Bivariate Linear Regression

Week 3

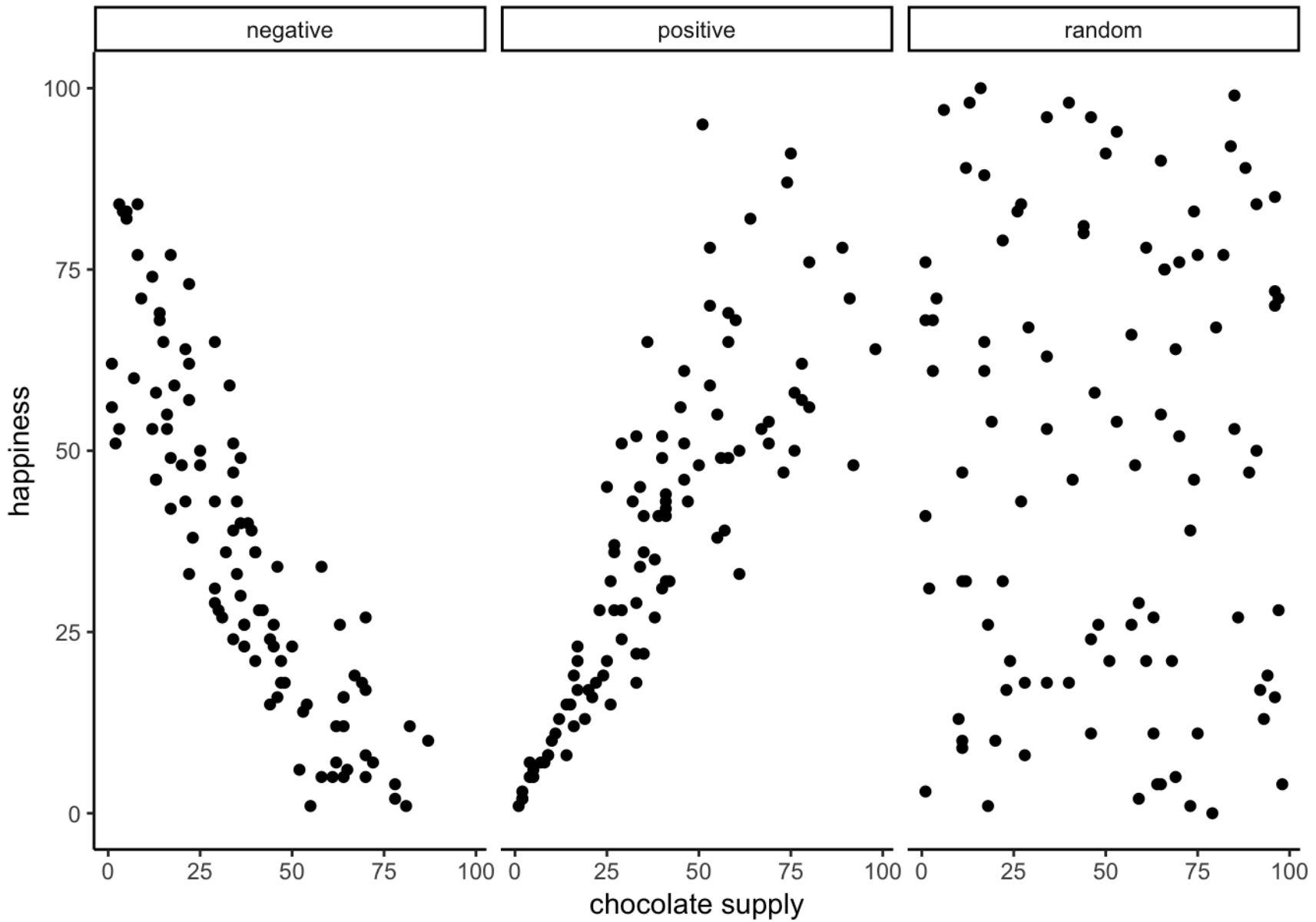
course website features

- copy icon in code blocks
- site-wide search bar
- `?` and `m` key during slides
- for the interested: site made using Quarto

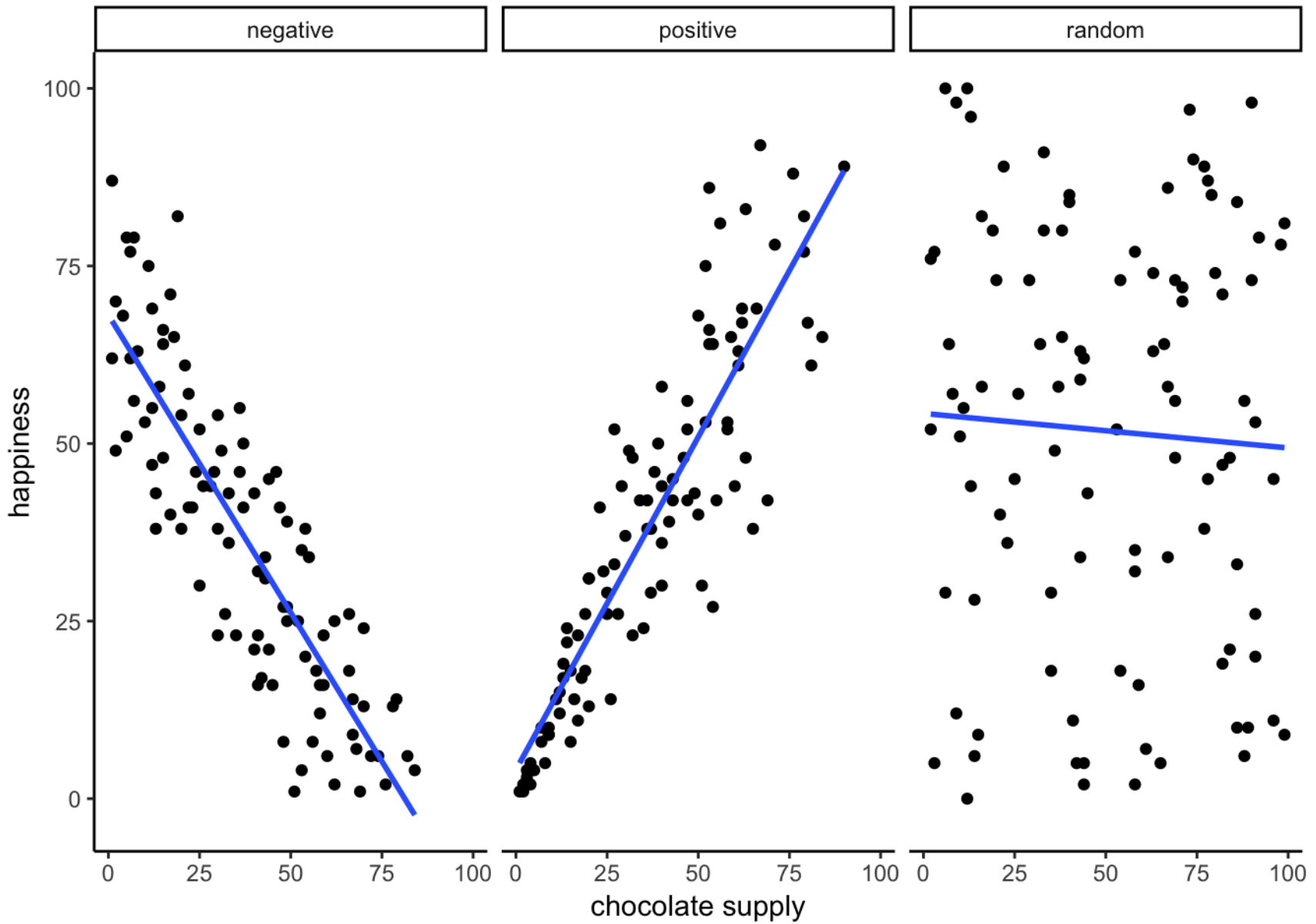
Linear Regression

- geometric interpretation of correlation
- can be used for prediction
- a **linear model** relating one variable to another variable

Examples of correlation



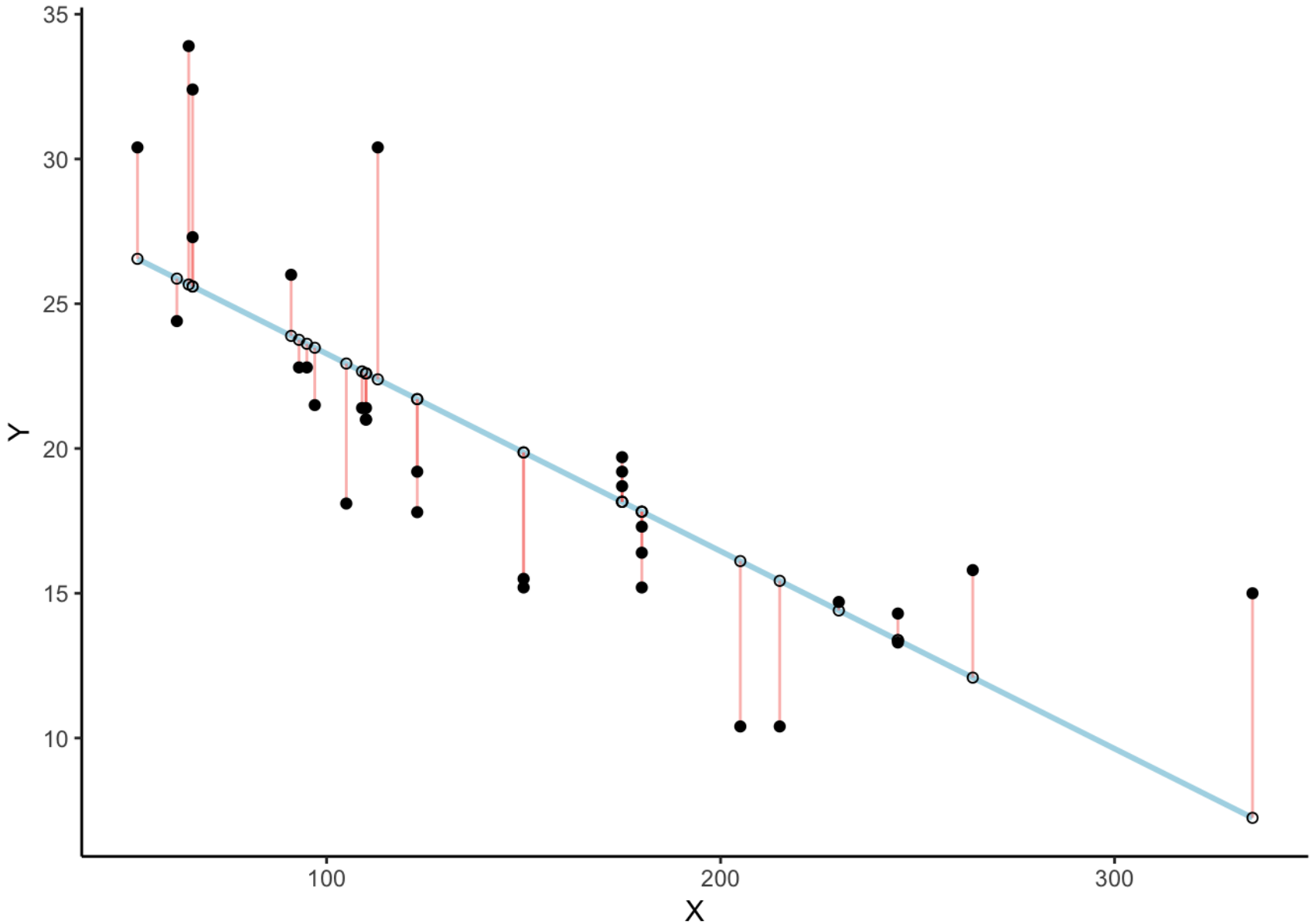
Correlation with Regression lines



What is a regression line?

- first: it's a **line** (we will need the equation)
- the **best fit** line
- how do we determine which line fits best?

Residuals and error



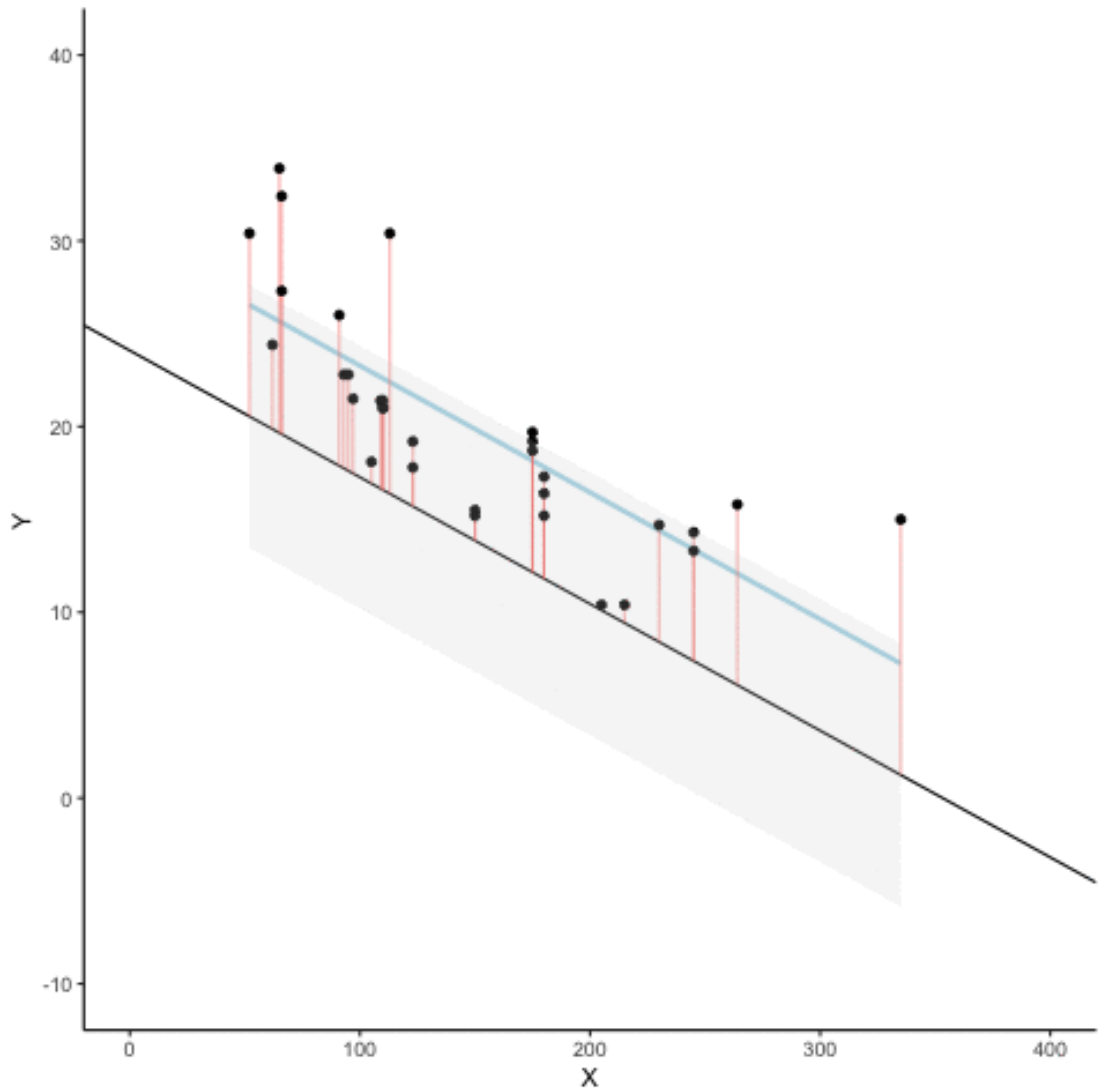
What is a regression line?

- first: it's a **line** (we will need the equation)
- the **best fit** line
- how do we determine which line fits best?

The regression line minimizes the sum of the (squared) residuals

Animated Residuals

regression minimizes the sum of the (squared) residuals



Finding the best fit line

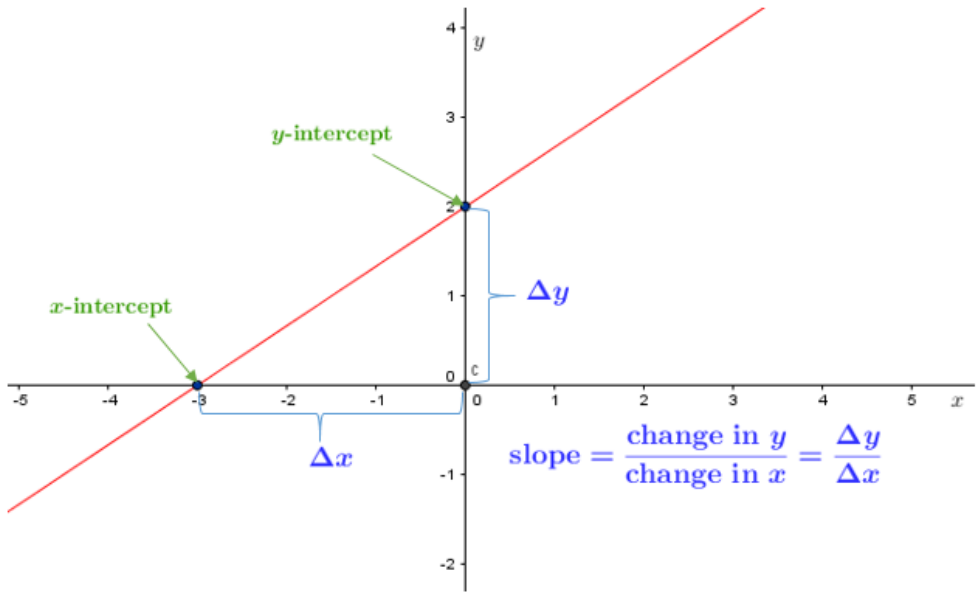
- how do we find the best fit line?
- First step, remember what lines are …

Equation for a line

$$y = mx + b$$

$$y = \text{slope} * x + \text{yintercept}$$

- y = value on y-axis
- m = slope of the line
- x = value on x-axis
- b = value on y-axis **when x = 0**



We will also use this form:

$$y = \beta_0 + \beta_1 x$$

solving for y

- predicting y based on x

$$y = .5x + 2$$

What is the value of y, when x is 0?

$$y = .5 * 0 + 2$$
$$y = 0 + 2$$
$$y = 2$$

Finding the best fit line

find m and b for:

$$Y = mX + b$$

so that the regression line minimizes the sum of the squared residuals

Finding the best fit line

$$Y = mX + b$$

$$intercept = b = \frac{\sum y \sum x^2 - \sum x \sum xy}{n \sum x^2 - (\sum x)^2}$$

$$slope = m = \frac{n \sum xy - \sum x \sum y}{n \sum x^2 - (\sum x)^2}$$

sample data

scores	x	y	x_squared	y_squared	xy
1	1	2	1	4	2
2	4	5	16	25	20
3	3	1	9	1	3
4	6	8	36	64	48
5	5	6	25	36	30
6	7	8	49	64	56
7	8	9	64	81	72
Sums	34	39	200	275	231

sample calculations

$$intercept = b = \frac{\sum y \sum x^2 - \sum x \sum xy}{n \sum x^2 - (\sum x)^2} = \frac{39 * 200 - 34 * 231}{7 * 200 - 34^2} = -.221$$

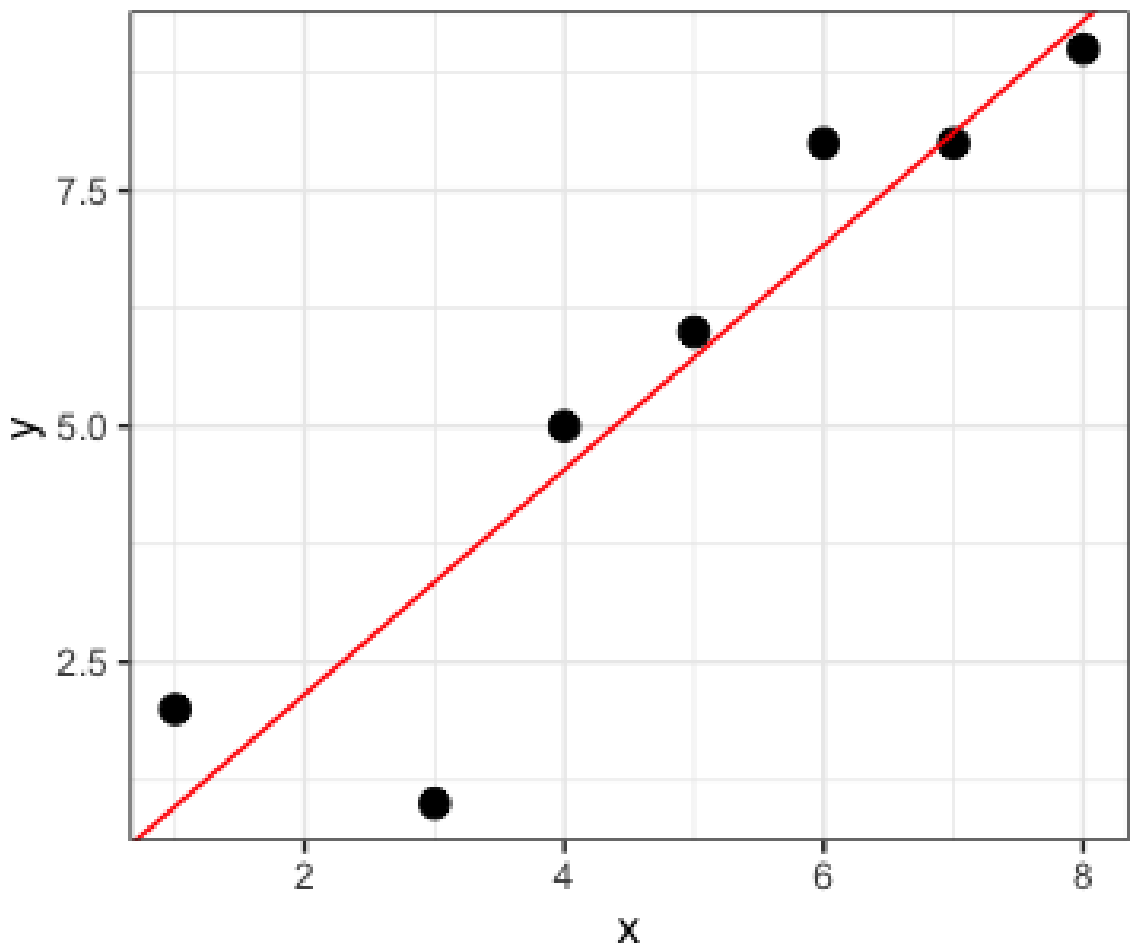
$$slope = m = \frac{n \sum xy - \sum x \sum y}{n \sum x^2 - (\sum x)^2} = \frac{7 * 231 - 34 * 39}{7 * 200 - 34^2} = 1.19$$

sample plot

$b = -0.221$

$m = 1.19$

$Y = (1.19)X - 0.221$



Linear Regression in R

```r
library(tidyverse)
x <- c(1,4,3,6,5,7,8)
y <- c(2,5,1,8,6,8,9)
n = 7
(b <- ((sum(y)*sum(x^2)) - (sum(x)*sum(x*y))) / ((n*sum(x^2)) - (sum(x))^2)
```

```
[1] -0.2213115
```

```r
(m = ((n*sum(x*y)) - (sum(x)*sum(y))) / ((n*sum(x^2)) - (sum(x))^2))
```

```
[1] 1.192623
```

Linear Regression in R using `lm()`

```
library(tidy
x <- c(1,4,3
y <- c(2,5,1
df <- tibble
df
```

```
# A tibble: 7 × 2
      x     y
  <dbl> <dbl>
1     1     2
2     4     5
3     3     1
4     6     8
5     5     6
6     7     8
7     8     9
```

```
mymod <- lm(y ~ x, data=df)
coef(mymod)
```

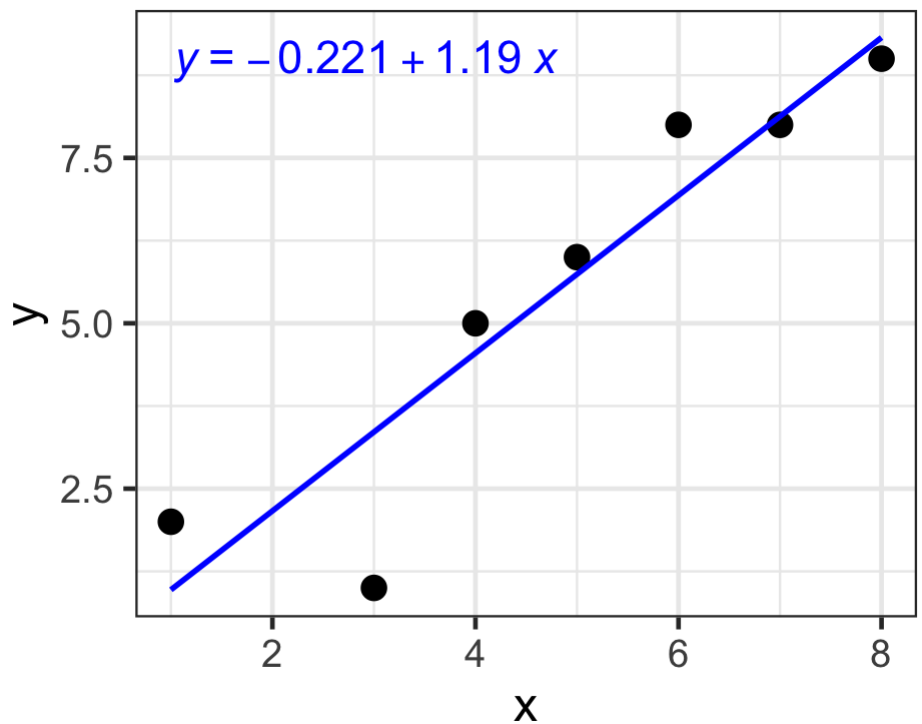
```
(Intercept)           x
 -0.2213115   1.1926230
```

Linear Regression in R using `lm()`

```r
library(tidy
x <- c(1,4,3
y <- c(2,5,1
df <- tibble
df
```

```
# A tibble: 7 × 2
      x     y
  <dbl> <dbl>
1     1     2
2     4     5
3     3     1
4     6     8
5     5     6
6     7     8
7     8     9
```

```r
library(ggpmisc) # for stat_poly_eq
ggplot(data=df, aes(x=x,y=y)) +
  geom_point(size=4, color="black") +
  geom_smooth(method="lm", se=FALSE, color="blue
  stat_poly_eq(use_label(c("eq")), size=6, colo
  theme_bw(base_size=18)
```



scatterplot plus regression line

R: intercept and slope

```
library(tidy
x <- c(55,61
y <- c(140,1
df <- tibble
df
```

```
# A tibble: 10 × 2
       x     y
   <dbl> <dbl>
 1    55   140
 2    61   150
 3    67   152
 4    83   220
 5    65   190
 6    82   195
 7    70   175
 8    58   130
 9    65   155
10    61   160
```

```
mymod <- lm(y~x, data=df)
coef(mymod)
```

```
(Intercept)           x
  -7.176930    2.606851
```

Reminders: y-intercept

What does the **y-intercept** mean?

It is the value where the line crosses the y-axis when x = 0

Reminders: slope

What does the **slope** mean?

The slope tells you the rate of change.

For every 1 unit of X, Y changes by “slope” amount

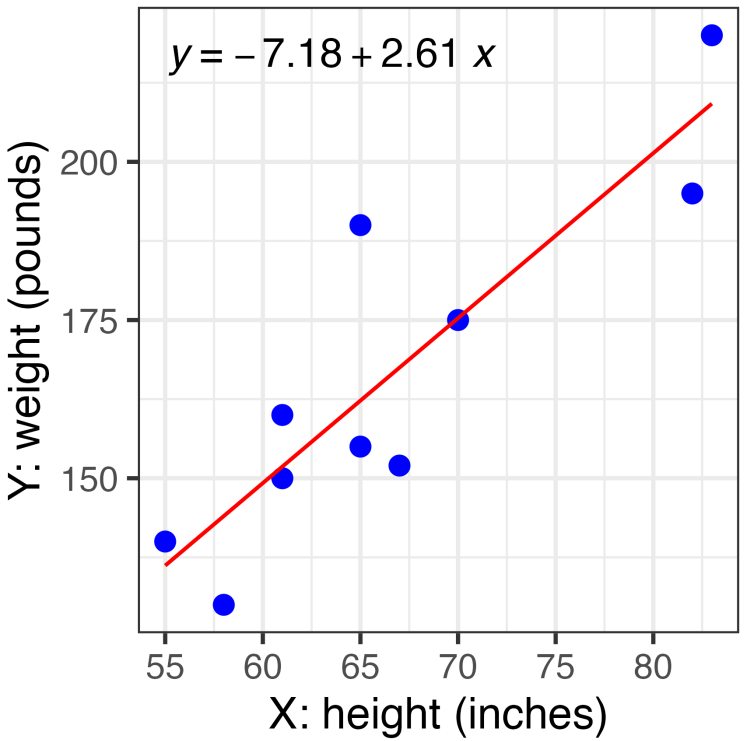
E.g., slope = 2.6
then for every 1 unit of X
Y increases by 2.6 units

Regression Equation

$$Y_i = \hat{Y}_i + \varepsilon_i$$

$$\hat{Y}_i = \beta_0 + \beta_1 X_i$$

- $\hat{Y}_i$ = **estimate** of the **dependent variable** Y_i
- β_0 = **y-intercept** of the regression line
- β_1 = **slope** of the regression line
- X_i = i^{th} value of the **independent variable** X
- ε_i = i^{th} **residual** of the regression line



Quantifying regression fit: R^2

Sum of squared residuals is SS_{res} : $SS_{res} = \sum_i (Y_i - \hat{Y}_i)^2$

Total variability in Y is SS_{tot} : $SS_{tot} = \sum_i (Y_i - \bar{Y})^2$

- Y_i are actual values of Y
- $\hat{Y}_i$ are predicted values of Y using the regression line
- $\bar{Y}$ is the mean Y across all observations Y_i for $i = 1 \ldots N$

Quantifying regression fit: R^2

Sum of squared residuals is SS_{res} : $SS_{res} = \sum_i (Y_i - \hat{Y}_i)^2$

Total variability in Y is SS_{tot} : $SS_{tot} = \sum_i (Y_i - \bar{Y})^2$

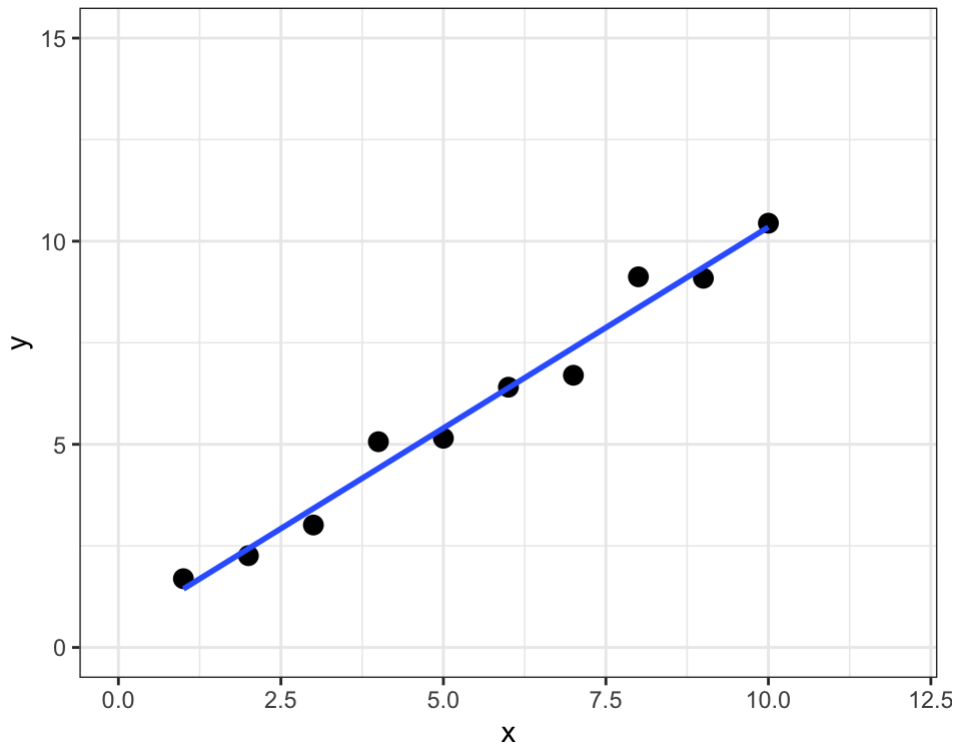
Coefficient of determination R^2 :

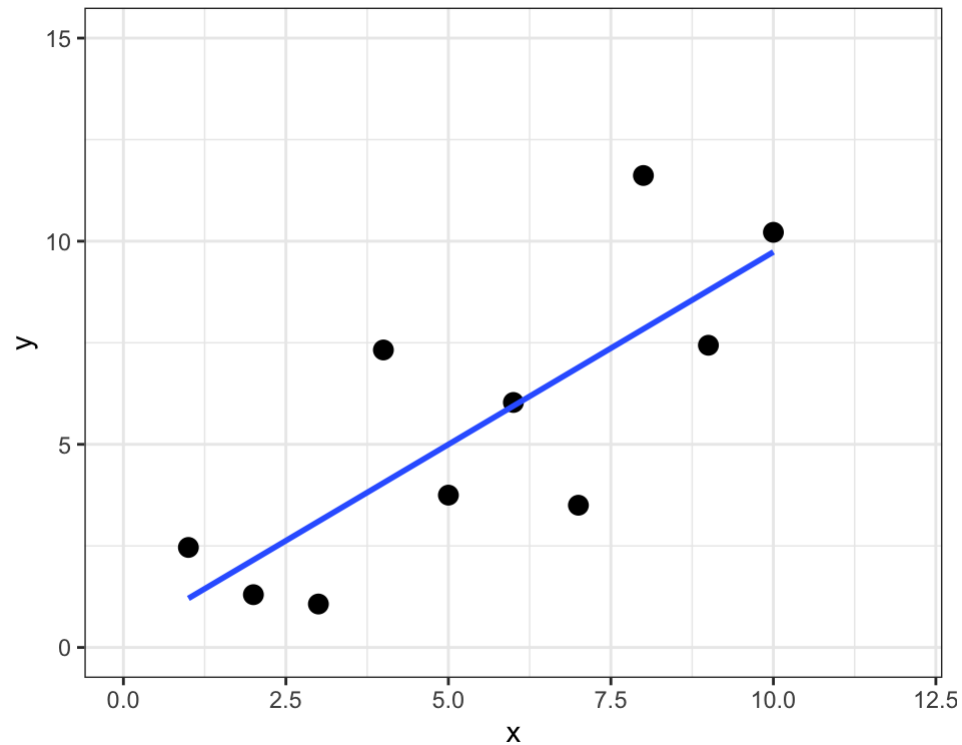
$$R^2 = 1 - \frac{SS_{res}}{SS_{tot}}$$

R^2 is the proportion of variance in the outcome variable Y that can be accounted for by the predictor variable X

R^2 always falls between 0.0 and 1.0

Quantifying regression fit: R^2






```
sum of squared residuals = 1.9
sum of squared total = 82.6
R-squared = 0.98
```

```
sum of squared residuals = 46.5
sum of squared total = 120.6
R-squared = 0.61
```

Quantifying regression fit: R^2

- $R^2 = 0.78$

  → 78% of the variance in Y can be accounted for by X

- $R = 0.22$

  → only 4.8% (0.22 x 0.22) of the variance in Y can be accounted for by X

Quantifying regression fit: σ_{est}

- σ_{est} is the **standard error of the estimate**
- σ_{est} is a measure of accuracy of predicting Y using X
- whereas R^2 is always between 0.0 and 1.0,
  → σ_{est} is **in units of Y**, the predicted variable
- this makes σ_{est} a useful measure of model fit

Quantifying regression fit: σ_{est}

$$\sigma_{est} = \sqrt{\frac{\sum(Y - \hat{Y})^2}{N}}$$

- σ_{est} is a measure of accuracy of predicting Y using X
- σ_{est} is **in units of Y**, the predicted variable

Quantifying regression fit: σ_{est}

$$\sigma_{est} = \sqrt{\frac{\sum(Y - \hat{Y})^2}{N}}$$

- σ_{est} is for a **population**
- For a **sample** the notation is s_{est}

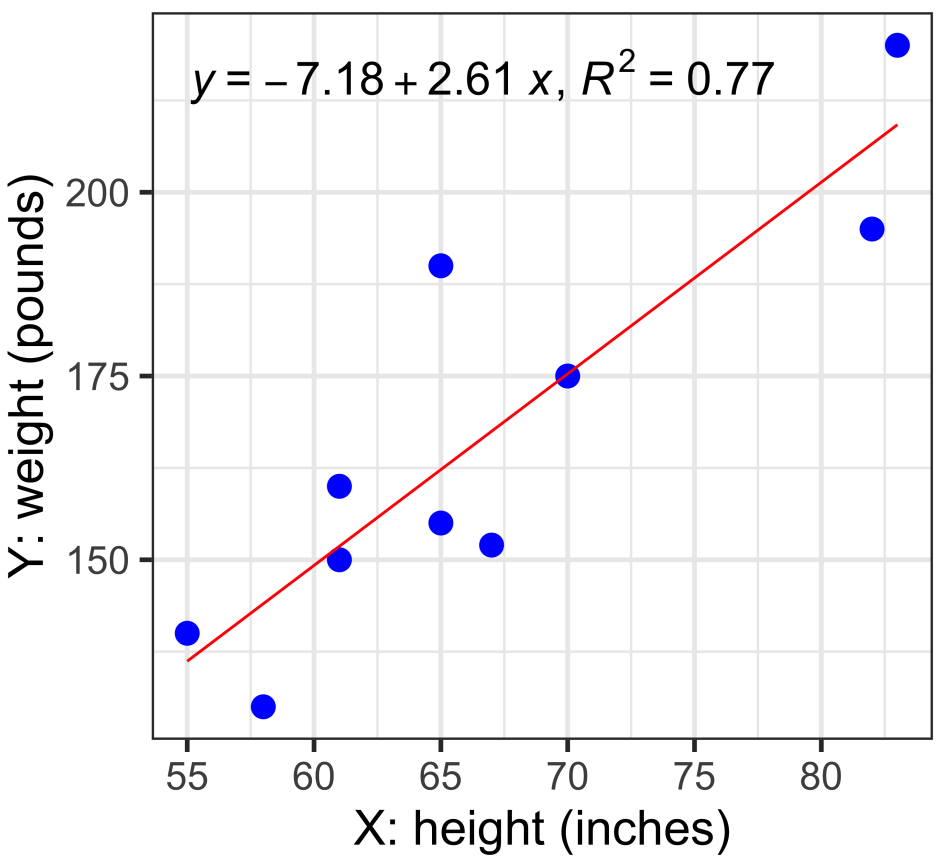
Quantifying regression fit: s_{est}

- For a **sample** the notation is s_{est}

$$s_{est} = \sqrt{\frac{\sum_i (Y_i - \hat{Y}_i)^2}{N - 2}}$$

- $N - 2$ because we estimate 2 parameters from our sample (slope and intercept of regression line)

Quantifying regression fit: s_{est}

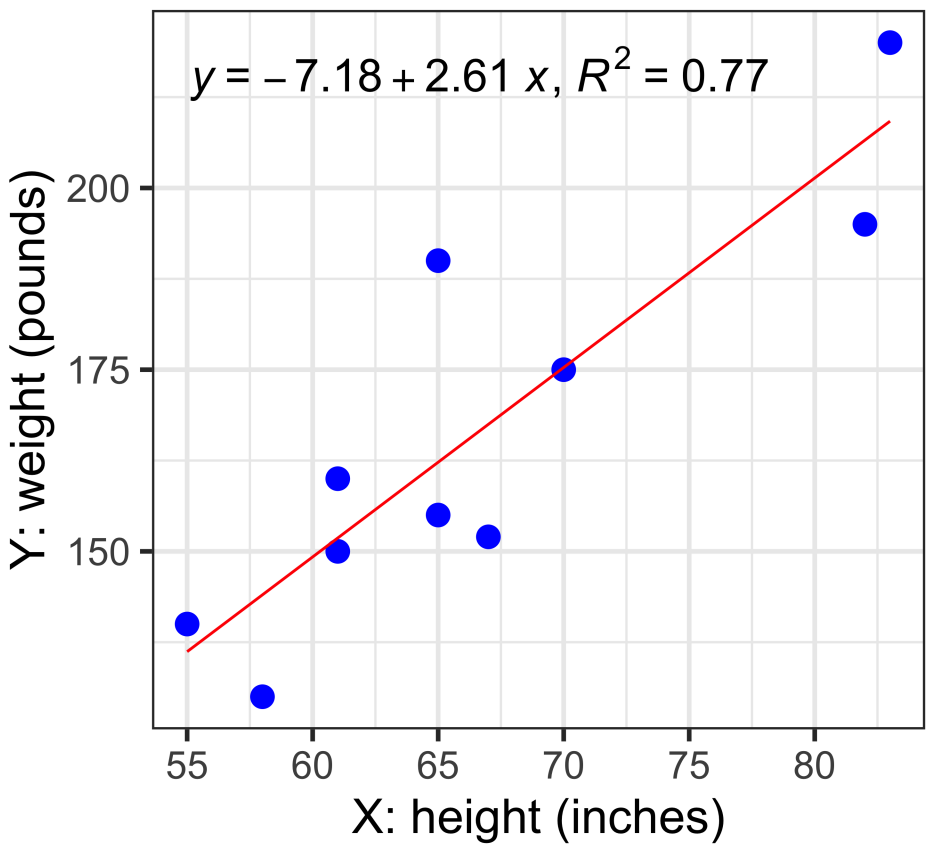


$$Y = -7.2 + 2.6X$$

$$R^2 = 0.772$$

$$s_{est} = 14.11 \text{ (pounds)}$$

Prediction



- predict values of y given:
  - a new value of x
  - the regression equation

Prediction

$$intercept = b = \frac{\sum y \sum x^2 - \sum x \sum xy}{n \sum x^2 - (\sum x)^2}$$

$$slope = m = \frac{n \sum xy - \sum x \sum y}{n \sum x^2 - (\sum x)^2}$$

- for a person with **height = 75 inches**
  → what is their weight?
- regression equation:
  weight = -7.18 + (2.61 * height)
  weight = -7.18 + (2.61 * 75)
  weight = -7.18 + 195.75
  weight = 188.57 pounds
- from `summary()` of our `lm()` in R:
- s_{est} = 14.11 pounds

Prediction

scores	x	y	x_squared	y_squared	xy
1	1	2	1	4	2
2	4	5	16	25	20
3	3	1	9	1	3
4	6	8	36	64	48
5	5	6	25	36	30
6	7	8	49	64	56
7	8	9	64	81	72
Sums	34	39	200	275	231

- for a person with **height = 57 inches**
  → what is their weight?
- regression equation:
  weight = -7.18 + (2.61 * height)
  weight = -7.18 + (2.61 * 57)
  weight = -7.18 + 148.77
  weight = 141.59 pounds
- from `summary()` of our `lm()` in R:
- s_{est} = 14.11 pounds

Prediction

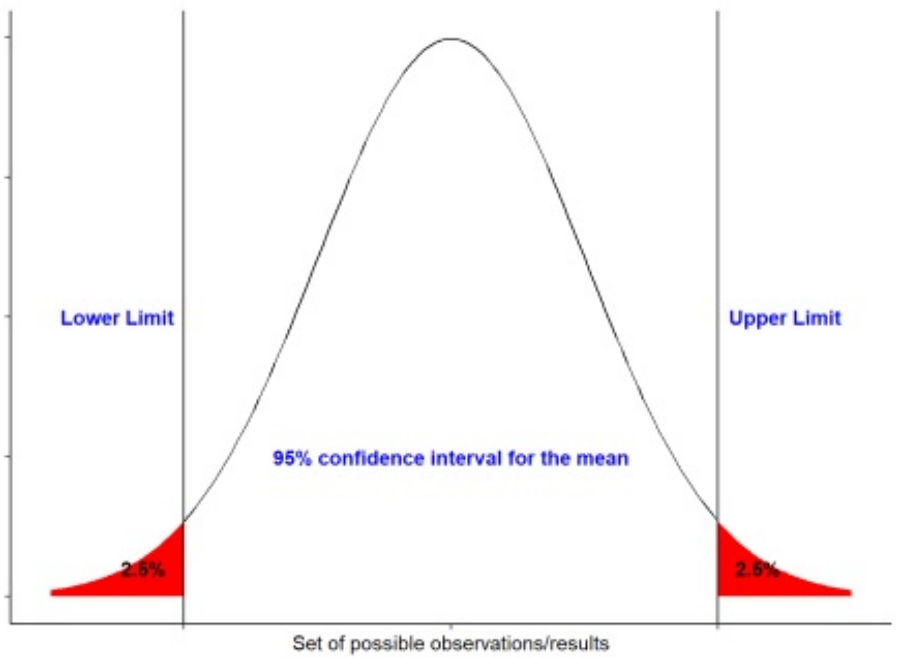
$$intercept = b = \frac{\sum y \sum x^2 - \sum x \sum xy}{n\sum x^2 - (\sum x)^2} = \frac{39*200 - 34*231}{7*200 - 34^2} = -.221$$

$$slope = m = \frac{n\sum xy - \sum x \sum y}{n\sum x^2 - (\sum x)^2} = \frac{7*231 - 34*39}{7*275 - 34^2} = 1.19$$

- for a person with **height = 0 inches**
  → what is their weight?
- regression equation:
  weight = -7.18 + (2.61 * height)
  weight = -7.18 + (2.61 * 0) weight = -7.18 + 0
  weight = -7.18 pounds
  ???
- predictions are only valid within the range of observed data
- extrapolate at your own risk!

Confidence Intervals

- another way of characterizing the **precision** of estimates
  - of model coefficients (slope, intercept)
  - of model prediction (predicted Y given new X)
- 95% CI



Confidence Intervals

- let's first quickly review confidence intervals of **the mean of a sample** before we talk about confidence intervals of regression coefficients

Confidence Intervals

- recall* that the 95% CI of the **mean** of a sample is:

$$\bar{X} \pm t_{(0.975, N-1)} \left(\frac{s}{\sqrt{N}} \right)$$

```
x <- c(55,61,67,83,65,82,70,58,65,61)
N <- length(x)
ci_lower <- mean(x) - (qt(.975,N-1) * (sd(x)/sqrt(N)))
ci_upper <- mean(x) + (qt(.975,N-1) * (sd(x)/sqrt(N)))
(mean(x))
```

```
[1] 66.7
```

```
(c(ci_lower, ci_upper))
```

```
[1] 59.98047 73.41953
```

*see section 10.5 of Navarro text for review

Confidence Intervals

- recall* that the 95% CI of the **mean** of a sample is:

$$\bar{X} \pm t_{(0.975, N-1)} \left(\frac{s}{\sqrt{N}} \right)$$

```r
x <- c(55,61,67,83,65,82,70,58,65,61)
library(lsr) # you may need to install this package
ciMean(x, 0.95)
```

```
      2.5%    97.5%
x 59.98047 73.41953
```

*see section 10.5 of Navarro text for review

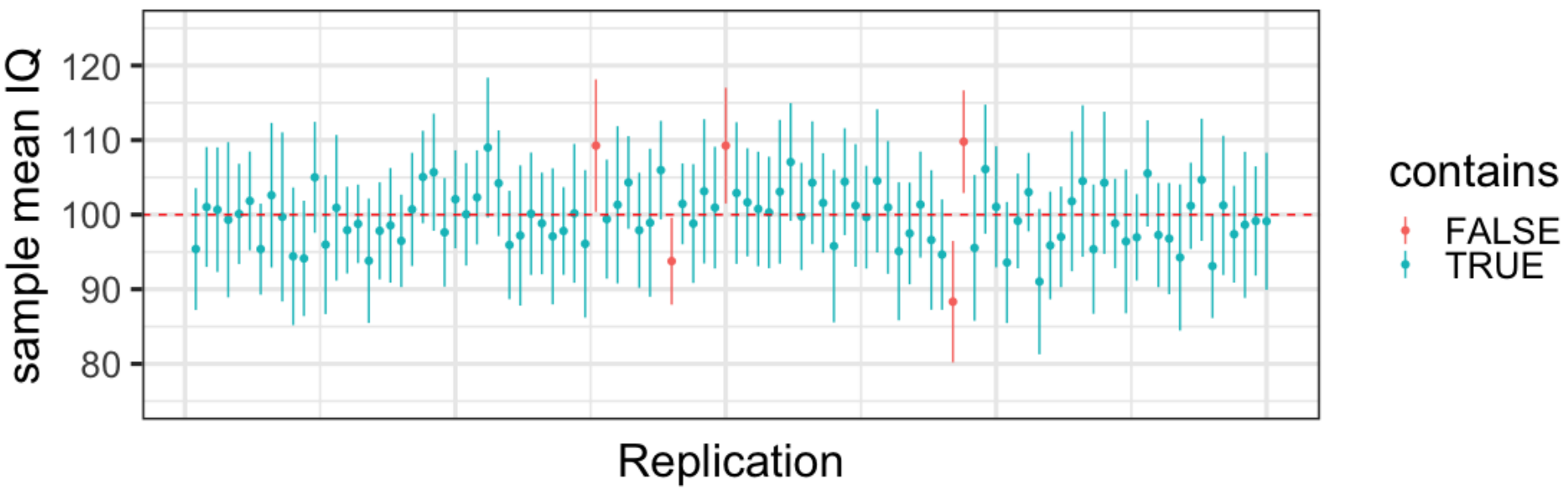
Confidence Intervals

- 95% CI of the mean of our **sample** is (59.98, 73.42)
- how to interpret* this?
- 95% chance the **population** mean is between 59.95 and 73.42
  → nope … but (subtly) close! → the “95% chance” is a statement about the **confidence interval**, not about the population mean

*also see section 10.5.2 of Navarro text

Confidence Intervals

- 95% CI of the mean of our sample is (59.98, 73.42)
- If we replicated the experiment over and over again, and computed a 95% confidence interval for each replication, then 95% of those confidence intervals would contain the population mean
- (but we will never know which ones!)



*also see section 10.5.2 of Navarro text

Confidence Intervals

- recall our regression model:
- $Y = -7.18 + 2.61X$
  - $\beta_0 = -7.18$
  - $\beta_1 = 2.61$
- we can also compute 95% CIs for coefficients (β_0, β_1)

Confidence Intervals

$$CI(b) = \hat{b} \pm \left(t_{crit} \times SE(\hat{b}) \right)$$

- $\hat{b}$ is each coefficient in the regression model
- t_{crit} is the critical value of t
- $SE(\hat{b})$ is the standard error of the regression coefficient

- in R it's easy, use `confint()`:

```
d <- tibble(x=c(55,61,67,83,65,82,70,58,65,61)
            y=c(140,150,152,220,190,195,175,13
mymod <- lm(y ~ x, data = d)
coef(mymod)
```

```
(Intercept)           x
  -7.176930    2.606851
```

```
confint(mymod)
```

```
                2.5 %    97.5 %
(Intercept) -84.898712 70.544852
x             1.451869  3.761832
```

Confidence Intervals

```r
d <- tibble(x=c(55,61,67,83,65,82,70,58,65,61),
            y=c(140,150,152,220,190,195,175,130,155,160))
mymod <- lm(y ~ x, data = d)
coef(mymod)
```

```
(Intercept)           x
  -7.176930    2.606851
```

```r
confint(mymod)
```

```
                2.5 %    97.5 %
(Intercept) -84.898712 70.544852
x             1.451869  3.761832
```

- interpretation of regression coefficient CIs is the same:
- If we were to replicate our sample a bunch of times, by resampling from the population and fitting a new regression model each time, and compute confidence intervals for the regression coefficients each time, then 95% of those CIs would contain the population value of the coefficients.

Hypothesis tests for Regression

```r
d <- tibble(x=c(55,61,67,83,65,82,70,58,65,61),
            y=c(140,150,152,220,190,195,175,130,155,160))
mymod <- lm(y ~ x, data = d)
summary(mymod)
```

```
Call:
lm(formula = y ~ x, data = d)

Residuals:
    Min      1Q  Median      3Q     Max
-15.482 -10.506  -1.072   7.069  27.732

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  -7.1769    33.7041  -0.213 0.836700
x             2.6069     0.5009   5.205 0.000818 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 14.11 on 8 degrees of freedom
Multiple R-squared:  0.772, Adjusted R-squared:  0.7435
F-statistic: 27.09 on 1 and 8 DF,  p-value: 0.0008176
```

??

$$F(1, 8) = 27.09$$

$$p = 0.0008176$$

→ what is this hypothesis test **of**?

Hypothesis tests for Regression

- This is a test of the model "as a whole"*
- specifically a test of the "full" regression model *versus* a "restricted" version of the model in which **there is no dependence of Y on X**
  - (i.e. the **slope** β_1 is zero)
- in this way the hypothesis test is essentially a test of whether $\beta_1 = 0$ or not
- $Y = \beta_0 + \beta_1 X$: *full model* (our alternate hypothesis H_1)
- $Y = \beta_0$: *restricted model* (our null hypothesis H_0)

*see section 15.5 of Navarro text

Hypothesis tests for Regression

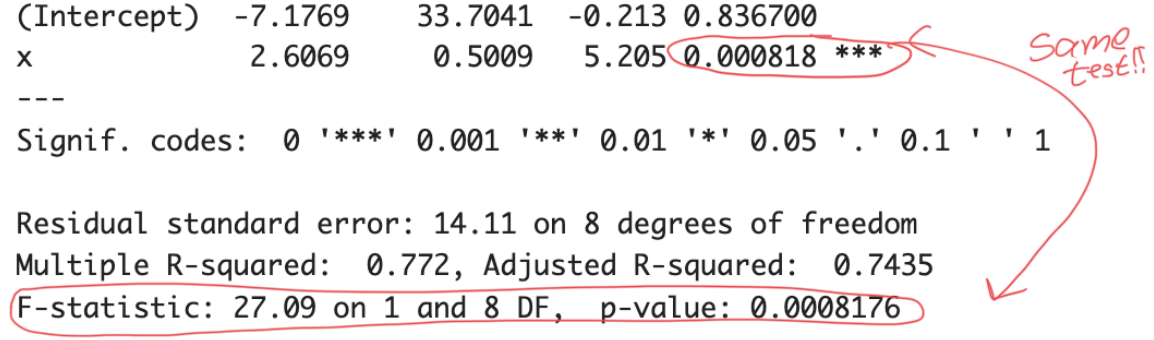
```r
d <- tibble(x=c(55,61,67,83,65,82,70,58,65,61),
            y=c(140,150,152,220,190,195,175,130,155,160))
mymod <- lm(y ~ x, data = d)
summary(mymod)
```

```
Call:
lm(formula = y ~ x, data = d)

Residuals:
    Min      1Q  Median      3Q     Max
-15.482 -10.506  -1.072   7.069  27.732

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  -7.1769    33.7041  -0.213 0.836700
x             2.6069     0.5009   5.205 0.000818 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 14.11 on 8 degrees of freedom
Multiple R-squared:  0.772, Adjusted R-squared:  0.7435
F-statistic: 27.09 on 1 and 8 DF,  p-value: 0.0008176
```



- F test of model as a whole tests full model vs restricted model
- $Y = \beta_0 + \beta_1 X$: full
- $Y = \beta_0$: restricted
- When there is only one dependent variable (X) in the regression model, (and hence only one slope β_1), then this is equivalent to a test of whether the slope is zero

Hypothesis tests for Regression

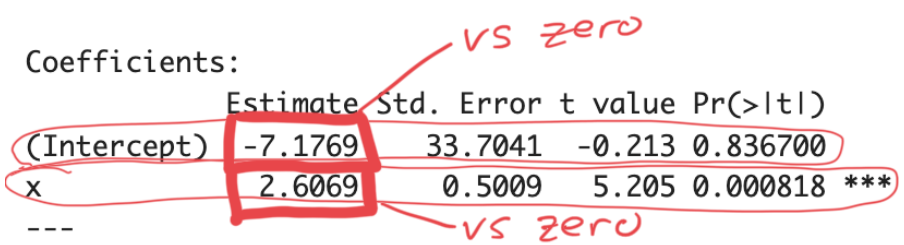
```r
d <- tibble(x=c(55,61,67,83,65,82,70,58,65,61),
            y=c(140,150,152,220,190,195,175,130,155,160))
mymod <- lm(y ~ x, data = d)
summary(mymod)
```

```
Call:
lm(formula = y ~ x, data = d)

Residuals:
    Min      1Q  Median      3Q     Max
-15.482 -10.506  -1.072   7.069  27.732

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  -7.1769    33.7041  -0.213 0.836700
x             2.6069     0.5009   5.205 0.000818 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 14.11 on 8 degrees of freedom
Multiple R-squared:  0.772, Adjusted R-squared:  0.7435
F-statistic: 27.09 on 1 and 8 DF,  p-value: 0.0008176
```



- also: hypothesis tests on **model coefficients**
- null hypothesis H_0: coefficient = zero
- alternate hypothesis H_1: coefficient is not zero
- intercept (β_0): p = 0.836700
- slope (β_1): p = 0.000818

Hypothesis tests for Regression

- intercept (β_0) = -7.1769
  p = 0.836700
- slope (β_1) = 2.6069
  p = 0.000818
- what do these p-values mean precisely?

Hypothesis tests for Regression

- intercept (β_0) = -7.1769
  p = 0.836700
- Under the null hypothesis H_0 the probability of obtaining an intercept as large (farthest from zero) as -7.1769 due to random sampling is 83.67%
- That is pretty high! We cannot really reject H_0
- We cannot reject H_0 (that the intercept = zero)
- We infer that the intercept is most likely = zero

Hypothesis tests for Regression

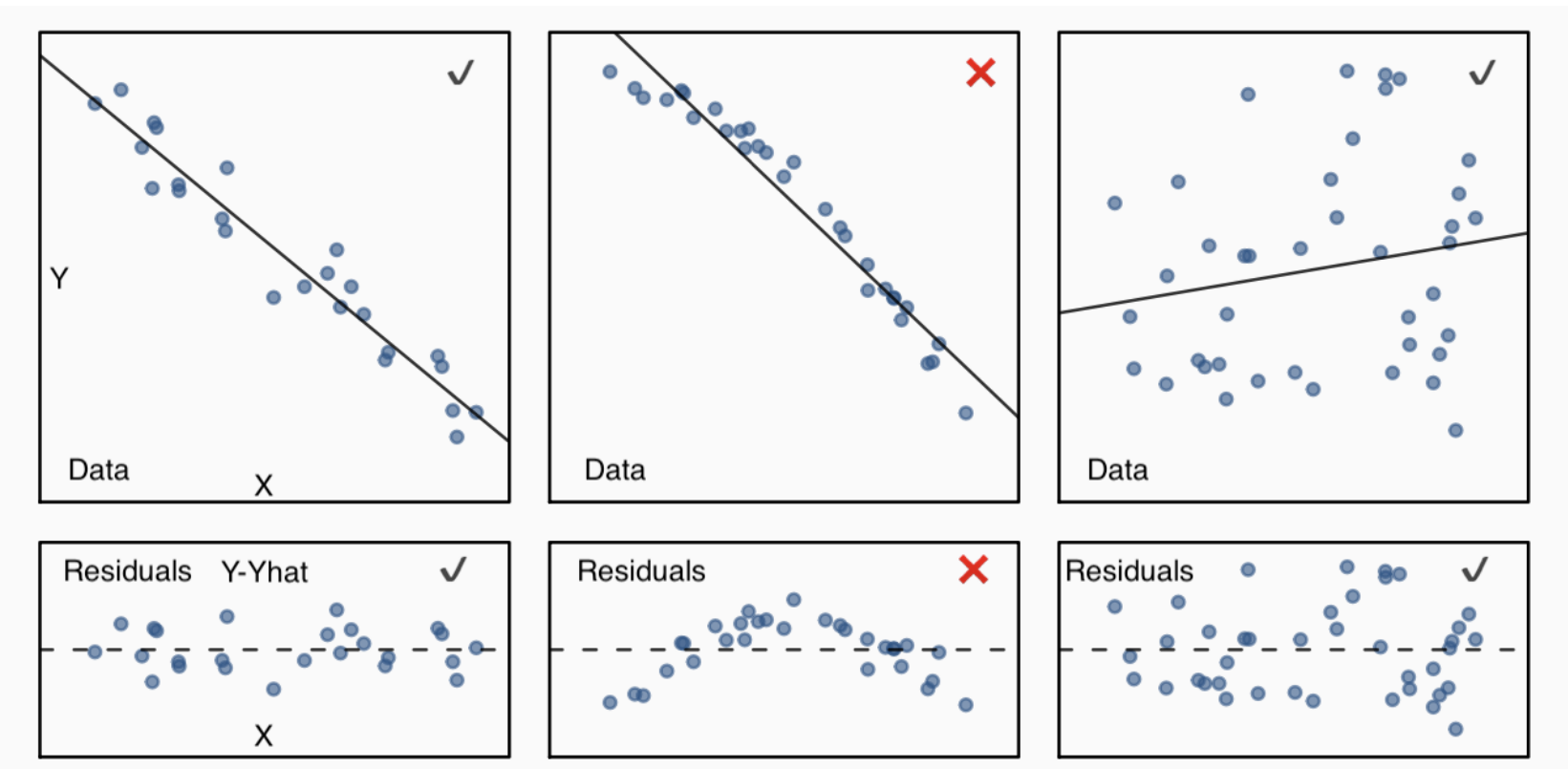
- slope (β_1) = 2.6069
  p = 0.000818
- Under the null hypothesis H_0 the probability of obtaining a slope as large (farthest from zero) as 2.6069 due to random sampling is 0.0818%
- That is pretty low! We will reject H_0 that the slope is zero
- The slope is not zero. What is it?
- Our estimate of the slope is $\beta_1 = 2.6069$
- Our 95% confidence interval is [1.451869, 3.761832] from `confint()`

Assumptions of Linear Regression

1. **Linearity**: linear relationship between X and Y
2. **Normality**: (nearly) normally distributed residuals
3. **Homoscedasticity**: Constant variance of Y across the range of X
4. **Outliers**: no extreme outliers
5. **Independence**: observations are independent of each other

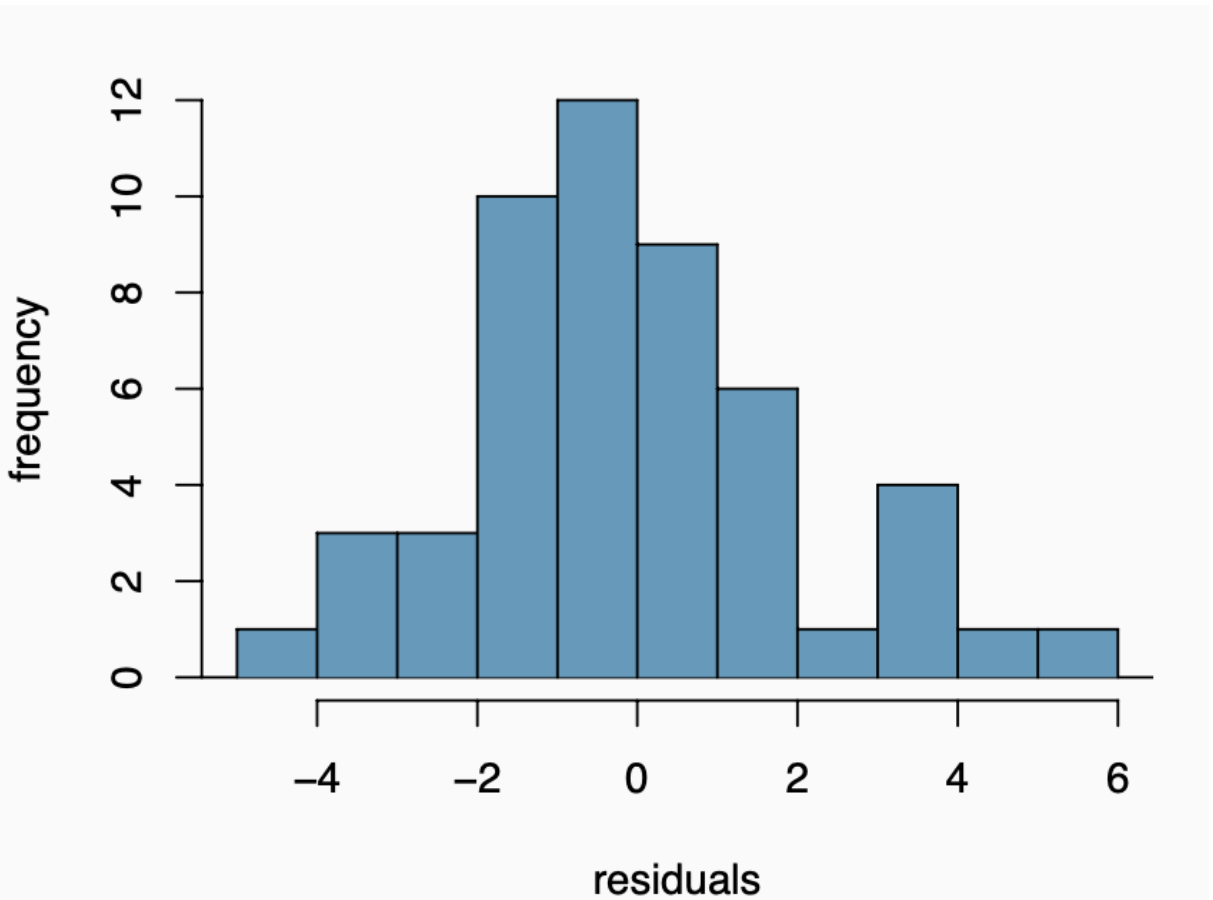
1. Linearity

- relationship between the predictor variable (X) and the predicted variable (Y) should be linear
- **check** with a scatterplot either of the data or residuals



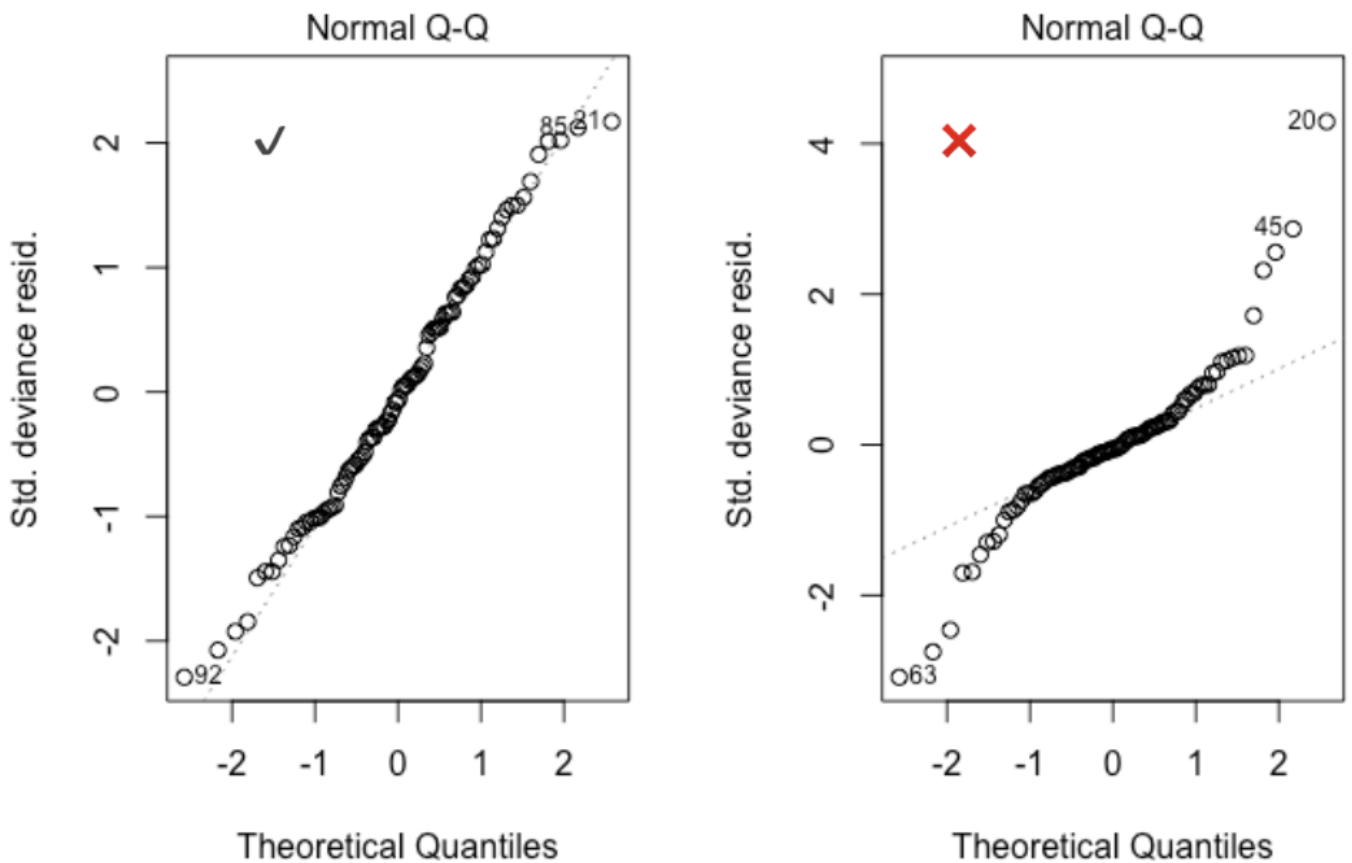
2. Normality

- the residuals should be nearly normally distributed
- can check using a histogram of the residuals:



2. Normality

- the residuals should be nearly normally distributed
- can also check using a Q-Q plot:

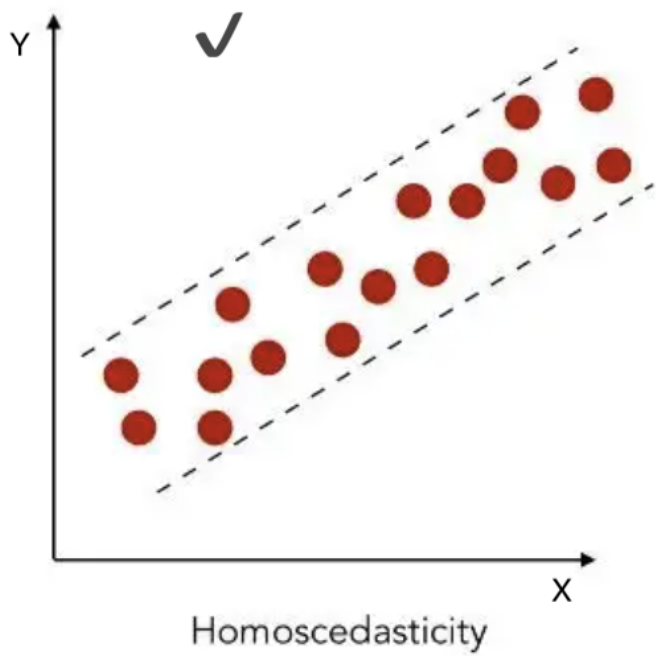


2. Normality

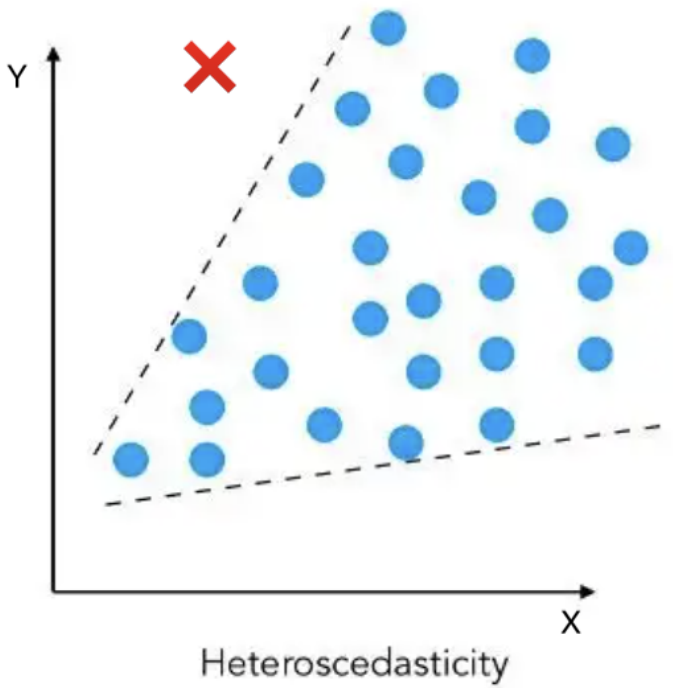
- statistical test for normality: `shapiro.test()` (Shapiro-Wilk test)
- null hypothesis H_0: data are normally distributed
- alternate hypothesis H_1: data are not normally distributed
- p < .05: reject the null hypothesis
  - conclude data are not normally distributed
- test the residuals: `shapiro.test(residuals(my.mod))`

3. Homoscedasticity

- homogeneity of variance
- variance of Y is the same across the range of X values
- plot X vs Y data:



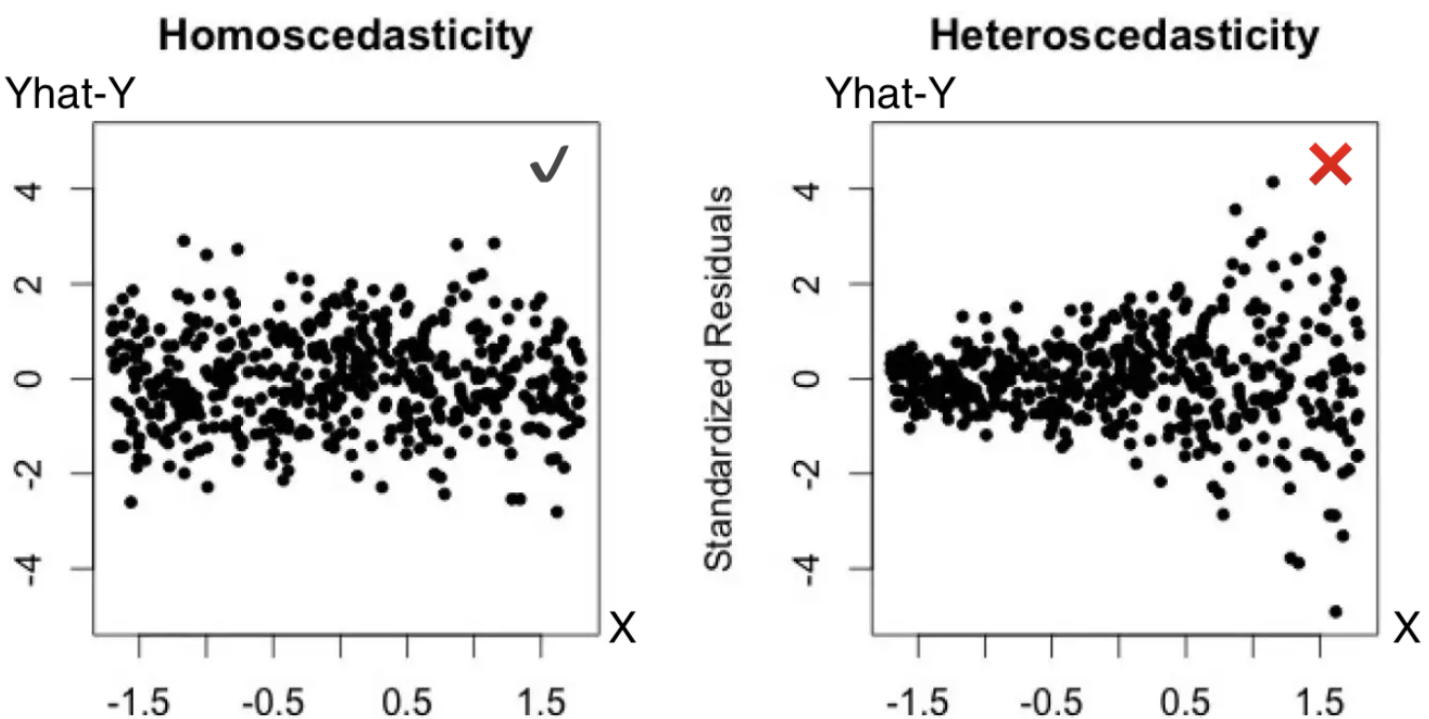
Homoscedasticity



Heteroscedasticity

3. Homoscedasticity

- homogeneity of variance
- variance of Y is the same across the range of X values
- or: plot X vs Residuals ($Y - \hat{Y}$)

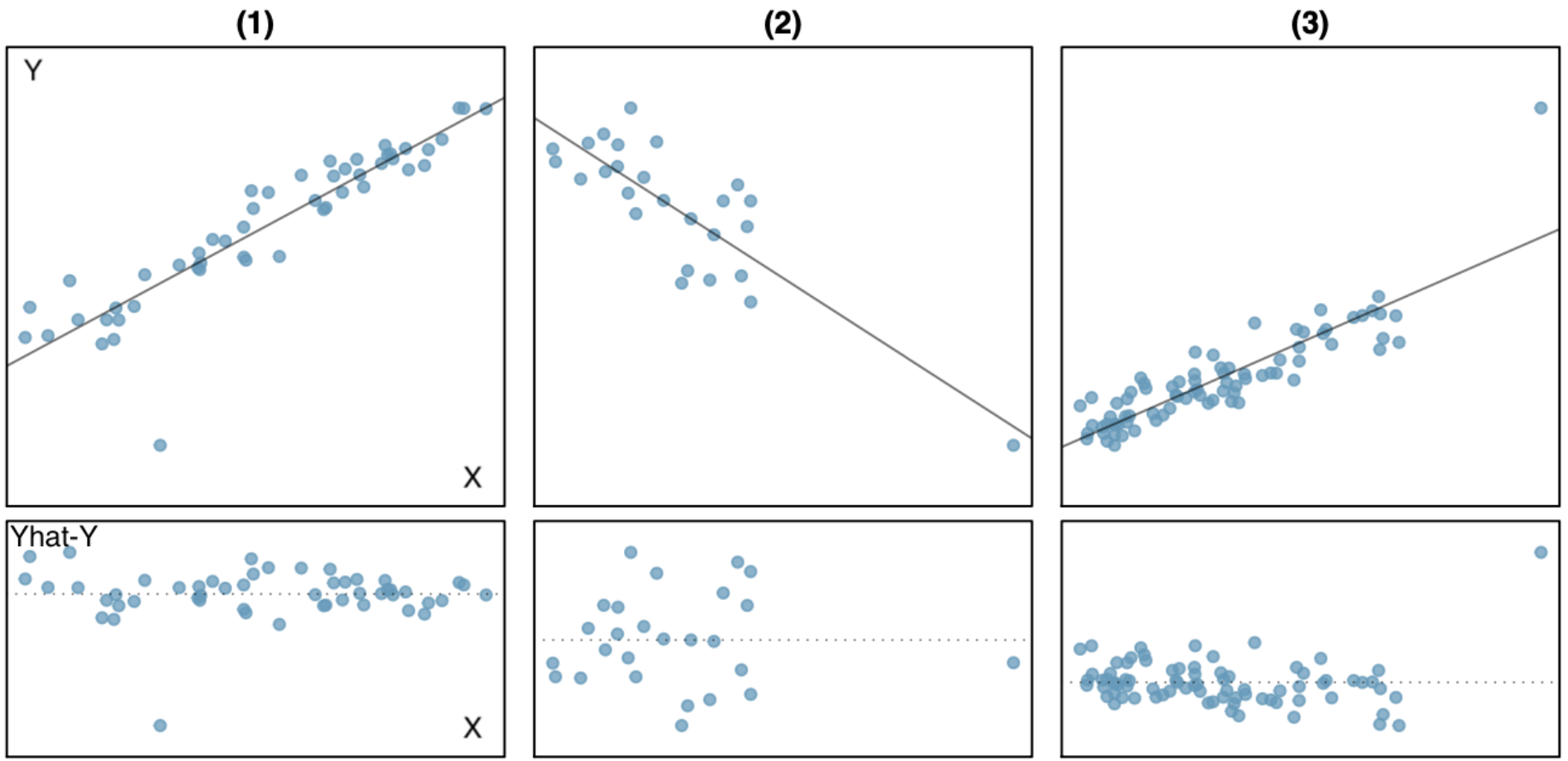


3. Homoscedasticity

- statistical test for homoscedasticity: `bptest()` (Breusch-Pagan test)
- in the `car` package: `ncvTest()` (non-constant variance test)
- null hypothesis H_0: homoscedasticity
- alternate hypothesis H_1: heteroscedasticity
- p < .05: reject the null hypothesis
  - conclude data are heteroscedastic
- apply the `ncvTest()` to the `lm()` model object: `ncvTest(my.mod)`

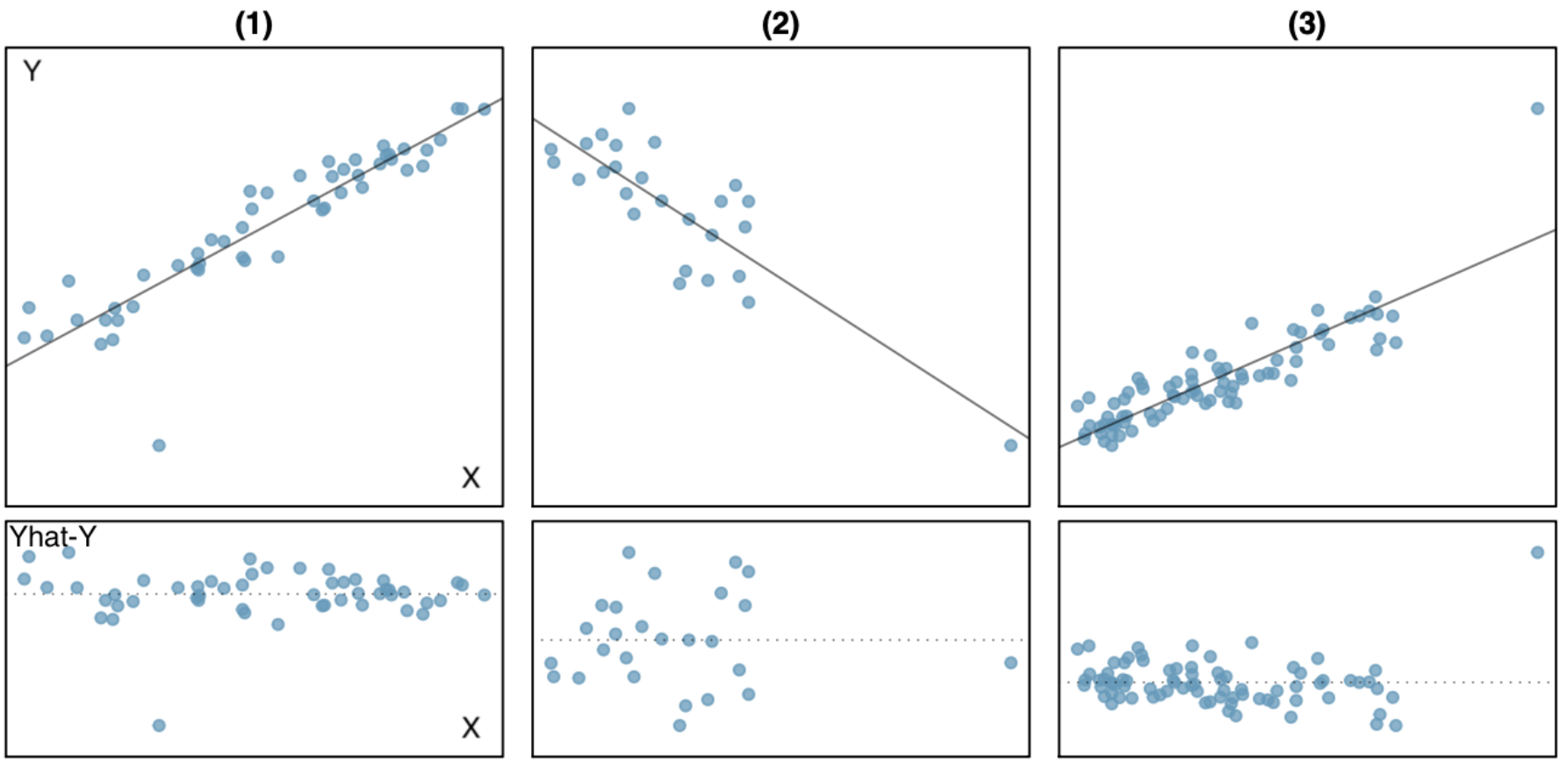
4. Outliers

- we should not fit our linear model to a dataset that includes extreme outliers



4. Outliers

- Cook's distance is one quantitative way of identifying observations that have a disproportionate influence on the regression line
- see Navarro text section 15.9



Violations: what to do?

- **Linearity**: If your data are not linear don't model it using a linear model! Non-linear methods do exist.
- **Normality**: if severe, consider transforming the dependent variable, e.g. using logarithm, square root, Box-Cox transformation, etc.
- **Homoscedasticity**: could consider transforming the dependent variable; could also use weighted regression
- **Outliers**: remove them!