431 Class 27

Thomas E. Love

2017-12-07

R Setup for Today

```
library(car); library(broom); library(magrittr)
library(tidyverse)

dm192 <- read_csv("data/dm192.csv")
```

Today's Agenda

- The `dm192` data
  - We have 7 regression inputs. How well can we predict today's systolic BP?
- Setting up Quiz 3
- So what have we learned?

Regression and the `dm192` data

Our Research Question

Can we predict a patient's `sbp` level today, if the seven features we can use to predict that are:

- their `sbp` level one year ago
- their `a1c` level now
- their `age`, `race`, `sex` and `insurance` type
- and the `practice` where they are seen

We want to use some or all of these seven regression inputs to do the best possible job of predicting today's `sbp`, regardless of which predictors fall in or out of the model.

For 431, we're working with complete cases only

```
dm192_work <- dm192 %>%
  select(pt.id, sbp, sbp_old, a1c, age, race,
         sex, insurance, practice) %>%
  filter(complete.cases(.))

head(dm192_work,3)
```

```
# A tibble: 3 x 9
  pt.id   sbp sbp_old   a1c   age  race    sex
  <int> <int>   <int> <dbl> <int> <chr>  <chr>
1     1   108     110   5.8    44 black   male
2     2   162     158  11.6    28 black female
3     4   133     145  12.7    56 black   male
# ... with 2 more variables: insurance <chr>,
#   practice <chr>
```

Change several character variables to factors

```
cols_temp <- c("race", "sex", "insurance", "practice")

dm192_work[cols_temp] <- lapply(dm192_work[cols_temp], factor)

head(dm192_work,3)
```

```
# A tibble: 3 x 9
  pt.id   sbp sbp_old   a1c   age   race    sex
  <int> <int>   <int> <dbl> <int> <fctr> <fctr>
1     1   108     110   5.8    44  black   male
2     2   162     158  11.6    28  black female
3     4   133     145  12.7    56  black   male
# ... with 2 more variables: insurance <fctr>,
#   practice <fctr>
```

Are the factor levels sensible and sensibly ordered? (1)

```
dm192_work %>% count(race)
```

```
# A tibble: 4 x 2
    race     n
  <fctr> <int>
1  asian     5
2  black   119
3  other    16
4  white    48
```

Auto-collapse to most common 2 levels, plus "Others"

```r
dm192_work$race <- dm192_work$race %>%
  fct_lump(n = 2, other_level = "Others")

table(dm192_work$race)
```

```
 black  white Others 
   119     48     21 
```

Are the factor levels sensible and sensibly ordered? (2)

```
dm192_work %>% count(sex)
```

```
# A tibble: 2 x 2
     sex     n
  <fctr> <int>
1 female    96
2   male    92
```

Are the factor levels sensible and sensibly ordered? (3)

```
dm192_work %>% count(insurance)
```

```
# A tibble: 4 x 2
   insurance     n
      <fctr> <int>
1 commercial    39
2   medicaid    67
3   medicare    76
4  uninsured     6
```

Collapse Medicaid and Uninsured together

```
dm192_work$insurance <-
  fct_collapse(dm192_work$insurance,
               Medicare = "medicare",
               Commercial = "commercial",
               Medicaid_Unins = c("medicaid", "uninsured"))

table(dm192_work$insurance)
```

```
    Commercial Medicaid_Unins       Medicare 
            39             73             76 
```

Reorder Factor Levels by Hand

```
dm192_work$insurance <-
  fct_relevel(dm192_work$insurance,
              "Medicare", "Commercial")

table(dm192_work$insurance)
```

```
     Medicare     Commercial Medicaid_Unins 
           76             39             73 
```

Are the factor levels sensible and sensibly ordered? (4)

```
dm192_work %>% count(practice)
```

```
# A tibble: 4 x 2
  practice     n
    <fctr> <int>
1        A    48
2        B    45
3        C    47
4        D    48
```

The tidyverse can do just about everything.

Except think.

Predict `sbp` as well as you can, in new data

Stage 1. Partition the Data

```
set.seed(43123)
dm192_train <-
  sample_frac(dm192_work, 0.8, replace = FALSE)
dm192_test <-
  anti_join(dm192_work, dm192_train)
```

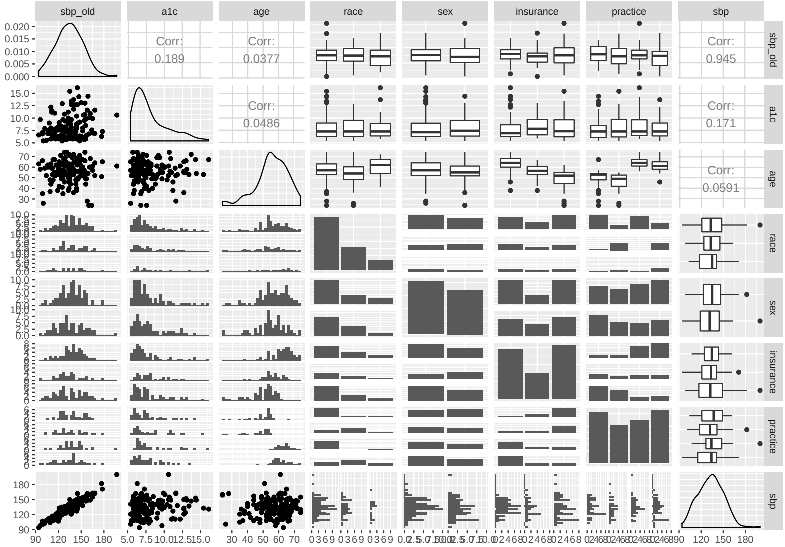
```
Joining, by = c("pt.id", "sbp", "sbp_old", "a1c", "age", "race
```

```
dim(dm192_train); dim(dm192_test)
```

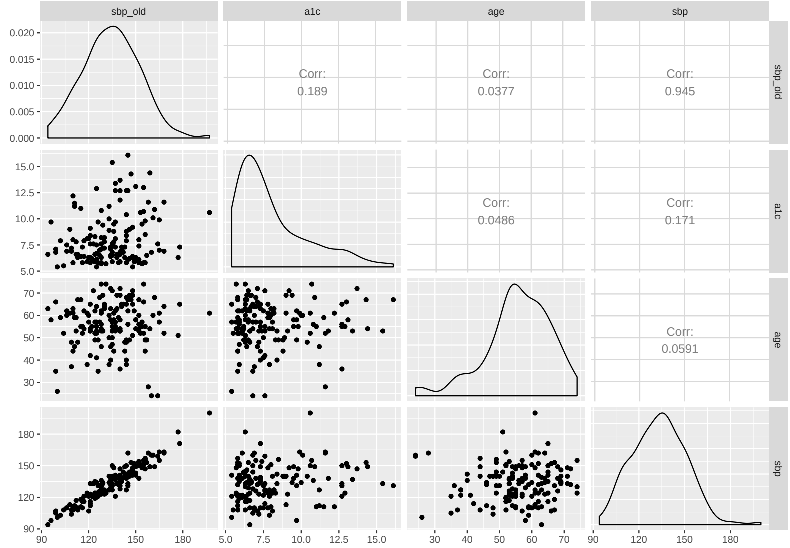
```
[1] 150   9
```

```
[1] 38  9
```

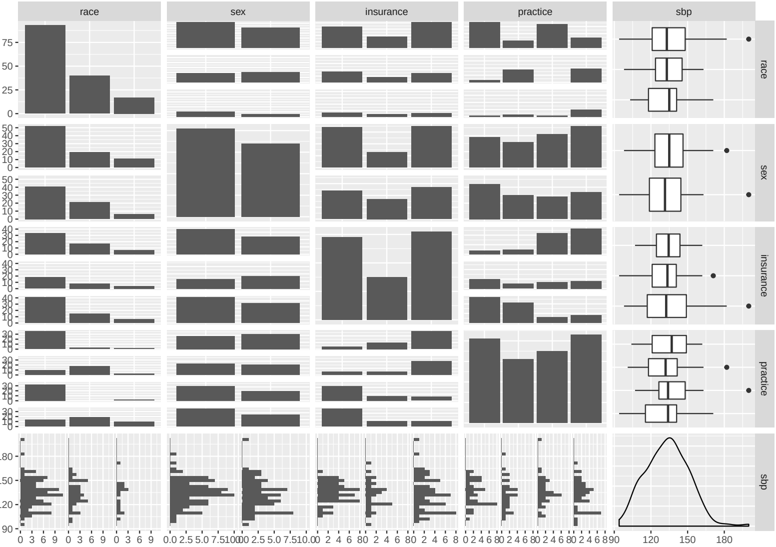
Stage 2. DTDP (everything in training set)



Stage 2. DTDP (quantitative predictors)



Stage 2. DTDP (categorical predictors)



Stage 3. Exploratory Data Analysis

```
mosaic::favstats(dm192_train$sbp)
```

```
 min      Q1 median     Q3 max    mean       sd   n
  94 121.25    133 145.5 200 133.42 17.48605 150
 missing
       0
```

```
mosaic::favstats(dm192_train$sbp ~ dm192_train$sex)
```

```
  dm192_train$sex min     Q1 median     Q3 max
1          female  98 123.25  135.0 146.25 182
2            male  94 118.75  131.5 144.50 200
      mean       sd  n missing
1 134.6463 16.19966 82       0
2 131.9412 18.93814 68       0
```

Usually, I stop myself from doing this.

Time to fit a Kitchen Sink Model

Stage 4. Fit Kitchen Sink Model in Training Sample

```
mod_ks1 <- lm(sbp ~ sbp_old + a1c + age + race +
                  sex + insurance + practice,
              data = dm192_train)

round(glance(mod_ks1),3)
```

```
  r.squared adj.r.squared sigma statistic p.value df
1     0.898          0.89 5.791   110.954       0 12
    logLik     AIC      BIC deviance df.residual
1 -470.034 966.067 1005.206  4627.97         138
```

arm::display(mod_ks1) (n = 150, r-sq = 0.90)

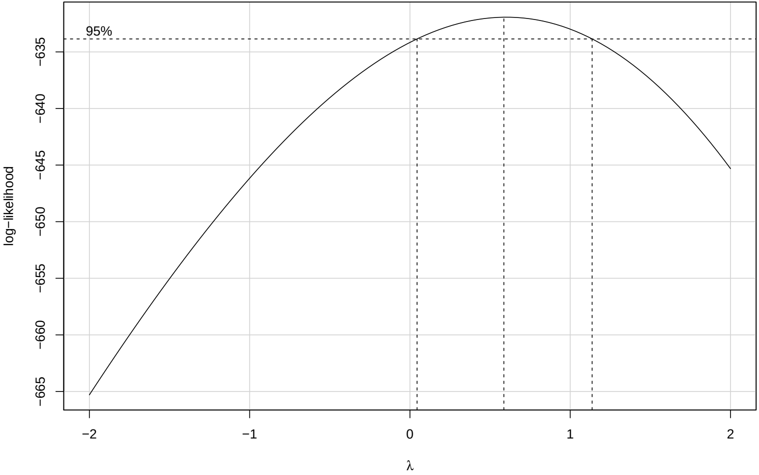
```
lm(formula = sbp ~ sbp_old + a1c + age + race + sex + insuranc
    practice, data = dm192_train)
                        coef.est coef.se
(Intercept)              6.70     5.70
sbp_old                  0.93     0.03
a1c                     -0.09     0.21
age                      0.02     0.08
racewhite               -1.16     1.39
raceOthers              -1.16     1.66
sexmale                 -0.67     0.97
insuranceCommercial      1.59     1.40
insuranceMedicaid_Unins  1.85     1.38
practiceB                1.29     1.59
practiceC                2.27     1.73
practiceD                2.91     1.75
---
n = 150, k = 12
```

Stage 5. Consider collinearity, residual plots, potential transformations of the outcome

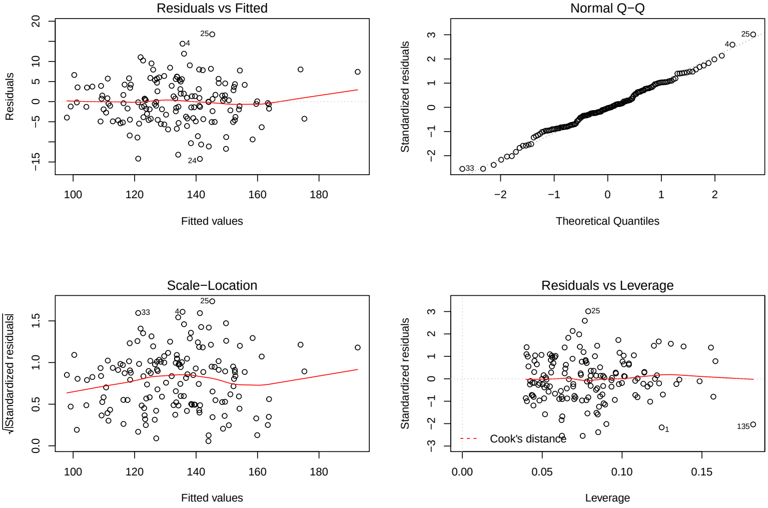
```
vif(mod_ks1)
```

```
              GVIF Df GVIF^(1/(2*Df))
sbp_old   1.082669  1        1.040514
a1c       1.075402  1        1.037016
age       2.645307  1        1.626440
race      1.745582  2        1.149437
sex       1.052235  1        1.025785
insurance 1.757957  2        1.151468
practice  4.032427  3        1.261618
```

boxCox(mod_ks1) ($\lambda = 0.6$, round to 1)



plot(mod_ks1)



Residuals vs Fitted
Residuals
Fitted values
25
4
24
Normal Q−Q
Standardized residuals
Theoretical Quantiles
25
4
33
Scale−Location
√|Standardized residuals|
Fitted values
25
33
4
Residuals vs Leverage
Standardized residuals
Leverage
Cook's distance
25
1
135

Stage 6. Consider stepwise regression to prune the model

```r
step(mod_ks1)
```

```
Start:  AIC=538.39
sbp ~ sbp_old + a1c + age + race + sex + insurance + practice

            Df Sum of Sq   RSS    AIC
- race       2        30  4658 535.35
- practice   3        98  4726 535.53
- age        1         1  4629 536.43
- a1c        1         7  4635 536.61
- insurance  2        69  4697 536.61
- sex        1        16  4644 536.89
<none>                    4628 538.39
- sbp_old    1     38268 42896 870.39
```

Suggested model from `step` is

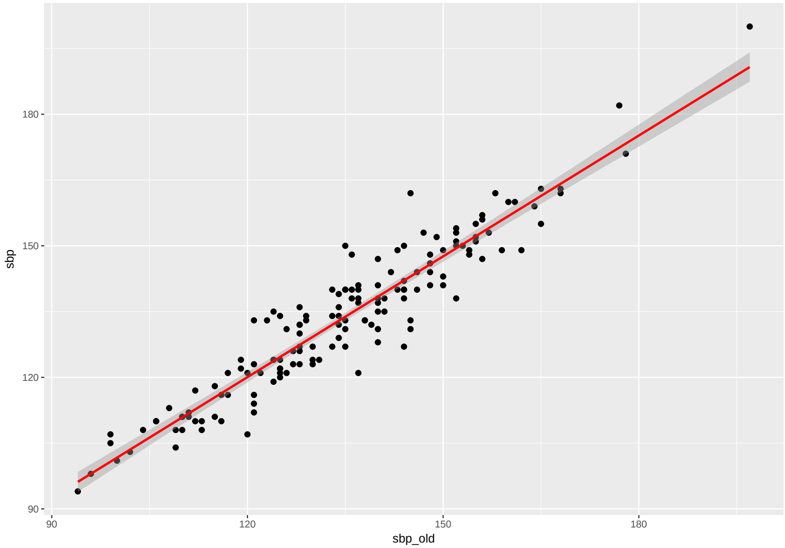
```
Step:  AIC=524.98
sbp ~ sbp_old

          Df Sum of Sq   RSS    AIC
<none>                  4836 524.98
- sbp_old  1     40722 45559 859.42

Call:
lm(formula = sbp ~ sbp_old, data = dm192_train)

Coefficients:
(Intercept)      sbp_old
     9.8485       0.9183
```

So that's just . . .



Stage 7. Compare potential models in-sample

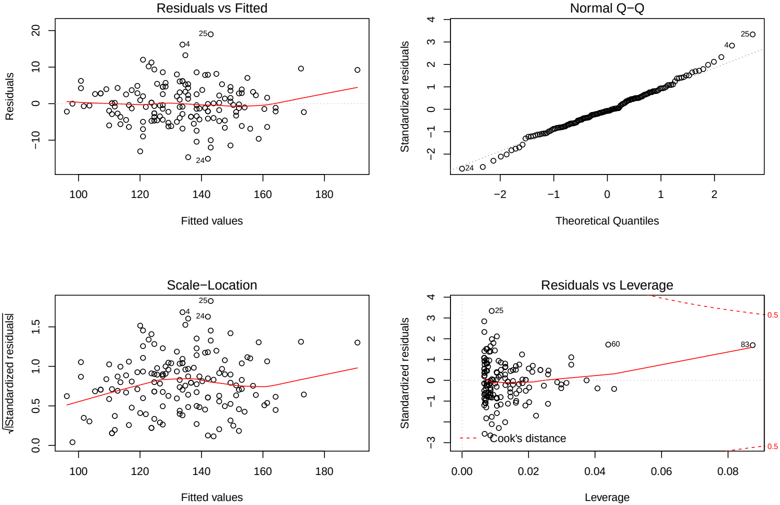
```
mod_simple <- lm(sbp ~ sbp_old, data = dm192_train)
glance(mod_simple) %>% select(r.squared, adj.r.squared, AIC, B
```

```
  r.squared adj.r.squared      AIC     BIC
1 0.8938499     0.8931326 952.6641 961.696
```

```
glance(mod_ks1) %>% select(r.squared, adj.r.squared, AIC, BIC)
```

```
  r.squared adj.r.squared      AIC      BIC
1 0.8984171     0.8903199 966.0673 1005.206
```

Residual Plots for Simple One-Predictor Model



Stage 8. Compare potential models on test data

```
pred_ks <- predict(mod_ks1, newdata = dm192_test)
err_ks <- dm192_test$sbp - pred_ks
round(summary(abs(err_ks)),3)
```

```
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
  0.578   2.876   4.287   5.130   6.848  14.480
```

```
round(summary(err_ks^2),3)
```

```
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
  0.334   8.271  18.396  37.102  46.906 209.672
```

```
round(cor(pred_ks, dm192_test$sbp)^2,4)
```

```
[1] 0.9163
```

Simple Model

```
pred_simple <- predict(mod_simple, newdata = dm192_test)
err_simple <- dm192_test$sbp - pred_simple
round(summary(abs(err_simple)),3)
```

```
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
  0.151   2.334   4.298   4.994   6.298  14.278
```

```
round(summary(err_simple^2),3)
```

```
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
  0.023   5.468  18.478  37.965  39.677 203.851
```

```
round(cor(pred_simple, dm192_test$sbp)^2,4)
```

```
[1] 0.9149
```

MAPE and MSPE results

Model	MAPE	MSPE	Max Abs. Error	Out of Sample R^2
Kitchen Sink	5.13	37.1	14.48	0.9163
Simple	4.99	38	14.28	0.9149

Remember that the training sample here has only 38 observations.

Stage 9. Re-combine sample and fit final model

```
model_all <- lm(sbp ~ sbp_old, data = dm192_work)

glance(model_all)
```

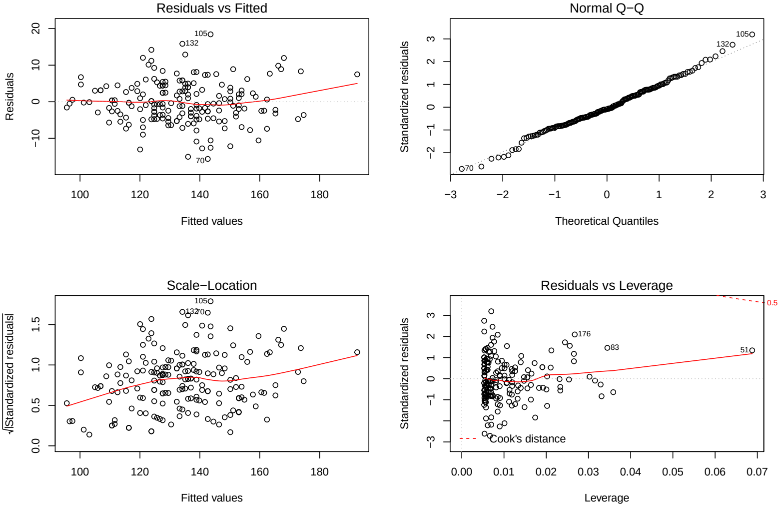
```
  r.squared adj.r.squared    sigma statistic
1 0.8962414     0.8956835 5.785442  1606.622
       p.value df    logLik     AIC      BIC deviance
1 1.907078e-93  2 -595.7599 1197.52 1207.229 6225.669
  df.residual
1         186
```

Tidied `model_all` Coefficients

```
tidy(model_all)
```

```
         term  estimate  std.error statistic
1 (Intercept) 7.1213074 3.19565828  2.228432
2     sbp_old 0.9410682 0.02347817 40.082692
       p.value
1 2.704951e-02
2 1.907078e-93
```

Residual Plot for `model_all`



So, what have we learned?

The Signal and The Noise

- Nature's laws do not change very much.
- There is no reason to conclude that the affairs of men are becoming more predictable. The opposite may well be true.

Thinking Probabilistically, and using the Bayesian way of thinking about prediction

- Don't fall into the comforting trap of binary thinking. Expressions of uncertainty are not admissions of weakness.
- Know Where You're Coming From - state explicitly how likely we believe an event is to occur *before* we begin to weigh the evidence.
- The volume of information is increasing exponentially. But the signal-to-noise ratio may be waning. We need better ways of distinguishing the two.

Our bias is to think that we are better at prediction than we really are.

The Course So Far

1. Statistics is too important to be left to statisticians.
2. Models and visualization are the big takeaways.
3. Reproducible research is the current wave.
4. Things are changing quickly. We live in interesting times.

That's all, folks!