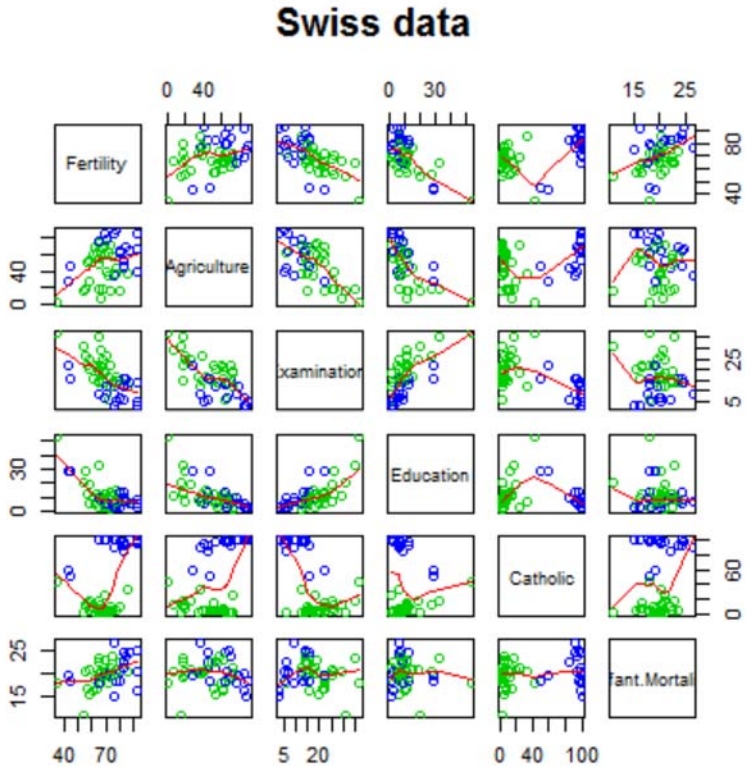
Multivariable regression examples

Regression Models

Brian Caffo, Jeff Leek and Roger Peng
Johns Hopkins Bloomberg School of Public Health

Swiss fertility data

```
library(datasets); data(swiss); require(stats); require(graphics)
pairs(swiss, panel = panel.smooth, main = "Swiss data", col = 3 + (swiss$Catholic > 50))
```



?swiss

Description

Standardized fertility measure and socio-economic indicators for each of 47 French-speaking provinces of Switzerland at about 1888.

A data frame with 47 observations on 6 variables, each of which is in percent, i.e., in [0, 100].

- [,1] Fertility Ig, ‘ common standardized fertility measure’
- [,2] Agriculture % of males involved in agriculture as occupation
- [,3] Examination % draftees receiving highest mark on army examination
- [,4] Education % education beyond primary school for draftees.
- [,5] Catholic % ‘catholic’ (as opposed to ‘protestant’).
- [,6] Infant.Mortality live births who live less than 1 year.

All variables but ‘Fertility’ give proportions of the population.

Calling `lm`

```
summary(lm(Fertility ~ . , data = swiss))
```

```
                 Estimate Std. Error t value  Pr(>|t|)
(Intercept)       66.9152   10.70604   6.250 1.906e-07
Agriculture       -0.1721    0.07030  -2.448 1.873e-02
Examination       -0.2580    0.25388  -1.016 3.155e-01
Education         -0.8709    0.18303  -4.758 2.431e-05
Catholic           0.1041    0.03526   2.953 5.190e-03
Infant.Mortality   1.0770    0.38172   2.822 7.336e-03
```

Example interpretation

- Agriculture is expressed in percentages (0 - 100)
- Estimate is -0.1721.
- We estimate an expected 0.17 decrease in standardized fertility for every 1\% increase in percentage of males involved in agriculture in holding the remaining variables constant.
- The t-test for $H_0 : \beta_{Agri} = 0$ versus $H_a : \beta_{Agri} \neq 0$ is significant.
- Interestingly, the unadjusted estimate is

```
summary(lm(Fertility ~ Agriculture, data = swiss))$coefficients
```

```
             Estimate Std. Error t value  Pr(>|t|)
(Intercept)  60.3044    4.25126  14.185 3.216e-18
Agriculture   0.1942    0.07671   2.532 1.492e-02
```

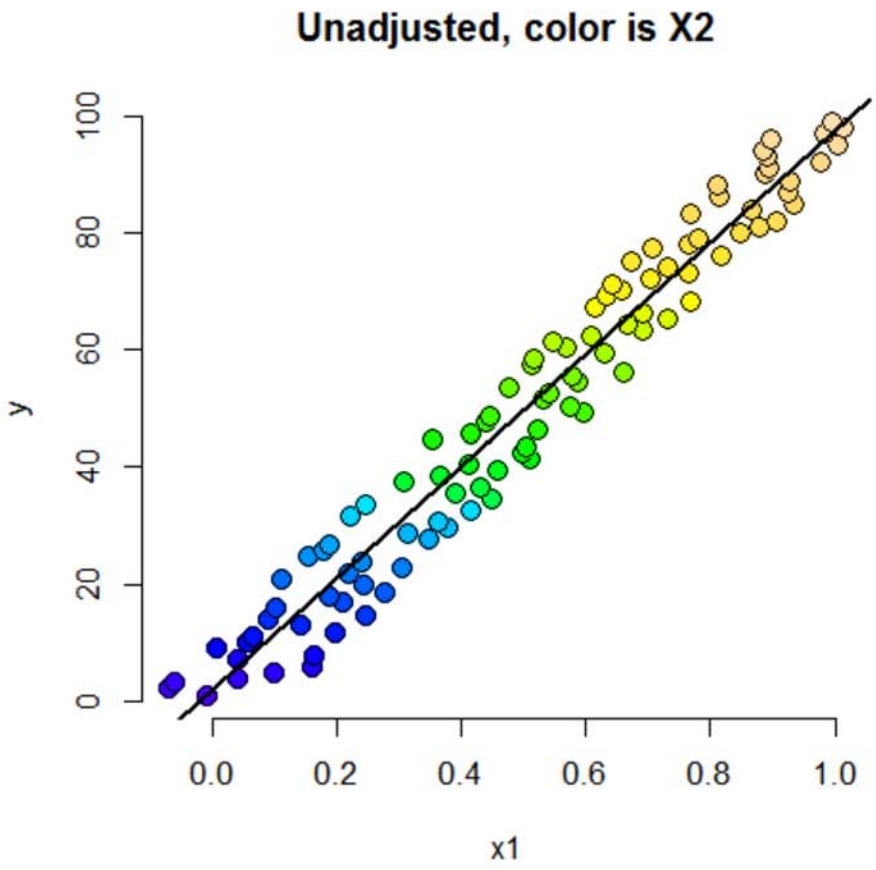
How can adjustment reverse the sign of an effect? Let's try a simulation.

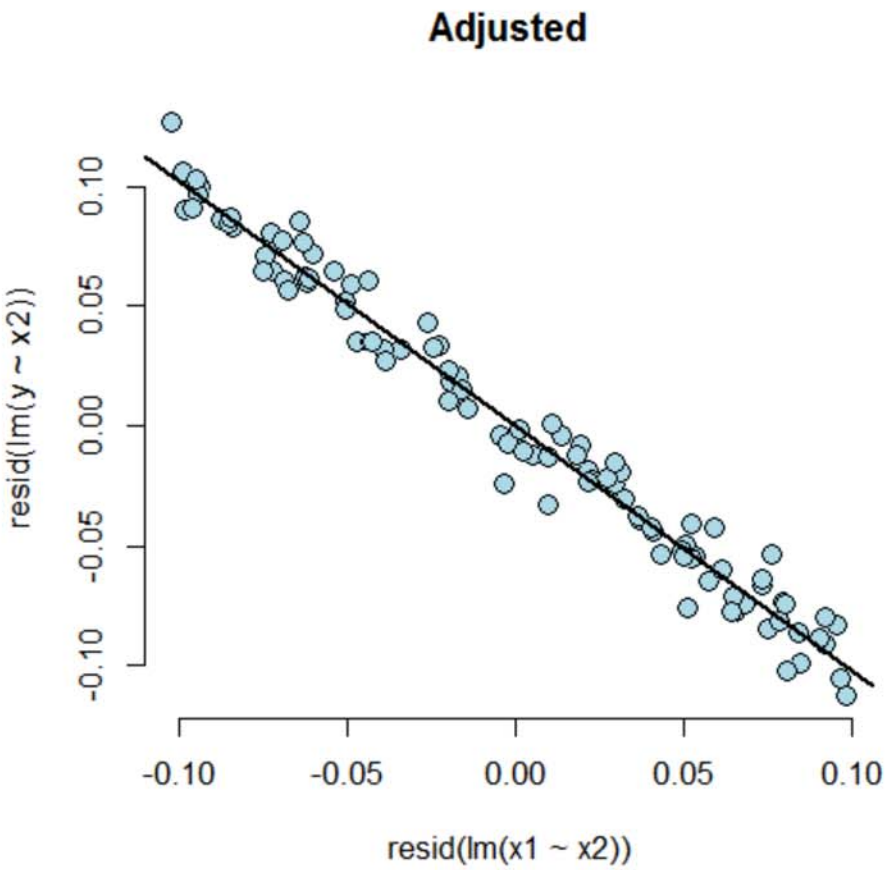
```
n <- 100; x2 <- 1 : n; x1 <- .01 * x2 + runif(n, -.1, .1); y = -x1 + x2 + rnorm(n, sd = .01)
summary(lm(y ~ x1))$coef
```

```
            Estimate Std. Error t value  Pr(>|t|)
(Intercept)    1.618      1.200   1.349 1.806e-01
x1            95.854      2.058  46.579 1.153e-68
```

```
summary(lm(y ~ x1 + x2))$coef
```

```
              Estimate Std. Error    t value   Pr(>|t|)
(Intercept)  0.0003683  0.0020141     0.1829  8.553e-01
x1          -1.0215256  0.0166372   -61.4001  1.922e-79
x2           1.0001909  0.0001681 5950.1818 1.369e-271
```


Unadjusted, color is X2
y
x1
0
20
40
60
80
100
0.0
0.2
0.4
0.6
0.8
1.0


Adjusted
resid(lm(y ~ x2))
resid(lm(x1 ~ x2))
-0.10
-0.05
0.00
0.05
0.10

Back to this data set

- The sign reverses itself with the inclusion of Examination and Education, but of which are negatively correlated with Agriculture.
- The percent of males in the province working in agriculture is negatively related to educational attainment (correlation of -0.6395) and Education and Examination (correlation of 0.6984) are obviously measuring similar things.
  - Is the positive marginal an artifact for not having accounted for, say, Education level? (Education does have a stronger effect, by the way.)
- At the minimum, anyone claiming that provinces that are more agricultural have higher fertility rates would immediately be open to criticism.

What if we include an unnecessary variable?

z adds no new linear information, since it's a linear combination of variables already included. R just drops terms that are linear combinations of other terms.

```
z <- swiss$Agriculture + swiss$Education
lm(Fertility ~ . + z, data = swiss)
```

```

Call:
lm(formula = Fertility ~ . + z, data = swiss)

Coefficients:
     (Intercept)       Agriculture       Examination         Education          Catholic
          66.915            -0.172            -0.258            -0.871             0.104
Infant.Mortality                 z
           1.077                NA
```

Dummy variables are smart

- Consider the linear model

$$Y_i = \beta_0 + X_{i1}\beta_1 + \epsilon_i$$

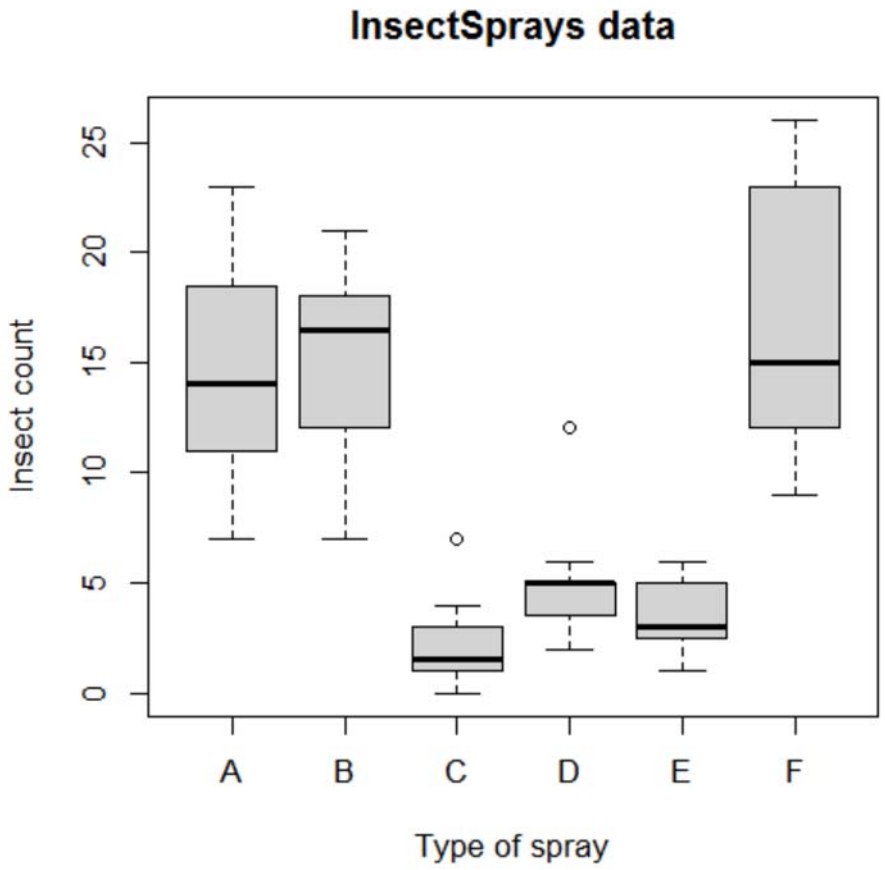
  where each X_{i1} is binary so that it is a 1 if measurement i is in a group and 0 otherwise. (Treated versus not in a clinical trial, for example.)

- Then for people in the group $E[Y_i] = \beta_0 + \beta_1$
- And for people not in the group $E[Y_i] = \beta_0$
- The LS fits work out to be $\hat{\beta}_0 + \hat{\beta}_1$ is the mean for those in the group and $\hat{\beta}_0$ is the mean for those not in the group.
- β_1 is interpretted as the increase or decrease in the mean comparing those in the group to those not.
- Note including a binary variable that is 1 for those not in the group would be redundant. It would create three parameters to describe two means.

More than 2 levels

- Consider a multilevel factor level. For didactic reasons, let's say a three level factor (example, US political party affiliation: Republican, Democrat, Independent)
- $Y_i = \beta_0 + X_{i1}\beta_1 + X_{i2}\beta_2 + \epsilon_i$.
- X_{i1} is 1 for Republicans and 0 otherwise.
- X_{i2} is 1 for Democrats and 0 otherwise.
- If i is Republican $E[Y_i] = \beta_0 + \beta_1$
- If i is Democrat $E[Y_i] = \beta_0 + \beta_2$.
- If i is Independent $E[Y_i] = \beta_0$.
- β_1 compares Republicans to Independents.
- β_2 compares Democrats to Independents.
- $\beta_1 - \beta_2$ compares Republicans to Democrats.
- (Choice of reference category changes the interpretation.)

Insect Sprays



Linear model fit, group A is the reference

```
summary(lm(count ~ spray, data = InsectSprays))$coef
```

```
            Estimate Std. Error t value  Pr(>|t|)
(Intercept)  14.5000      1.132 12.8074 1.471e-19
sprayB        0.8333      1.601  0.5205 6.045e-01
sprayC      -12.4167      1.601 -7.7550 7.267e-11
sprayD       -9.5833      1.601 -5.9854 9.817e-08
sprayE      -11.0000      1.601 -6.8702 2.754e-09
sprayF        2.1667      1.601  1.3532 1.806e-01
```

Hard coding the dummy variables

```
summary(lm(count ~
             I(1 * (spray == 'B')) + I(1 * (spray == 'C')) +
             I(1 * (spray == 'D')) + I(1 * (spray == 'E')) +
             I(1 * (spray == 'F'))
           , data = InsectSprays))$coef
```

```
                    Estimate Std. Error t value  Pr(>|t|)
(Intercept)          14.5000      1.132 12.8074 1.471e-19
I(1 * (spray == "B"))  0.8333      1.601  0.5205 6.045e-01
I(1 * (spray == "C")) -12.4167      1.601 -7.7550 7.267e-11
I(1 * (spray == "D"))  -9.5833      1.601 -5.9854 9.817e-08
I(1 * (spray == "E")) -11.0000      1.601 -6.8702 2.754e-09
I(1 * (spray == "F"))  2.1667      1.601  1.3532 1.806e-01
```

What if we include all 6?

```
lm(count ~
   I(1 * (spray == 'B')) + I(1 * (spray == 'C')) +
   I(1 * (spray == 'D')) + I(1 * (spray == 'E')) +
   I(1 * (spray == 'F')) + I(1 * (spray == 'A')), data = InsectSprays)
```

```
Call:
lm(formula = count ~ I(1 * (spray == "B")) + I(1 * (spray ==
    "C")) + I(1 * (spray == "D")) + I(1 * (spray == "E")) + I(1 *
    (spray == "F")) + I(1 * (spray == "A")), data = InsectSprays)

Coefficients:
          (Intercept)  I(1 * (spray == "B"))  I(1 * (spray == "C"))  I(1 * (spray == "D"))
               14.500                 0.833                -12.417                -9.583
I(1 * (spray == "E"))  I(1 * (spray == "F"))  I(1 * (spray == "A"))
              -11.000                 2.167                     NA
```

What if we omit the intercept?

```
summary(lm(count ~ spray - 1, data = InsectSprays))$coef
```

```
       Estimate Std. Error t value  Pr(>|t|)
sprayA   14.500      1.132  12.807 1.471e-19
sprayB   15.333      1.132  13.543 1.002e-20
sprayC    2.083      1.132   1.840 7.024e-02
sprayD    4.917      1.132   4.343 4.953e-05
sprayE    3.500      1.132   3.091 2.917e-03
sprayF   16.667      1.132  14.721 1.573e-22
```

```
unique(ave(InsectSprays$count, InsectSprays$spray))
```

```
[1] 14.500 15.333  2.083  4.917  3.500 16.667
```

Summary

- If we treat Spray as a factor, R includes an intercept and omits the alphabetically first level of the factor.
  - All t-tests are for comparisons of Sprays versus Spray A.
  - Emprirical mean for A is the intercept.
  - Other group means are the itc plus their coefficient.
- If we omit an intercept, then it includes terms for all levels of the factor.
  - Group means are the coefficients.
  - Tests are tests of whether the groups are different than zero. (Are the expected counts zero for that spray.)
- If we want comparisons between, Spray B and C, say we could refit the model with C (or B) as the reference level.

Reordering the levels

```
spray2 <- relevel(InsectSprays$spray, "C")
summary(lm(count ~ spray2, data = InsectSprays))$coef
```

```
            Estimate Std. Error t value  Pr(>|t|)
(Intercept)    2.083      1.132  1.8401 7.024e-02
spray2A       12.417      1.601  7.7550 7.267e-11
spray2B       13.250      1.601  8.2755 8.510e-12
spray2D        2.833      1.601  1.7696 8.141e-02
spray2E        1.417      1.601  0.8848 3.795e-01
spray2F       14.583      1.601  9.1083 2.794e-13
```

Doing it manually

Equivalently

$$Var(\hat{\beta}_B - \hat{\beta}_C) = Var(\hat{\beta}_B) + Var(\hat{\beta}_C) - 2Cov(\hat{\beta}_B, \hat{\beta}_C)$$

```
fit <- lm(count ~ spray, data = InsectSprays) #A is ref
bbmbc <- coef(fit)[2] - coef(fit)[3] #B - C
temp <- summary(fit)
se <- temp$sigma * sqrt(temp$cov.unscaled[2, 2] + temp$cov.unscaled[3,3] - 2 *temp$cov.unscaled[2,3
t <- (bbmbc) / se
p <- pt(-abs(t), df = fit$df)
out <- c(bbmbc, se, t, p)
names(out) <- c("B - C", "SE", "T", "P")
round(out, 3)
```

```
 B - C     SE      T      P
13.250  1.601  8.276  0.000
```

Other thoughts on this data

- Counts are bounded from below by 0, violates the assumption of normality of the errors.
  - Also there are counts near zero, so both the actual assumption and the intent of the assumption are violated.
- Variance does not appear to be constant.
- Perhaps taking logs of the counts would help.
  - There are 0 counts, so maybe log(Count + 1)
- Also, we'll cover Poisson GLMs for fitting count data.

Example - Millenium Development Goal 1

http://www.un.org/millenniumgoals/pdf/MDG_FS_1_EN.pdf

http://apps.who.int/gho/athena/data/GHO/WHOSIS_000008.csv?profile=text&filter=COUNTRY:;*SEX:*

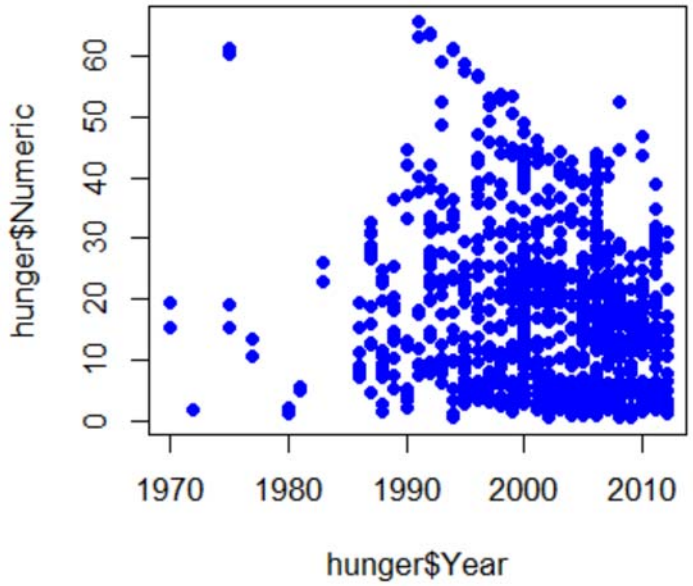
WHO childhood hunger data

```
#download.file("http://apps.who.int/gho/athena/data/GHO/WHOSIS_000008.csv?profile=text&filter=COUNT
hunger <- read.csv("hunger.csv")
hunger <- hunger[hunger$Sex!="Both sexes",]
head(hunger)
```

```
                                Indicator Data.Source PUBLISH.STATES Year            WHO.region
1 Children aged <5 years underweight (%) NLIS_310044      Published 1986                Africa
2 Children aged <5 years underweight (%) NLIS_310233      Published 1990              Americas
3 Children aged <5 years underweight (%) NLIS_312902      Published 2005              Americas
5 Children aged <5 years underweight (%) NLIS_312522      Published 2002 Eastern Mediterranean
6 Children aged <5 years underweight (%) NLIS_312955      Published 2008                Africa
8 Children aged <5 years underweight (%) NLIS_312963      Published 2008                Africa
       Country    Sex Display.Value Numeric Low High Comments
1       Senegal   Male          19.3    19.3  NA   NA       NA
2      Paraguay   Male           2.2     2.2  NA   NA       NA
3     Nicaragua   Male           5.3     5.3  NA   NA       NA
5        Jordan Female           3.2     3.2  NA   NA       NA
6 Guinea-Bissau Female          17.0    17.0  NA   NA       NA
8         Ghana   Male          15.7    15.7  NA   NA       NA
```

Plot percent hungry versus time

```
lm1 <- lm(hunger$Numeric ~ hunger$Year)
plot(hunger$Year,hunger$Numeric,pch=19,col="blue")
```



Remember the linear model

$$Hu_i = b_0 + b_1 Y_i + e_i$$

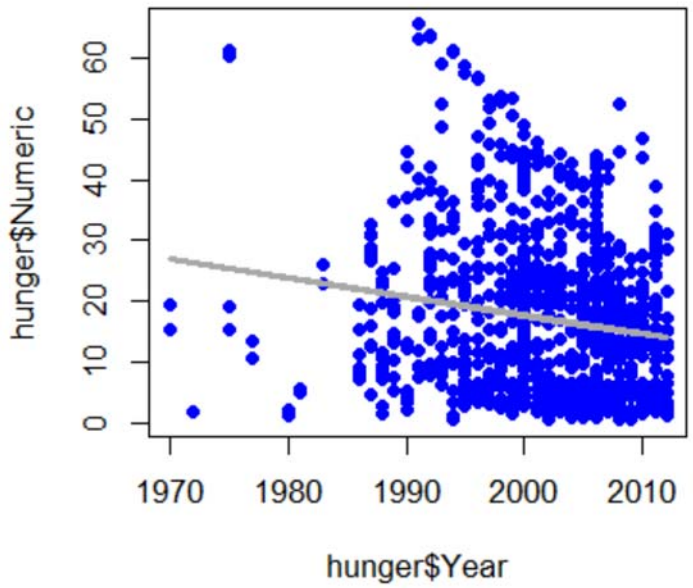
b_0 = percent hungry at Year 0

b_1 = decrease in percent hungry per year

e_i = everything we didn't measure

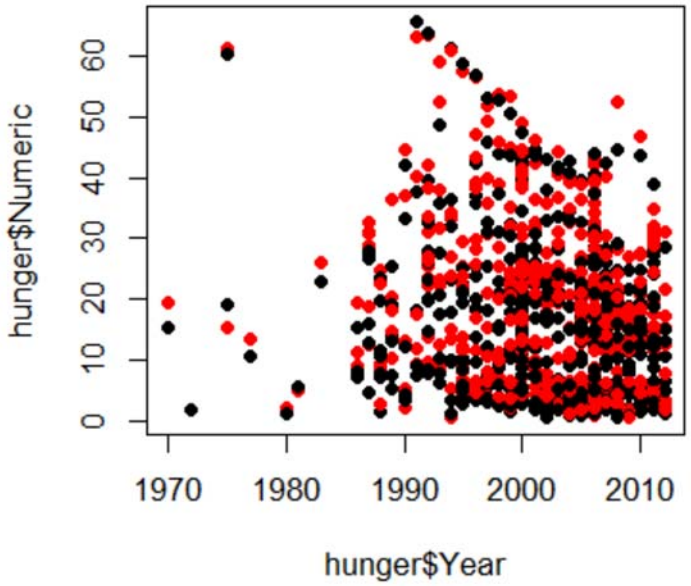
Add the linear model

```
lm1 <- lm(hunger$Numeric ~ hunger$Year)
plot(hunger$Year,hunger$Numeric,pch=19,col="blue")
lines(hunger$Year,lm1$fitted,lwd=3,col="darkgrey")
```



Color by male/female

```
plot(hunger$Year,hunger$Numeric,pch=19)
points(hunger$Year,hunger$Numeric,pch=19,col=((hunger$Sex=="Male")*1+1))
```



Now two lines

$$HuF_i = bf_0 + bf_1 YF_i + ef_i$$

bf_0 = percent of girls hungry at Year 0

bf_1 = decrease in percent of girls hungry per year

ef_i = everything we didn't measure

$$HuM_i = bm_0 + bm_1 YM_i + em_i$$

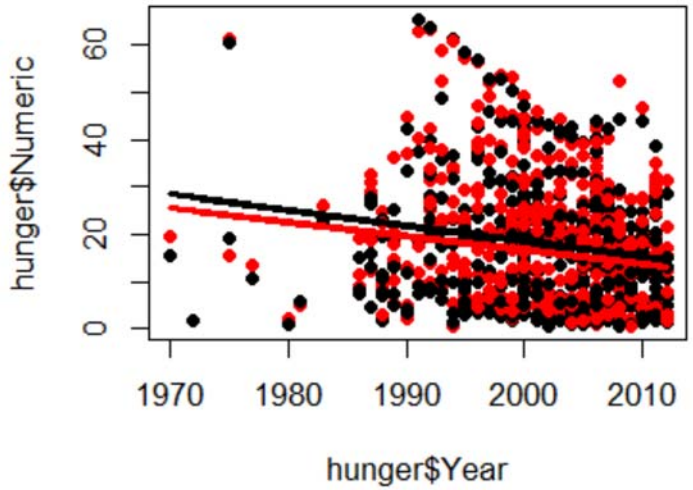
bm_0 = percent of boys hungry at Year 0

bm_1 = decrease in percent of boys hungry per year

em_i = everything we didn't measure

Color by male/female

```
lmM <- lm(hunger$Numeric[hunger$Sex=="Male"] ~ hunger$Year[hunger$Sex=="Male"])
lmF <- lm(hunger$Numeric[hunger$Sex=="Female"] ~ hunger$Year[hunger$Sex=="Female"])
plot(hunger$Year,hunger$Numeric,pch=19)
points(hunger$Year,hunger$Numeric,pch=19,col=((hunger$Sex=="Male")*1+1))
lines(hunger$Year[hunger$Sex=="Male"],lmM$fitted,col="black",lwd=3)
lines(hunger$Year[hunger$Sex=="Female"],lmF$fitted,col="red",lwd=3)
```



Two lines, same slope

$$Hu_i = b_0 + b_1 1(Sex_i = \text{" } Male \text{ "}) + b_2 Y_i + e_i^*$$

b_0 - percent hungry at year zero for females

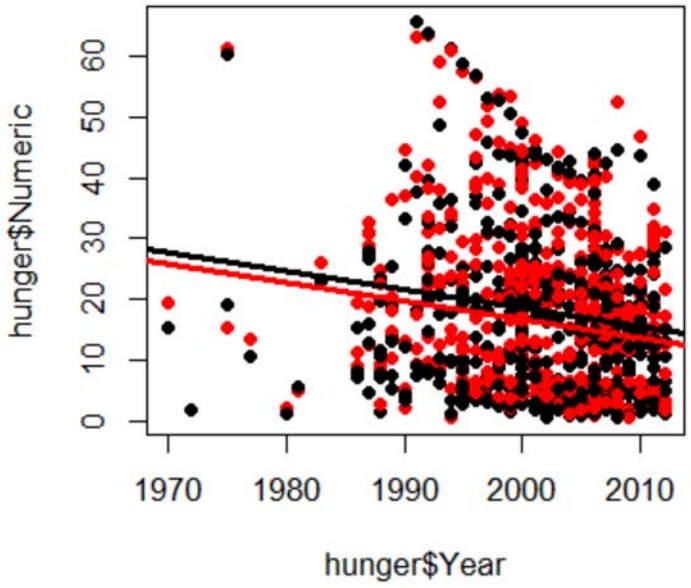
$b_0 + b_1$ - percent hungry at year zero for males

b_2 - change in percent hungry (for either males or females) in one year

e_i^* - everything we didn't measure

Two lines, same slope in R

```
lmBoth <- lm(hunger$Numeric ~ hunger$Year + hunger$Sex)
plot(hunger$Year,hunger$Numeric,pch=19)
points(hunger$Year,hunger$Numeric,pch=19,col=((hunger$Sex=="Male")*1+1))
abline(c(lmBoth$coeff[1],lmBoth$coeff[2]),col="red",lwd=3)
abline(c(lmBoth$coeff[1] + lmBoth$coeff[3],lmBoth$coeff[2] ),col="black",lwd=3)
```



Two lines, different slopes (interactions)

$$Hu_i = b_0 + b_1 1(Sex_i = \text{" } Male \text{ "}) + b_2 Y_i + b_3 1(Sex_i = \text{" } Male \text{ "}) \times Y_i + e_i^+$$

b_0 - percent hungry at year zero for females

$b_0 + b_1$ - percent hungry at year zero for males

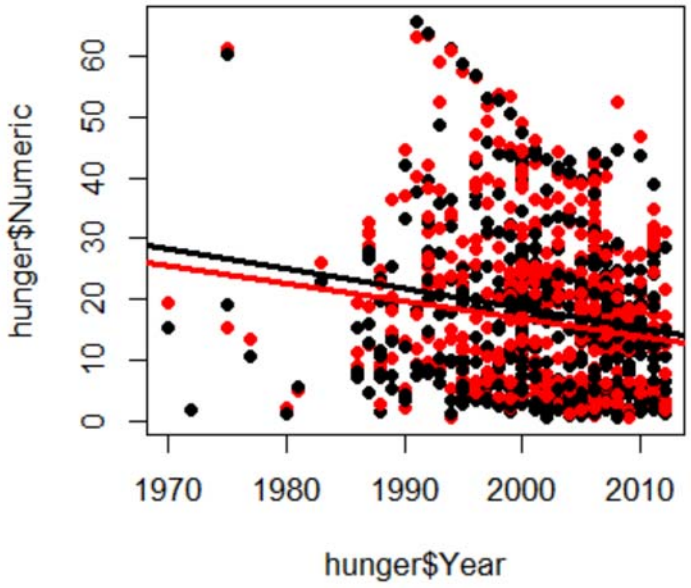
b_2 - change in percent hungry (females) in one year

$b_2 + b_3$ - change in percent hungry (males) in one year

e_i^+ - everything we didn't measure

Two lines, different slopes in R

```
lmBoth <- lm(hunger$Numeric ~ hunger$Year + hunger$Sex + hunger$Sex*hunger$Year)
plot(hunger$Year,hunger$Numeric,pch=19)
points(hunger$Year,hunger$Numeric,pch=19,col=((hunger$Sex=="Male")*1+1))
abline(c(lmBoth$coeff[1],lmBoth$coeff[2]),col="red",lwd=3)
abline(c(lmBoth$coeff[1] + lmBoth$coeff[3],lmBoth$coeff[2] +lmBoth$coeff[4]),col="black",lwd=3)
```



Two lines, different slopes in R

```
summary(lmBoth)
```

```
Call:
lm(formula = hunger$Numeric ~ hunger$Year + hunger$Sex + hunger$Sex *
    hunger$Year)

Residuals:
   Min     1Q Median     3Q    Max
-25.91 -11.25  -1.85   7.09  46.15

Coefficients:
                          Estimate Std. Error t value Pr(>|t|)
(Intercept)               603.5058   171.0552    3.53  0.00044 ***
hunger$Year                -0.2934     0.0855   -3.43  0.00062 ***
hunger$SexMale             61.9477   241.9086    0.26  0.79795
hunger$Year:hunger$SexMale -0.0300     0.1209   -0.25  0.80402
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 13.2 on 944 degrees of freedom
Multiple R-squared:  0.0318,    Adjusted R-squared:  0.0287
F-statistic: 10.3 on 3 and 944 DF,  p-value: 1.06e-06
```

Interpretting a continuous interaction

$$E[Y_i|X_{1i} = x_1, X_{2i} = x_2] = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \beta_3 x_1 x_2$$

Holding X_2 constant we have

$$E[Y_i|X_{1i} = x_1 + 1, X_{2i} = x_2] - E[Y_i|X_{1i} = x_1, X_{2i} = x_2] = \beta_1 + \beta_3 x_2$$

And thus the expected change in Y per unit change in X_1 holding all else constant is not constant. β_1 is the slope when $x_2 = 0$. Note further that:

$$E[Y_i|X_{1i} = x_1 + 1, X_{2i} = x_2 + 1] - E[Y_i|X_{1i} = x_1, X_{2i} = x_2 + 1]$$

$$-E[Y_i|X_{1i} = x_1 + 1, X_{2i} = x_2] - E[Y_i|X_{1i} = x_1, X_{2i} = x_2]$$

$$= \beta_3$$

Thus, β_3 is the change in the expected change in Y per unit change in X_1, per unit change in X_2.

Or, the change in the slope relating X_1 and Y per unit change in X_2.

Example

$$Hu_i = b_0 + b_1 In_i + b_2 Y_i + b_3 In_i \times Y_i + e_i^+$$

b_0 - percent hungry at year zero for children with whose parents have no income

b_1 - change in percent hungry for each dollar of income in year zero

b_2 - change in percent hungry in one year for children whose parents have no income

b_3 - increased change in percent hungry by year for each dollar of income - e.g. if income is $10,000, then change in percent hungry in one year will be

$$b_2 + 1e4 \times b_3$$

e_i^+ - everything we didn't measure

Lot's of care/caution needed!

Multivariable regression

Regression

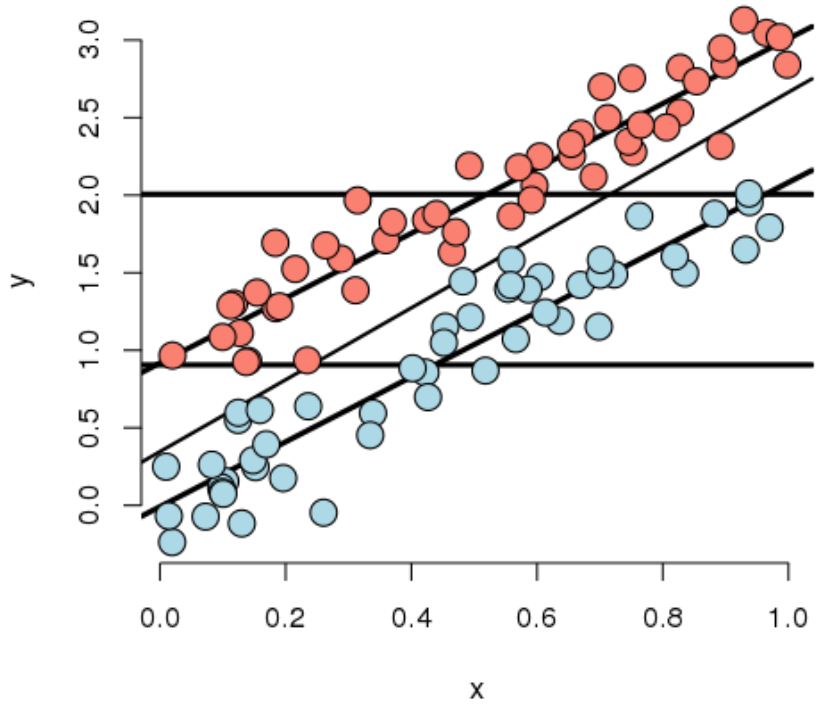
Brian Caffo, Jeff Leek, Roger Peng
Johns Hopkins Bloomberg School of Public Health

Consider the following simulated data

Code for the first plot, rest omitted (See the git repo for the rest of the code.)

```
n <- 100; t <- rep(c(0, 1), c(n/2, n/2)); x <- c(runif(n/2), runif(n/2));
beta0 <- 0; beta1 <- 2; tau <- 1; sigma <- .2
y <- beta0 + x * beta1 + t * tau + rnorm(n, sd = sigma)
plot(x, y, type = "n", frame = FALSE)
abline(lm(y ~ x), lwd = 2)
abline(h = mean(y[1 : (n/2)]), lwd = 3)
abline(h = mean(y[(n/2 + 1) : n]), lwd = 3)
fit <- lm(y ~ x + t)
abline(coef(fit)[1], coef(fit)[2], lwd = 3)
abline(coef(fit)[1] + coef(fit)[3], coef(fit)[2], lwd = 3)
points(x[1 : (n/2)], y[1 : (n/2)], pch = 21, col = "black", bg = "lightblue", cex = 2)
points(x[(n/2 + 1) : n], y[(n/2 + 1) : n], pch = 21, col = "black", bg = "salmon", cex = 2)
```

Simulation 1

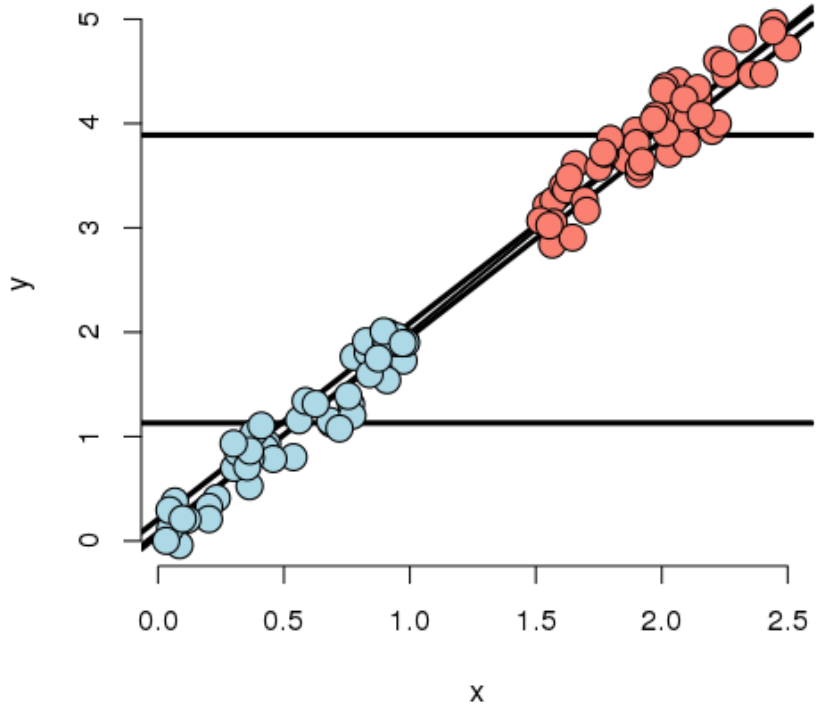


Discussion

Some things to note in this simulation

- The X variable is unrelated to group status
- The X variable is related to Y, but the intercept depends on group status.
- The group variable is related to Y.
    - The relationship between group status and Y is constant depending on X.
    - The relationship between group and Y disregarding X is about the same as holding X constant

Simulation 2

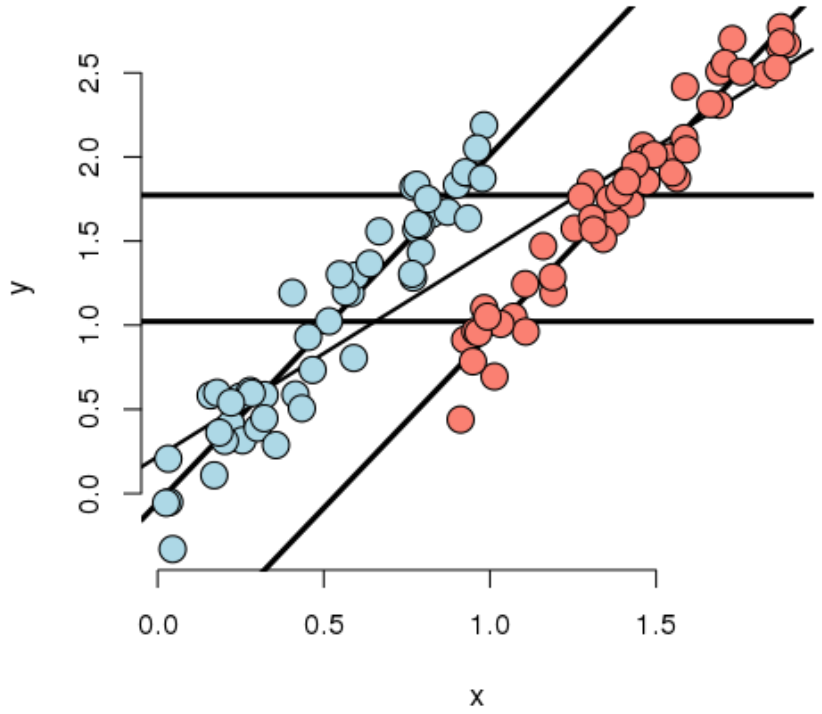


Discussion

Some things to note in this simulation

- The X variable is highly related to group status
- The X variable is related to Y, the intercept doesn't depend on the group variable.
    - The X variable remains related to Y holding group status constant
- The group variable is marginally related to Y disregarding X.
- The model would estimate no adjusted effect due to group.
    - There isn't any data to inform the relationship between group and Y.
    - This conclusion is entirely based on the model.

Simulation 3

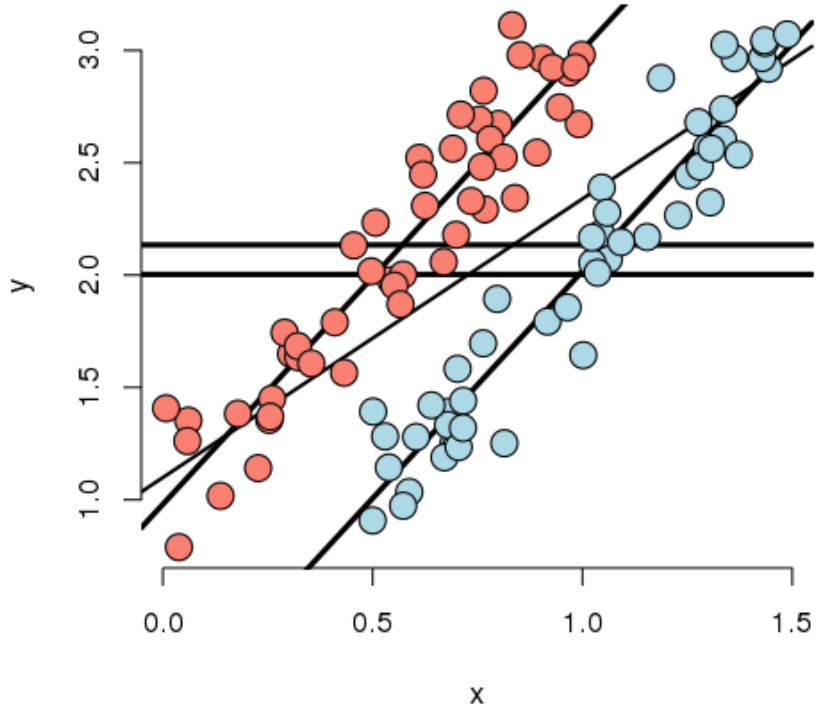


Discussion

Some things to note in this simulation

- Marginal association has red group higher than blue.
- Adjusted relationship has blue group higher than red.
- Group status related to X.
- There is some direct evidence for comparing red and blue holding X fixed.

Simulation 4

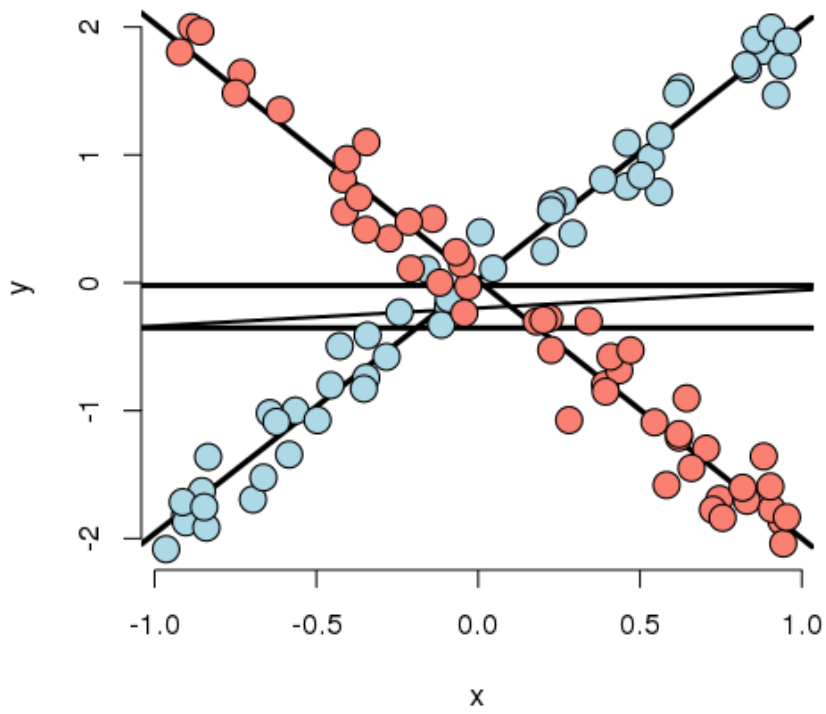


Discussion

Some things to note in this simulation

- No marginal association between group status and Y.
- Strong adjusted relationship.
- Group status not related to X.
- There is lots of direct evidence for comparing red and blue holding X fixed.

Simulation 5

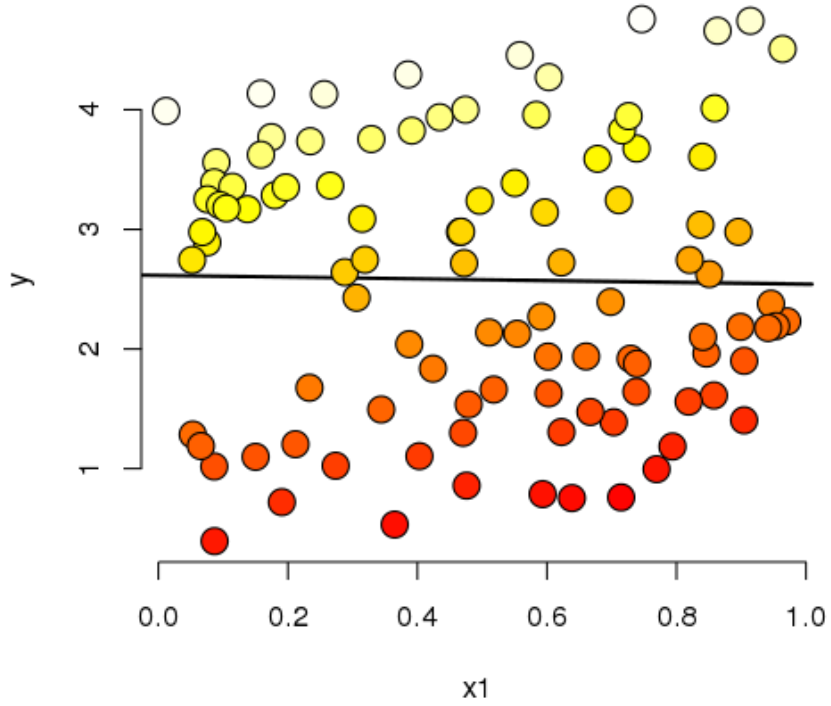


Discussion

Some things to note from this simulation

- There is no such thing as a group effect here.
  - The impact of group reverses itself depending on X.
  - Both intercept and slope depends on group.
- Group status and X unrelated.
  - There's lots of information about group effects holding X fixed.

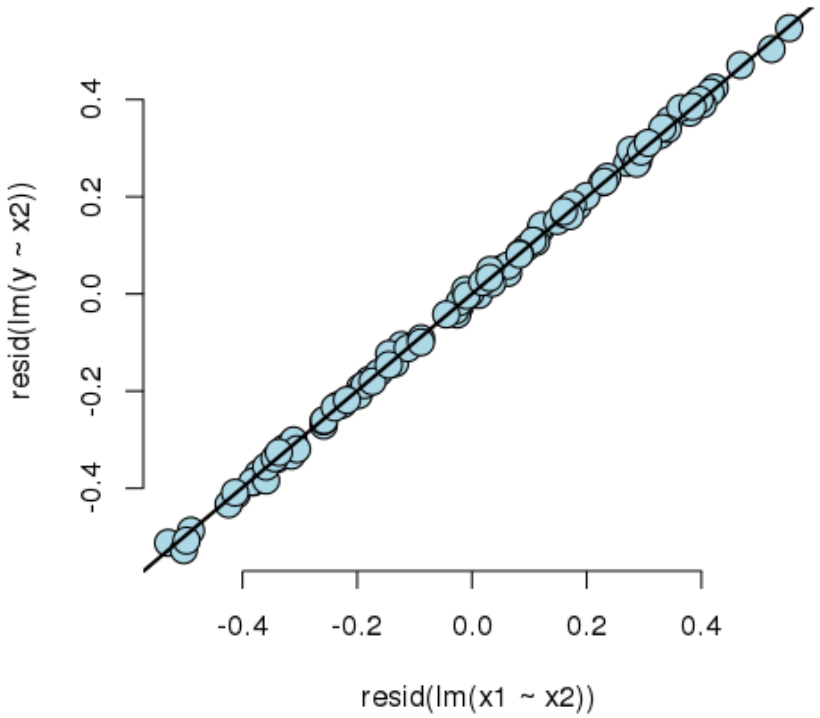
Simulation 6



Do this to investigate the bivariate relationship

```
library(rgl)
plot3d(x1, x2, y)
```

Residual relationship



Discussion

Some things to note from this simulation

- X1 unrelated to X2
- X2 strongly related to Y
- Adjusted relationship between X1 and Y largely unchanged by considering X2.
    - Almost no residual variability after accounting for X2.

Some final thoughts

- Modeling multivariate relationships is difficult.
- Play around with simulations to see how the inclusion or exclustion of another variable can change analyses.
- The results of these analyses deal with the impact of variables on associations.
    - Ascertaining mechanisms or cause are difficult subjects to be added on top of difficulty in understanding multivariate associations.

Residuals, diagnostics, variation

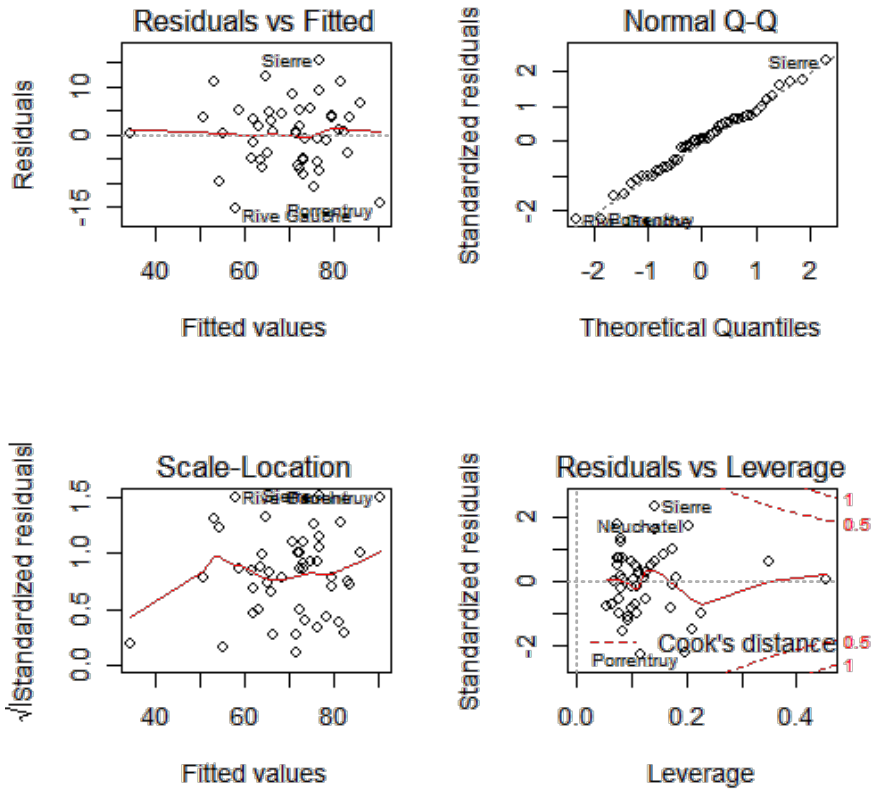
Regression

Brian Caffo, Jeff Leek, Roger Peng
Johns Hopkins Bloomberg School of Public Health

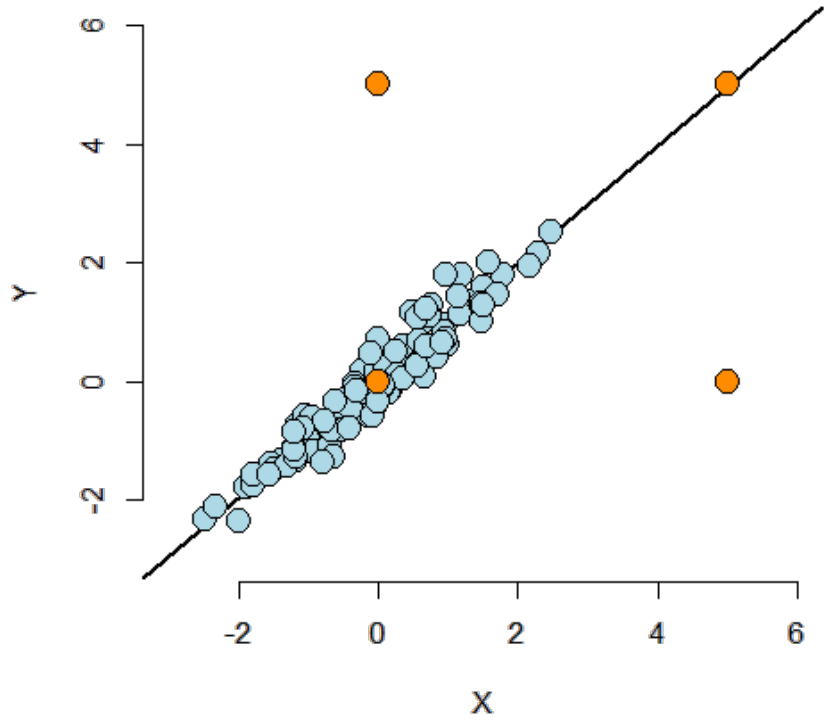
The linear model

- Specified as $Y_i = \sum_{k=1}^{p} X_{ik} \beta_j + \epsilon_i$
- We'll also assume here that $\epsilon_i \overset{iid}{\sim} N(0, \sigma^2)$
- Define the residuals as $e_i = Y_i - \hat{Y}_i = Y_i - \sum_{k=1}^{p} X_{ik} \hat{\beta}_j$
- Our estimate of residual variation is $\hat{\sigma}^2 = \frac{\sum_{i=1}^{n} e_i^2}{n-p}$, the $n - p$ so that $E[\hat{\sigma}^2] = \sigma^2$

```
data(swiss); par(mfrow = c(2, 2))
fit <- lm(Fertility ~ . , data = swiss); plot(fit)
```



Influential, high leverage and outlying points



Summary of the plot

Calling a point an outlier is vague.

- Outliers can be the result of spurious or real processes.
- Outliers can have varying degrees of influence.
- Outliers can conform to the regression relationship (i.e being marginally outlying in X or Y, but not outlying given the regression relationship).
    - Upper left hand point has low leverage, low influence, outlies in a way not conforming to the regression relationship.
    - Lower left hand point has low leverage, low influence and is not to be an outlier in any sense.
    - Upper right hand point has high leverage, but chooses not to extert it and thus would have low actual influence by conforming to the regresison relationship of the other points.
    - Lower right hand point has high leverage and would exert it if it were included in the fit.

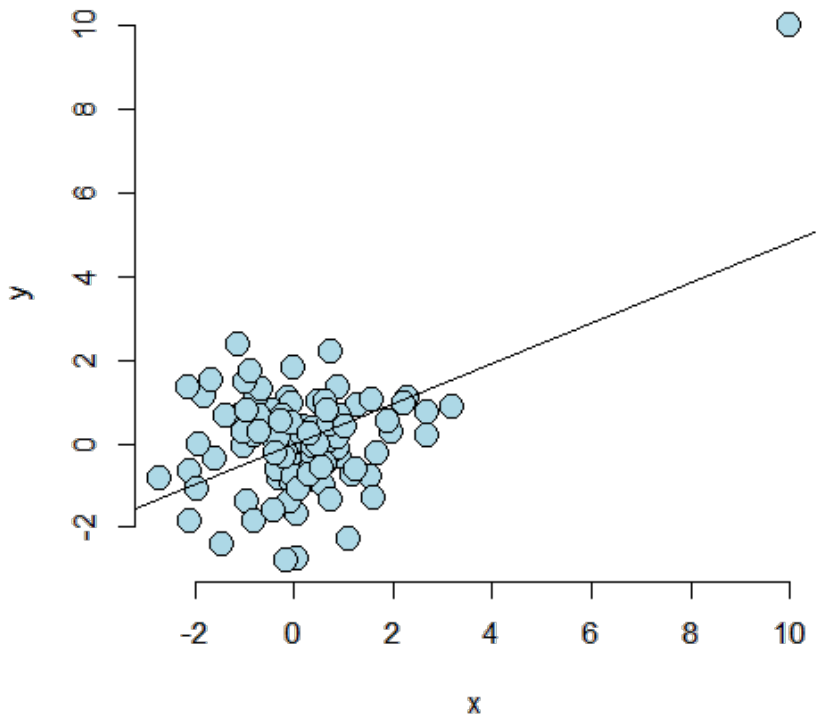
Influence measures

- Do `?influence.measures` to see the full suite of influence measures in stats. The measures include
  - `rstandard` - standardized residuals, residuals divided by their standard deviations)
  - `rstudent` - standardized residuals, residuals divided by their standard deviations, where the ith data point was deleted in the calculation of the standard deviation for the residual to follow a t distribution
  - `hatvalues` - measures of leverage
  - `dffits` - change in the predicted response when the i^{th} point is deleted in fitting the model.
  - `dfbetas` - change in individual coefficients when the i^{th} point is deleted in fitting the model.
  - `cooks.distance` - overall change in teh coefficients when the i^{th} point is deleted.
  - `resid` - returns the ordinary residuals
  - `resid(fit) / (1 - hatvalues(fit))` where `fit` is the linear model fit returns the PRESS residuals, i.e. the leave one out cross validation residuals - the difference in the response and the predicted response at data point i, where it was not included in the model fitting.

How do I use all of these things?

- Be wary of simplistic rules for diagnostic plots and measures. The use of these tools is context specific. It's better to understand what they are trying to accomplish and use them judiciously.
- Not all of the measures have meaningful absolute scales. You can look at them relative to the values across the data.
- They probe your data in different ways to diagnose different problems.
- Patterns in your residual plots generally indicate some poor aspect of model fit. These can include:
    - Heteroskedasticity (non constant variance).
    - Missing model terms.
    - Temporal patterns (plot residuals versus collection order).
- Residual QQ plots investigate normality of the errors.
- Leverage measures (hat values) can be useful for diagnosing data entry errors.
- Influence measures get to the bottom line, 'how does deleting or including this point impact a particular aspect of the model'.

Case 1



The code

```
n <- 100; x <- c(10, rnorm(n)); y <- c(10, c(rnorm(n)))
plot(x, y, frame = FALSE, cex = 2, pch = 21, bg = "lightblue", col = "black")
abline(lm(y ~ x))
```

- The point `c(10, 10)` has created a strong regression relationship where there shouldn't be one.

Showing a couple of the diagnostic values

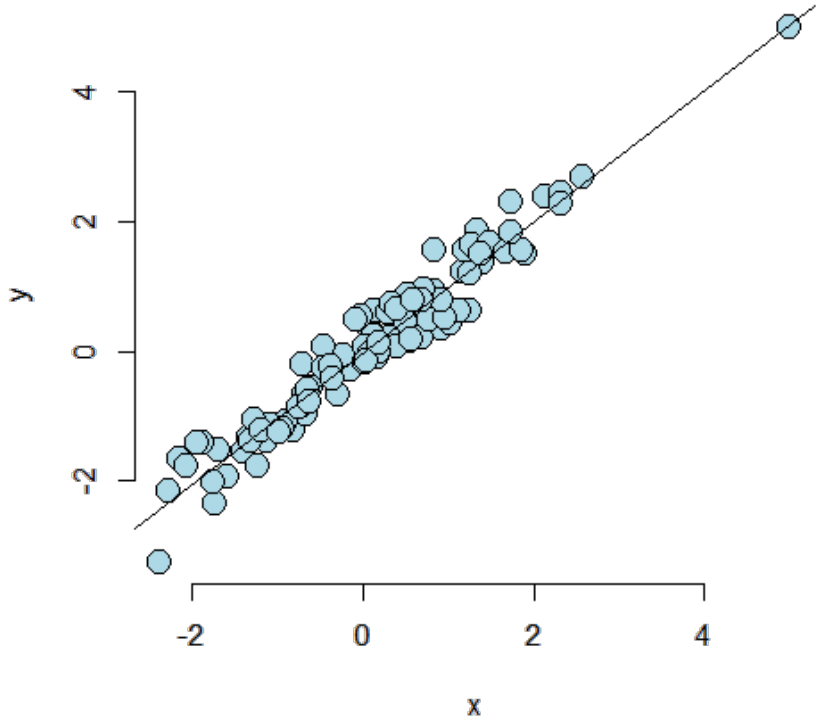
```
fit <- lm(y ~ x)
round(dfbetas(fit)[1 : 10, 2], 3)
```

```
     1      2      3      4      5      6      7      8      9     10
 6.007 -0.019 -0.007  0.014 -0.002 -0.083 -0.034 -0.045 -0.112 -0.008
```

```
round(hatvalues(fit)[1 : 10], 3)
```

```
    1     2     3     4     5     6     7     8     9    10
0.445 0.010 0.011 0.011 0.030 0.017 0.012 0.033 0.021 0.010
```

Case 2



Looking at some of the diagnostics

```
round(dfbetas(fit2)[1 : 10, 2], 3)
```

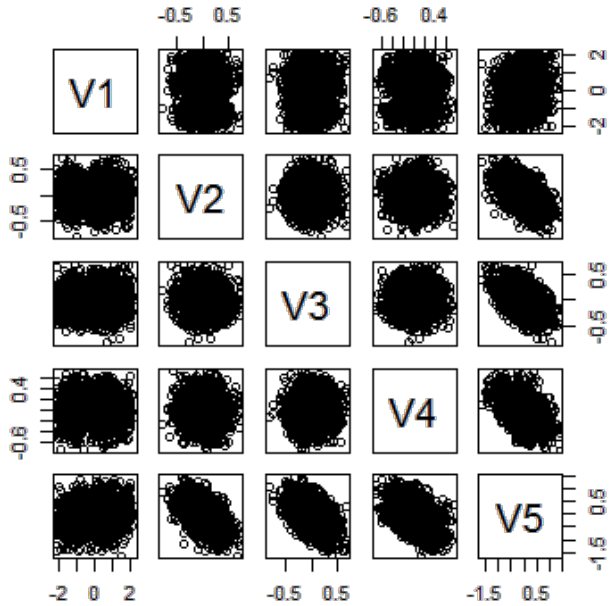
```
     1      2      3      4      5      6      7      8      9     10
-0.072 -0.041 -0.007  0.012  0.008 -0.187  0.017  0.100 -0.059  0.035
```

```
round(hatvalues(fit2)[1 : 10], 3)
```

```
    1     2     3     4     5     6     7     8     9    10
0.164 0.011 0.014 0.012 0.010 0.030 0.017 0.017 0.013 0.021
```

Example described by Stefanski TAS 2007 Vol 61.

```
## Don't everyone hit this server at once.  Read the paper first.
dat <- read.table('http://www4.stat.ncsu.edu/~stefanski/NSF_Supported/Hidden_Images/orly_owl_files/orly
pairs(dat)
```



Got our P-values, should we bother to do a residual plot?

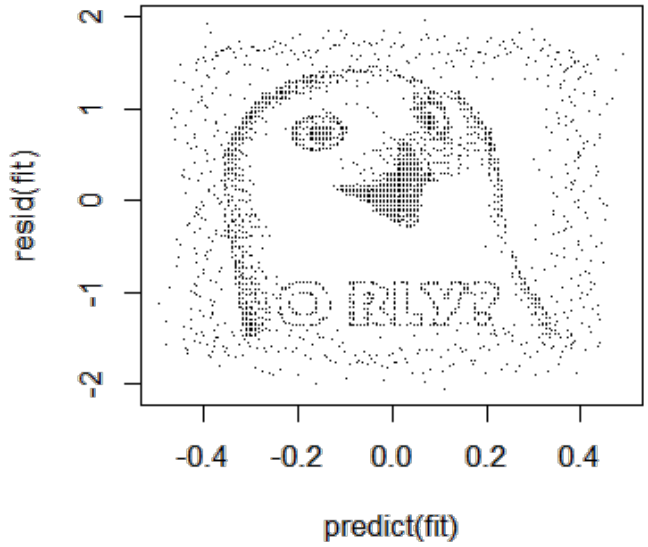
```
summary(lm(V1 ~ . -1, data = dat))$coef
```

```
   Estimate Std. Error t value  Pr(>|t|)
V2   0.9856    0.12798   7.701 1.989e-14
V3   0.9715    0.12664   7.671 2.500e-14
V4   0.8606    0.11958   7.197 8.301e-13
V5   0.9267    0.08328  11.127 4.778e-28
```

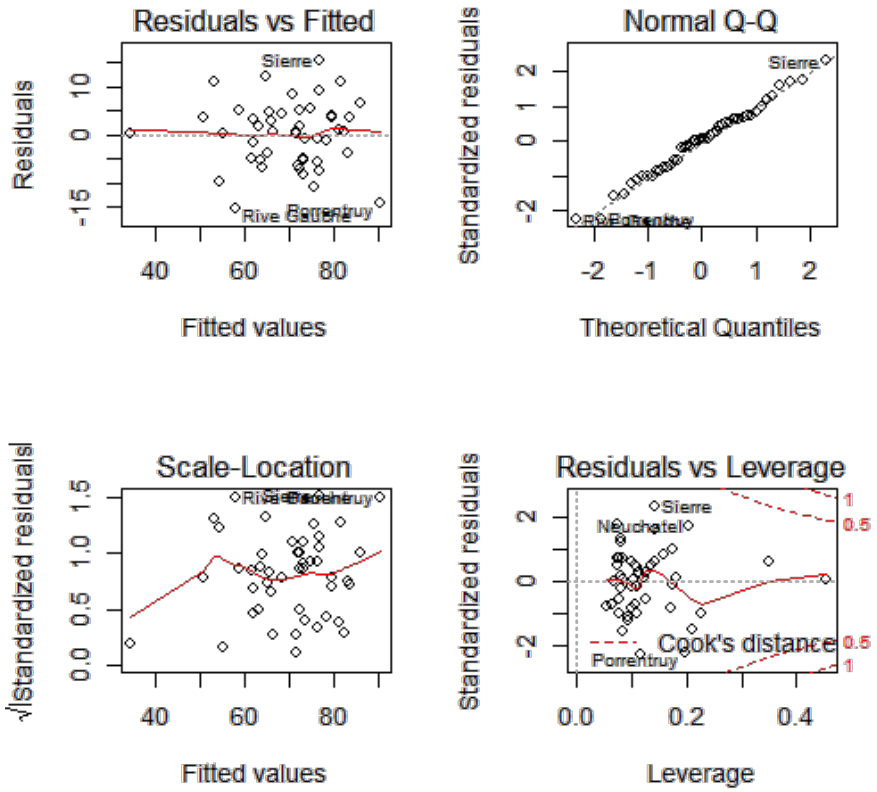
Residual plot

P-values significant, O RLY?

```
fit <- lm(V1 ~ . - 1, data = dat); plot(predict(fit), resid(fit), pch = '.')
```



Back to the Swiss data



Multiple variables

Regression

Brian Caffo, Jeff Leek, Roger Peng
Johns Hopkins Bloomberg School of Public Health

Multivariable regression

- We have an entire class on prediction and machine learning, so we'll focus on modeling.
  - Prediction has a different set of criteria, needs for interpretability and standards for generalizability.
  - In modeling, our interest lies in parsimonious, interpretable representations of the data that enhance our understanding of the phenomena under study.
  - A model is a lense through which to look at your data. (I attribute this quote to Scott Zeger)
  - Under this philosophy, what's the right model? Whatever model connects the data to a true, parsimonious statement about what you're studying.
- There are nearly uncontable ways that a model can be wrong, in this lecture, we'll focus on variable inclusion and exclusion.
- Like nearly all aspects of statistics, good modeling decisions are context dependent.
  - A good model for prediction versus one for studying mechanisms versus one for trying to establish causal effects may not be the same.

The Rumsfeldian triplet

There are known knowns. These are things we know that we know. There are known unknowns. That is to say, there are things that we know we don't know. But there are also unknown unknowns. There are things we don't know we don't know. Donald Rumsfeld

In our context

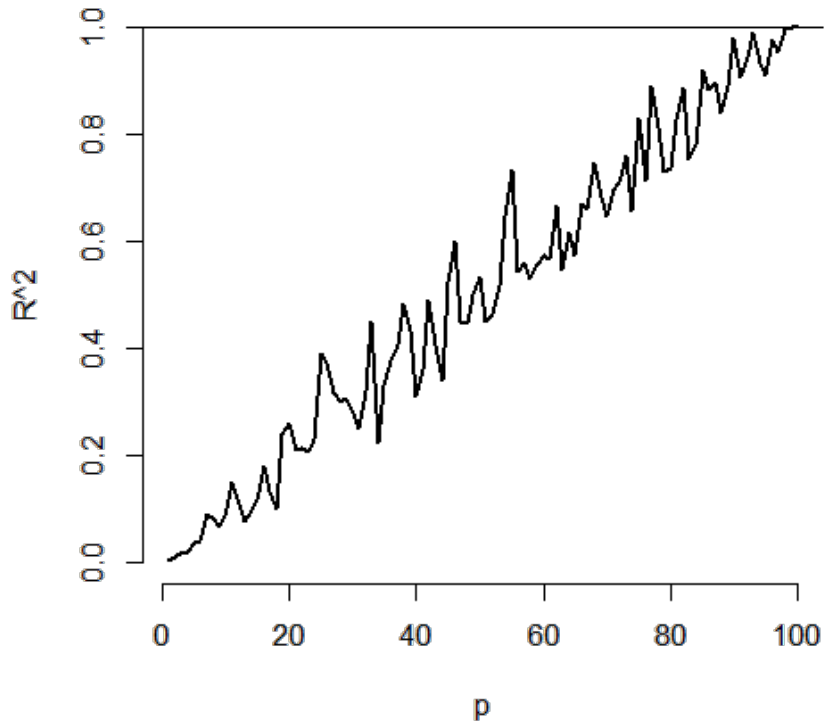
- (Known knowns) Regressors that we know we should check to include in the model and have.
- (Known Unknowns) Regressors that we would like to include in the model, but don't have.
- (Unknown Unknowns) Regressors that we don't even know about that we should have included in the model.

General rules

- Omitting variables results in bias in the coeficients of interest - unless their regressors are uncorrelated with the omitted ones.
  - This is why we randomize treatments, it attempts to uncorrelate our treatment indicator with variables that we don't have to put in the model.
  - (If there's too many unobserved confounding variables, even randomization won't help you.)
- Including variables that we shouldn't have increases standard errors of the regression variables.
  - Actually, including any new variables increasese (actual, not estimated) standard errors of other regressors. So we don't want to idly throw variables into the model.
- The model must tend toward perfect fit as the number of non-redundant regressors approaches n.
- R^2 increases monotonically as more regressors are included.
- The SSE decreases monotonically as more regressors are included.

Plot of R^2 versus n

For simulations as the number of variables included equals increases to $n = 100$. No actual regression relationship exist in any simulation



Variance inflation

```
n <- 100; nosim <- 1000
x1 <- rnorm(n); x2 <- rnorm(n); x3 <- rnorm(n);
betas <- sapply(1 : nosim, function(i){
  y <- x1 + rnorm(n, sd = .3)
  c(coef(lm(y ~ x1))[2],
    coef(lm(y ~ x1 + x2))[2],
    coef(lm(y ~ x1 + x2 + x3))[2])
})
round(apply(betas, 1, sd), 5)
```

```
     x1      x1      x1
0.02839 0.02872 0.02884
```

Variance inflation

```
n <- 100; nosim <- 1000
x1 <- rnorm(n); x2 <- x1/sqrt(2) + rnorm(n) /sqrt(2)
x3 <- x1 * 0.95 + rnorm(n) * sqrt(1 - 0.95^2);
betas <- sapply(1 : nosim, function(i){
  y <- x1 + rnorm(n, sd = .3)
  c(coef(lm(y ~ x1))[2],
    coef(lm(y ~ x1 + x2))[2],
    coef(lm(y ~ x1 + x2 + x3))[2])
})
round(apply(betas, 1, sd), 5)
```

```
     x1      x1      x1
0.03131 0.04270 0.09653
```

Variance inflation factors

- Notice variance inflation was much worse when we included a variable that was highly related to `x1`.
- We don't know σ, so we can only estimate the increase in the actual standard error of the coefficients for including a regressor.
- However, σ drops out of the relative standard errors. If one sequentially adds variables, one can check the variance (or sd) inflation for including each one.
- When the other regressors are actually orthogonal to the regressor of interest, then there is no variance inflation.
- The variance inflation factor (VIF) is the increase in the variance for the ith regressor compared to the ideal setting where it is orthogonal to the other regressors.
  - (The square root of the VIF is the increase in the sd ...)
- Remember, variance inflation is only part of the picture. We want to include certain variables, even if they dramatically inflate our variance.

Revisting our previous simulation

```
##doesn't depend on which y you use,
y <- x1 + rnorm(n, sd = .3)
a <- summary(lm(y ~ x1))$cov.unscaled[2,2]
c(summary(lm(y ~ x1 + x2))$cov.unscaled[2,2],
  summary(lm(y~ x1 + x2 + x3))$cov.unscaled[2,2]) / a
```

```
[1] 1.895 9.948
```

```
temp <- apply(betas, 1, var); temp[2 : 3] / temp[1]
```

```
   x1    x1
1.860 9.506
```

Swiss data

```
data(swiss);
fit1 <- lm(Fertility ~ Agriculture, data = swiss)
a <- summary(fit1)$cov.unscaled[2,2]
fit2 <- update(fit, Fertility ~ Agriculture + Examination)
fit3 <- update(fit, Fertility ~ Agriculture + Examination + Education)
  c(summary(fit2)$cov.unscaled[2,2],
    summary(fit3)$cov.unscaled[2,2]) / a
```

```
[1] 1.892 2.089
```

Swiss data VIFs,

```
library(car)
fit <- lm(Fertility ~ . , data = swiss)
vif(fit)
```

```
     Agriculture      Examination        Education         Catholic Infant.Mortality 
           2.284            3.675            2.775            1.937            1.108 
```

```
sqrt(vif(fit)) #I prefer sd
```

```
     Agriculture      Examination        Education         Catholic Infant.Mortality 
           1.511            1.917            1.666            1.392            1.052 
```

What about residual variance estimation?

- Assuming that the model is linear with additive iid errors (with finite variance), we can mathematically describe the impact of omitting necessary variables or including unnecessary ones.
  - If we underfit the model, the variance estimate is biased.
  - If we correctly or overfit the model, including all necessary covariates and/or unnecessary covariates, the variance estimate is unbiased.
  - However, the variance of the variance is larger if we include unnecessary variables.

Covariate model selection

- Automated covariate selection is a difficult topic. It depends heavily on how rich of a covariate space one wants to explore.
    - The space of models explodes quickly as you add interactions and polynomial terms.
- In the prediction class, we'll cover many modern methods for traversing large model spaces for the purposes of prediction.
- Principal components or factor analytic models on covariates are often useful for reducing complex covariate spaces.
- Good design can often eliminate the need for complex model searches at analyses; though often control over the design is limited.
- If the models of interest are nested and without lots of parameters differentiating them, it's fairly uncontroversial to use nested likelihood ratio tests. (Example to follow.)
- My favoriate approach is as follows. Given a coefficient that I'm interested in, I like to use covariate adjustment and multiple models to probe that effect to evaluate it for robustness and to see what other covariates knock it out. This isn't a terribly systematic approach, but it tends to teach you a lot about the the data as you get your hands dirty.

How to do nested model testing in R

```
fit1 <- lm(Fertility ~ Agriculture, data = swiss)
fit3 <- update(fit, Fertility ~ Agriculture + Examination + Education)
fit5 <- update(fit, Fertility ~ Agriculture + Examination + Education + Catholic + Infant.Mortality)
anova(fit1, fit3, fit5)
```

```
Analysis of Variance Table

Model 1: Fertility ~ Agriculture
Model 2: Fertility ~ Agriculture + Examination + Education
Model 3: Fertility ~ Agriculture + Examination + Education + Catholic +
    Infant.Mortality
  Res.Df  RSS Df Sum of Sq    F  Pr(>F)
1     45 6283
2     43 3181  2      3102 30.2 8.6e-09 ***
3     41 2105  2      1076 10.5 0.00021 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```