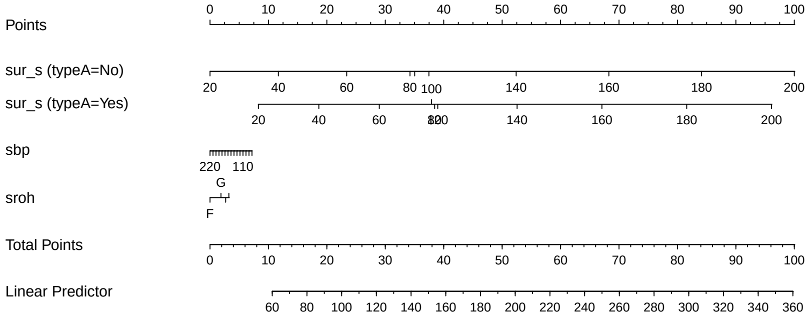
What does `modI2` look like?

```
ggplot(Predict(modI2))
```



Nomogram for `modI2`

```
plot(nomogram(modI2))
```



Comparing Models?

```
set.seed(4321); validate(modA)
```

```
          index.orig training     test optimism
R-square      0.5239   0.5485   0.5035   0.0450
MSE         279.4736 309.0788 291.4385  17.6403
g            19.6922  20.4852  19.5334   0.9518
Intercept     0.0000   0.0000   5.5331  -5.5331
Slope         1.0000   1.0000   0.9670   0.0330
          index.corrected  n
R-square           0.4788 40
MSE              261.8333 40
g                 18.7404 40
Intercept          5.5331 40
Slope              0.9670 40
```

- Ran validate for other models (see next slide)

Table of `validate` Results

Model	Raw R^2	Corrected R^2	Corrected MSE
`modA` (Main Effects)	0.5239	0.4788	261.8
`modP2` (Quadr. Pol.)	0.5863	0.4756	293.8
`modP3` (Cubic Pol.)	0.9535	0.9684	28.5
`modC3` (RCS, 3 knots)	0.5399	0.4510	313.0
`modC4` (RCS, 4 knots)	0.7864	0.7294	162.8
`modC5` (RCS, 5 knots)	0.8106	0.7580	137.6
`modI1` (interaction)	0.5422	0.4413	339.2
`modI2` (int + RCS4)	0.7953	0.7337	161.4

Making Predictions

Suppose we want to predict the `result` for these new subjects:

```
new_people <- tibble(
    name = c("Dave", "Edna"),
    sur_s = c(100, 115), typeA = c("Yes", "No"),
    sbp = c(140, 125), sroh = c("G", "E"))

new_people %>% kable()
```

name	sur_s	typeA	sbp	sroh
Dave	100	Yes	140	G
Edna	115	No	125	E

Predicting Dave and Edna with `modA`

- Individual Prediction Intervals

```
predict(modA, newdata = data.frame(new_people),
        conf.int = 0.95, conf.type = "individual")
```

```
$linear.predictors
       1        2
172.7522 187.9628

$lower
       1        2
139.4297 154.4792

$upper
       1        2
206.0747 221.4463
```

Predicting mean of people just like Dave and Edna with modA

- Mean Prediction Intervals

```
predict(modA, newdata = data.frame(new_people),
        conf.int = 0.95, conf.type = "mean")
```

```
$linear.predictors
       1        2
172.7522 187.9628

$lower
       1        2
169.4477 183.3068

$upper
       1        2
176.0567 192.6188
```

Predicting Dave and Edna with other models

```
predict(modP3, newdata = data.frame(new_people))
```

```
       1        2 
173.8945 173.7410 
```

```
predict(modC4, newdata = data.frame(new_people))
```

```
       1        2 
171.7091 167.9763 
```

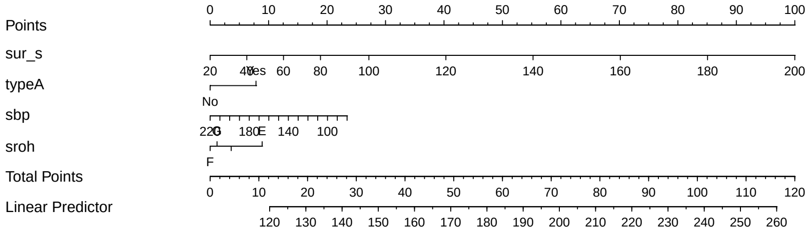
```
predict(modI2, newdata = data.frame(new_people))
```

```
       1        2 
171.8875 169.4290 
```

Predicting Dave via the Nomogram for model `modC3`

- Dave has `sur_s` = 100, is typeA, Good `sroh`, `sbp` = 140.

```
plot(nomogram(modC3))
```



Dave's Actual Predicted Value (from `modC3`)

```
predict(modC3, newdata = data.frame(new_people))[1]
```

```
       1 
170.2422 
```

Running the `lm` version of `modC5`

```
modC5_lm <- lm(result ~ rcs(sur_s,5) + typeA + sbp + sroh,
               data = dat12)

anova(modC5_lm)
```

```
Analysis of Variance Table

Response: result
               Df Sum Sq Mean Sq F value    Pr(>F)
rcs(sur_s, 5)   4 171275   42819 375.551 < 2.2e-16 ***
typeA           1   8141    8141  71.402 5.856e-16 ***
sbp             1   7124    7124  62.479 2.789e-14 ***
sroh            3   3779    1260  11.048 5.627e-07 ***
Residuals     390  44466     114
---
Signif. codes:
0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

The `modC5_lm` model equation

```r
extract_eq(modC5_lm, use_coefs = TRUE, wrap = TRUE,
           terms_per_line = 2)
```

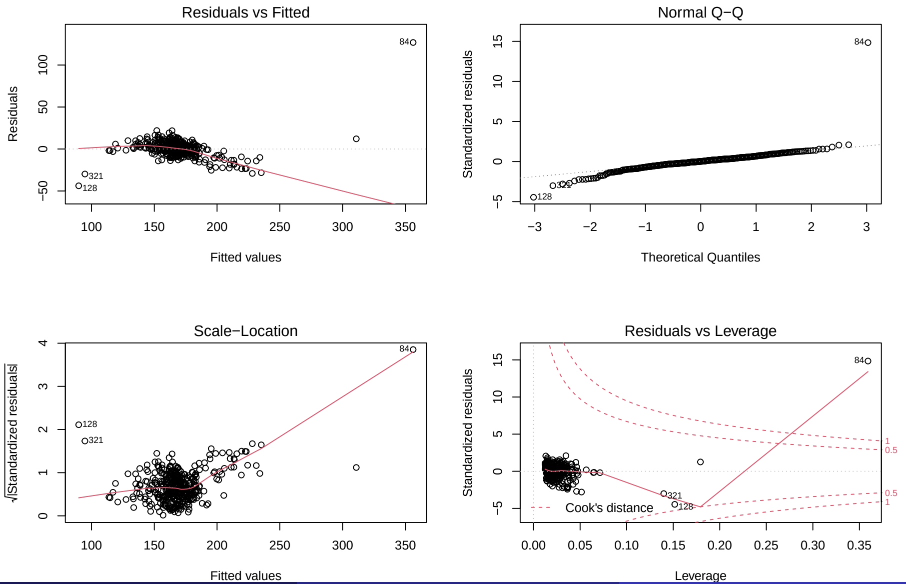
$$
\begin{aligned}
\widehat{\text{result}} &= 57.33 + 1.67(\operatorname{rcs(sur\_s,\ 5)}_{\operatorname{sur\_s}}) - \\
&\quad 3.25(\operatorname{rcs(sur\_s,\ 5)}_{\operatorname{sur\_s'}}) + 1.62(\operatorname{rcs(sur\_s,\ 5)}_{\operatorname{sur\_s''}}) + \\
&\quad 27.1(\operatorname{rcs(sur\_s,\ 5)}_{\operatorname{sur\_s'''}}) + 9.75(\operatorname{typeA}_{\operatorname{Yes}}) - \\
&\quad 0.18(\operatorname{sbp}) - 2.23(\operatorname{sroh}_{\operatorname{VG}}) - \\
&\quad 4.62(\operatorname{sroh}_{\operatorname{G}}) - 11.02(\operatorname{sroh}_{\operatorname{F}})
\end{aligned} \tag{1}
$$

Residual Plots for `modC5`

```
par(mfrow = c(2,2)); plot(modC5_lm); par(mfrow = c(1,1))
```

- Results shown on next slide (not for the faint of heart)

Residual Plots for `modC5`



Oh dear...

```r
dat12 %>% slice(84) %>% kable()
```

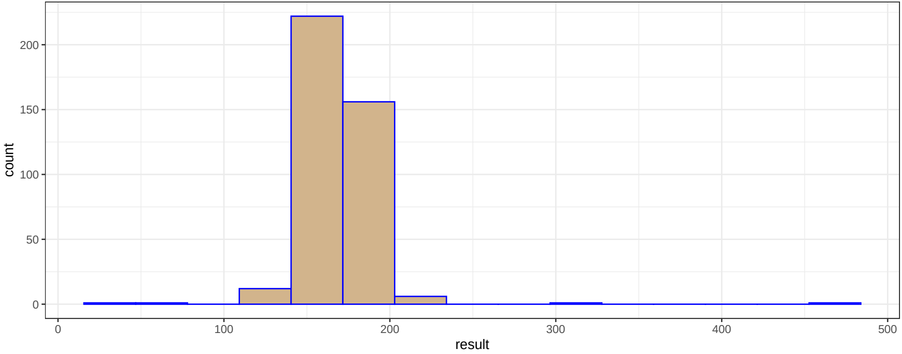
subj	result	sur_s	typeA	sbp	sroh
84	483	185	No	148	E

```r
summary(dat12 %>% select(result, sur_s, sbp, sroh, typeA))
```

```
     result           sur_s             sbp           sroh
 Min.   : 46.0   Min.   : 39.0   Min.   : 85.0   E : 68
 1st Qu.:160.0   1st Qu.: 87.0   1st Qu.:132.0   VG:147
 Median :168.0   Median :101.0   Median :147.0   G :139
 Mean   :168.8   Mean   :100.2   Mean   :148.1   F : 46
 3rd Qu.:177.0   3rd Qu.:114.0   3rd Qu.:165.0
 Max.   :483.0   Max.   :185.0   Max.   :215.0
 typeA
 No :203
 Yes:197
```

Was this foreseeable?

```
ggplot(data = dat12, aes(x = result)) +
    geom_histogram(bins = 15, col = "blue", fill = "tan")
```



Logistic Regression

Framingham Data (from Class 10)

```
fram_raw <- read_csv(here("data/framingham.csv")) %>%
    type.convert(as.is = FALSE) %>%
    clean_names()
```

The variables describe n = 4238 adults examined at baseline, then followed for 10 years to see if they developed incident coronary heart disease. The binary outcome (below) has no missing values.

```
fram_raw %>% tabyl(ten_year_chd)
```

```
 ten_year_chd    n   percent
            0 3594 0.8480415
            1  644 0.1519585
```

Data Cleanup

```
fram_new <- fram_raw %>%
    rename(cigs = "cigs_per_day",
           stroke = "prevalent_stroke",
           hrate = "heart_rate",
           sbp = "sys_bp",
           chd10_n = "ten_year_chd") %>%
    mutate(educ = fct_recode(factor(education),
                         "Some HS" = "1",
                         "HS grad" = "2",
                         "Some Coll" = "3",
                         "Coll grad" = "4")) %>%
    mutate(chd10_f = fct_recode(factor(chd10_n),
                         "chd" = "1", "chd_no" = "0")) %>%
    select(subj_id, chd10_n, chd10_f, age,
           cigs, educ, hrate, sbp, stroke)
```

Data Descriptions

Today, we'll only use the `chd` variables, plus `age`.

Variable	Description
`subj_id`	identifying code added by Dr. Love
`chd10_n`	(numeric) 1 = coronary heart disease in next 10 years
`chd10_f`	(factor) "chd" or "chd_no" in next ten years
`age`	in years (range is 32 to 70)
`cigs`	number of cigarettes smoked per day
`educ`	4-level factor: educational attainment
`hrate`	heart rate in beats per minute
`sbp`	systolic blood pressure in mm Hg
`stroke`	1 = history of stroke, else 0

Missing Data?

```
miss_var_summary(fram_new)
```

```
# A tibble: 9 x 3
  variable n_miss pct_miss
  <chr>     <int>    <dbl>
1 educ        105   2.48
2 cigs         29   0.684
3 hrate         1   0.0236
4 subj_id       0   0
5 chd10_n       0   0
6 chd10_f       0   0
7 age           0   0
8 sbp           0   0
9 stroke        0   0
```

Prepare our outcome.

We have our binary outcome as both a factor variable and a numeric (0/1) variable

```
fram_new %$% str(chd10_f)
```

```
 Factor w/ 2 levels "chd_no","chd": 1 1 1 2 1 1 2 1 1 1 ...
```

```
fram_new %$% str(chd10_n)
```

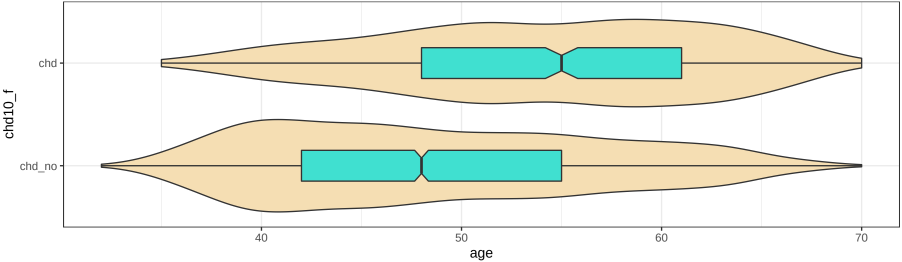
```
 int [1:4238] 0 0 0 1 0 0 1 0 0 0 ...
```

```
fram_new %>% tabyl(chd10_f, chd10_n)
```

```
 chd10_f    0   1
  chd_no 3594   0
     chd    0 644
```

Working with Binary Outcome Models

Does Pr(CHD in next ten years) look higher for *older* or *younger* people?



chd10_f	n	mean(age)	sd(age)	median(age)
chd_no	3594	48.77	8.41	48
chd	644	54.15	8.01	55

So what do we expect in this model?

Pr(CHD in next ten years) looks higher for *older* people?

If we predict log(odds(CHD in next ten years)), we want to ensure that value will be **rising** with increased age.

So, for the `mage_1` model below, what sign do we expect for the slope of age?

```
mage_1 <- glm(chd10_f ~ age, family = binomial,
              data = fram_new)
```

Results for `mage_1`

```
tidy(mage_1) %>% kable(digits = 3)
```

term	estimate	std.error	statistic	p.value
(Intercept)	-5.558	0.284	-19.585	0
age	0.075	0.005	14.166	0

```
tidy(mage_1, exponentiate = TRUE) %>% kable(digits = 3)
```

term	estimate	std.error	statistic	p.value
(Intercept)	0.004	0.284	-19.585	0
age	1.077	0.005	14.166	0

Six ways to specify the outcome for this model

```
x1 <- glm(chd10_f ~ age,
          family = binomial, data = fram_new)
x2 <- glm(chd10_n ~ age,
          family = binomial, data = fram_new)
x3 <- glm((chd10_n == "1") ~ age,
          family = binomial, data = fram_new)
x4 <- glm((chd10_n == "0") ~ age,
          family = binomial, data = fram_new)
x5 <- glm((chd10_f == "chd") ~ age,
          family = binomial, data = fram_new)
x6 <- glm((chd10_f == "chd_no") ~ age,
          family = binomial, data = fram_new)
```

What will happen to the age coefficient in these models?

Age Models `x1` and `x2`

```
x1 <- glm(chd10_f ~ age,
          family = binomial, data = fram_new)
extract_eq(x1, use_coefs = TRUE)
```

$$\log\left[\frac { \widehat{P(\operatorname{chd10\_f} = \operatorname{chd})} }{ 1 - \widehat{P(\operatorname{chd10\_f} = \operatorname{chd})} } \right] = -5.56 + 0.07(\operatorname{age}) \tag{2}$$

```
x2 <- glm(chd10_n ~ age,
          family = binomial, data = fram_new)
extract_eq(x2, use_coefs = TRUE)
```

$$\log\left[\frac { \widehat{P(\operatorname{chd10\_n} = 1)} }{ 1 - \widehat{P(\operatorname{chd10\_n} = 1)} } \right] = -5.56 + 0.07(\operatorname{age}) \tag{3}$$

Age Models x3 and x4

```
x3 <- glm((chd10_n == "1") ~ age,
          family = binomial, data = fram_new)
extract_eq(x3, use_coefs = TRUE)
```

$$\log\left[\frac{\widehat{P(\text{chd10\_n} = 1)}}{1 - \widehat{P(\text{chd10\_n} = 1)}}\right] = -5.56 + 0.07(\text{age}) \tag{4}$$

```
x4 <- glm((chd10_n == "0") ~ age,
          family = binomial, data = fram_new)
extract_eq(x4, use_coefs = TRUE)
```

$$\log\left[\frac{\widehat{P(\text{chd10\_n} = 0)}}{1 - \widehat{P(\text{chd10\_n} = 0)}}\right] = 5.56 - 0.07(\text{age}) \tag{5}$$

Age Models x5 and x6

```
x5 <- glm((chd10_f == "chd") ~ age,
          family = binomial, data = fram_new)
extract_eq(x5, use_coefs = TRUE)
```

$$\log\left[\frac { \widehat{P(\operatorname{chd10\_f} = \operatorname{chd})} }{ 1 - \widehat{P(\operatorname{chd10\_f} = \operatorname{chd})} } \right] = -5.56 + 0.07(\operatorname{age}) \tag{6}$$

```
x6 <- glm((chd10_f == "chd_no") ~ age,
          family = binomial, data = fram_new)
extract_eq(x6, use_coefs = TRUE)
```

$$\log\left[\frac { \widehat{P(\operatorname{chd10\_f} = \operatorname{chd\_no})} }{ 1 - \widehat{P(\operatorname{chd10\_f} = \operatorname{chd\_no})} } \right] = 5.56 - 0.07(\operatorname{age}) \tag{7}$$

Making Predictions with a `glm` model

```
modelL1 <- glm(chd10_f == "chd" ~ age,
               family = binomial, data = fram_new)

new_folks <- tibble(name = c("Frank", "Grace"),
                    age = c(42, 56))

new_folks %>% kable()
```

name	age
Frank	42
Grace	56

Predictions from a glm model (modelL1)

predictions on the logit scale

```
predict(modelL1, newdata = data.frame(new_folks))
```

```
        1         2 
-2.424935 -1.380563 
```

or on the probability scale (reminder: glm fit)

```
predict(modelL1, newdata = data.frame(new_folks),
        type = "response")
```

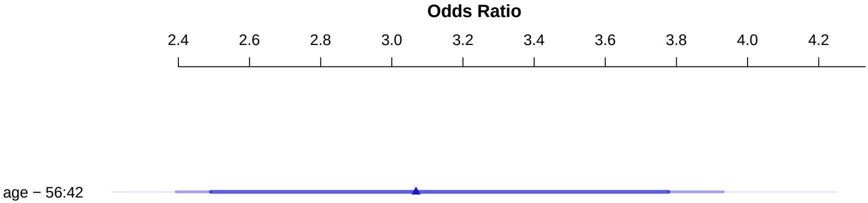
```
        1         2 
0.0812909 0.2009186 
```

Building a different model with lrm

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

modelL2 <- lrm(chd10_f == "chd" ~ rcs(age, 4),
               data = fram_new, x = TRUE, y = TRUE)
```

Plot Effect Sizes from `modelL2`

```
plot(summary(modelL2))
```



Making Predictions with `lrm` (`modelL2`)

```
new_folks %>% kable()
```

name	age
Frank	42
Grace	56

- Predictions on the logit scale

```
predict(modelL2, newdata = data.frame(new_folks))
```

```
        1         2
-2.465157 -1.344158
```

Useful Predictions with `lrm` (`modelL2`)

```
new_folks %>% kable()
```

name	age
Frank	42
Grace	56

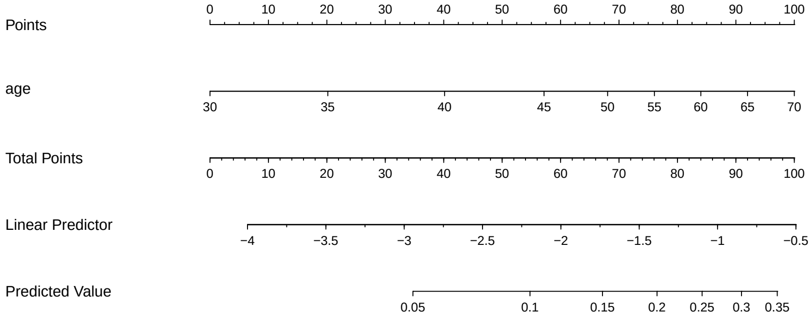
- Predicted probabilities after an `lrm` fit…

```
predict(modelL2, newdata = data.frame(new_folks),
        type = "fitted")
```

```
         1          2
0.07833722 0.20682714
```

Using the Nomogram to predict for Age 50

```
plot(nomogram(modelL2, fun = plogis))
```



Compare our results from the nomogram...

- Predicted probabilities after an `lrm` fit...

```
predict(modelL2, newdata = data.frame(age = 50),
        type = "fitted")
```

```
        1
0.1542483
```

Validate C statistic, Nagelkerke R^2, Brier score

```r
set.seed(2022)
validate(modelL2, B = 50)
```

```
> set.seed(2022)
> validate(modelL2, B = 50)
          index.orig training   test optimism index.corrected  n
Dxy           0.3581   0.3548 0.3581  -0.0033          0.3615 50
R2            0.0891   0.0887 0.0883   0.0004          0.0887 50
Intercept     0.0000   0.0000 0.0077  -0.0077          0.0077 50
Slope         1.0000   1.0000 1.0057  -0.0057          1.0057 50
Emax          0.0000   0.0000 0.0026   0.0026          0.0026 50
D             0.0522   0.0520 0.0517   0.0003          0.0519 50
U            -0.0005  -0.0005 0.0000  -0.0005          0.0000 50
Q             0.0527   0.0525 0.0517   0.0008          0.0519 50
B             0.1223   0.1225 0.1224   0.0001          0.1222 50
g             0.8169   0.8116 0.8124  -0.0008          0.8176 50
gp            0.0925   0.0918 0.0917   0.0001          0.0924 50
```

Next Time

- Logistic Regression using `tidymodels`
- Quiz 1 will be made available today at 5 PM. Good luck!