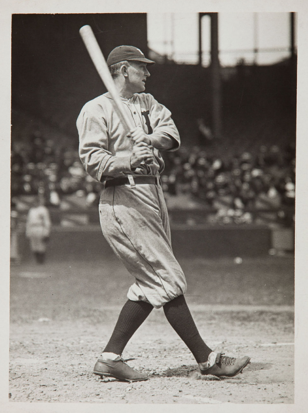
THE UNIVERSITY OF TEXAS AT AUSTIN
McCOMBS SCHOOL OF BUSINESS

Model Building 2

Lecture 15

STA 371G

Let's model the batting averages of baseball players



- All of the data here came from `http://seanlahman.com/baseball-archive/statistics/`
- Some data cleaning was done, mostly to calculate averages
- We are going to explore this dataset with best subsets regression

The response variable

- **AVG**: Batting average

The potential predictors

- **YEAR**: Year this entry calculated for
- **LG**: League, either NL or AL
- **OBP**: On base percentage
- **SLG**: Slugging average
- **EXP**: Years of experience
- **PAYR**: Plate appearances per year
- **MLAVG**: Batting average for the leauge for the year
- **MLOBP**: On base percentage for the leaugue for the year
- **MLSLG**: Slugging percentage for the leaugue for the year
- **AVGcumLag1**: Player's cumulative batting average for previous years
- **OBPcumLag1**: Player's cumulative on base percentage for previous years
- **SLGcumLag1**: Player's cumulative slugging percentage for previous years
- **G**: Games played (must have been at least 98)
- **YRINDEX**: Number of years since 1958

Build model full and check for multicollinearity

```r
full <- lm(AVG ~ OBP + SLG + EXP + PAYR + MLAVG
                 + MLOBP + MLSLG + AVGcumLag1 + OBPcumLag1
                 + SLGcumLag1 + G + YRINDEX, data=baseball)

vif(full)
```

```
       OBP        SLG        EXP       PAYR      MLAVG      MLOBP 
      3.71       4.32       1.20       1.37      11.07      12.69 
     MLSLG AVGcumLag1 OBPcumLag1 SLGcumLag1          G    YRINDEX 
      7.39       2.09       3.95       3.82       1.12       2.18 
```

Build model full and check for multicollinearity

```
full <- lm(AVG ~ OBP + SLG + EXP + PAYR + MLAVG
                  + MLOBP + MLSLG + AVGcumLag1 + OBPcumLag1
                  + SLGcumLag1 + G + YRINDEX, data=baseball)

vif(full)

       OBP        SLG        EXP       PAYR      MLAVG      MLOBP
      3.71       4.32       1.20       1.37      11.07      12.69
     MLSLG AVGcumLag1 OBPcumLag1 SLGcumLag1          G    YRINDEX
      7.39       2.09       3.95       3.82       1.12       2.18
```

Uh oh. Houston, we have a problem!

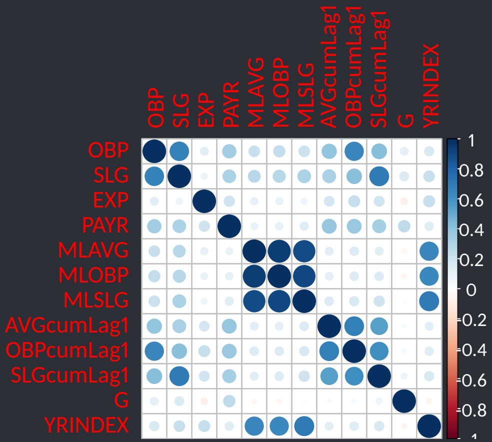
Look at the correlations to find the problem

Columns 8-19 in the data set are numeric. Let's pull those out and look at the correlation matrix.

```
numeric.predictors <- baseball[,8:19]
cor(numeric.predictors)
```

A correlation plot is easier to read!

```
library(corrplot)
corrplot(cor(numeric.predictors))
```



Reduce multicollinearity by dropping variables

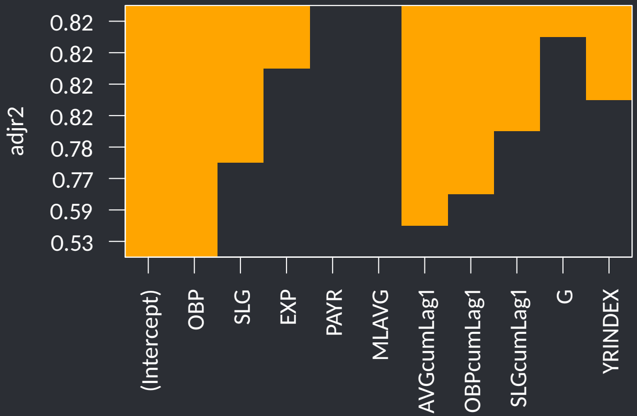
The Major League averages are highly correlated with each other; let's keep just MLAVG and drop MLOBP and MLSLG. (This choice depends on our preference of which variable would make the most sense to keep.)

```
full <- lm(AVG ~ OBP + SLG + EXP + PAYR + MLAVG
                  + AVGcumLag1 + OBPcumLag1
                  + SLGcumLag1 + G + YRINDEX, data=baseball)

vif(full)

       OBP        SLG        EXP       PAYR      MLAVG AVGcumLag1
      3.62       4.29       1.16       1.37       1.86       2.09
OBPcumLag1 SLGcumLag1          G    YRINDEX
      3.92       3.79       1.12       1.85
```

Much better!

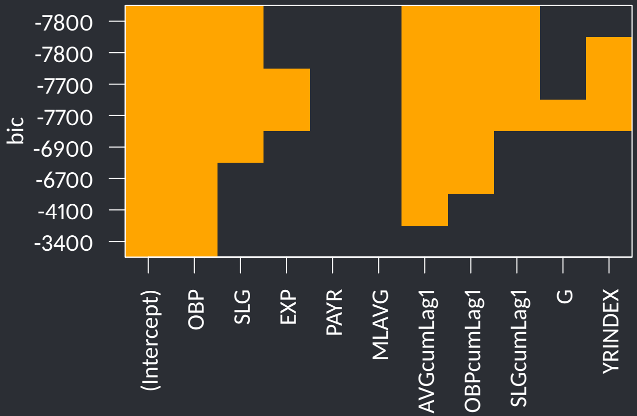
Use best-subsets regression to get a sense of the best predictors

```
bestsubsets <- regsubsets(AVG ~ OBP + SLG + EXP + PAYR + MLAVG
  + AVGcumLag1 + OBPcumLag1 + SLGcumLag1 + G + YRINDEX, data=baseball)
plot(bestsubsets, scale="adjr2")
```



Use best-subsets regression to get a sense of the best predictors

```
plot(bestsubsets, scale="bic")
```



Generate the best candidate model

```
Call:
lm(formula = AVG ~ OBP + SLG + AVGcumLag1 + OBPcumLag1 + SLGcumLag1,
    data = baseball)

Residuals:
     Min       1Q   Median       3Q      Max 
-0.05601 -0.00772  0.00026  0.00818  0.04051 

Coefficients:
            Estimate Std. Error t value Pr(>|t|)    
(Intercept)  0.02787    0.00250    11.2   <2e-16 ***
OBP          0.49821    0.00909    54.8   <2e-16 ***
SLG          0.16083    0.00470    34.2   <2e-16 ***
AVGcumLag1   0.88035    0.01195    73.7   <2e-16 ***
OBPcumLag1  -0.47626    0.01211   -39.3   <2e-16 ***
SLGcumLag1  -0.17183    0.00555   -31.0   <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.0121 on 4529 degrees of freedom
Multiple R-squared:  0.821,Adjusted R-squared:  0.821 
F-statistic: 4.15e+03 on 5 and 4529 DF,  p-value: <2e-16
```

Does the National League's Designated Hitter Rule Matter?

Let's first look at only the cases where LG is either NL or AL, to simplify the analysis (other rows correspond to a player that switched teams during the season). Then we'll add LG to the model.

```
base1 <- baseball[baseball$LG == "NL" |
                  baseball$LG == "AL",]
modelLG <- lm(AVG ~ OBP + SLG + AVGcumLag1 + OBPcumLag1
                     + SLGcumLag1 + LG, data=base1)
```

```
summary(modelLG)



Call:
lm(formula = AVG ~ OBP + SLG + AVGcumLag1 + OBPcumLag1 + SLGcumLag1 +
    LG, data = base1)

Residuals:
     Min       1Q   Median       3Q      Max
-0.05583 -0.00782  0.00026  0.00822  0.04022

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept)  0.028177   0.002559   11.01   <2e-16 ***
OBP          0.499326   0.009356   53.37   <2e-16 ***
SLG          0.159058   0.004830   32.93   <2e-16 ***
AVGcumLag1   0.877759   0.012311   71.30   <2e-16 ***
OBPcumLag1  -0.476465   0.012464  -38.23   <2e-16 ***
SLGcumLag1  -0.170083   0.005708  -29.80   <2e-16 ***
LGNL         0.000303   0.000372    0.81     0.42
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.0122 on 4306 degrees of freedom
Multiple R-squared:  0.821,Adjusted R-squared:  0.821
F-statistic: 3.29e+03 on 6 and 4306 DF,  p-value: <2e-16
```

Is this model really useful?

- Automated regression model selection methods cannot make something out of nothing.

Is this model really useful?

- Automated regression model selection methods cannot make something out of nothing.
- If you omit some important variables or fail to use data transformations when they are needed, or if the assumption of linear or linearizable relationships is simply wrong, the model is a bad one, no matter what the R^2.

Is this model really useful?

- Automated regression model selection methods cannot make something out of nothing.
- If you omit some important variables or fail to use data transformations when they are needed, or if the assumption of linear or linearizable relationships is simply wrong, the model is a bad one, no matter what the R^2.
- Use your own judgment and intuition about your data to try to fine-tune whatever the computer comes up with.

Surprise!

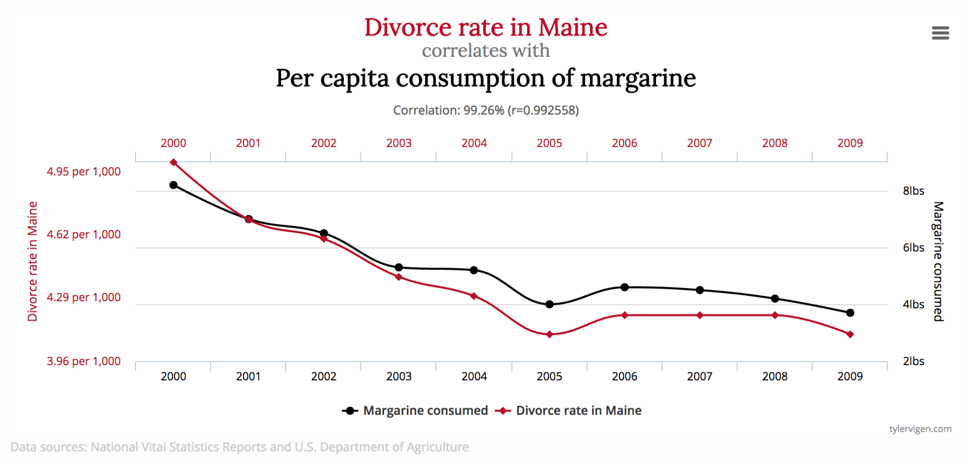
This data is all random numbers! Here's how it was generated:

```
y <- rnorm(100)
x1 <- rnorm(100)
x2 <- rnorm(100)
# etc
```

$R^2 = 0.21$, so 21% of the variance in Y is explained by random numbers!

Be careful of spurious correlations and overfitting!

- If you have more than 1 predictor for 10-15 cases, you are likely to see spurious correlations.
- If you fit models with meaningless variables, you are fitting noise and will end up with an overfit model that is not predictive on new data.


Divorce rate in Maine
correlates with
Per capita consumption of margarine
Correlation: 99.26% (r=0.992558)
2000
2001
2002
2003
2004
2005
2006
2007
2008
2009
4.95 per 1,000
4.62 per 1,000
4.29 per 1,000
3.96 per 1,000
Divorce rate in Maine
8lbs
6lbs
4lbs
2lbs
Margarine consumed
Margarine consumed
Divorce rate in Maine
tylervigen.com
Data sources: National Vital Statistics Reports and U.S. Department of Agriculture