STAT 516 Lec 04

Multiple linear regression (part 2/2)

Karl Gregory

2024-02-06

Rental rates of commercial properties example

As in part 1/2, consider these data from Kutner et al. (2005).

```
link <- url("https://people.stat.sc.edu/gregorkb/data/KNLIcp.txt")
commprop <- read.table(link,col.names=c("rent","age","optx","vac","sqft"))
commprop$sqft <- commprop$sqft/10000 # rescale sqft
head(commprop)
```

```
  rent age  optx  vac    sqft
1 13.5   1  5.02 0.14 12.3000
2 12.0  14  8.19 0.27 10.4079
3 10.5  16  3.00 0.00  3.9998
4 15.0   4 10.70 0.05  5.7112
5 14.0  11  8.97 0.07  6.0000
6 10.5  15  9.45 0.24 10.1385
```

```
n <- nrow(commprop)
p <- ncol(commprop) - 1
```

There are $n = 81$ rows and $p = 4$ predictors.

Setup

Consider data $(Y_1, \mathbf{x}_1), \ldots, (Y_n, \mathbf{x}_n)$, with each $\mathbf{x}_i = (x_{i1}, \ldots, x_{ip})^T$.

The multiple linear regression model is

$$Y_i = \beta_0 + x_{i1}\beta_1 + \cdots + x_{ip}\beta_p + \varepsilon_i, \quad i = 1, \ldots, n,$$

where

- $\mathbf{x}_1, \ldots, \mathbf{x}_n$ are vectors in $\mathbb{R}^p$ of covariate or predictor values.
- $Y_1, \ldots, Y_n$ are the response values
- $\beta_0, \beta_1, \ldots, \beta_p$ are the regression coefficients.
- $\varepsilon_1, \ldots, \varepsilon_n$ are iid Normal$(0, \sigma^2)$ error terms.
- σ^2 is the error term variance.

Goals in multiple linear regression

In part 1/2, we addressed these goals:

1. Estimate the regression coefficients β_0 and $\beta_1, \ldots, \beta_p$.
2. Estimate the error term variance σ^2.
3. Perform inference on $\beta_1, \ldots, \beta_p$.
4. Build a CI for $\beta_0 + \beta_1 x_{\text{new},1} + \cdots + \beta_p x_{\text{new},p}$ at any $\mathbf{x}_{\text{new}}$.
5. Build a prediction interval for Y at any $\mathbf{x}_{\text{new}}$.
6. Decompose the variation in Y into (sums of) sums of squares.
7. Check whether the model assumptions are satisfied.
8. Identify outliers and understand their effects.

In part 2/2 we focus on these:

9. Test for significance of a subset of covariates
10. Understand how correlations among the covariates affect inferences
11. Do variable selection

$$Y_i = \beta_0 + \beta_1 x_{i1} + \cdots + \beta_p x_{ip} + \varepsilon_i$$

Let $M \subset \{1, \ldots, p\}$ (subset of) be a subset of $\{1, \ldots, p\}$.

Want to test

$$H_0: \quad \beta_j = 0 \quad \text{for all} \quad j \notin M \quad (\notin \text{ not in})$$

Example:

$$M = \{1, 2\} \Rightarrow \quad Y_i = \beta_0 + \beta_1 x_{i1} + \beta_2 x_{i2} + \varepsilon_i$$

$$H_0: \beta_j = 0 \quad \text{for all} \quad j \notin \{1, 2\}$$

i.e. $H_0: \beta_3 = 0, \ldots, \beta_p = 0.$

Review of F distributions

For $W_1 \sim \chi^2_{\nu_1}(\phi)$, $W_2 \sim \chi^2_{\nu_2}$ independent, $R = \dfrac{W_1/\nu_1}{W_2/\nu_2} \sim F_{\nu_1,\nu_2}(\phi)$.

$F_{\nu_1,\nu_2}(\phi)$ denotes the F distribution with

- numerator degrees of freedom ν_1
- denominator degrees of freedom ν_2
- noncentrality parameter $\phi \geq 0$

If $\phi > 0$ the distribution is a non-central F distribution.

When $\phi = 0$ we just write F_{ν_1,ν_2} to denote the "central" F distribution.

We will encounter ratios of sums of squares which have F distributions.

Plot of some F distribution pdfs

$Z \sim N(0,1) \Rightarrow Z^2 \sim \chi^2_1$

$Z_1, \ldots, Z_n \overset{ind}{\sim} N(0,1) \Rightarrow Z_1^2 + \cdots + Z_n^2 \sim \chi^2_n$

```
nu1 <- c(1,2,3,5,5,5,50,50)
nu2 <- c(3,3,3,10,10,10,50,50)
phi <- c(0,0,0,0,4,8,0,4)
f <- seq(.01,4,length=200)
dfmat <- matrix(0,length(f),200)
for(j in 1:length(nu1)){

  dfmat[j,] <- df(f,df1 = nu1[j],df2=nu2[j],ncp=phi[j])

}
lab <- paste("(df1,df2,phi) = (",
      apply(cbind(nu1,nu2,phi),1,paste,collapse = ","),
      ")",sep="")
```

$Z \sim N(0,1)$, $W \sim \chi^2_\nu$ ("nu"), Z, W independent, then $T = \dfrac{Z}{\sqrt{W/\nu}} \sim t_\nu$

$W_1 \sim \chi^2_{\nu_1}$, $W_2 \sim \chi^2_{\nu_2}$, W_1, W_2 independent, then

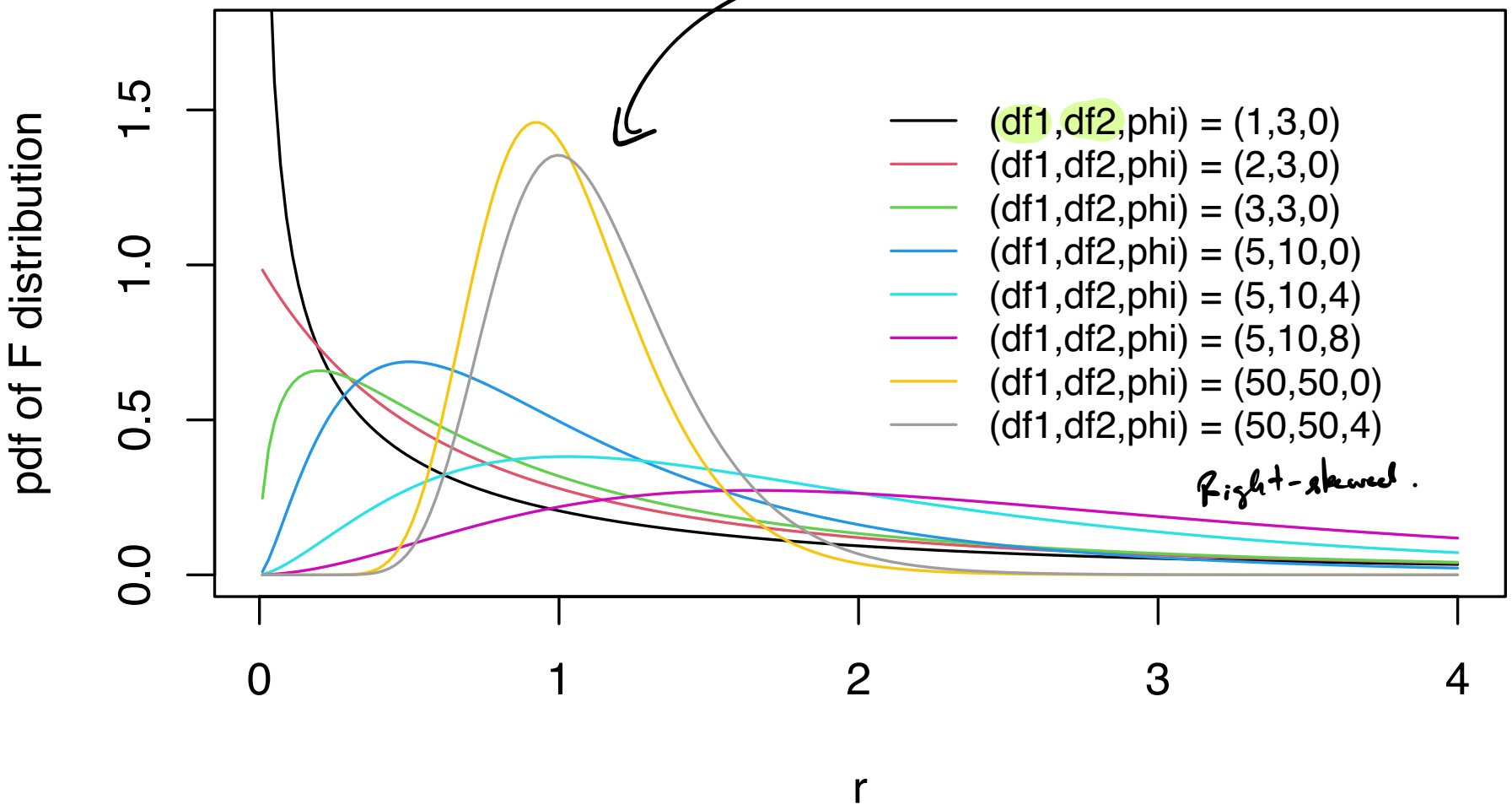
$$R = \frac{W_1/\nu_1}{W_2/\nu_2} \sim F_{\nu_1, \nu_2}$$

(ν_1: "numerator" df; ν_2: "denominator" df)

↳ "numerator" df

```
plot(NA,xlim = range(f),ylim = c(0,1.2*max(dfmat[-1,])),
     xlab = "r",
     ylab = "pdf of F distribution")
for(j in 1:length(nu1)) lines(dfmat[j,]~f, col = j)
legend(x = .5*max(f),y = 1.1*max(dfmat[-1,]),legend = lab,
       col = 1:length(nu1), lty = 1,bty = "n", cex = .8)
```

pdfs of F distributions.



Right-skewed.

$$Y_i = \beta_0 + \beta_1 x_{i1} + \cdots + \beta_p x_{ip} + \varepsilon_i$$

H_0: $\beta_1 = 0, \beta_2 = 0, \ldots, \beta_p = 0$

H_1: At least one β_j is not zero.

$$SS_{Reg} = \sum_{i=1}^{n} (\hat{y}_i - \bar{y}_n)^2 \overset{\text{Under } H_0}{\sim} \chi^2_p$$

$$SS_{Error} = \sum_{i=1}^{n} (y_i - \hat{y}_i)^2 \sim \chi^2_{n-(p+1)}$$

$\hat{y}_i = \hat{\beta}_0 + \hat{\beta}_1 x_{i1} + \cdots + \hat{\beta}_p x_{ip}$

Under H_0, $\hat{y}_i \approx \bar{y}_n$

$$F_{stat} = \frac{MS_{Reg}}{MS_{Error}}$$

Should be small under H_0

$$= \frac{\sum_{i=1}^{n} (\hat{y}_i - \bar{y}_n)^2 / p}{\sum_{i=1}^{n} (y_i - \hat{y}_i)^2 / (n-(p+1))}$$

Reject H_0 when this is large enough.

$$\rightarrow MS_{Reg} = \frac{SS_{Reg}}{p}$$

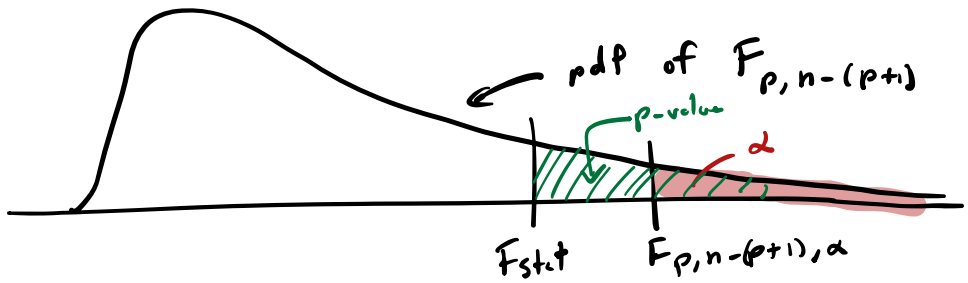
$$\rightarrow MS_{Error} = \frac{SS_{Error}}{n-(p+1)}$$

$$\rightarrow F_{stat} = \frac{MS_{Reg}}{MS_{Error}} \sim F_{p,\, n-(p+1)}$$

$\frac{\chi^2_p}{p}$ over MS_{Reg}

$\frac{\chi^2_{n-(p+1)}}{n-(p+1)}$ over MS_{Error}

So Reject H_0 when $F_{stat} > F_{p,n-(p+1),\alpha}$



Testing for significance of a subset of covariates

Full-Reduced Model F-test

We may wish to keep only a subset $M \subset \{1, \ldots, p\}$ of the covariates.

Before discarding the covariates in M^c we should test

$$H_0\text{: } \beta_j = 0 \text{ for all } j \in M^c.$$

We can test the above with the full-reduced model F-test:

- Let s be the number of covariates in M^c and compute

$$F_{\text{stat}} = \frac{(\text{SS}_{\text{Error}}(\text{Reduced}) - \text{SS}_{\text{Error}}(\text{Full}))/s}{\text{SS}_{\text{Error}}(\text{Full})/(n-(p+1))}$$

  where "Full" means from the model with all p covariates and "Reduced" means from the model with only the covariates in M.

- Reject H_0 if $F_{\text{stat}} > F_{s,n-(p+1),\alpha}$.
- Obtain p-value as $P(F > F_{\text{stat}})$, where $F \sim F_{s,n-(p+1)}$.

$$Y_i = \beta_0 + age_i \beta_{age} + optx_i \beta_{optx} + vac_i \beta_{vac} + sgift_i \beta_{sgift} + \varepsilon_i \quad \text{(Full)}$$

get rid of these?

$$Y_i = \beta_0 + age_i \beta_{age} + sgift_i \beta_{sgift} + \varepsilon_i \quad \text{(Reduced)}$$

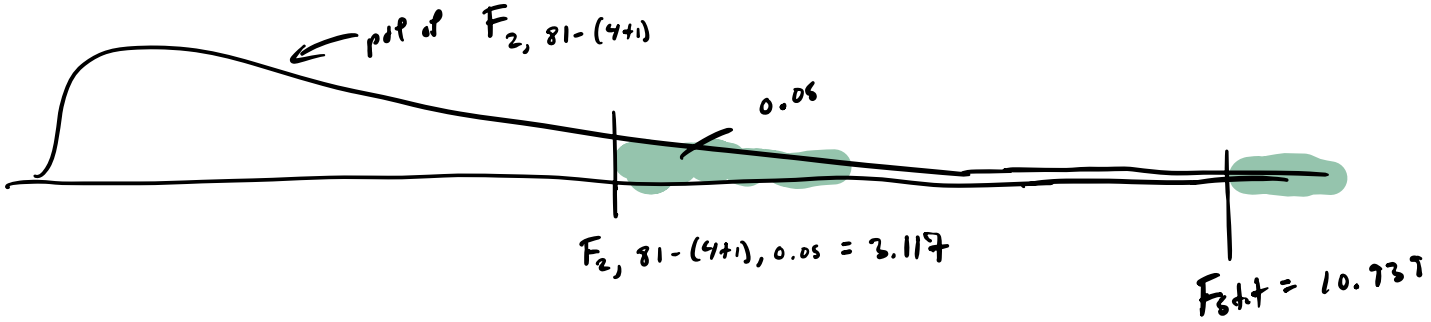
Rental rates of commercial properties example (cont)

Check whether vac and optx contribute significantly to the rent.

That is test H_0: $\beta_{\text{vac}} = 0$ and $\beta_{\text{optx}} = 0$.

```
lm_red <- lm(rent ~ age + sqft, data = commprop)
lm_full <- lm(rent ~ age + optx + vac + sqft, data = commprop)
SSE_red <- sum(lm_red$residuals^2)
SSE_full <- sum(lm_full$residuals^2)
s <- 2 # significance of two covariates being tested
Fstat <- (SSE_red - SSE_full)/s / ( SSE_full / (n - (p + 1)))
alpha <- 0.05
F_crit <- qf(1 - alpha,s,n-(p+1))
pval <- 1 - pf(Fstat, 2, n - (p+1))
```

We obtain $F_{\text{stat}} = 10.939$ and $F_{s,n-(p+1),0.05} = 3.117$, and the p-value is 0.



$\Rightarrow$ Reject H_0: $\beta_{Vac} = 0$ and $\beta_{Opt\times} = 0$.

The overall F-test

We may wish to test whether *any* covariates are important, that is

$$H_0\text{: } \beta_j = 0 \text{ for all } j = 1, \ldots, p.$$

H_1: at least one β_j is non zero.

- Compute on the full model the value

$$F_{\text{stat}} = \frac{\text{MS}_{\text{Reg}}}{\text{MS}_{\text{Error}}} \left(= \frac{\text{SS}_{\text{Reg}}/p}{\text{SS}_{\text{Error}}/(n-(p+1))} \right)$$

$\sim \chi^2_p$ ← num df

$\sim \chi^2_{n-(p+1)}$ ↑ den df

← Larger values carry stronger evidence against H_0.

- Reject H_0 if $F_{\text{stat}} > F_{p,n-(p+1),\alpha}$.
- The p-value is $P(F > F_{\text{stat}})$, where $F \sim F_{p,n-(p+1)}$.

This is called the overall F-test of significance.

This test statistic and p-value are reported by `summary()` on `lm()`.

```
lm_out <- lm(rent ~ age + optx + vac + sqft, data = cp)
summary(lm_out)
```

```
Call:
lm(formula = rent ~ age + optx + vac + sqft, data = cp)

Residuals:
    Min      1Q  Median      3Q     Max
-3.1872 -0.5911 -0.0910  0.5579  2.9441

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept) 12.20059  0.57796  21.110  < 2e-16 ***
age         -0.14203  0.02134  -6.655 3.89e-09 ***
optx         0.28202  0.06317   4.464 2.75e-05 ***
vac          0.61934  1.08681   0.570     0.57
sqft         0.07924  0.01385   5.722 1.98e-07 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 1.137 on 76 degrees of freedom
Multiple R-squared:  0.5847,    Adjusted R-squared:  0.5629
F-statistic: 26.76 on 4 and 76 DF,  p-value: 7.272e-14
```

$\hat{\beta}_0$ $\hat{\beta}_1$ $\hat{\beta}_p$ $\hat{\sigma}$

$T_{stat} = \dfrac{\hat{\beta}_1}{\hat{\sigma}/\sqrt{S_{xx}}}$

$2 \cdot P(T > |-6.655|)$

Overall F-test.

$F_{stat} = \dfrac{MS_{Reg}}{MS_{Error}}$

p $n-(p+1)$

$P(F > F_{stat}), \; F \sim F_{p,\,n-(p+1)}$

$\hat{\sigma}^2 = \dfrac{1}{n-(p+1)} \sum_{i=1}^{n} \hat{e}_i^2 = \dfrac{SS_{Error}}{n-(p+1)} = MS_{Error}$

$$R^2 = \frac{SS_{Reg}}{SS_{Total}} = \frac{\sum_{i=1}^{n} (\hat{y}_i - \bar{y}_n)^2}{\sum_{i=1}^{n} (y_i - \bar{y}_n)^2}$$

EXAM!!

$$SS_{Total} = SS_{Reg} + SS_{Error}$$

Exercise: Show that the test statistic of the overall F test can be written

$$F_{\text{stat}} = \frac{\text{MS}_{\text{Reg}}}{\text{MS}_{\text{Error}}} = \frac{(n-(p+1))}{p} \frac{R^2}{1-R^2},$$

Don't need to memorize.

where R^2 is the coefficient of determination.

$$F_{stat} = \frac{MS_{Reg}}{MS_{Error}} = \frac{SS_{Reg} / p}{SS_{Error} / (n-(p+1))}$$

$$= \frac{SS_{Reg} / p}{(SS_{Total} - SS_{Reg}) / (n-(p+1))}$$

$$= \frac{n-(p+1)}{p} \frac{SS_{Reg}}{SS_{Total} - SS_{Reg}}$$

$$= \frac{n-(p+1)}{p} \frac{SS_{Reg}/SS_{Total}}{SS_{Total}/SS_{Total} - SS_{Reg}/SS_{Total}}$$

$$= \frac{n-(p+1)}{p} \frac{R^2}{1 - R^2}$$

Exercise: Suppose you fit a regression model with 3 predictors on a data set with 81 observations, and you obtain $\hat{\sigma} = 1.132$ and $R^2 = 0.583$. Use this information to fill in the entire ANOVA table:

Source	Df	SS	MS	F value	p-value
Regression	p	SS_{Reg}	MS_{Reg}	F_{stat}	$P(F > F_{\text{stat}})$
Error	$n-(p+1)$	SS_{Error}	MS_{Error}		
Total	$n-1$	SS_{Tot}			

$n = 81$

$p = 3$

$R^2 = 0.583 = \frac{SS_{Reg}}{SS_{total}}$

$\hat{\sigma} = 1.132$

If you know one of these and you know R^2, then you can get the other two values.

This: $R^2 = \frac{SS_{Reg}}{SS_{total}}$

$\Leftrightarrow R^2 = \frac{SS_{Reg}}{SS_{Reg} + SS_{Error}}$

$\Leftrightarrow (SS_{Reg} + SS_{Error}) R^2 = SS_{Reg}$

$\Leftrightarrow SS_{Reg} R^2 + SS_{Error} R^2 = SS_{Reg}$

$\Leftrightarrow SS_{Reg} R^2 - SS_{Reg} = -SS_{Error} R^2$

$\Leftrightarrow SS_{Reg} (R^2 - 1) = -SS_{Error} R^2$

$\Leftrightarrow SS_{Reg} = \frac{-SS_{Error} R^2}{(R^2 - 1)} = SS_{Error} \frac{R^2}{1 - R^2}$

So I can write $SS_{Reg} = SS_{Error} \frac{R^2}{1 - R^2}$

Or this:

$R^2 = \frac{SS_{Reg}}{SS_{total}}$

$\Leftrightarrow R^2 = \frac{SS_{total} - SS_{Error}}{SS_{total}} = 1 - \frac{SS_{Error}}{SS_{total}}$

$\Leftrightarrow 1 - R^2 = \frac{SS_{Error}}{SS_{total}}$

$\Leftrightarrow SS_{total} = \frac{SS_{Error}}{1 - R^2}$

$$SS_{total} = \frac{(1.132)^2 \cdot (81 - (3+1))}{1 - 0.583}$$

$$SS_{Reg} = (1.132)^2 \cdot (81 - (3+1)) \cdot \frac{0.583}{1 - 0.583}$$

$$SS_{total} = SS_{Reg} + SS_{Error}$$

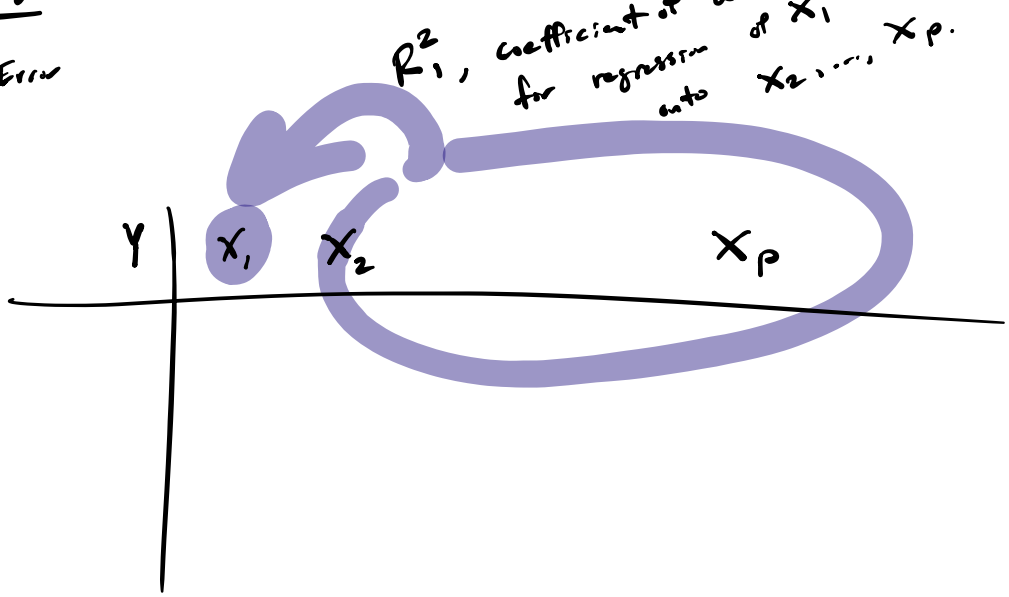
$$\left(MS_{Error} = \frac{SS_{Error}}{n - (p+1)} \right)$$

$$MS_{Error} = \hat{\sigma}^2 = (1.132)^2$$

$$SS_{Error} = MS_{Error} \cdot (n - (p+1)) = (1.132)^2 \cdot (81 - (3+1))$$

$$MS_{Reg} = \frac{SS_{Reg}}{p} = \frac{(1.132)^2 \cdot (81 - (3+1)) \cdot \frac{0.583}{1 - 0.583}}{3}$$

$$F_{stat} = \frac{MS_{Reg}}{MS_{Error}}$$



$$Y_i = \beta_0 + \beta_1 x_{i1} + \cdots + \beta_p x_{ip} + \varepsilon_i$$

Effect of correlations among the covariates

$\hat{\beta}_j \sim \text{Normal}\left(\beta_j, \frac{\sigma^2}{n}\Omega_{jj}\right)$

From before $\operatorname{Var}\hat{\beta}_j = \sigma^2\Omega_{jj}/n$. An alternate expression gives

large if x_j highly correlated with other covariates

$$\operatorname{Var}\hat{\beta}_j = \frac{1}{1-R_j^2}\frac{\sigma^2}{\sum_{i=1}^n (x_{ji}-\bar{x}_j)^2},$$

$\sum_{i=1}^n (x_{ji}-\bar{x}_j)^2 = S_{xx,j}$, spread of covariate j values

where R_j^2 is the R^2 from regressing x_j on the other covariates.

So multicollinearity of x_j with the other covariates "inflates" $\operatorname{Var}\hat{\beta}_j$:

- Makes confidence intervals for β_j wider.
- Makes tests of H_0: $\beta_j = 0$ vs H_1: $\beta_j \neq 0$ less powerful.

Call $\frac{1}{1-R_j^2}$ the variance inflation factor (VIF) for x_j, $j = 1, \ldots, p$.

VIFs in commercial properties example

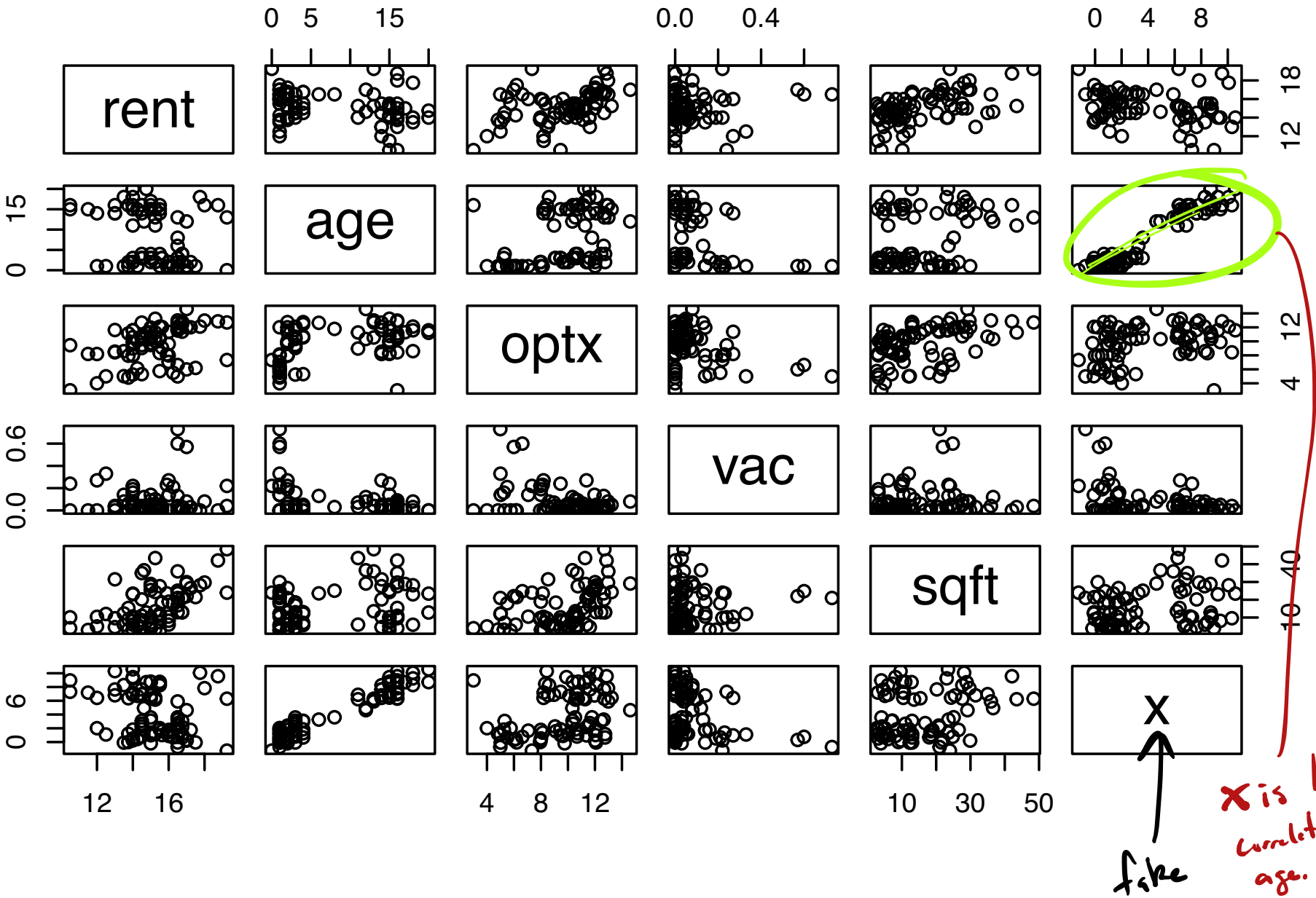
Add to the data set a spurious predictor highly correlated with age.

Check the effect of this on our inferences for β_{age}.

```
# make new x correlated with age
x <- .5*commprop$age + rnorm(n)
commpropx <- cbind(commprop,x)
round(cor(commpropx),4)
```

```
        rent     age    optx     vac   sqft       x
rent  1.0000 -0.2503  0.4138  0.0665 0.5353 -0.2588
age  -0.2503  1.0000  0.3888 -0.2527 0.2886  0.9580
optx  0.4138  0.3888  1.0000 -0.3798 0.4407  0.3412
vac   0.0665 -0.2527 -0.3798  1.0000 0.0806 -0.2968
sqft  0.5353  0.2886  0.4407  0.0806 1.0000  0.2407
x    -0.2588  0.9580  0.3412 -0.2968 0.2407  1.0000
```

```
plot(commpropx)
```



```r
lm_out <- lm(rent ~ age + optx + vac + sqft, data = commpropx)
confint(lm_out)
```

without x

```
                 2.5 %      97.5 %
(Intercept) 11.04948640 13.35168536
age         -0.18454113 -0.09952615
optx         0.15619789  0.40783517
vac         -1.54523184  2.78391885
sqft         0.05166283  0.10682321
```

C.I. for β_{age}, width: 0.085

```r
lmx_out <- lm(rent ~ age + optx + vac + sqft + x, data = commpropx)
confint(lmx_out)
```

with x.

```
                 2.5 %      97.5 %
(Intercept) 10.89425319 13.28353725
age         -0.33146367 -0.05048364
optx         0.16181151  0.41790047
vac         -1.42385089  3.04187798
sqft         0.05171573  0.10706511
x           -0.17248415  0.37129156
```

C.I. for β_{age}, width = 0.281

The width of the CI for β_{age} *was* 0.085.

With the new covariate the width of the CI for β_{age} becomes 0.281.

So including `x` in the model makes our estimation of β_{age} less accurate.

```
summary(lm_out)
```

```
Call:
lm(formula = rent ~ age + optx + vac + sqft, data = commpropx)

Residuals:
    Min      1Q  Median      3Q     Max
-3.1872 -0.5911 -0.0910  0.5579  2.9441

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept) 12.20059    0.57796  21.110  < 2e-16 ***
age         -0.14203    0.02134  -6.655 3.89e-09 ***
optx         0.28202    0.06317   4.464 2.75e-05 ***
vac          0.61934    1.08681   0.570     0.57
sqft         0.07924    0.01385   5.722 1.98e-07 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 1.137 on 76 degrees of freedom
Multiple R-squared:  0.5847,    Adjusted R-squared:  0.5629
F-statistic: 26.76 on 4 and 76 DF,  p-value: 7.272e-14
```

without x

p-value for overall F-test

P n-(p+1)

numerator / denom. degrees of Freedom.

$$F_{stat} = \frac{MS_{Reg}}{MS_{Error}}$$

The p-value for age is very small.

```
summary(lmx_out)
```

```
Call:
lm(formula = rent ~ age + optx + vac + sqft + x, data = commpropx)

Residuals:
    Min      1Q  Median      3Q     Max
-3.1884 -0.5502 -0.0471  0.6345  3.0769

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept) 12.08890    0.59969  20.159  < 2e-16 ***
age         -0.19097    0.07052  -2.708  0.00838 **
optx         0.28986    0.06428   4.510 2.36e-05 ***
vac          0.80901    1.12086   0.722  0.47267
sqft         0.07939    0.01389   5.715 2.10e-07 ***
x            0.09940    0.13648   0.728  0.46868
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 1.14 on 75 degrees of freedom
Multiple R-squared:  0.5877,    Adjusted R-squared:  0.5602
F-statistic: 21.38 on 5 and 75 DF,  p-value: 3.134e-13
```

include x

not as small as before.

The p-value for age is not nearly as small when x is included!

Getting VIFs with `vif()` from the car package

We can use the R package `car` from Fox and Weisberg (2019).

First time must install the package with `install.package("car")`.

```
library(car)
vif(lm_out)  without  x
```

```
     age      optx       vac      sqft
1.240348 1.648225 1.323552 1.412722
```

Some say:
Don't want VIF > 10.

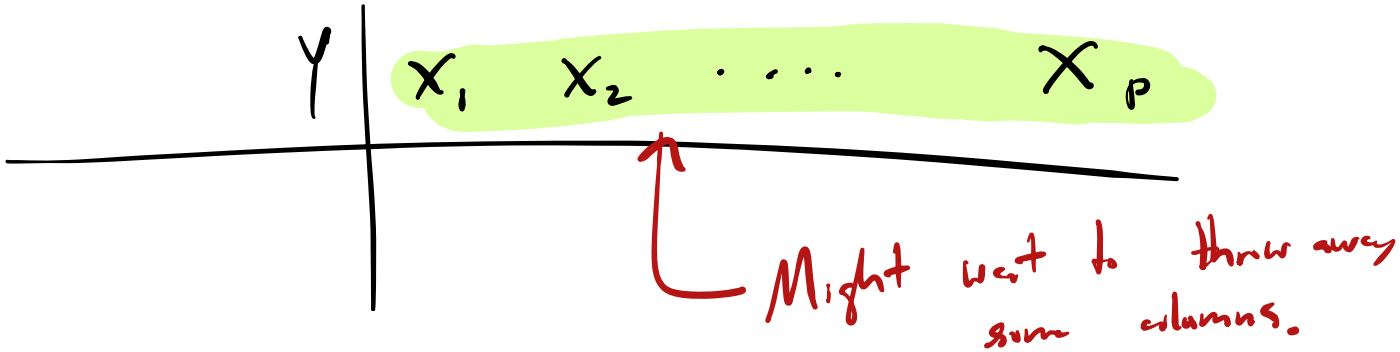
```
vif(lmx_out)  with  x
```

```
      age       optx        vac       sqft          x
13.459371  1.695779  1.399077  1.413022 13.158690
```

Note the change in VIF for age due to including `x` in the model!

End exam 1 material

Variable selection



Sometimes the number of potentially important predictors is quite large.

Large p tends to increase the VIFs, leading to low power.

So we may wish to discard some predictors. We briefly discuss:

- Forward and backward stepwise selection with AIC
- Best subset selection with Mallow's $C(p)$
- LASSO selection

And most importantly:

- The dangers of naïve post-selection inference!!

赤池弘次

aka ike

Introduced Akaike's Information Criterion (AIC).

Akaike's Information Criterion (AIC) for comparing models

$Y \mid X_1 \ldots X_p$

For example, compare these:

Model 1: $Y = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \varepsilon$

Model 2: $Y = \beta_1 + \beta_1 X_1 + \cdots + \beta_p X_p + \varepsilon$

For a given model, i.e. set of covariates, AIC is defined as

$$\text{AIC} = 2(p+1) - 2\underbrace{\ell(\hat{\sigma}^2, \hat{\beta}_0, \hat{\beta}_1, \ldots, \hat{\beta}_p)}_{\text{log-likelihood}}.$$

p = # predictors in model

The log-likelihood is the log of the joint pdf of the data (STAT 512).

AIC can be used to compare several models for the same data.

The "best" model is the one which minimizes AIC.

The extractAIC() function

The extractAIC function in R returns a modified version of AIC:

$$\text{AIC}^* = 2(p+1) + n\log(\text{SS}_{\text{Error}}/n)$$

```
lm_out <- lm(rent ~ age + optx + vac + sqft, data = commprop)
extractAIC(lm_out) # gives value p + 1 as well as AIC value
```

```
[1]  5.00000 25.62235
```

```
# compute it "manually"
p <- 4
2*(p+1) + n * log(sum(lm_out$residuals^2)/n)
```

```
[1] 25.62235
```

Comparing models using AIC

Continuing with the rental property data, compare the model with all the covariates to the model with only the covariate age.

```
lm_all <- lm(rent ~ age + optx + vac + sqft, data = commprop)
extractAIC(lm_all)
```

```
[1]  5.00000 25.62235
```

AIC

```
lm_age <- lm(rent ~ age, data = commprop)
extractAIC(lm_age)
```

```
[1]  2.00000 85.57119
```

AIC

The model with all the covariates has AIC value 25.622, while the model with only the covariate age has AIC value 85.571.

So according to AIC the model with all the covariates is superior.

Stepwise selection based on AIC

$Y \mid X_1 \quad X_2 \quad \ldots \quad X_{100}$

2^{100} different models possible.

Stepwise selection:

- Backward: Begin with all the predictors and remove one at a time.
- Forward: Begin with no predictors and add one at a time.

In each step remove/add predictor to get largest decrease in AIC.

If a decrease in AIC is not possible, stop.

Stepwise selection with commercial property data

```
# BACKWARD: fit model with all covariates
lm0 <- lm(rent ~ age + optx + vac + sqft, data = commprop)
extractAIC(lm0)[2]
```

```
[1] 25.62235
```

```
# find which variable to remove
lm1_1 <- lm(rent ~ optx + vac + sqft, data = commprop)
lm1_2 <- lm(rent ~ age + vac + sqft, data = commprop)
lm1_3 <- lm(rent ~ age + optx + sqft, data = commprop)
lm1_4 <- lm(rent ~ age + optx + vac, data = commprop)

c(extractAIC(lm1_1)[2],extractAIC(lm1_2)[2],extractAIC(lm1_3)[2],extractAIC(lm1_4)[2])
```

```
[1] 60.81404 42.48560 23.96773 52.64345
```

```
# removing vac decreases AIC the most. Now find next variable to remove:

lm2_1 <- lm(rent ~ optx + sqft, data = commprop)
lm2_2 <- lm(rent ~ age + sqft, data = commprop)
lm2_3 <- lm(rent ~ age + optx, data = commprop)

c(extractAIC(lm2_1)[2],extractAIC(lm2_2)[2],extractAIC(lm2_3)[2])
```

```
[1] 60.88082 42.11426 55.33500
```

```
# could not decrease the AIC, so stick with the model: rent ~ sqft + age + vac
```

```r
# FORWARD: fit model with no covariates (intercept only)
lm0
```

```
Call:
lm(formula = rent ~ age + optx + vac + sqft, data = commprop)

Coefficients:
(Intercept)          age         optx          vac         sqft
    12.2059      -0.1420       0.2820       0.6193       0.0792
```

```r
extractAIC(lm0)[2]
```

```
[1] 25.62235
```

$y = 1 + x_1 + 3x_2 + (0)x_6 + \varepsilon$

$+ x_4 - 3x_5$

```r
# find which variable to add
lm1_1 <- lm(rent ~ age, data = commprop)
lm1_2 <- lm(rent ~ optx, data = commprop)
lm1_3 <- lm(rent ~ vac, data = commprop)
lm1_4 <- lm(rent ~ sqft, data = commprop)

c(extractAIC(lm1_1)[2],extractAIC(lm1_2)[2],extractAIC(lm1_3)[2],extractAIC(lm1_4)[2])
```

```
[1] 85.57119 75.59929 90.45183 63.46705
```

```r
# adding sqft decreases AIC the most. Now find next variable to add:
lm2_1 <- lm(rent ~ sqft + age, data = commprop)
lm2_2 <- lm(rent ~ sqft + optx, data = commprop)
lm2_3 <- lm(rent ~ sqft + vac, data = commprop)

c(extractAIC(lm2_1)[2],extractAIC(lm2_2)[2],extractAIC(lm2_3)[2])
```

```
[1] 42.11426 60.88082 65.40458
```

```r
# adding age decreases AIC the most.
lm3_1 <- lm(rent ~ sqft + age + optx, data = commprop)
lm3_2 <- lm(rent ~ sqft + age + vac, data = commprop)

c(extractAIC(lm3_1)[2],extractAIC(lm3_2)[2])
```

The `step()` function

Use `step()` function to do forward and backward stepwise selection.

```
# first fit two models: intercept-only and full
lm_intercept <- lm(rent ~ 1, data = commprop)
lm_all <- lm(rent ~ vac + age + optx + sqft, data = commprop)

# backward selection
step_back <- step(lm_all,
                  direction = "backward",
                  scope = formula(lm_all),
                  trace = 0) # suppress printed output

# forward selection
step_forw <- step(lm_intercept,
                  direction = "forward",
                  scope = formula(lm_all),
                  trace = 0) # suppress printed output
```

Set `trace = 1` to see output for each step.

Forward and backward stepwise chose the same model for these data:

```
step_back
```

```
Call:
lm(formula = rent ~ age + optx + sqft, data = commprop)

Coefficients:
(Intercept)          age         optx         sqft
   12.37058     -0.14416      0.26717      0.08178
```

```
step_forw
```

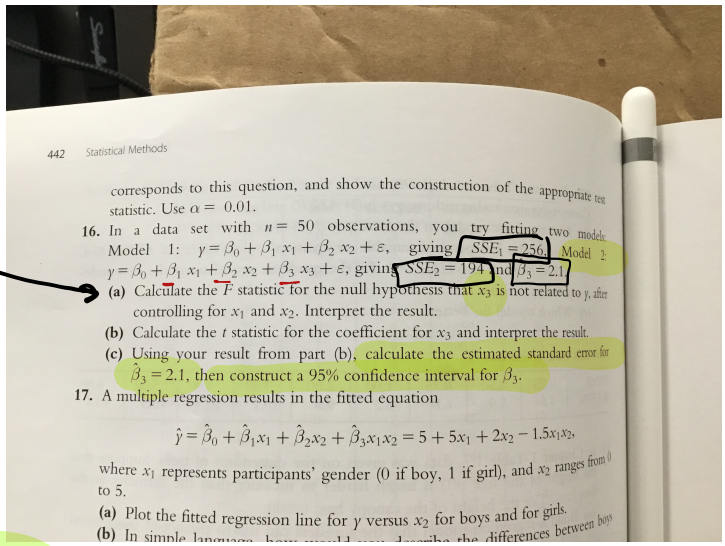
```
Call:
lm(formula = rent ~ sqft + age + optx, data = commprop)

Coefficients:
(Intercept)         sqft          age         optx
   12.37058      0.08178     -0.14416      0.26717
```

But this need not be the case.

Ch. 8
#16

a)

Full-reduced



442 Statistical Methods

corresponds to this question, and show the construction of the appropriate test statistic. Use $\alpha = 0.01$.

16. In a data set with $n = 50$ observations, you try fitting two models: Model 1: $y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \varepsilon$, giving $SSE_1 = 256$. Model 2: $y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \beta_3 x_3 + \varepsilon$, giving $SSE_2 = 194$ and $\hat{\beta}_3 = 2.1$.

(a) Calculate the F statistic for the null hypothesis that x_3 is not related to y, after controlling for x_1 and x_2. Interpret the result.

(b) Calculate the t statistic for the coefficient for x_3 and interpret the result.

(c) Using your result from part (b), calculate the estimated standard error for $\hat{\beta}_3 = 2.1$, then construct a 95% confidence interval for β_3.

17. A multiple regression results in the fitted equation

$$\hat{y} = \hat{\beta}_0 + \hat{\beta}_1 x_1 + \hat{\beta}_2 x_2 + \hat{\beta}_3 x_1 x_2 = 5 + 5x_1 + 2x_2 - 1.5x_1 x_2,$$

where x_1 represents participants' gender (0 if boy, 1 if girl), and x_2 ranges from 0 to 5.

(a) Plot the fitted regression line for y versus x_2 for boys and for girls.

(b) In simple language, how would you describe the differences between boys

$$F_{stat} = \frac{(SSE(Red) - SSE(full)) / \#\text{ variables removing}}{SSE(full) / (n - (p+1))}$$

SSE_1 SSE_2 SSE_2 1 5 3

Reject H_0 if $F_{stat} > F_{1,\, n-(p+1),\, 0.05}$

$qf(.95,\ 1,\ n-(p+1))$

b) $T_{stat} = \dfrac{\hat{\beta}_3}{\hat{\sigma}\sqrt{\mathcal{L}_{33}}}$ ← not given

$(T_{stat})^2 = F_{stat}$, since we are removing only one predictor.

c) $T_{stat} = \dfrac{\hat{\beta}_3}{\hat{se}\{\hat{\beta}_3\}}$

known ↑ solve for this

$\Rightarrow$ $\hat{\beta}_3 \pm t_{n-(p+1),\alpha/2}\ \hat{se}\{\hat{\beta}_3\}$

Best subset selection with Mallow's C_p

Given q available covariates, there are 2^q possible subset models (why?).

Mallow's C_p can be used to compare subset models: Let

$$C_p = (n - (p+1)) \left[\frac{\mathrm{MS}_{\mathrm{Error}}(\mathrm{subset})}{\mathrm{MS}_{\mathrm{Error}}(\mathrm{all})} - 1 \right] + (p+1),$$

where

- p is the number of predictors *in the subset model*.
- $\mathrm{MS}_{\mathrm{Error}}(\mathrm{subset})$ is the $\mathrm{MS}_{\mathrm{Error}}$ of the subset model.
- $\mathrm{MS}_{\mathrm{Error}}(\mathrm{all})$ is the $\mathrm{MS}_{\mathrm{Error}}$ of the model with all the covariates.

If the subset model is adequate, $\mathrm{MS}_{\mathrm{Error}}(\mathrm{subset})$ estimates the same target as $\mathrm{MS}_{\mathrm{Error}}(\mathrm{all})$, so the first term should be small and $C_p \approx p + 1$.

Can look at C_p values for all subset models of each size $p = 0, 1, 2, \ldots, q$

Want smallest model such that $C_p \approx p + 1$.

skip

Mallow's C_p on the rental properties data

Compute Mallow's C_p for a single subset model:

```
lm_all <- lm(rent ~ vac + age + optx + sqft, data = commprop)
lm_sub <- lm(rent ~ age + sqft, data = commprop)
MSE_sub <- sum(lm_sub$residuals^2) / (n - 3)
MSE_all <- sum(lm_all$residuals^2) / (n - 5)
Csub <- (MSE_sub / MSE_all - 1)*(n - 3) + 3
Csub
```

```
[1] 22.87781
```

This value is too large; the subset is not a good one.

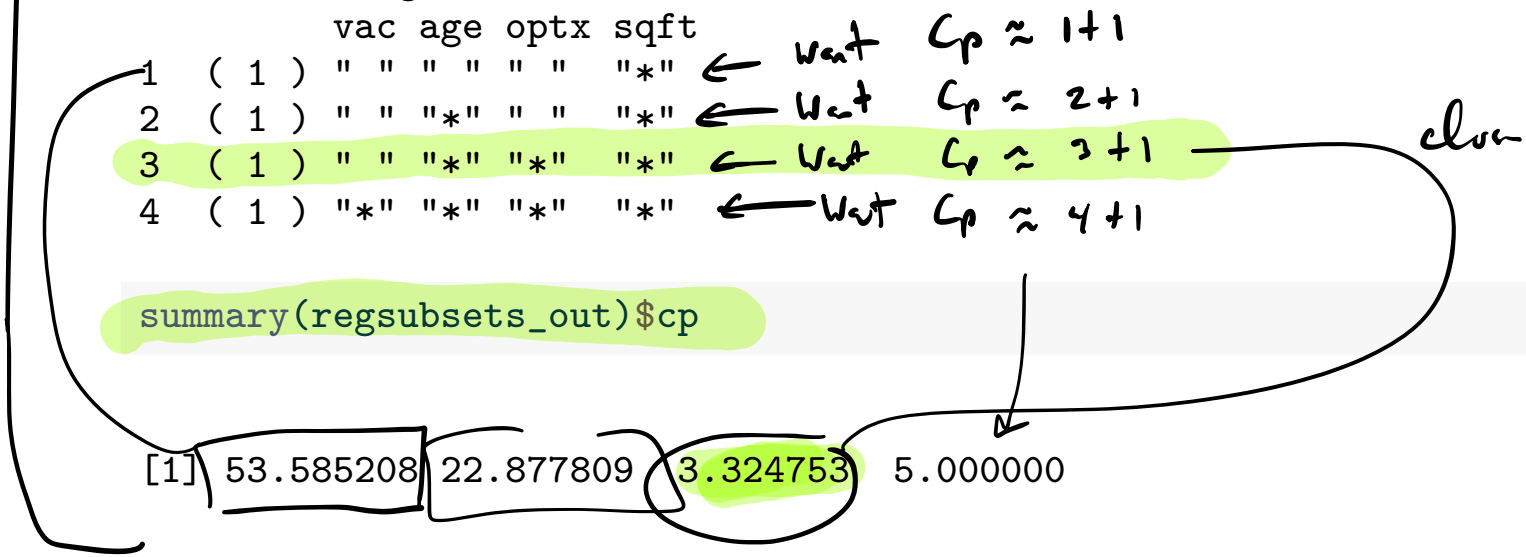
$Y \mid X_1 \; X_2 \; X_3 \; X_4$

$2^4 = 16$ possible models

The regsubsets() function from the R package leaps

```
library(leaps) # first time run install.packages("leaps")
regsubsets_out <- regsubsets(rent ~ vac + age + optx + sqft, data = commprop)
summary(regsubsets_out)
```

```
Subset selection object
Call: regsubsets.formula(rent ~ vac + age + optx + sqft, data = commprop)
4 Variables  (and intercept)
     Forced in Forced out
vac      FALSE      FALSE
age      FALSE      FALSE
optx     FALSE      FALSE
sqft     FALSE      FALSE
1 subsets of each size up to 4
Selection Algorithm: exhaustive
         vac age optx sqft
1  ( 1 ) " " " " " "  "*"
2  ( 1 ) " " "*" " "  "*"
3  ( 1 ) " " "*" "*"  "*"
4  ( 1 ) "*" "*" "*"  "*"
```




```
summary(regsubsets_out)$cp
```

```
[1] 53.585208 22.877809  3.324753  5.000000
```

LASSO selection

For $\lambda > 0$, define the LASSO objective function as

$$Q_\lambda(b_0, b_1, \dots, b_p) = \sum_{i=1}^{n} (Y_i - (b_0 + b_1 x_{i1} + \cdots + b_p x_{ip}))^2 + \lambda \sum_{j=1}^{p} |b_j|$$

Denote by $\tilde{\beta}_0, \tilde{\beta}_1, \dots, \tilde{\beta}_p$ the values of $b_0, b_1, \dots, b_p$ which minimize Q_λ.

Then $\tilde{\beta}_0, \tilde{\beta}_1, \dots, \tilde{\beta}_p$ are the LASSO estimators of $\beta_0, \beta_1, \dots, \beta_p$.

The penalty $\lambda \sum_{j=1}^{p} |b_j|$ can cause $\tilde{\beta}_j = 0$ for some j.

So LASSO performs variable selection and estimation simultaneously.

Nice, but trouble is: Hard to build CIs based on $\tilde{\beta}_0, \tilde{\beta}_1, \dots, \tilde{\beta}_p$.

LASSO on the commercial properties data

Use `cv.ncvreg()` function from R package `ncvreg`.

Runs crossvalidation to choose the best value of λ.

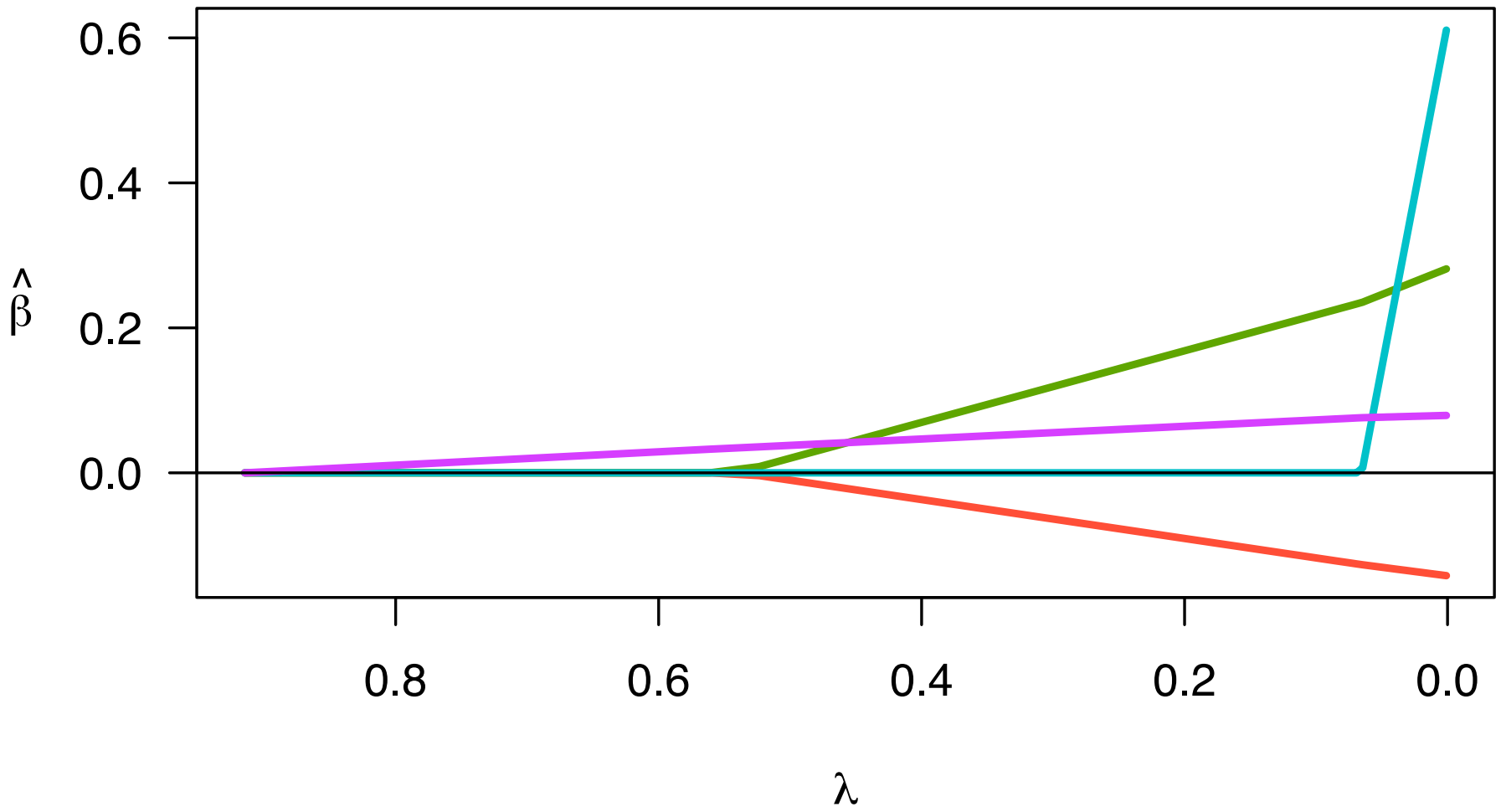
```
library(ncvreg) # first time run install.packages("ncvreg")

# prepare response vector and design matrix
y <- commprop$rent
X <- cbind(commprop$age,commprop$optx,commprop$vac,commprop$sqft)

# crossvalidation to choose lambda
lasso <- cv.ncvreg(X,y,penalty = "lasso")
lasso$lambda[lasso$min] # the chosen value of lambda
```

```
[1] 0.04881635
```

```
lasso$fit$beta[,lasso$min] # estimates under the "best" lambda
```

```
(Intercept)          V1          V2          V3          V4
12.52706502 -0.13053297  0.24680823  0.15642618  0.07685691
```

```
plot(lasso$fit)
```



optional

```
lasso$lambda[25] # a larger value of lambda
```

```
[1] 0.1714036
```

```
lasso$fit$beta[,25] # the estimates under the larger value of lambda
```

```
(Intercept)          V1          V2          V3          V4
13.07091918 -0.09820832  0.18250805  0.00000000  0.06674422
```

The dangers of post-selection inference

$Y \mid X_1 \; X_2 \; \cdots \; X_{50}$

Double-dipping

It is dangerous to:

1. Ask the data what hypotheses to test (what model to build).
2. Use afterwards the same data to perform inference (get p values).

Illustration:

Add 50 spurious predictors to the commercial properties data.

See how many we find to be significant.

```
X <- matrix(rnorm(n*50),n,50)
colnames(X) <- paste("x",1:50,sep="")
commpropX <- cbind(commprop,X)

lmX_out <- lm(rent ~ ., data = commpropX)
```

unrelated to rent.

rent ~ vac + age + sqft + optx + $X_1 + \cdots + X_{50}$

```
summary(lmX_out)
```

```
Call:
lm(formula = rent ~ ., data = commpropX)

Residuals:
     Min       1Q   Median       3Q      Max
-1.59062 -0.28456  0.05265  0.37467  1.32721

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept) 12.240240   0.804196  15.220 1.83e-14 ***
age         -0.159799   0.032199  -4.963 3.71e-05 ***
optx         0.306053   0.088151   3.472 0.001821 **
vac         -1.019504   1.626735  -0.627 0.536309
sqft         0.075606   0.019462   3.885 0.000631 ***
x1           0.081065   0.202076   0.401 0.691580
x2          -0.013859   0.215785  -0.064 0.949282
x3           0.353769   0.156937   2.254 0.032835 *
x4          -0.246326   0.214493  -1.148 0.261255
x5           0.094743   0.217846   0.435 0.667219
x6          -0.255471   0.179398  -1.424 0.166325
x7           0.044972   0.249455   0.180 0.858330
x8          -0.093089   0.173642  -0.536 0.596451
x9          -0.173610   0.200781  -0.865 0.395125
x10         -0.456824   0.171506  -2.664 0.013095 *
x11         -0.198532   0.182312  -1.089 0.286157
x12          0.167599   0.225407   0.744 0.463821
x13         -0.042958   0.158114  -0.272 0.788006
x14         -0.149825   0.157784  -0.950 0.351082
x15          0.044798   0.206143   0.217 0.829660
x16         -0.085366   0.179704  -0.475 0.638728
x17          0.409642   0.198006   2.069 0.048639 *
x18         -0.014995   0.168287  -0.089 0.929681
x19          0.310235   0.233058   1.331 0.194696
x20         -0.095293   0.177272  -0.538 0.595460
x21         -0.241792   0.201412  -1.200 0.240774
x22         -0.187829   0.178686  -1.051 0.302854
x23         -0.057653   0.150114  -0.384 0.704054
x24          0.015410   0.160248   0.096 0.924129
x25          0.045967   0.212807   0.216 0.830669
x26         -0.163131   0.206006  -0.792 0.435601
x27          0.298277   0.166657   1.790 0.085146 .
```

$H_0: \beta_j = 0$ vs $H_1: \beta_j \neq 0$

$x_1, \ldots, x_{50}$ have nothing to do with response.

We reject H_0: $\beta_j = 0$ at $\alpha = 0.05$ for 3 of the spurious predictors.

So the Type I error rate was $3/50 = 0.06$.

Now do backwards stepwise selection to throw some variables away.

Then see how many of the spurious predictors we find "significant".

```r
stepX_out <- step(lmX_out, data = commpropX, trace = 0)
summary(stepX_out)
```

```
Call:
lm(formula = rent ~ age + optx + vac + sqft + x3 + x4 + x6 +
    x9 + x10 + x11 + x14 + x17 + x19 + x21 + x22 + x27 + x30 +
    x33 + x35 + x38 + x39 + x40 + x43 + x45, data = commpropX)

Residuals:
     Min       1Q   Median       3Q      Max
-1.56212 -0.38254  0.00396  0.45158  1.45098

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept) 12.30833    0.45134  27.271  < 2e-16 ***
age         -0.14311    0.01677  -8.535 1.03e-11 ***
optx         0.28698    0.04891   5.868 2.49e-07 ***
vac         -1.20706    0.88517  -1.364 0.178134
sqft         0.07511    0.01105   6.800 7.41e-09 ***
x3           0.34486    0.10129   3.405 0.001231 **
x4          -0.15437    0.10145  -1.522 0.133739
x6          -0.19038    0.09232  -2.062 0.043839 *
x9          -0.22542    0.11113  -2.029 0.047268 *
x10         -0.38397    0.09581  -4.008 0.000183 ***
x11         -0.29839    0.10528  -2.834 0.006377 **
x14         -0.17005    0.09444  -1.801 0.077133 .
x17          0.41548    0.10696   3.885 0.000273 ***
x19          0.32996    0.11166   2.955 0.004567 **
x21         -0.15718    0.10393  -1.512 0.136084
x22         -0.15554    0.10417  -1.493 0.141020
x27          0.26847    0.08775   3.060 0.003398 **
x30         -0.31647    0.10519  -3.009 0.003927 **
x33         -0.15086    0.09853  -1.531 0.131348
x35         -0.20031    0.10194  -1.965 0.054391 .
x38          0.16086    0.10969   1.466 0.148108
x39          0.12000    0.09388   1.278 0.206457
x40          0.20019    0.10926   1.832 0.072230 .
x43          0.16494    0.10597   1.557 0.125213
x45          0.20619    0.08264   2.495 0.015574 *
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.7825 on 56 degrees of freedom
```

Reject $H_0: \beta_j = 0$ for 10 of these

Keep these, even though they are unrelated to rent.

Backwards stepwise selection keeps 20 of the 50 spurios predictors.

Among these 20, we reject H_0: $\beta_j = 0$ at $\alpha = 0.05$ for 10 of them.

So the post-selection Type I error rate was $10/20 = 0.5$ 😱.

WARNING: Selecting variables and then getting p-values in the selected model often leads to astonishingly inflated Type I error rates.

References

Fox, John, and Sanford Weisberg. 2019. *An R Companion to Applied Regression*. Third. Thousand Oaks CA: Sage. https://socialsciences.mcmaster.ca/jfox/Books/Companion/.

Kutner, Michael H, Christopher J Nachtsheim, John Neter, and William Li. 2005. *Applied Linear Statistical Models*. McGraw-hill.