

Simple linear regression : $\underline{y_i} = \beta_0 + \beta_1 \underline{x_i} + \varepsilon_i$, $\varepsilon_i \sim N(0, \sigma^2)$
(SLR)

Multiple linear regression : $y_i = \beta_0 + \beta_1 x_{1i} + \beta_2 x_{2i} + \dots + \beta_p x_{pi} + \varepsilon_i$
(MLR) $\varepsilon_i \sim N(0, \sigma^2)$

STAT 516 Lec 03

Multiple linear regression (part 1/2)

Karl Gregory

2026-09-02

SC apartments example

These data are a subset of “Apartment for Rent Classified” (2019).

```
link <- url("https://gregorkb.github.io/data/scapts.csv")
scapts <- read.csv(link)
head(scapts)
```

$\log(\text{price})$

	price	nbath	nbed	sqft	city	pets
1	1199	2	3	1200	Columbia	yes
2	1199	2	3	1450	Columbia	yes
3	1299	2	3	1950	Columbia	yes
4	1175	2	3	1312	Columbia	yes
5	1349	2	4	2015	Columbia	yes
6	1225	2	3	1336	Columbia	yes

```
n <- nrow(scapts)
```

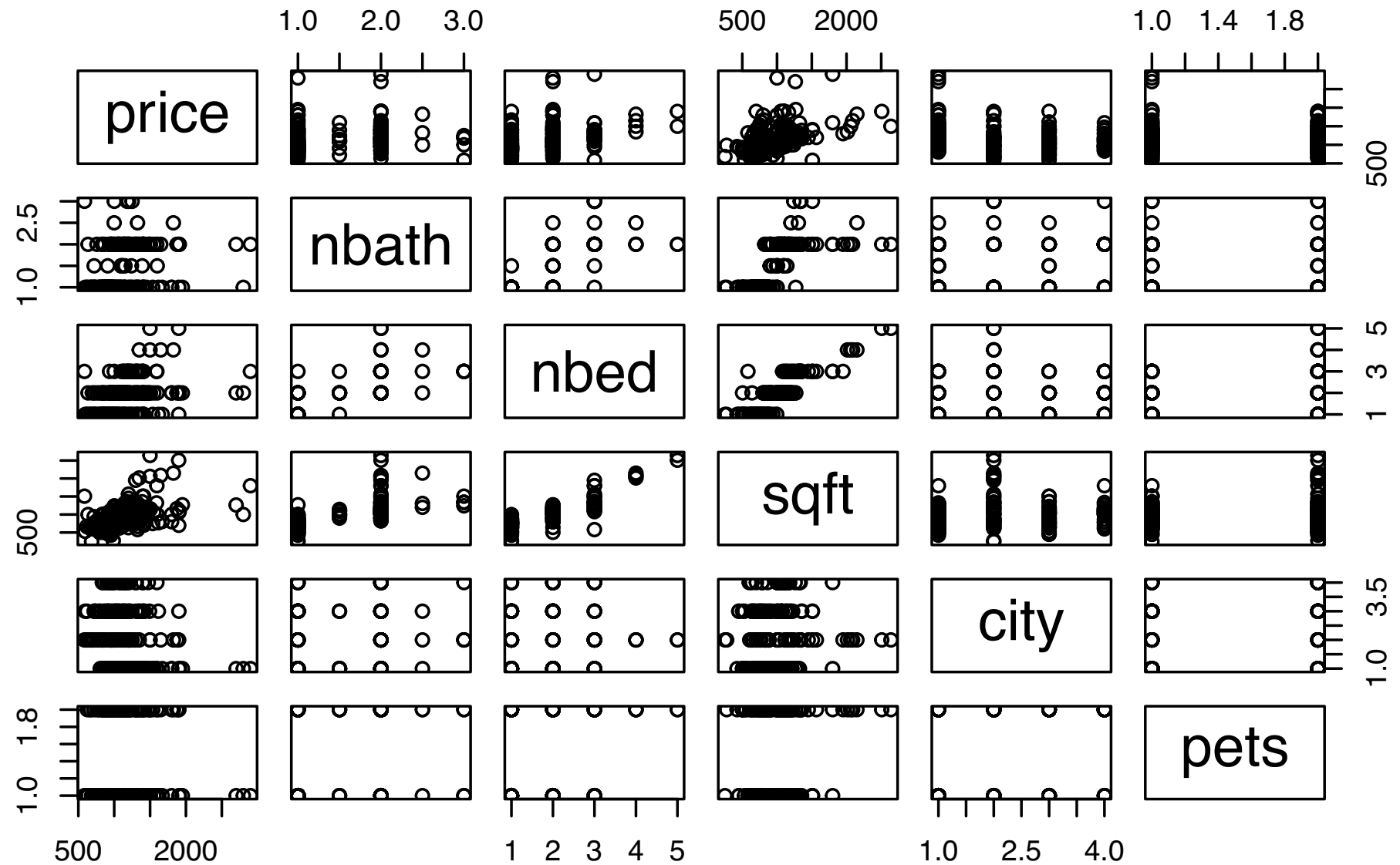
$X =$

nbath	nbed	log(sqft)	Col.	Grn	Rde	pets
2	3	...				
2	3	...				
2	3	...				
2	3	...				
2	4	...				
2	3	...				
...						

There are $n = 214$ data points.

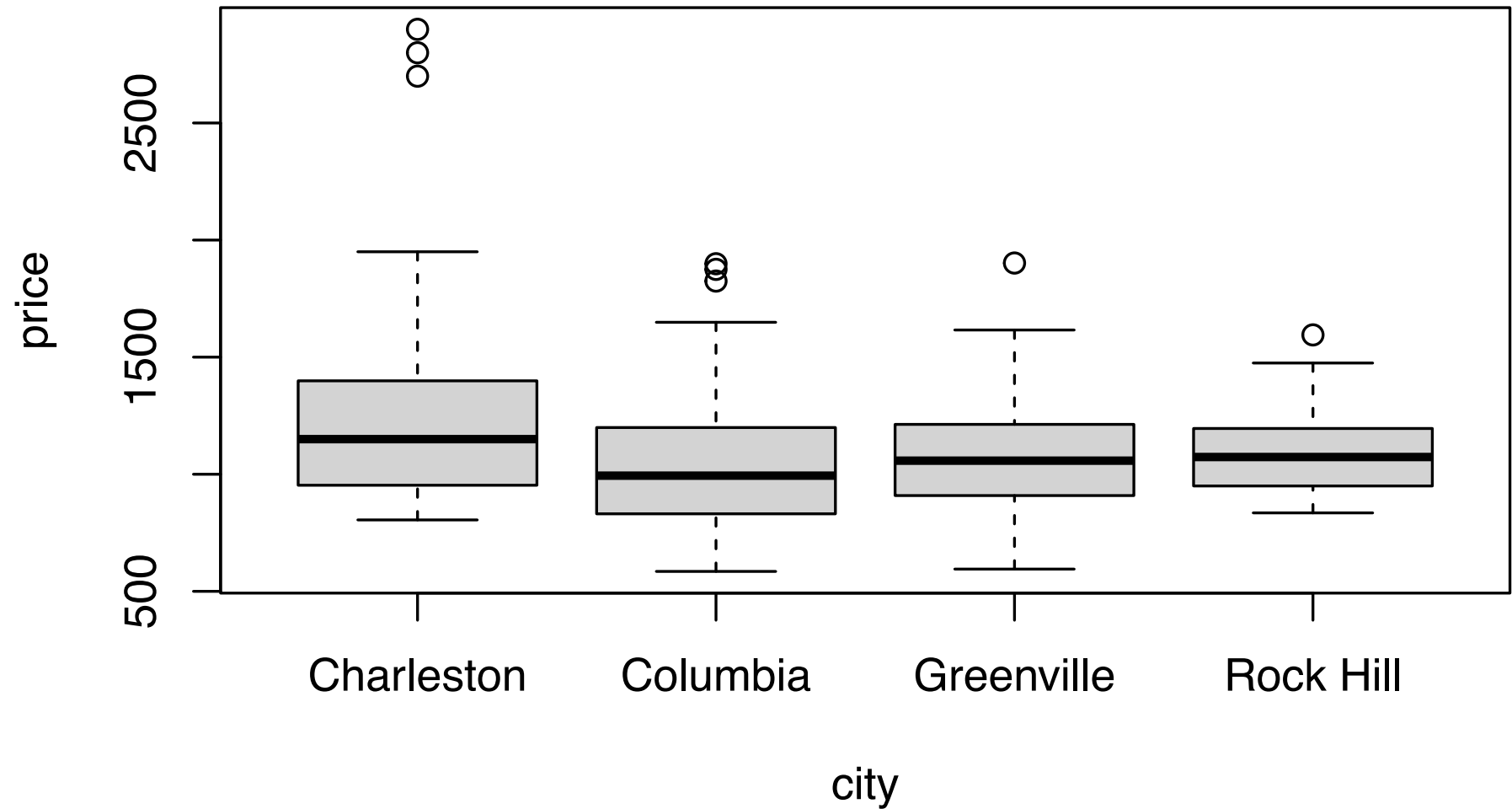
$x_{\text{pets}} = \begin{cases} 1 & \text{yes} \\ 0 & \text{no} \end{cases}$

```
plot(scaps)
```



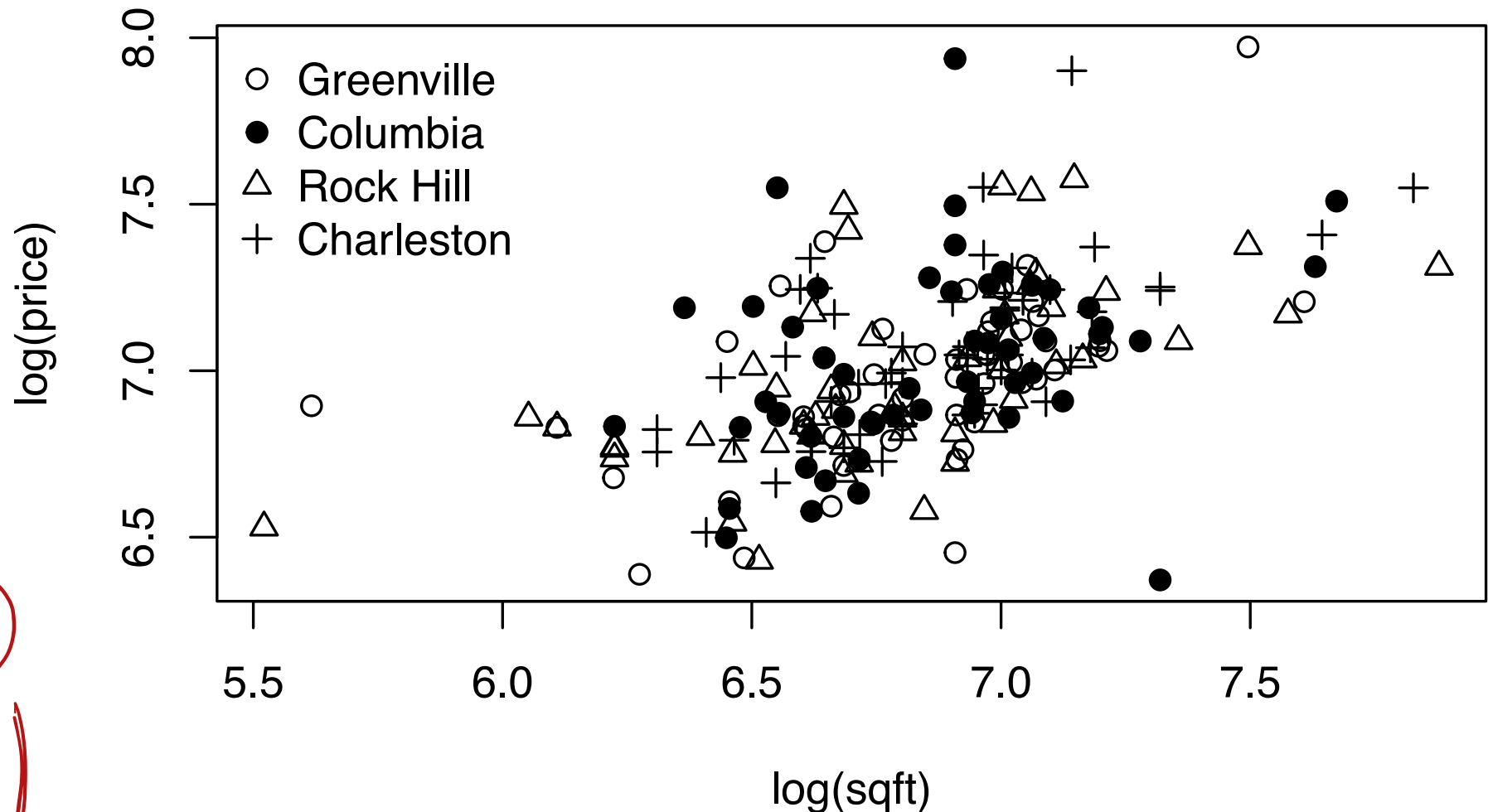
SC apartments (cont)

```
boxplot(price ~ city, data = scapts)
```



SC apartments (cont)

```
symbols <- c("Greenville" = 1, "Columbia" = 19, "Rock Hill" = 2, "Charleston" = 3)
plot(log(price) ~ log(sqft), data = scapts, pch = symbols)
legend("topleft", legend = names(symbols), pch = symbols, bty = "n")
```



Setup

$$\underset{\sim}{\mathbf{x}}_1 = \begin{bmatrix} x_{11} \\ \vdots \\ x_{1p} \end{bmatrix} \dots \underset{\sim}{\mathbf{x}}_n = \begin{bmatrix} x_{n1} \\ \vdots \\ x_{np} \end{bmatrix} \quad \underset{\sim}{\mathbf{x}}_i = \begin{bmatrix} x_{i1} \\ \vdots \\ x_{ip} \end{bmatrix}$$

"in" $\mathbb{R}^p$
real number

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

The multiple linear regression model is

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

where

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

Goals in multiple linear regression

As in *simple* linear regression, will learn how to

1. Estimate the regression coefficients β_0 and $\beta_1, \dots, \beta_p$.
2. Estimate the error term variance σ^2 .
3. Perform inference on $\beta_1, \dots, \beta_p$.
4. Build a CI for $\beta_0 + \beta_1 x_{\text{new},1} + \dots + \beta_p x_{\text{new},p}$ 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.

8.5 Interactions

Beyond the above, in *multiple* linear regression we wish to

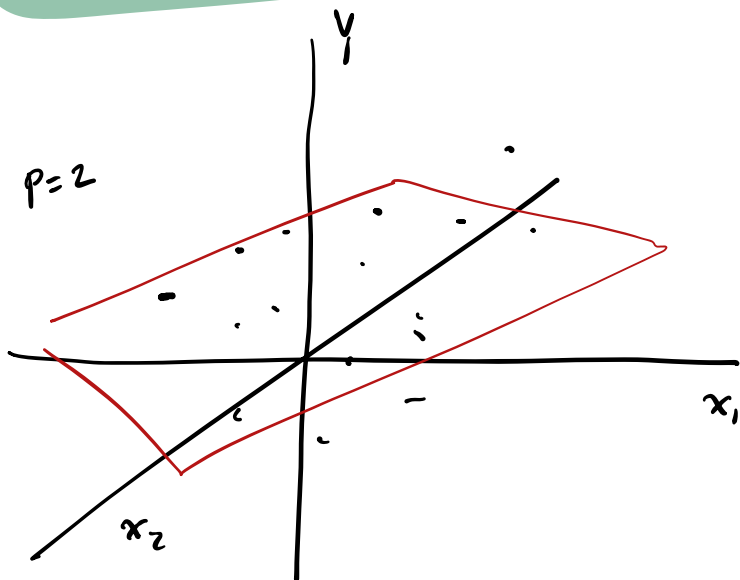
9. Test for significance of a subset of covariates
10. Understand how correlations among the covariates affect inferences
11. Do variable selection

Latter goals considered in part 2/2.

Estimating $\beta_0, \beta_1, \dots, \beta_p$:

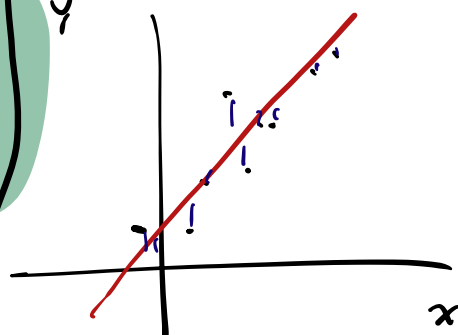
Let

$$Q(b_0, b_1, \dots, b_p) = \sum_{i=1}^n \left(y_i - (b_0 + b_1 x_{i1} + \dots + b_p x_{ip}) \right)^2$$



Recall: In SLR

$$Q(b_0, b_1) = \sum_{i=1}^n \left(y_i - (b_0 + b_1 x_i) \right)^2$$



$p=1$

$$\hat{\beta}_1 = \frac{\sum_{i=1}^n (x_i - \bar{x}_n)(y_i - \bar{y}_n)}{\sum_{i=1}^n (x_i - \bar{x}_n)^2}$$

$$\hat{\beta}_0 = \bar{y}_n - \hat{\beta}_1 \bar{x}_n$$

Estimating σ^2 :

Just as in SLR, get

Fitted values: $\hat{y}_i = \hat{\beta}_0 + \hat{\beta}_1 x_{i1} + \dots + \hat{\beta}_p x_{ip}, \quad i=1, \dots, n$

Residuals: $\hat{\varepsilon}_i = y_i - \hat{y}_i, \quad i=1, \dots, n.$

$$\hat{\sigma}^2 = \frac{1}{n-(p+1)} \sum_{i=1}^n \hat{\varepsilon}_i^2$$

$$\left(\text{SLR: } \hat{\sigma}^2 = \frac{1}{\underset{\uparrow}{n-2}} \sum_{i=1}^n \hat{\varepsilon}_i^2 \right)$$

Inference for $\beta_1, \dots, \beta_p$.

Confidence interval for β_j :

Turns out
$$\hat{\beta}_j \sim N\left(\beta_j, \sigma^2 \Omega_{jj}\right)$$

So

Ω_{jj} is the diagonal entry of $(X^T X)^{-1}$ corresponding to $\hat{\beta}_j$.

$$\frac{\hat{\beta}_j - \beta_j}{\hat{\sigma} \sqrt{\Omega_{jj}}} \sim t_{n-(p+1)}$$

This gives a $(1-\alpha) \times 100\%$ C.I. for β_j as:

$$\hat{\beta}_j \pm t_{n-(p+1), \alpha/2} \hat{\sigma} \sqrt{\Omega_{jj}}.$$

Use `confint()`.

Least-squares estimation of regression coefficients

Define the squared error criterion as

$$Q(b_0, b_1, \dots, b_p) = \sum_{i=1}^n (Y_i - (b_0 + b_1 x_{i1} + \dots + b_p x_{ip}))^2.$$

Suppose $Q(b_0, b_1, \dots, b_p)$ is uniquely minimized at $(\hat{\beta}_0, \hat{\beta}_1, \dots, \hat{\beta}_p)$.

Then we call $\hat{\beta}_0, \hat{\beta}_1, \dots, \hat{\beta}_p$ the least-squares estimators of $\beta_0, \beta_1, \dots, \beta_p$.

The best way to compute $\hat{\beta}_0, \hat{\beta}_1, \dots, \hat{\beta}_p$ is with matrix calculations...

Linear regression model in matrix form

Write equations $Y_i = \beta_0 + x_{i1}\beta_1 + \dots + x_{ip}\beta_p + \varepsilon_i$, for $i = 1, \dots, n$, as

A diagram showing a system of linear regression equations for $i = 1, \dots, n$. The equations are:

$$\begin{aligned} Y_1 &= \beta_0 + \beta_1 x_{11} + \dots + \beta_p x_{1p} + \varepsilon_1 \\ Y_2 &= \beta_0 + \beta_1 x_{21} + \dots + \beta_p x_{2p} + \varepsilon_2 \\ &\vdots \\ Y_n &= \beta_0 + \beta_1 x_{n1} + \dots + \beta_p x_{np} + \varepsilon_n \end{aligned}$$

The equations are grouped into four vectors:

- $\underline{Y}$ (n x 1): A column vector containing $Y_1, Y_2, \dots, Y_n$.
- $\underline{X}$ (n x (p+1)): A matrix where each row represents the coefficients for a given i . The first column is all ones, and the subsequent columns are $x_{i1}, \dots, x_{ip}$.
- $\underline{b}$ ((p+1) x 1): A column vector containing the coefficients $\beta_0, \beta_1, \dots, \beta_p$.
- $\underline{e}$ (n x 1): A column vector containing the error terms $\varepsilon_1, \varepsilon_2, \dots, \varepsilon_n$.

An arrow points from the $\underline{X}$ matrix to the $\underline{b}$ vector, indicating their relationship in the matrix equation.

Now set

$$\underset{n \times 1}{\underline{Y}} = \underset{n \times (p+1)}{\underline{X}} \underset{(p+1) \times 1}{\underline{b}} + \underset{n \times 1}{\underline{e}}$$

Then the above equations can be written in matrix form as $\underline{Y} = \underline{X}\underline{b} + \underline{e}$.

Least-squares estimators in matrix form

Provided $\mathbf{X}^T \mathbf{X}$ is invertible, the entries of the vector

$$\hat{\mathbf{b}} = (\mathbf{X}^T \mathbf{X})^{-1} \mathbf{X}^T \mathbf{Y}$$

give the least-squares estimators $\hat{\beta}_0, \hat{\beta}_1, \dots, \hat{\beta}_p$.

Important: Can only compute $\hat{\mathbf{b}}$ if no column of $\mathbf{X}$ can be constructed as a linear combination of other columns (equivalent to $\mathbf{X}^T \mathbf{X}$ invertible).

Estimating the error term variance

After obtaining $\hat{\beta}_0, \hat{\beta}_1, \dots, \hat{\beta}_p$, define the

- ▶ fitted values as $\hat{Y}_i = \hat{\beta}_0 + \hat{\beta}_1 x_{i1} + \dots + \hat{\beta}_p x_{ip}$
- ▶ residuals as $\hat{\varepsilon}_i = Y_i - \hat{Y}_i$

for $i = 1, \dots, n$.

Then an unbiased estimator of σ^2 is given by

$$\hat{\sigma}^2 = \frac{1}{n - (p + 1)} \sum_{i=1}^n \hat{\varepsilon}_i^2.$$

SLR

$$\hat{\sigma}^2 = \frac{1}{n-2} \sum_{i=1}^n \hat{\varepsilon}_i^2$$

SC apartments (cont)

Make indicator or dummy variables to encode the four cities:

$$\left[\begin{array}{l} x_{\text{Columbia}} = \begin{cases} 1 & \text{if city is Columbia} \\ 0 & \text{otherwise} \end{cases} \\ x_{\text{Greenville}} = \begin{cases} 1 & \text{if city is Greenville} \\ 0 & \text{otherwise} \end{cases} \\ x_{\text{Rock Hill}} = \begin{cases} 1 & \text{if city is Rock Hill} \\ 0 & \text{otherwise} \end{cases} \end{array} \right.$$

Then Charleston is indicated by $(x_{\text{Columbia}}, x_{\text{Greenville}}, x_{\text{Rock Hill}}) = (0, 0, 0)$.

Note: If we included in $\mathbf{X}$ an indicator for all four levels, $\mathbf{X}^T\mathbf{X}$ would not be invertible due to the presence of the intercept column (this is linear algebra knowledge)!

SC apartments (cont)

The pets column contains the values “yes” and “no”.

Use a dummy variable to encode it:

$$x_{\text{pets}} = \begin{cases} 1 & \text{if yes} \\ 0 & \text{if no} \end{cases}$$

SC apartments (cont)

Construct the response vector and the design matrix.

```
Y <- log(scpts$price)
X <- cbind("Intercept" = rep(1,n),
           "Columbia" = ifelse(scpts$city == "Columbia",1,0),
           "Greenville" = ifelse(scpts$city == "Greenville",1,0),
           "Rock Hill" = ifelse(scpts$city == "Rock Hill",1,0),
           "nbath" = scpts$nbath,
           "nbed" = scpts$nbed,
           "log_sqft" = log(scpts$sqft),
           "pets" = ifelse(scpts$pets == "yes",1,0))
```

"Design matrix"

SC apartments (cont)

Estimate the regression coefficients and the error term variance:

```
bhat <- solve(t(X) %*% X) %*% t(X) %*% Y
bhat
```

$$\hat{b} = (X^T X)^{-1} X^T Y$$

```
          [,1]
Intercept  3.93274698
Columbia   -0.24090102
Greenville -0.11950394
Rock Hill  -0.14535978
nbath       -0.10728344
nbed         0.06453526
log_sqft    0.47283468
pets        -0.01857260
```

```
Yhat <- X %*% bhat
ehat <- Y - Yhat
p <- ncol(X) - 1
sgsqhat <- sum(ehat^2) / (n - (p + 1))
sgsqhat
```

```
[1] 0.04336654
```

Interpretation of the slope parameters

Consider what stories $\beta_0, \beta_1, \dots, \beta_p$ tell in the MLR model

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

- ▶ β_0 , as in SLR, just gives the function the right “height”.
- ▶ β_j is the amount by which the mean of Y changes due to a 1-unit increase in covariate j , with all other variables held fixed.

SC apartments

bhat

$\log \hat{Y}_i =$

	[,1]
Intercept	3.93274698
Columbia	-0.24090102
Greenville	-0.11950394
Rock Hill	-0.14535978
nbath	-0.10728344
nbed	0.06453526
log_sqft	0.47283468
pets	-0.01857260

- ▶ The effect of an additional bedroom (all else being equal) is to increase the expected log price by 0.0645353, or to increase the expected price by a factor of 1.0666632, or by 6.6663193 percent.
- ▶ An apartment which allows pets (all else being equal) is expected to have a log price 0.0185726 lower, that is, to be cheaper by a factor of 0.9815988, i.e. to be 1.8401195 percent cheaper.
- ▶ In which city are apartments most expensive (all else being equal)?

Do not omit *all else being equal* (or *ceteris paribus* in the Latin ;-))!

Confidence intervals for the slope parameters

Let $\Omega = (\mathbf{X}^T \mathbf{X})^{-1}$ with Ω_{jj} the diagonal entry corresponding to β_j .

- ▶ Then the estimator $\hat{\beta}_j$ is distributed as

$$\hat{\beta}_j \sim \text{Normal}(\beta_j, \sigma^2 \Omega_{jj}).$$

- ▶ “Studentizing” the above gives

$$\frac{\hat{\beta}_j - \beta_j}{\hat{\sigma} \sqrt{\Omega_{jj}}} \sim t_{n-(p+1)}$$

- ▶ So a $(1 - \alpha)100\%$ confidence interval for β_j is

$$\hat{\beta}_j \pm t_{n-(p+1), \alpha/2} \hat{\sigma} \sqrt{\Omega_{jj}}.$$

- ▶ We often write $\widehat{\text{s.e.}}(\hat{\beta}_1) = \hat{\sigma} \sqrt{\Omega_{jj}}$, s.e. for standard error.

SC apartments (cont)

Construct 95% confidence intervals for the slope coefficients.

```
alpha <- 0.05
Om <- solve(t(X) %*% X)
om <- diag(Om)
ta2 <- qt(1-alpha/2,n - (p + 1))
se <- sqrt(sgsqhat * om)
lo <- bhat - ta2 * se
up <- bhat + ta2 * se
cis <- round(cbind(bhat,lo,up),4)
colnames(cis) <- c("estimate","lower","upper")
rownames(cis) <- colnames(X)
print(cis)
```

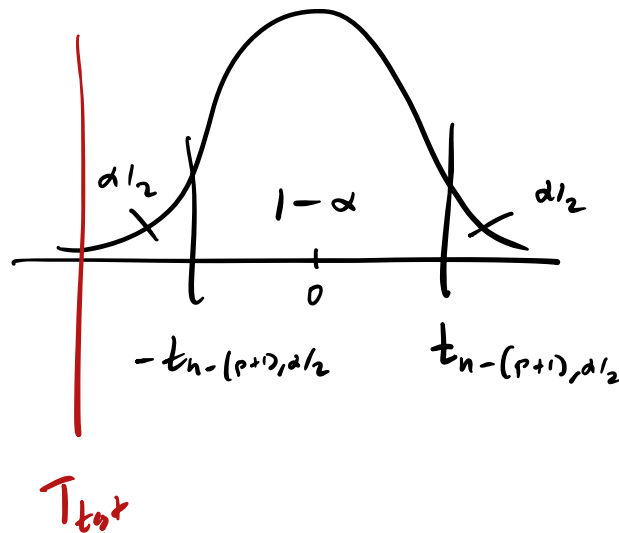
	estimate	lower	upper
Intercept	3.9327	2.9839	4.8816
Columbia	-0.2409	-0.3207	-0.1611
Greenville	-0.1195	-0.1909	-0.0481
Rock Hill	-0.1454	-0.2397	-0.0510
nbath	-0.1073	-0.1927	-0.0219
nbed	0.0645	-0.0026	0.1317
log_sqft	0.4728	0.3188	0.6268
pets	-0.0186	-0.0783	0.0411

Testing hypothesis about $\beta_1, \dots, \beta_p$

Mostly interested in testing $H_0: \beta_j = 0$
 $H_1: \beta_j \neq 0$.

Test statistic:

$$T_{\text{test}} = \frac{\hat{\beta}_j - 0}{\hat{\sigma} \sqrt{\mathcal{L}_{jj}}} \underset{H_0}{\sim} t_{n-(p+1)}$$



Reject $H_0: \beta_j = 0$ if $|T_{\text{test}}| > t_{n-(p+1), \alpha/2}$

Tests of hypotheses about the slope coefficients

We most often test hypotheses about the β_j of the form

$$\begin{array}{lll} H_0: \beta_j \geq 0 & \text{or} & H_0: \beta_j = 0 & \text{or} & H_0: \beta_j \leq 0 \\ H_1: \beta_j < 0 & & H_1: \beta_j \neq 0 & & H_1: \beta_j > 0. \end{array}$$

Reject or fail to reject H_0 based on the value of the test statistic

$$T_{\text{test}} = \frac{\hat{\beta}_j}{\hat{\sigma} \sqrt{\Omega_{jj}}}.$$

Rejection rules for the above at significance level α are

$$T_{\text{test}} < -t_{n-(p+1),\alpha} \quad \text{or} \quad |T_{\text{test}}| > t_{n-(p+1),\alpha/2} \quad \text{or} \quad T_{\text{test}} > t_{n-(p+1),\alpha}.$$

The corresponding p-values are, with $T \sim t_{n-(p+1)}$, the probabilities

$$P(T < T_{\text{test}}) \quad \text{or} \quad 2 \times P(T > |T_{\text{test}}|) \quad \text{or} \quad P(T > T_{\text{test}}).$$

SC apartments (cont)

Obtain p-values for testing $H_0: \beta_j = 0$ vs $H_1: \beta_j \neq 0$ for each j .

```
sehat <- sqrt(sgsqhat * om)
Tstat <- bhat / sehat
pval <- 2*(1 - pt(abs(Tstat),df = n - (p + 1)))
summ <- round(cbind(bhat,sehat,Tstat,pval),4)
colnames(summ) <- c("estimate","sehat","Tstat","pval")
rownames(summ) <- colnames(X)
print(summ)
```

	estimate	sehat	Tstat	pval
Intercept	3.9327	0.4812	8.1720	0.0000
Columbia	-0.2409	0.0405	-5.9508	0.0000
Greenville	-0.1195	0.0362	-3.2987	0.0011
Rock Hill	-0.1454	0.0479	-3.0371	0.0027
nbath	-0.1073	0.0433	-2.4762	0.0141
nbed	0.0645	0.0341	1.8950	0.0595
log_sqft	0.4728	0.0781	6.0535	0.0000
pets	-0.0186	0.0303	-0.6135	0.5402

The lm(), summary(), and confint() functions in R

```
lm_out <- lm(log(price) ~ city + nbath + nbed + log(sqft) + pets, data = scapts)
summary(lm_out)
```

$$\widehat{\log y} = 3.93 - 0.24(\text{Colo}) - 0.12(\text{Grn}) - 0.15(\text{RockHill}) - 0.11(\text{nbath}) + 0.065(\text{nbed})$$

Call:

```
lm(formula = log(price) ~ city + nbath + nbed + log(sqft) + pets,
    data = scapts)
```

Residuals:

Min	1Q	Median	3Q	Max
-0.65276	-0.12428	-0.01501	0.08756	0.71661

Coefficients:

	Estimate	Std. Error	t value	Pr(> t)
(Intercept)	3.93275	0.48125	8.172	3.04e-14 ***
cityColumbia	-0.24090	0.04048	-5.951	1.13e-08 ***
cityGreenville	-0.11950	0.03623	-3.299	0.00114 **
cityRock Hill	-0.14536	0.04786	-3.037	0.00270 **
nbath	-0.10728	0.04333	-2.476	0.01409 *
nbed	0.06454	0.03406	1.895	0.05950 .
log(sqft)	0.47283	0.07811	6.053	6.61e-09 ***
petsyes	-0.01857	0.03027	-0.614	0.54021

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.2082 on 206 degrees of freedom

Multiple R-squared: 0.414, Adjusted R-squared: 0.3941

F-statistic: 20.79 on 7 and 206 DF, p-value: < 2.2e-16

$$+ 0.47 \log(\text{sqft})$$

$$- 0.62 \text{ pets}$$

$$T_{\text{test}} = \frac{\hat{\beta}_{\text{pets}}}{\hat{\sigma} \sqrt{\Omega_{\text{pets}}}} = \frac{-0.01857}{0.03027} = -0.614$$

$$\hat{\beta}_{\text{pets}} = -0.01857$$

$$\hat{\sigma} = \sqrt{\hat{\sigma}^2} = \sqrt{\frac{1}{n-(p+1)} \sum_{i=1}^n \hat{\epsilon}_i^2}$$

$$n = 214, \quad p = 7$$

$$214 - (7 + 1) = 206$$

Since the price is $\log(\text{price})$ we interpret $\hat{\beta}_{\text{pets}} = -0.01857$ as follows:

An apartment allowing pets has a price equal to $e^{\hat{\beta}_{\text{pets}}}$ $\times$ price of apt which does not, all else being equal.

The price changes by the factor 0.9816 when pets are allowed.

Because response is $\log(\text{price})$.

n bath coefficient is -0.107 .

let

$Y_1 =$ price of apartment 1

$Y_2 =$ price of apartment 2 (has one more bathroom, but otherwise equal)

$$\log(Y_2) - \log(Y_1) = -0.107$$

$$\Rightarrow \log\left(\frac{Y_2}{Y_1}\right) = -0.107$$

$$\Rightarrow \frac{Y_2}{Y_1} = e^{-0.107} = 0.899$$

$$\Rightarrow Y_2 = 0.899 \cdot Y_1$$

↑

$$\log(a) - \log(b) = \log\left(\frac{a}{b}\right)$$

```
confint(lm_out)
```

	2.5 %	97.5 %
(Intercept)	2.983942705	4.88155125
cityColumbia	-0.320713599	-0.16108844
cityGreenville	-0.190928040	-0.04807984
cityRock Hill	-0.239721140	-0.05099842
nbath	-0.192703165	-0.02186372
nbed	-0.002607853	0.13167838
log(sqft)	0.318837614	0.62683175
petsyes	-0.078255964	0.04111076

```
confint(lm_out, level = .99)
```

	0.5 %	99.5 %
(Intercept)	2.68154577	5.183948180
cityColumbia	-0.34615096	-0.135651076
cityGreenville	-0.21369188	-0.025315996
cityRock Hill	-0.26979540	-0.020924159
nbath	-0.21992760	0.005360712
nbed	-0.02400729	0.153077815
log(sqft)	0.26975664	0.675912723
petsyes	-0.09727787	0.060132663

CI for the mean and PI for Y_{new} at $\mathbf{x}_{\text{new}}$

For a new vector of covariate values $\mathbf{x}_{\text{new}}$, let

$$\hat{Y}_{\text{new}} = \hat{\beta}_0 + \hat{\beta}_1 x_{\text{new},1} + \cdots + \hat{\beta}_p x_{\text{new},p}$$

► A $(1 - \alpha) \times 100$ CI for $\beta_0 + \beta_1 x_{\text{new},1} + \cdots + \beta_p x_{\text{new},p}$ is given by

$$\hat{Y}_{\text{new}} \pm t_{n-(p+1), \alpha/2} \hat{\sigma} \sqrt{\Omega_{\text{new}}},$$

► A $(1 - \alpha) \times 100$ PI for Y_{new} corresponding to $\mathbf{x}_{\text{new}}$ is given by

$$\hat{Y}_{\text{new}} \pm t_{n-(p+1), \alpha/2} \hat{\sigma} \sqrt{1 + \Omega_{\text{new}}},$$

where $\Omega_{\text{new}} = \tilde{\mathbf{x}}_{\text{new}}^T \Omega \tilde{\mathbf{x}}_{\text{new}}$ with $\tilde{\mathbf{x}}_{\text{new}}^T = (1 \ x_{\text{new},1} \ \cdots \ x_{\text{new},p})^T$.

SC apartments (cont)

Build 95% CI for the average rent of apartments in Greenville with one bathroom, two bedrooms, nine-hundred square feet, and which allow pets.

```
xnew <- c(1,0,1,0,1,2,log(900),1)
om_new <- t(xnew) %*% Om %*% xnew
Ynew_hat <- t(xnew) %*% bhat
seci <- sqrt(sgsqhat) * sqrt(om_new)
loci <- Ynew_hat - ta2 * segi
upci <- Ynew_hat + ta2 * segi

# exponentiate since we have used the log of the price
eloci <- exp(loci)
eupci <- exp(upci)
```

The confidence interval is (1040.545, 1234.266).

Now build a 95% PI for the rent of a single such a property.

```
sepi <- sqrt(sgsqhat) * sqrt( 1 + om_new)
lopi <- Ynew_hat - ta2 * sepi
uppi <- Ynew_hat + ta2 * sepi

# exponentiate since we have used the log of the price
elopi <- exp(lopi)
euppi <- exp(uppi)
```

The prediction interval is (745.099, 1723.675).

The predict() function in R

```
newdata <- data.frame(city = "Greenville",  
                      nbath = 1,  
                      nbed = 2,  
                      sqft = 900,  
                      pets = "yes")  
ci <- predict(lm_out, newdata = newdata, int = "conf")  
exp(ci)
```

	fit	lwr	upr
1	1133.274	1040.545	1234.266

```
pi <- predict(lm_out, newdata = newdata, int = "pred")  
exp(pi)
```

	fit	lwr	upr
1	1133.274	745.0992	1723.675

Sums of squares in multiple linear regression

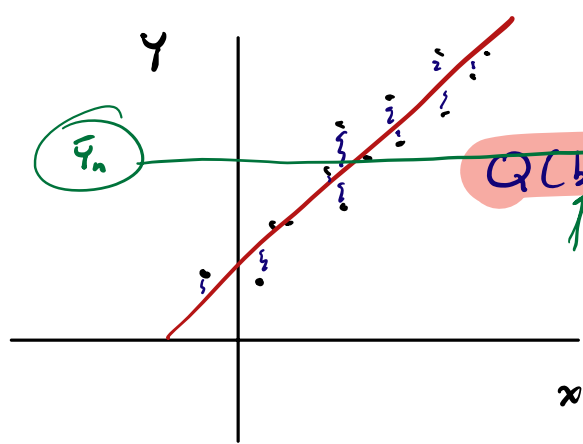
We decompose the variation in $Y_1, \dots, Y_n$ by defining the:

- ▶ Total sum of squares: $SS_{\text{Tot}} = \sum_{i=1}^n (Y_i - \bar{Y}_n)^2$
- ▶ Regression sum of squares: $SS_{\text{Reg}} = \sum_{i=1}^n (\hat{Y}_i - \bar{Y}_n)^2$
- ▶ Error sum of squares: $SS_{\text{Error}} = \sum_{i=1}^n (Y_i - \hat{Y}_i)^2$

We have $SS_{\text{Tot}} = SS_{\text{Reg}} + SS_{\text{Error}}$.

The coefficient of determination is defined as $R^2 = \frac{SS_{\text{Reg}}}{SS_{\text{Tot}}}$.

- ▶ $R^2 \in [0, 1]$
- ▶ Proportion of variation in Y “explained” by the covariates $x_1, \dots, x_p$.



$$Q(b_0, b_1) = \sum_{i=1}^n (y_i - (b_0 + b_1 x_i))^2$$

$\hat{\beta}_0, \hat{\beta}_1$ the "least-squares" estimator

$$\hat{\varepsilon}_i = y_i - (\hat{\beta}_0 + \hat{\beta}_1 x_i) = y_i - \hat{y}_i$$

$$SS_{Tot} = \sum_{i=1}^n (y_i - \bar{y}_n)^2$$

$$SS_{Error} = \sum_{i=1}^n \hat{\varepsilon}_i^2 = \sum_{i=1}^n (y_i - \hat{y}_i)^2$$

$$\hat{\sigma}^2 = \frac{1}{n-2} SS_{Error}$$

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

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

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

$$MS_{Error} = \frac{SS_{Error}}{n-(p+1)} = \hat{\sigma}^2$$

ANOVA

Source	Df	SS	MS	F_{test}	p-value
Reg	p	SS_{Reg}	MS_{Reg}	$\frac{MS_{Reg}}{MS_{Error}}$	
Error	$n - (p+1)$	SS_{Error}	MS_{Error}		
Total	$n - 1$	SS_{Total}			

The mean squares in multiple linear regression

The SS, appropriately scaled, follow chi-square distributions:

- ▶ $SS_{\text{Tot}} / \sigma^2 \sim \chi_{n-1}^2(\phi_{\text{Tot}})$
- ▶ $SS_{\text{Reg}} / \sigma^2 \sim \chi_p^2(\phi_{\text{Reg}})$
- ▶ $SS_{\text{Error}} / \sigma^2 \sim \chi_{n-(p+1)}^2$,

where ϕ_{Tot} and ϕ_{Reg} are noncentrality parameters.

Dividing SS_{Reg} and SS_{Error} by their dfs, we define:

- ▶ Regression mean square: $MS_{\text{Reg}} = \frac{SS_{\text{Reg}}}{p}$
- ▶ Error mean square: $MS_{\text{Error}} = \frac{SS_{\text{Error}}}{n - (p + 1)}$

Moreover, define the adjusted R squared as $\bar{R}^2 = 1 - \frac{MS_{\text{Error}}}{SS_{\text{Tot}} / (n - 1)}$.

Adjustment “penalizes” the inclusion of additional covariates.

The Analysis of Variance (ANOVA) table

We often present the SS, df, and MS values in a table like this:

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

This is an example of an ANOVA table.

The F-value and the p-value we will discuss later in these slides.

Building the ANOVA table

```
Ybar <- mean(Y)
SST <- sum((Y - Ybar)^2)
SSR <- sum((Yhat - Ybar)^2)
SSE <- sum((Y - Yhat)^2)
MSR <- SSR / p
MSE <- SSE / (n-(p+1))
Fstat <- MSR / MSE
pval <- 1 - pf(Fstat,p,n-(p+1))
```

Source	Df	SS	MS	F value	p-value
Regression	7	6.31	0.9	20.79	0
Error	206	8.93	0.04		
Total	213	15.24			

Moreover $R^2 = 0.414$ and $\bar{R}^2 = 0.394$.

ANOVA quantities in output from `lm()` with `summary()`

```
summary(lm_out)
```

Call:

```
lm(formula = log(price) ~ city + nbath + nbed + log(sqft) + pets,  
    data = scapts)
```

Residuals:

Min	1Q	Median	3Q	Max
-0.65276	-0.12428	-0.01501	0.08756	0.71661

Coefficients:

	Estimate	Std. Error	t value	Pr(> t)	
(Intercept)	3.93275	0.48125	8.172	3.04e-14	***
cityColumbia	-0.24090	0.04048	-5.951	1.13e-08	***
cityGreenville	-0.11950	0.03623	-3.299	0.00114	**
cityRock Hill	-0.14536	0.04786	-3.037	0.00270	**
nbath	-0.10728	0.04333	-2.476	0.01409	*
nbed	0.06454	0.03406	1.895	0.05950	.
log(sqft)	0.47283	0.07811	6.053	6.61e-09	***
petsyes	-0.01857	0.03027	-0.614	0.54021	

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.2082 on 206 degrees of freedom

Multiple R-squared: 0.414, Adjusted R-squared: 0.3941

F-statistic: 20.79 on 7 and 206 DF, p-value: < 2.2e-16

Sequential SS with anova() function (seldom use)

Sequential SS report the changes in SS_{Reg} from adding new variables.

```
lm_out <- lm(log(price) ~ city + nbath + nbed + log(sqft) + pets, data = scapts)
anova(lm_out)
```

Analysis of Variance Table

Response: log(price)

	Df	Sum Sq	Mean Sq	F value	Pr(>F)	
city	3	1.0337	0.34458	7.9458	4.869e-05	***
nbath	1	1.7918	1.79184	41.3185	8.832e-10	***
nbed	1	1.8666	1.86660	43.0425	4.250e-10	***
log(sqft)	1	1.6021	1.60213	36.9439	5.804e-09	***
pets	1	0.0163	0.01632	0.3764	0.5402	
Residuals	206	8.9335	0.04337			

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

The sequential SS depend on the order in which variables are added:

```
lm2 <- lm(log(price) ~ nbath + city + nbed + log(sqft) + pets, data = scapts)
anova(lm2)
```

Analysis of Variance Table

Response: log(price)

	Df	Sum Sq	Mean Sq	F value	Pr(>F)	
nbath	1	1.3087	1.30872	30.1782	1.155e-07	***
city	3	1.5169	0.50562	11.6592	4.344e-07	***
nbed	1	1.8666	1.86660	43.0425	4.250e-10	***
log(sqft)	1	1.6021	1.60213	36.9439	5.804e-09	***
pets	1	0.0163	0.01632	0.3764	0.5402	
Residuals	206	8.9335	0.04337			

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Sequential model fits to obtain sequential SS

```
lm1 <- lm(log(price) ~ nbath, data = scapts)
lm2 <- lm(log(price) ~ nbath + city, data = scapts)
lm3 <- lm(log(price) ~ nbath + city + nbed, data = scapts)
lm4 <- lm(log(price) ~ nbath + city + nbed + log(sqft), data = scapts)
lm5 <- lm(log(price) ~ nbath + city + nbed + log(sqft) + pets, data = scapts)

SSR1 <- SST - sum(lm1$residuals^2)
SSR2 <- SST - sum(lm2$residuals^2)
SSR3 <- SST - sum(lm3$residuals^2)
SSR4 <- SST - sum(lm4$residuals^2)
SSR5 <- SST - sum(lm5$residuals^2)
seqSS <- c(SSR1, SSR2 - SSR1, SSR3 - SSR2, SSR4 - SSR3, SSR5 - SSR4)
names(seqSS) <- c("city", "nbath", "nbed", "log(sqft)", "pets")
round(seqSS, 3)
```

city	nbath	nbed	log(sqft)	pets
1.309	1.517	1.867	1.602	0.016

Interactions

- ▶ When the effect of one predictor depends on the value of another it is called an interaction.
 - ▶ Does the effect of number of bedrooms depend on the number of bathrooms?
 - ▶ Does the effect of square feet differ between Charleston and Rock Hill?
- ▶ We can include interactions by including in $\mathbf{X}$ products of pairs of predictors.
- ▶ Most often we consider interactions between dummy variables and numeric variables.
- ▶ Such interactions act as “slope modifiers”, allowing different slope coefficients for different subsets of the data.

$$x_{\text{columbia}} = \begin{cases} 1 & \text{if columbia} \\ 0 & \text{otherwise} \end{cases}$$

$$\log(\text{Price}) = \beta_0 + \beta_1 x_{\text{columbia}} + \beta_2 \log(\text{sqft}) + \beta_3 \left(\underbrace{x_{\text{columbia}} \times \log(\text{sqft})}_{\text{Interaction term.}} \right)$$

$$= \begin{cases} \beta_0 + \beta_1 + (\beta_2 + \beta_3) \log(\text{sqft}) & \text{in columbia} \\ \beta_0 + \beta_2 \log(\text{sqft}) & \text{otherwise} \end{cases}$$

SC apartments (cont)

Add interactions between:

- ▶ Number of bedrooms/bathrooms.
- ▶ Square feet and each city indicator.

```
Y <- log(scapts$price)
X <- cbind("Intercept" = rep(1,n),
  "Columbia" = ifelse(scapts$city == "Columbia",1,0),
  "Greenville" = ifelse(scapts$city == "Greenville",1,0),
  "Rock Hill" = ifelse(scapts$city == "Rock Hill",1,0),
  "nbath" = scapts$nbath,
  "nbed" = scapts$nbed,
  "log_sqft" = log(scapts$sqft),
  "pets" = ifelse(scapts$pets == "yes",1,0),
  "nbath:nbed" = scapts$nbath*scapts$nbed,
  "log_sqft:Columbia" = log(scapts$sqft)*ifelse(scapts$city == "Columbia",1,0),
  "log_sqft:Greenville" = log(scapts$sqft)*ifelse(scapts$city == "Greenville",1,0),
  "log_sqft:Rock Hill" = log(scapts$sqft)*ifelse(scapts$city == "Rock Hill",1,0))
```

SC apartments (cont)

Include interactions in the `lm()` function as below:

```
lm_out <- lm(log(price) ~ city + nbath + nbed + nbath:nbed + log(sqft) + pets + log(sqft):city, data = scapts)
summary(lm_out)
```

Call:

```
lm(formula = log(price) ~ city + nbath + nbed + nbath:nbed +
    log(sqft) + pets + log(sqft):city, data = scapts)
```

Residuals:

Min	1Q	Median	3Q	Max
-0.59947	-0.13123	-0.00646	0.09455	0.69055

Coefficients:

	Estimate	Std. Error	t value	Pr(> t)
(Intercept)	2.878148	0.686715	4.191	4.15e-05 ***
cityColumbia	1.677294	0.819015	2.048	0.0419 *
cityGreenville	0.891876	0.888803	1.003	0.3168
cityRock Hill	0.429247	1.384079	0.310	0.7568
nbath	-0.181440	0.102964	-1.762	0.0796 .
nbed	0.042290	0.082471	0.513	0.6087
log(sqft)	0.639417	0.108495	5.894	1.56e-08 ***
petsyes	-0.005982	0.031216	-0.192	0.8482
nbath:nbed	0.024299	0.048527	0.501	0.6171
cityColumbia:log(sqft)	-0.282220	0.120906	-2.334	0.0206 *
cityGreenville:log(sqft)	-0.149093	0.130962	-1.138	0.2563
cityRock Hill:log(sqft)	-0.086384	0.200685	-0.430	0.6673

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.2073 on 202 degrees of freedom

Multiple R-squared: 0.4307, Adjusted R-squared: 0.3997

F-statistic: 13.89 on 11 and 202 DF, p-value: < 2.2e-16

$$\begin{aligned} \log(\text{price}) = & \dots 1.68 x_{\text{columbia}} + \dots \\ & + \dots + 0.639 \log(\text{sqft}) + \dots \\ & + (-0.282)(x_{\text{columbia}} \times \log(\text{sqft})) \\ & \dots \end{aligned}$$

Check for significance of any interaction effects and give interpretations.

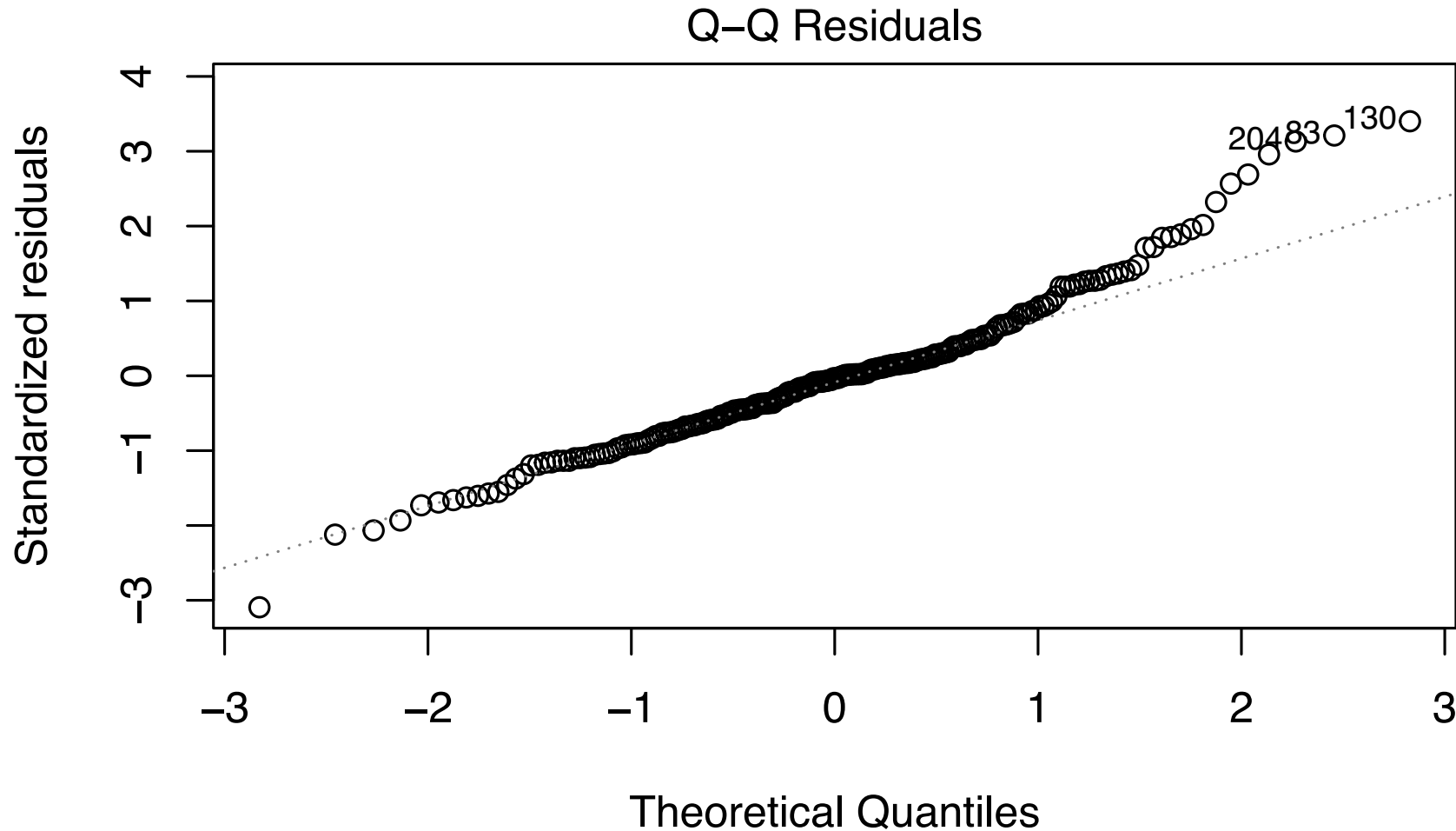
Checking model assumptions

Validity of the foregoing analyses depends on these assumptions:

1. The responses are normally distributed around the regression line (Check QQ plot of residuals). *If n is large this only matters for prediction intervals.*
2. The response has the same variance for all covariate values (Check residuals vs fitted values plot).
3. The covariates and the response are linearly related (Check residuals vs fitted values plot).
4. The response values are independent of each other (No way to check; must trust experimental design).

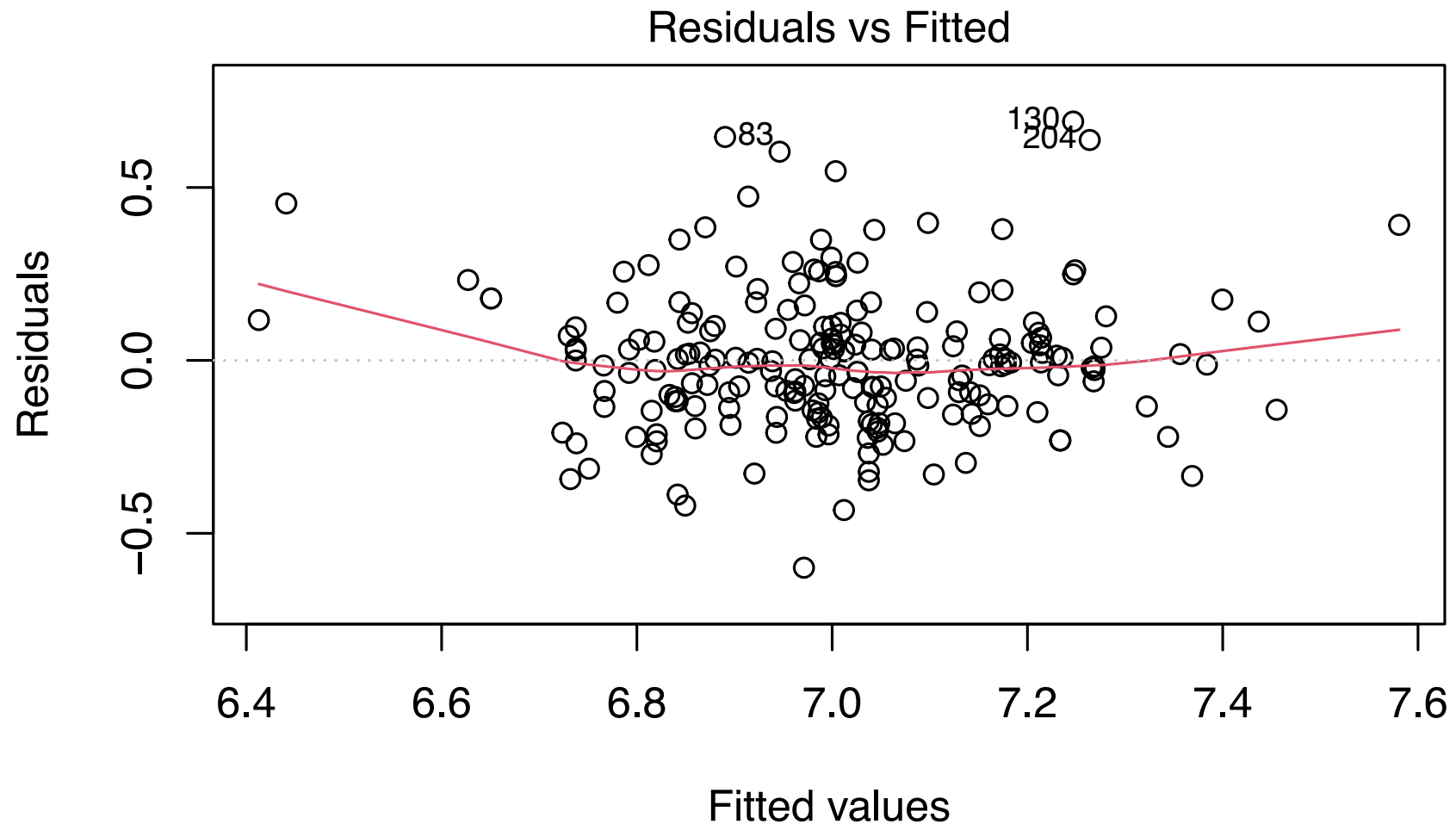
Generating diagnostic plots from `lm()` with `plot()`

```
plot(lm_out, which = 2)
```



$\text{lm}(\log(\text{price}) \sim \text{city} + \text{nbath} + \text{nbed} + \text{nbath}:\text{nbed} + \log(\text{sqft}) + \text{pets} + \log(\text{s} \dots$

```
plot(lm_out, which = 1)
```



$\text{lm}(\log(\text{price}) \sim \text{city} + \text{nbath} + \text{nbed} + \text{nbath}:\text{nbed} + \log(\text{sqft}) + \text{pets} + \log(\text{s} \dots$

Leverage and Cook's distance in MLR

The leverage of a point $(Y_i, \mathbf{x}_i)$ among $(Y_1, \mathbf{x}_1), \dots, (Y_n, \mathbf{x}_n)$ is

$$\text{lev}_i = \text{entry } i \text{ on the diagonal of the matrix } \mathbf{X}(\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}.$$

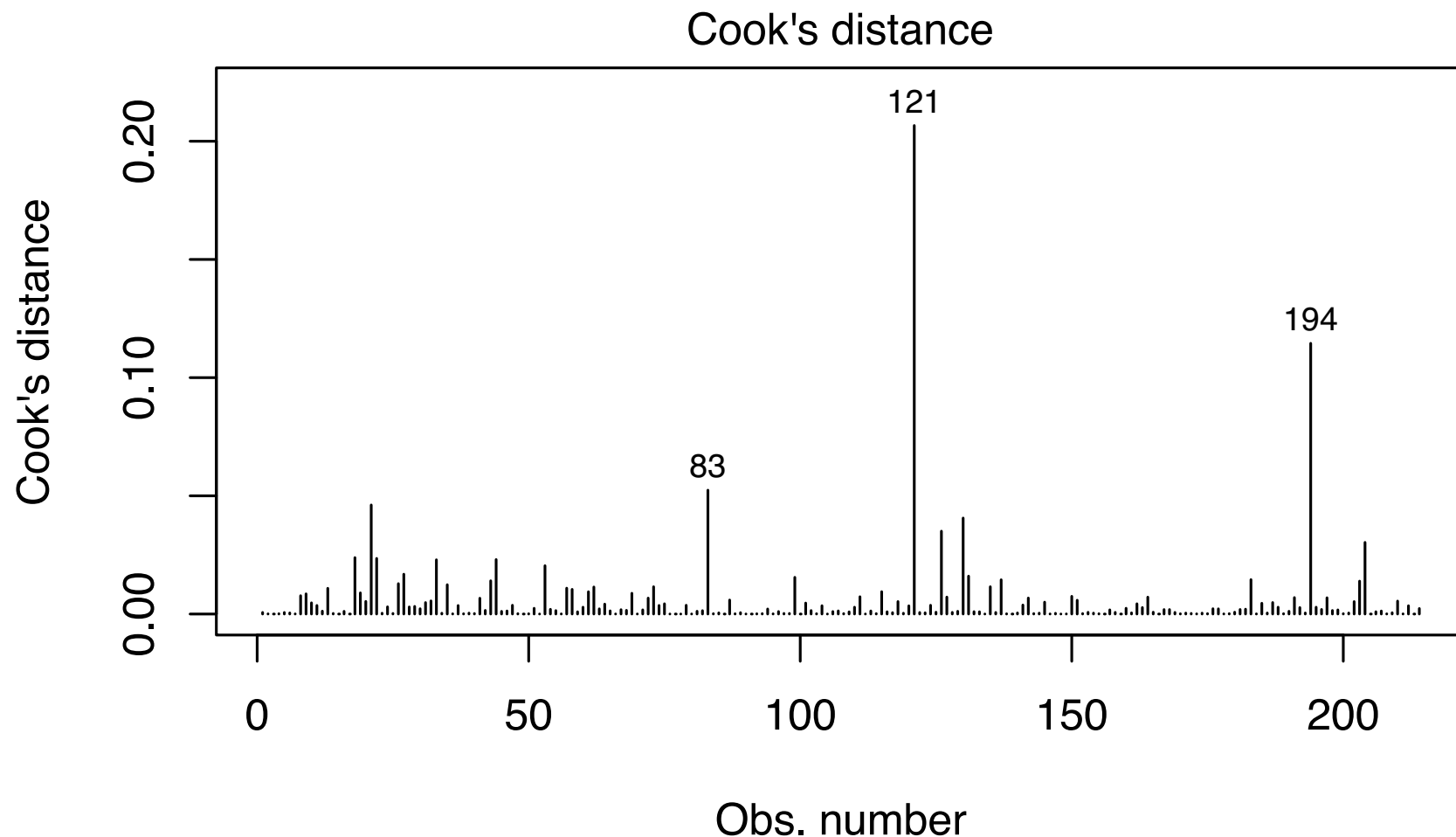
Leverage only shows outlying-ness in the covariate space.

Cook's Distance measures how much each data point changes the fit:

$$D_i = \frac{1}{(p+1)\hat{\sigma}^2} \sum_{j=1}^n (\hat{Y}_j - \hat{Y}_{j(i)})^2 = \frac{\hat{e}_i^2}{(p+1)\hat{\sigma}^2} \frac{\text{lev}_i}{(1 - \text{lev}_i)^2} \quad \text{for } i = 1, \dots, n,$$

where $\hat{Y}_{j(i)}$ is the j th fitted value from the model fitted without obs i .

```
plot(lm_out, which = 4)
```



$\text{lm}(\log(\text{price}) \sim \text{city} + \text{nbath} + \text{nbed} + \text{nbath}:\text{nbed} + \log(\text{sqft}) + \text{pets} + \log(\text{s} \dots$

References

“Apartment for Rent Classified.” 2019. UCI Machine Learning Repository.