11/17 Collinearity in regression

Definition: Predictors $X_1, \ldots, X_p$ are linearly dependent if for some j

$$X_j = \sum_{i \neq j} c_i X_i \quad \text{for some constants } \{c_i\} \qquad \bigstar$$

Example: $p = 3$, $X_1 + X_2 = X_3$, then

$$\begin{aligned} E[y|X] &= \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \beta_3 X_3 \\ &= \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \beta_3 (X_1 + X_2) \\ &= \beta_0 + (\beta_1 + \beta_3) X_1 + (\beta_2 + \beta_3) X_2 \end{aligned}$$

$\rightsquigarrow$ infinitely many β satisfy $\beta^T X = E[y|X]$, e.g. $(\beta_0, \beta_1, \beta_2, \beta_3)$ and $(\beta_0, \beta_1 + \beta_3, \beta_2 + \beta_3, 0)$ $\rightsquigarrow$ β coefficients are "non-identifiable"

Matrix perspective: Columns of X ($n \times (p+1)$ design matrix) are linearly dependent upon each other $\rightsquigarrow$ some $c \neq 0$ with $Xc = 0$

So $X^T X c = 0 \Rightarrow (X^T X)^{-1}$ and thus $\hat{\beta}^{OLS}$ are not uniquely defined (both c and 0 get mapped to 0 by $X^T X$)

Collinearity: Near Linear dependence

Suppose $\bigstar$ holds only approximately $\rightsquigarrow$ mathematically, $(X^T X)^{-1}$ and $\hat{\beta} = (X^T X)^{-1} X^T y$ are well-defined, but they are unstable numerically (small changes in x $\rightsquigarrow$ big changes in $\hat{\beta}$)

Intuition: $\hat{\beta}_j$ = effect of X_j holding other X_i's fixed

Once X_i's are fixed, the range of X_j values is very limited

e.g. SLR with all $x_i \approx \bar{x} \Rightarrow S_{xx} \approx 0 \Rightarrow \hat{\beta}_1 = S_{xy}/S_{xx}$ unstable

Example (Minnesota water usage)

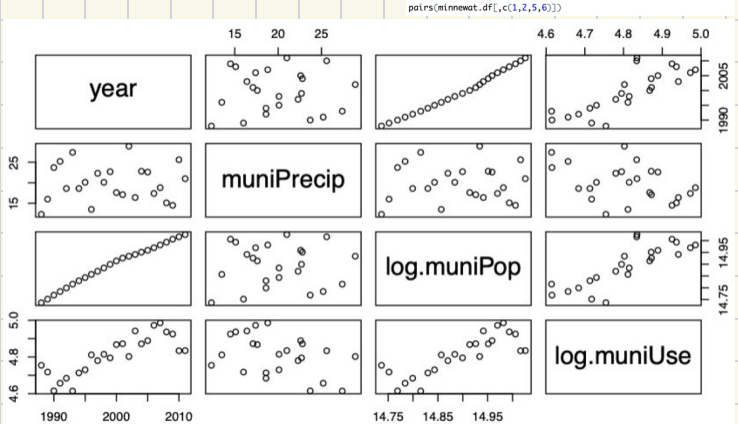
Context: Yearly water usage in Minnesota metropolitan areas, 1988-2011

- log(muniUse) = log of water used (billions of gallons) (response y)
- year = year of measurement
- muniPrecip = # inches of rain during growing season
- log.muniPop = log of population in metro areas

Q: How has water usage changed over time?

```
minnewat.df <- read.csv("MinneWater.csv",header=T)
head(minnewat.df)
minnewat.df$log.muniPop <- log(minnewat.df$muniPop)
minnewat.df$log.muniUse <- log(minnewat.df$muniUse)
head(minnewat.df)
pairs(minnewat.df[,c(1,2,5,6)])
```

Pairwise scatterplots:



1) Water usage increases with time & population

2) year & log(pop) are ≈ linearly dependent

⇝ Water usage increasing with time is just a reflection of "more people ⇒ more water usage"

SLR: $y = \beta_0 + \beta_1 \text{year} + \varepsilon$

```
lmod <- lm(log.muniUse~year,minnewat.df)
summary(lmod)
```

```
Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept) -20.048043   3.745726  -5.352 2.25e-05 ***
year          0.012432   0.001873   6.636 1.13e-06 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.06353 on 22 degrees of freedom
Multiple R-squared:  0.6669,    Adjusted R-squared:  0.6517
F-statistic: 44.04 on 1 and 22 DF,  p-value: 1.132e-06
```

⇝ each year, water usage increases by ≈ 1.25% (not controlling for population increase)

Model 2: Adding precipitation variable

$y = \beta_0 + \beta_1 \text{year} + \beta_2 \text{precip} + \varepsilon$

```
lmod2 <- lm(log.muniUse~year+muniPrecip,minnewat.df)
summary(lmod2)
```

```
Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept) -20.158353   2.726451  -7.394 2.85e-07 ***
year          0.012586   0.001364   9.228 7.77e-09 ***
muniPrecip   -0.009932   0.002192  -4.531 0.000183 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.04624 on 21 degrees of freedom
Multiple R-squared:  0.8315,    Adjusted R-squared:  0.8155
F-statistic: 51.83 on 2 and 21 DF,  p-value: 7.557e-09
```

$\hat{\beta}_1$ is ≈ unchanged

$\hat{\beta}_2 < 0$ (More rain means less need for watering)

Model 3: Control for population (problematic)

```
lmod3 <- lm(log.muniUse~year+muniPrecip+log.muniPop,minnewat.df)
summary(lmod3)
```

```
Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept) -1.278394  11.508965  -0.111    0.913
year        -0.011132   0.014141  -0.787    0.440
muniPrecip  -0.010559   0.002135  -4.946 7.78e-05 ***
log.muniPop  1.917355   1.138236   1.684    0.108
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.04434 on 20 degrees of freedom
Multiple R-squared:  0.8525,    Adjusted R-squared:  0.8303
F-statistic: 38.52 on 3 and 20 DF,  p-value: 1.682e-08
```

Sign of $\hat{\beta}_1$ flipped, log pop. effect is large & positive, but neither are significant

1% inc. in pop ⇝ 1.9% inc. in water usage

Compare with nested model: $y = \beta_0 + \beta_1 \text{muniPrecip} + \varepsilon$

```
> lmod0 <- lm(log.muniUse~muniPrecip,minnewat.df)
> anova(lmod0,lmod3)
Analysis of Variance Table

Model 1: log.muniUse ~ muniPrecip
Model 2: log.muniUse ~ year + muniPrecip + log.muniPop
  Res.Df     RSS Df Sum of Sq      F    Pr(>F)
1     22 0.22695
2     20 0.03932  2   0.18763 47.719 2.437e-08 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

Jointly, year & log(muniPop) have a strong association with $y = \log(\text{muniUse})$, but collinearity prevents us from separating the joint effect into 2 individual effects (both marginal t-tests are not significant)

```
> cor(minnewat.df$year,minnewat.df$log.muniPop)
[1] 0.9955863
```

Intuition: To estimate change in water usage over time holding population fixed, we'd need observations with same pop. but different year (don't exist in data)

⇝ Reframe question: Has water usage per-person changed over time? ⇝ New response variable:

$$\log(\text{perCapitaUse}) = \log\left(\frac{10^6 \cdot \text{muniUse}}{\text{muniPop}}\right)$$

thousands gal / person

and remove log(muniPop) as a predictor in the model

$$\log(\text{perCapitaUse}) = \beta_0 + \beta_1 \text{year} + \beta_2 \text{muniPrecip} + \varepsilon$$

```
> cor(minnewat.df$muniPrecip,minnewat.df$year)
[1] 0.02494691
```

Since year & muniPrecip are ≈ uncorrelated, the collinearity issue doesn't arise ⇝ coefficients can be reliably estimated

```
> minnewat.df$log.perCapitaUse <- log(10^6*minnewat.df$muniUse)-log(minnewat.df$muniPop)
> lmod5 <- lm(log.perCapitaUse~year+muniPrecip,minnewat.df)
> summary(lmod5)

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept)  3.5040361  2.5925575   1.352    0.191
year         0.0002155  0.0012969   0.166    0.870
muniPrecip  -0.0102590  0.0020845  -4.922 7.21e-05 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.04397 on 21 degrees of freedom
Multiple R-squared:  0.5356,    Adjusted R-squared:  0.4914
F-statistic: 12.11 on 2 and 21 DF,  p-value: 0.0003176
```

Interpretation: After appropriately controlling for population growth, little evidence remains for increased water usage

Summary (signs of collinearity)

- $\hat{\beta}$ changes dramatically when predictors added/removed
- High pairwise correlations between predictors
- Counter-intuitive signs or magnitudes of $\hat{\beta}$
- F test is significant, but individual t-tests aren't ⇔ inflated standard errors

Definition: In the MLR model $y = \beta_0 + \beta_1 X_1 + \cdots + \beta_p X_p + \varepsilon$, the Variance Inflation Factor (VIF_j) for a predictor X_j is $VIF_j = \frac{1}{1 - R_j^2}$ where R_j^2 is the R^2-value from regressing X_j onto all other predictors

Interpretation

Recall for SLR, $Var(\hat{\beta}_1) = \sigma^2 / \sum_{i=1}^{n}(x_i - \bar{x})^2$

For MLR, $Var(\hat{\beta}_j) = \frac{\sigma^2}{\sum_{i=1}^{n}(x_{ij} - \bar{x}_j)^2} VIF_j$

$\rightsquigarrow$ VIF_j quantifies how much larger $Var(\hat{\beta}_j)$ is due to the collinearity with other predictors (relative to the variance of $\hat{\beta}_1$ in the SLR fit $y \sim x_j$)

e.g. $VIF_j = 1 \iff R_j^2 = 0$ (X_j has no correlation with other predictors)

$VIF_j = 5 \iff R_j^2 = 0.8$ (Moderate correlation)

$VIF_j = 10 \iff R_j^2 = 0.9 \rightsquigarrow$ Strong correlation

Rule of thumb: $VIF_j > 10$ suggests collinearity

Example (Minnesota water data)

$$VIF_{year} = \frac{1}{1 - R^2_{year \sim others}}$$

```
> lmod <- lm(year~muniPrecip+log.muniPop,minnewat.df)
> summary(lmod)
```

```
Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept) 806.29777   24.17908  33.347   <2e-16 ***
muniPrecip   -0.02587    0.03246  -0.797    0.434
log.muniPop  80.14721    1.62459  49.334   <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.6842 on 21 degrees of freedom
Multiple R-squared:  0.9915,    Adjusted R-squared:  0.9906
F-statistic:  1218 on 2 and 21 DF,  p-value: < 2.2e-16
```

```
> 1/(1-summary(lmod)$r.squared)
[1] 116.9698
```

This computation is automated in R via the vif() function in library(car)

```
> lmod3 <- lm(log.muniUse~year+muniPrecip+log.muniPop,minnewat.df)
> library(car)
> vif(lmod3)
       year  muniPrecip log.muniPop
 116.969782    1.032013  117.095745
```

⇝ year & log(muniPop) are highly collinear, but muniPrecip is not correlated with either

```
Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept) -20.048043   3.745726  -5.352 2.25e-05 ***
year          0.012432   0.001873   6.636 1.13e-06 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.06353 on 22 degrees of freedom
Multiple R-squared:  0.6669,    Adjusted R-squared:  0.6517
F-statistic: 44.04 on 1 and 22 DF,  p-value: 1.132e-06
```

```
Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept) -1.278394  11.508965  -0.111    0.913
year        -0.011132   0.014141  -0.787    0.440
muniPrecip  -0.010559   0.002135  -4.946 7.78e-05 ***
log.muniPop  1.917355   1.138236   1.684    0.108
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.04434 on 20 degrees of freedom
Multiple R-squared:  0.8525,    Adjusted R-squared:  0.8303
F-statistic: 38.52 on 3 and 20 DF,  p-value: 1.682e-08
```

$$\frac{se(\hat{\beta}_1 \text{ in full model})}{se(\hat{\beta}_1 \text{ in reduced model})} \cdot \frac{\hat{\sigma}_{full}}{\hat{\sigma}_{reduced}} = \frac{0.014141}{0.001873\left(\frac{0.04434}{0.06353}\right)} = 10.82 \approx \sqrt{117}$$

Higher-order collinearity example (Sales data)

Context: A firm records its aggregate sales over 23 years

Data:

```
> sale.df <- read.table("sales.txt",header=TRUE)
> head(sale.df)
       S_t      A_t P_t  E_t A_.t.1. P_.t.1.
1 20.11371 1.98786 1.0 0.30 2.01722     0.0
2 15.10439 1.94418 0.0 0.30 1.98786     1.0
3 18.68375 2.19954 0.8 0.35 1.94418     0.0
4 16.05173 2.00107 0.0 0.35 2.19954     0.8
5 21.30101 1.69292 1.3 0.30 2.00107     0.0
6 17.85004 1.74334 0.3 0.32 1.69292     1.3
```

Variables: S_t = Aggregate sales in year t (response)

A_t = advertising expenses A_{t-1} = advertising expenses in previous year

P_t = promotion expenses P_{t-1} = promotion expenses in previous year

E_t = sales expense

MLR model: $S_t = \beta_0 + \beta_1 E_t + \beta_2 A_t + \beta_3 P_t + \beta_4 A_{t-1} + \beta_5 P_{t-1} + \varepsilon$

```
> lmod <- lm(S_t~E_t+A_t+P_t+A_.t.1.+P_.t.1.,sale.df)
> summary(lmod)

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  -14.194     18.715  -0.758   0.4592
E_t           22.521      2.142  10.512 1.36e-08 ***
A_t            5.361      4.028   1.331   0.2019
P_t            8.372      3.586   2.334   0.0329 *
A_.t.1.        3.855      3.578   1.077   0.2973
P_.t.1.        4.125      3.895   1.059   0.3053
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 1.32 on 16 degrees of freedom
Multiple R-squared:  0.9169,    Adjusted R-squared:  0.8909
F-statistic:  35.3 on 5 and 16 DF,  p-value: 4.289e-08
```

Most standard errors are large relative to coefficient estimates + advertising has no significant association with sales (suspicious)

Check F-test for X_j's other than E_t

⇒ F-test rejects but individual t-tests don't.

Drop A_t from model

Signs flip & $\hat{\beta}_j$'s magnitudes change a lot

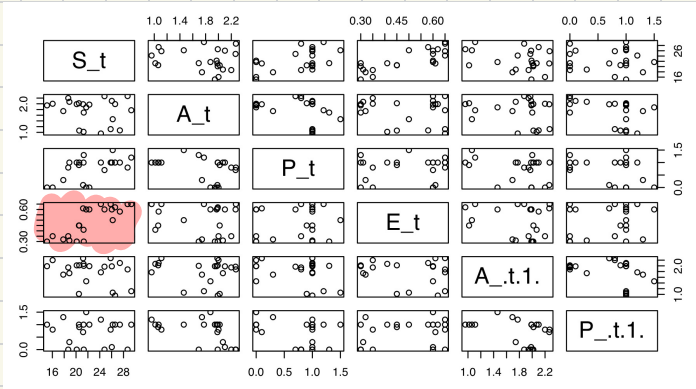
```
> lmod0 <- lm(S_t~E_t,sale.df)
> anova(lmod0,lmod)
Analysis of Variance Table

Model 1: S_t ~ E_t
Model 2: S_t ~ E_t + A_t + P_t + A_.t.1. + P_.t.1.
  Res.Df     RSS Df Sum of Sq      F    Pr(>F)
1     20 114.887
2     16  27.879  4    87.008 12.484 8.487e-05 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

```
> lmod2 <- lm(S_t~E_t+P_t+A_.t.1.+P_.t.1.,sale.df)
> summary(lmod2)

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  10.5094     2.4576   4.276 0.000510 ***
E_t          22.7942     2.1804  10.454 8.04e-09 ***
P_t           3.7018     0.7571   4.889 0.000138 ***
A_.t.1.      -0.7692     0.8746  -0.880 0.391388
P_.t.1.      -0.9687     0.7423  -1.305 0.209273
```

pairs(sale.df) ⇝




```
> round(cor(sale.df),2)
        S_t   A_t   P_t   E_t A_.t.1. P_.t.1.
S_t    1.00 -0.17  0.54  0.81   -0.31   -0.05
A_t   -0.17  1.00 -0.36 -0.13   -0.14   -0.50
P_t    0.54 -0.36  1.00  0.06   -0.32   -0.30
E_t    0.81 -0.13  0.06  1.00   -0.17    0.21
A_.t.1. -0.31 -0.14 -0.32 -0.17    1.00   -0.36
P_.t.1. -0.05 -0.50 -0.30  0.21   -0.36    1.00
```

Pairwise correlations are all small to moderate

Compute VIF scores:

```
> vif(lmod)
     E_t       A_t       P_t   A_.t.1.   P_.t.1.
1.075962 36.941513 33.473514 25.915651 43.520965
```

⇝ collinearity between A_t, P_t, A_{t-1}, P_{t-1}

Try $A_t \sim A_{t-1} + P_t + P_{t-1}$

```
> lmod3 <- lm(A_t~P_t+A_.t.1.+P_.t.1.,sale.df)
> summary(lmod3)
```

$A_t \approx 4.63 - 0.87 P_t - 0.86 A_{t-1} - 0.95 P_{t-1}$

```
Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept)  4.63124    0.12937   35.80  < 2e-16 ***
P_t         -0.86953    0.04333  -20.07 9.08e-14 ***
A_.t.1.     -0.86340    0.05024  -17.18 1.30e-12 ***
P_.t.1.     -0.94689    0.04192  -22.59 1.17e-14 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.0776 on 18 degrees of freedom
Multiple R-squared:  0.9727,    Adjusted R-squared:  0.9681
F-statistic: 213.6 on 3 and 18 DF,  p-value: 2.951e-14
```

Explanation: The firm had a budget constraint on advertising & promotion expenses $A_t + A_{t-1} + P_t + P_{t-1} \approx 5$

```
> sale.df$A_t+sale.df$A_.t.1.+sale.df$P_t+sale.df$P_.t.1.
 [1] 5.00508 4.93204 4.94372 5.00061 4.99399 5.03626 5.11241 5.08616 4.93615 4.98045 5.02138
[12] 4.83510 4.94620 5.18300 5.08598 4.93285 4.94808 4.96041 4.93714 5.04949 5.18400 5.16869
```

```
> summary(sale.df$A_t+sale.df$A_.t.1.+sale.df$P_t+sale.df$P_.t.1.)
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
  4.835   4.944   4.997   5.013   5.077   5.184
```

⇝ VIF scores can detect higher order collinearity not visible in pairwise correlations

Prediction

Claim: Under collinearity, individual $\hat{\beta}_i$ estimates are unreliable, but prediction estimates are stable within the collinear structure

Demonstration: Suppose we want to predict S_t for

$$(E_t, A_t, A_{t-1}, P_t, P_{t-1}) = (0.5, \underbrace{1.5, 1.5, 1, 1}_{\text{sums to 5}}) \quad (*)$$

$$\text{and} = (0.5, \underbrace{2.5, 2.5, 2, 2}_{\text{sums to 9}}) \quad (**)$$

Compare predictions for (*) and (**) using full model (lmod) & model after dropping A_t (lmod2)

```
> ex1 <- data.frame(E_t=0.5,A_t=1.5,A_.t.1.=1.5,P_t=1,P_.t.1.=1) (*)
> ex2 <- data.frame(E_t=0.5,A_t=2.5,A_.t.1.=2.5,P_t=2,P_.t.1.=2) (**)
> c(predict(lmod,ex1),predict(lmod2,ex1))
       1        1
23.38694 23.48578     ≈ same
> c(predict(lmod,ex2),predict(lmod2,ex2))
       1        1
45.09937 25.44968     very different
```

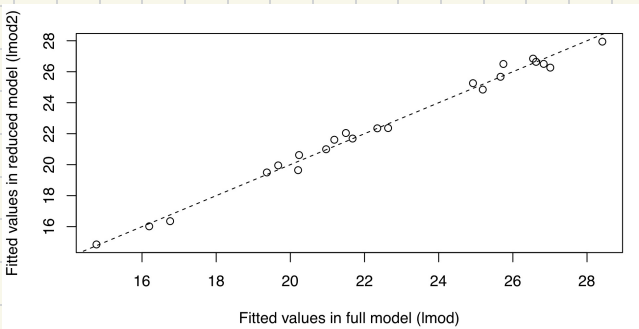
Intuition: $E[S_t \mid E_t, A_t, A_{t-1}, P_t, P_{t-1}] \approx E[S_t \mid E_t, A_{t-1}, P_t, P_{t-1}]$ since A_t is nearly redundant on top of A_{t-1}, P_t, P_{t-1} if

i.e. doesn't add new information

it obeys the relation $A_t + A_{t-1} + P_t + P_{t-1} \approx 5$

⇝ Dropping A_t doesn't lead to big changes in the estimated mean function for the observed range of x values

Comparing fitted values between full & reduced (dropping A_t) models:




```
plot(lmod$fitted.values,lmod2$fitted.values,
     xlab="Fitted values in full model (lmod)",
     ylab="Fitted values in reduced model (lmod2)")
abline(a=0,b=1,lty=2)
```

Summary for collinearity

1) Detection methods:

- Scatter plot matrix (for pairwise collinearity)
- VIF scores bigger than 10
- lm output: large SEs, non-significant coefficient estimates despite significant F-test.

2) What is the goal? Inference on β or prediction of $E[y^*|x^*]$

Task	Status	Why?
Inference on β_js	Problematic	$\hat{\beta}$ is unstable, tests have low power.
Predict within collinearity	O.K.	If constraint holds, $\hat{\beta}^T x_i$ is stable
Predict outside collinearity	Problematic	If it doesn't, $\hat{\beta}^T x_i$ can be unstable

11/21 Bias variance trade-off + Variable Selection

Motivating example

Consider a MLR with $p=2$ predictors and $\text{cor}(X_1, X_2) > 0.9$

Full model: $y = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \varepsilon \rightsquigarrow (\hat{\beta}_0, \hat{\beta}_1, \hat{\beta}_2)$

Subset model: $y = \beta_0' + \beta_1' X_1 + \varepsilon \rightsquigarrow (\hat{\beta}_0', \hat{\beta}_1')$

For simplicity, assume $S_{x_1x_1} = S_{x_2x_2} = 1$

Goal: Estimate effect of X_1 on y, accounting for X_2

(β_1 in the full model)

Error criterion: $\underbrace{E[(\hat{\beta}_1 - \beta_1)^2]}_{MSE(\hat{\beta}_1)} \overset{*}{=} Var(\hat{\beta}_1) + \underbrace{(E\hat{\beta}_1 - \beta_1)^2}_{\text{bias}}$ for any $\hat{\beta}_1$

$\hat{\beta}_1$ from full model fit has bias $= 0$, but high variance,

$$Var(\hat{\beta}_1) = \frac{\sigma^2}{S_{x_1x_1}} \underbrace{VIF_1}_{1/(1-R_{12}^2)}$$

for regressing $X_2 \sim X_1$

Consider estimating β_1 with $\hat{\beta}_1'$ (from subset fit)

$\rightsquigarrow$ incurs a bias, but may have much lower variance

(i.e, a better bias/variance tradeoff)

Claim: Bias $= E[\hat{\beta}_1'] - \beta_1 = R_{12}\beta_2$ assuming full model is true

because $\hat{\beta}_1' = \frac{S_{x_1 y}}{S_{x_1 x_1}} = \sum_{i=1}^{n} (x_{1i} - \bar{x}_1)(y_i - \bar{y})$

$y_i - \bar{y} = \beta_1 (x_{1i} - \bar{x}_1) + \beta_2 (x_{2i} - \bar{x}_2) + \varepsilon_i - \bar{\varepsilon}$

$= \beta_1 \underbrace{S_{x_1 x_1}}_{1} + \beta_2 \underbrace{S_{x_1 x_2}}_{R_{12}} + \underbrace{\sum_{i=1}^{n} (x_{1i} - \bar{x}_1)(\varepsilon_i - \bar{\varepsilon})}_{\sum_{i=1}^{n} (x_{1i} - \bar{x}_1)\varepsilon_i}$

since $\bar{\varepsilon} \sum_{i=1}^{n} (x_{1i} - \bar{x}_1) = 0$

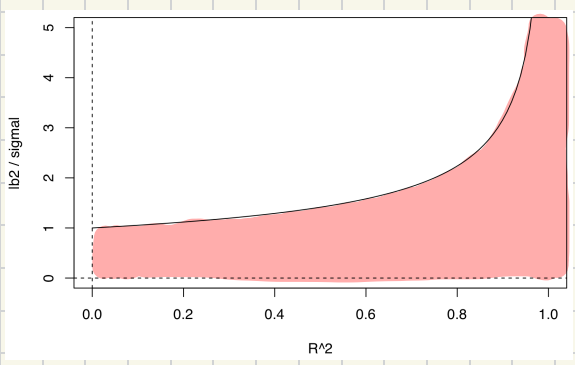
$\Rightarrow E[\hat{\beta}_1'] = \beta_1 + \underbrace{\beta_2 R_{12}}_{\text{bias}}$ □

Claim: $Var(\hat{\beta}_1') = \sigma^2$

because $\sum_{i=1}^{n} (x_{1i} - \bar{x})^2 Var(\varepsilon_i) = \sigma^2 S_{x_1 x_1} = \sigma^2$ □

★ $\Rightarrow MSE(\hat{\beta}_1') = \sigma^2 + (\beta_2 R_{12})^2 \overset{?}{<} \frac{\sigma^2}{1 - R_{12}^2} = MSE(\hat{\beta}_1)$

happens when $\left| \frac{\beta_2}{\sigma} \right| < \frac{1}{\sqrt{1 - R_{12}^2}}$




```
r.sq <- seq(0,1,length=100)
plot(r.sq,1/sqrt(1-r.sq),
     xlim=c(0,1),ylim=c(0,5),type="l",
     xlab="R^2",ylab="|b2 / sigma|")
abline(h=0,lty=2)
abline(v=0,lty=2)
```

If $R_{12}^2 \approx 1$, then the subset model provides a better estimate for β_1. For any $R_{12}^2 > 0$, the subset model estimate has lower MSE if $|\beta_2/\sigma| < 1$

⇝ Dropping X_2 can lead to better

estimation of β_1, even if the true $\beta_2 \neq 0$ (as long as it is small, relative to σ)

For larger models $p > 2$ (not necessarily collinear), still want to select a subset of predictors sufficiently informative about the response (fewer variables $\rightsquigarrow$ more interpretable) and delete variables with small coefficients $|\beta_j|/\sigma$.

$|\beta_j|$ and σ are unknown $\rightsquigarrow$ selection procedures try to estimate which variables are worth keeping

Example (Highway data) Minnesota, 1973

Civil engineering data from n=39 highway segments

Response: rate of accidents (per million vehicle miles) $\frac{a}{V \times len}$

11 potential predictor variables: segment length, daily traffic volume, speed limit, lane width, highway type ... $y = \log(\text{rate})$

```
> highway.df <- read.csv("highway.csv",header=TRUE)
> head(highway.df)
  adt trks lane acpt sigs  itg slim   len lwid shld htype rate
1  69    8    8  4.6 0.00 1.20   55  4.99   12   10   fai 4.58
2  73    8    4  4.4 0.00 1.43   60 16.11   12   10   fai 2.86
3  49   10    4  4.7 0.00 1.54   60  9.75   12   10   fai 3.02
4  61   13    6  3.8 0.00 0.94   65 10.65   12   10   fai 2.29
5  28   12    4  2.2 0.00 0.65   70 20.01   12   10   fai 1.61
6  30    6    4 24.8 1.84 0.34   55  5.97   12   10    pa 6.87
```

access points per mile

interchanges per mile

See pg 192 & 241 in textbook for variable descriptions

```
Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept)  4.668229   1.795502   2.600  0.01543 *
log(adt)    -0.062081   0.112063  -0.554  0.58451
trks        -0.027997   0.026450  -1.058  0.29996
lane         0.002819   0.061625   0.046  0.96388
acpt         0.004326   0.009989   0.433  0.66872
sigs         0.183682   0.127522   1.440  0.16216
itg         -0.084730   0.262868  -0.322  0.74988
slim        -0.035406   0.019298  -1.835  0.07848 .
log(len)    -0.289590   0.103505  -2.798  0.00976 **
lwid        -0.016581   0.147717  -0.112  0.91152
shld         0.001702   0.039750   0.043  0.96619
htypema     -0.270216   0.385565  -0.701  0.48988
htypemc     -0.092198   0.428052  -0.215  0.83121
htypepa     -0.372000   0.307428  -1.210  0.23758
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.2829 on 25 degrees of freedom
Multiple R-squared:  0.7543,    Adjusted R-squared:  0.6266
F-statistic: 5.905 on 13 and 25 DF,  p-value: 7.572e-05
```

Goal: Understand which design variables are associated with accident rate $\rightsquigarrow$ Select a small subset of predictors that are most important / informative for modeling accident rate

Information Criteria

Want a measure that is big when model fit is poor or the model is too complex (too many predictor variables) that we can evaluate for candidate subsets

(Smaller values are preferred)

AIC (Akaike Information Criterion, Sakamoto et. al 1986)

$$AIC = n \log(RSS(\text{subset model})/n) + 2p_s$$

p_s: # betas in subset model (including intercept β_0)

Bayesian Information Criterion (BIC, Schwarz 1978)

$$BIC = n \log(RSS(\text{subset model})/n) + \log(n)\, p_s$$

$\rightsquigarrow$ penalizes complexity more heavily

Asymptotic justifications ($n \to \infty$, see references)

AIC or BIC are used to select a model, but we

can't evaluate AIC / BIC for all 2^p candidate subset models.

Two efficient methods

Backward Elimination (e.g. using AIC)

1) Start with full model as the "current" model

2) Consider candidate models differing from the current model by deletion of one variable, and compute their AIC

3) Accept the current model if AIC is higher for all candidates. Else, set the current model to the one with minimal AIC

If no regressors remain, stop. Else, repeat steps 2 & 3

Forward selection

1) Start with only intercept in current model

2) Consider candidate models differing from the current model by addition of one variable, and compute their AIC

3) Same as in backwards selection

(These don't necessarily lead to the same model)

Both algorithms consider at most $p(p+1)/2$ candidate subsets of predictors, where p = total # of potential

Predictors, e.g. $p = 11 \Rightarrow \frac{p(p+1)}{2} = 66$ vs. $2^p = 2048$ possible subsets (latter is computationally challenging as p grows)

Example (highway data)

```
lmod0 <- lm(log(rate)~1,highway.df)
lmod1 <- lm(log(rate)~log(adt)+trks+lane+acpt+sigs+
              itg+slim+log(len)+lwid+shld+htype,highway.df)
```

```
> step(lmod0,scope=list(upper=lmod1),direction="forward")
Start:  AIC=-59.08
log(rate) ~ 1

             Df Sum of Sq    RSS     AIC
+ slim        1    3.8806 4.2635 -82.325
+ acpt        1    3.5719 4.5723 -79.599
+ log(len)    1    2.6604 5.4838 -72.509
+ log(trks)   1    2.4224 5.7218 -70.852
+ sigs        1    2.3478 5.7964 -70.347
+ shld        1    1.3230 6.8212 -63.998
<none>                    8.1442 -59.084
+ htype       3    0.8727 7.2715 -57.504
+ lane        1    0.0066 8.1376 -57.116
+ log(adt)    1    0.0063 8.1379 -57.114
+ itg         1    0.0056 8.1386 -57.111
+ lwid        1    0.0040 8.1402 -57.103
```

First step of forward selection

⇝ Choose to add speed limit first (lowest AIC)

```
Step:  AIC=-82.33
log(rate) ~ slim

             Df Sum of Sq    RSS     AIC
+ log(len)    1   1.32693 2.9366 -94.866
+ log(trks)   1   0.96560 3.2979 -90.340
+ sigs        1   0.63047 3.6331 -86.566
+ acpt        1   0.55954 3.7040 -85.812
<none>                    4.2635 -82.325
+ lane        1   0.20789 4.0556 -82.275
+ log(adt)    1   0.17195 4.0916 -81.931
+ itg         1   0.17021 4.0933 -81.914
+ shld        1   0.08165 4.1819 -81.079
+ lwid        1   0.06687 4.1967 -80.942
+ htype       3   0.17421 4.0893 -77.952
```

Second step

⇝ Choose log(len) next

⋮

```
Call:
lm(formula = log(rate) ~ slim + log(len) + acpt + log(trks),
    data = highway.df)

Coefficients:
(Intercept)         slim     log(len)         acpt    log(trks)
    4.16654     -0.03185     -0.23573      0.01100     -0.32904
```

(Final model)

```
Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept)  4.166541   0.741065   5.622 2.67e-06 ***
slim        -0.031852   0.010262  -3.104  0.00383 **
log(len)    -0.235735   0.084897  -2.777  0.00887 **
acpt         0.011004   0.006669   1.650  0.10815
log(trks)   -0.329037   0.213484  -1.541  0.13251
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.2698 on 34 degrees of freedom
Multiple R-squared:  0.6961,    Adjusted R-squared:  0.6603
F-statistic: 19.47 on 4 and 34 DF,  p-value: 2.067e-08
```

Interpretation: Lower accident rates on longer highway segments with higher speed limits & truck traffic, more access points ⇝ higher rate

(Highway designers may put lower speed limit where there have been more accidents + len appears in response $\log\left(\frac{\text{accidents}}{\text{vehicles} \times \text{len}}\right)$)

Backward selection:

```
> step(lmod1,scope=list(lower=lmod0),direction="backward")
Start:  AIC=-87.72
log(rate) ~ log(adt) + log(trks) + lane + acpt + sigs + itg +
    slim + log(len) + lwid + shld + htype

            Df Sum of Sq    RSS     AIC
- htype      3   0.17951 2.1860 -90.379
- shld       1   0.00004 2.0065 -89.720
- lane       1   0.00040 2.0068 -89.713
- lwid       1   0.00113 2.0076 -89.699
- itg        1   0.00620 2.0126 -89.600
- acpt       1   0.01548 2.0219 -89.421
- log(adt)   1   0.02522 2.0317 -89.234
- log(trks)  1   0.08384 2.0903 -88.124
<none>                   2.0064 -87.721
- sigs       1   0.13553 2.1420 -87.172
- slim       1   0.26232 2.2688 -84.929
- log(len)   1   0.61993 2.6264 -79.220
```

First step of backward elimination ⇝ remove htype (categorical) first

⋮

```
Call:
lm(formula = log(rate) ~ log(adt) + acpt + sigs + itg + slim +
    log(len), data = highway.df)

Coefficients:
(Intercept)    log(adt)        acpt        sigs         itg        slim     log(len)
   3.631945   -0.120181    0.009558    0.161297    0.226373   -0.028792   -0.308160
```

(final model, different from forward selection)

↳ higher traffic associated w/ lower accident rate

```
> n <- nrow(highway.df)
> lmod.f <- lm(log(rate)~slim+log(len)+acpt+log(trks),highway.df)
> k.f <- 4+1
> lmod.b <- lm(log(rate)~log(adt)+acpt+sigs+itg+slim+log(len),highway.df)
> k.b <- 6+1
> # Compute AICs
> n*log(sum(residuals(lmod.f)^2)/n)+2*k.f
[1] -97.53195
> n*log(sum(residuals(lmod.b)^2)/n)+2*k.b
[1] -95.8069
```

⇝ Choose forward selected model (fewer predictors, more intuitive, lower AIC)