Simple Linear Regression

Prof Wells

STA 295: Stat Learning

February 6th, 2024

Outline

In today's class, we will. . .

- Discuss theoretical foundation for linear regression
- Perform inference for simple linear models
- Implement simple linear regression in R

Section 1

Foundations

Linear Regression

- Suppose we have one or more predictors $(X_1, X_2, \ldots, X_p)$ and a *quantitative* response variable Y, and that

$$Y = f(X_1, \ldots, X_p) + \epsilon$$

Linear Regression

- Suppose we have one or more predictors $(X_1, X_2, \dots, X_p)$ and a *quantitative* response variable Y, and that
$$Y = f(X_1, \dots, X_p) + \epsilon$$
- The function f could theoretically take many forms. But the simplest form assumes f is a linear function:
$$f(x_1, x_2, \dots, x_p) = \beta_0 + \beta_1 x_1 + \dots + \beta_p x_p$$

Linear Regression

- Suppose we have one or more predictors $(X_1, X_2, \ldots, X_p)$ and a *quantitative* response variable Y, and that

$$Y = f(X_1, \ldots, X_p) + \epsilon$$

- The function f could theoretically take many forms. But the simplest form assumes f is a linear function:

$$f(x_1, x_2, \ldots, x_p) = \beta_0 + \beta_1 x_1 + \cdots + \beta_p x_p$$

  - Note: a change in f is constant per unit change in any of the inputs.

Linear Regression

- Suppose we have one or more predictors $(X_1, X_2, \ldots, X_p)$ and a *quantitative* response variable Y, and that
$$Y = f(X_1, \ldots, X_p) + \epsilon$$
- The function f could theoretically take many forms. But the simplest form assumes f is a linear function:
$$f(x_1, x_2, \ldots, x_p) = \beta_0 + \beta_1 x_1 + \cdots + \beta_p x_p$$
  - Note: a change in f is constant per unit change in any of the inputs.
- If Y depends on only 1 predictor X, then the linear model reduces to
$$y = \hat{f}(x) = \beta_0 + \beta_1 x$$

Linear Regression

- Suppose we have one or more predictors $(X_1, X_2, \ldots, X_p)$ and a *quantitative* response variable Y, and that
$$Y = f(X_1, \ldots, X_p) + \epsilon$$
- The function f could theoretically take many forms. But the simplest form assumes f is a linear function:
$$f(x_1, x_2, \ldots, x_p) = \beta_0 + \beta_1 x_1 + \cdots + \beta_p x_p$$
    - Note: a change in f is constant per unit change in any of the inputs.
- If Y depends on only 1 predictor X, then the linear model reduces to
$$y = \hat{f}(x) = \beta_0 + \beta_1 x$$
- We'll use **Simple Linear Regression** (SLR) to build intuition about all linear models

Approximations and Estimates

- In reality, the relationship f between Y and $X_1, \ldots, X_p$ may not be linear

Approximations and Estimates

- In reality, the relationship f between Y and $X_1, \ldots, X_p$ may not be linear
- But many functions can be well-approximated by linear ones (especially when inputs are restricted to a small range)

Approximations and Estimates

- In reality, the relationship f between Y and $X_1, \ldots, X_p$ may not be linear
- But many functions can be well-approximated by linear ones (especially when inputs are restricted to a small range)
- But even if f is truly linear, we still have problems: We do not know the parameters of the linear model.

Approximations and Estimates

- In reality, the relationship f between Y and $X_1, \ldots, X_p$ may not be linear
- But many functions can be well-approximated by linear ones (especially when inputs are restricted to a small range)
- But even if f is truly linear, we still have problems: We do not know the parameters of the linear model.
- Based on data, we estimate the parameters to create an estimated linear model

$$\hat{f} = \hat{\beta}_0 + \hat{\beta}_1 x_1 + \cdots + \hat{\beta}_p x_p$$

Approximations and Estimates

- In reality, the relationship f between Y and $X_1, \ldots, X_p$ may not be linear
- But many functions can be well-approximated by linear ones (especially when inputs are restricted to a small range)
- But even if f is truly linear, we still have problems: We do not know the parameters of the linear model.
- Based on data, we estimate the parameters to create an estimated linear model

$$\hat{f} = \hat{\beta}_0 + \hat{\beta}_1 x_1 + \cdots + \hat{\beta}_p x_p$$

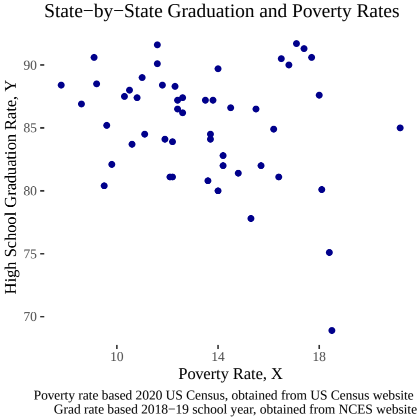
- So we are **estimating** an **approximation** to a relationship between response and predictors.

SLR Review

Consider the relationship between a state's high school grad rate Y and its poverty rate X.

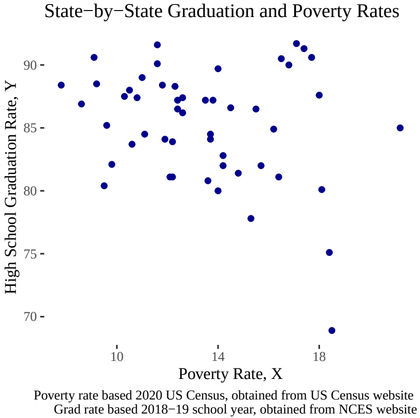
SLR Review

Consider the relationship between a state's high school grad rate Y and its poverty rate X.



SLR Review

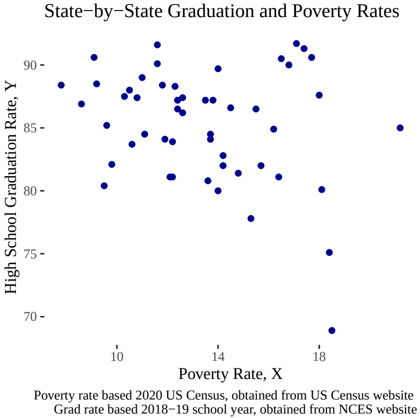
Consider the relationship between a state's high school grad rate Y and its poverty rate X.



- Suppose we want to model Y as a function of X

SLR Review

Consider the relationship between a state's high school grad rate Y and its poverty rate X.

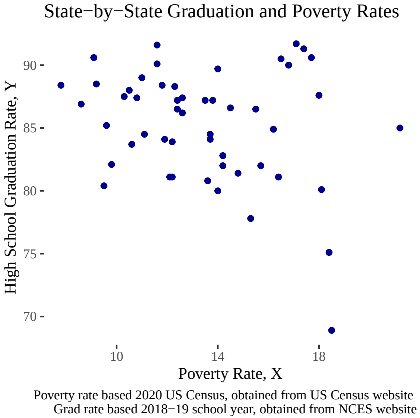


- Suppose we want to model Y as a function of X
- Let's assume a linear relationship

$$Y = \beta_0 + \beta_1 X + \epsilon$$

SLR Review

Consider the relationship between a state's high school grad rate Y and its poverty rate X.



- Suppose we want to model Y as a function of X
- Let's assume a linear relationship

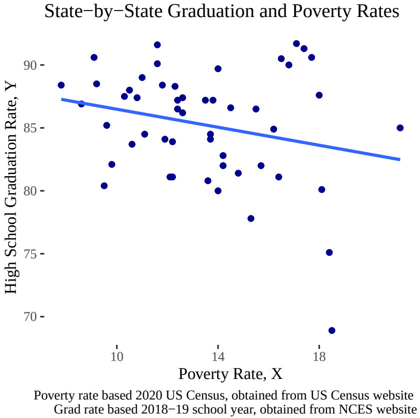
$$Y = \beta_0 + \beta_1 X + \epsilon$$

- Fitted Model:

$$\hat{Y} = \hat{\beta}_0 + \hat{\beta}_1 X = 90 - 0.4X$$

SLR Review

Consider the relationship between a state's high school grad rate Y and its poverty rate X.



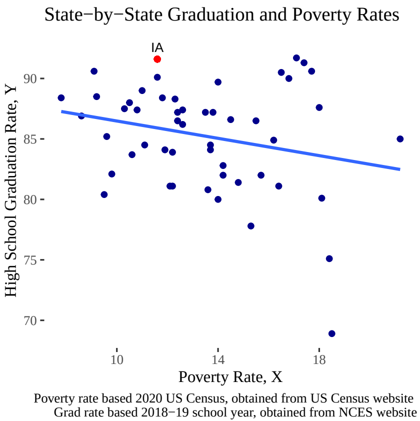
- Suppose we want to model Y as a function of X
- Let's assume a linear relationship

$$Y = \beta_0 + \beta_1 X + \epsilon$$

- Fitted Model:

$$\hat{Y} = \hat{\beta}_0 + \hat{\beta}_1 X = 90 - 0.4X$$

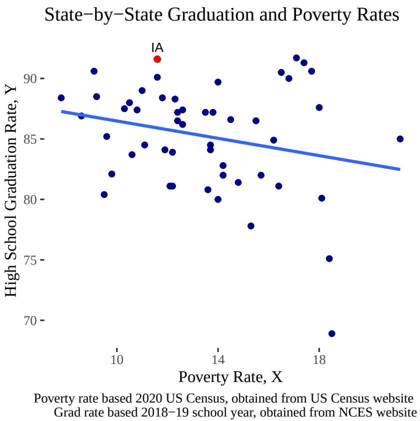
Model Predictions



- Model:

$$\hat{Y} = 90 - 0.4 \cdot X$$

Model Predictions

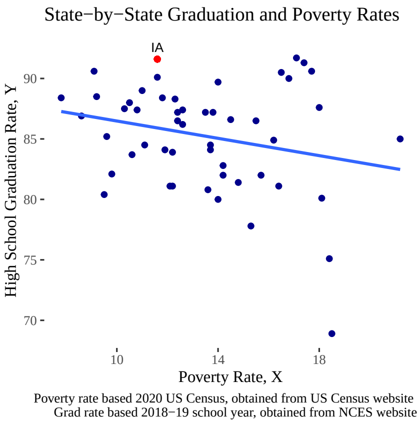


- Model:

$$\hat{Y} = 90 - 0.4 \cdot X$$

- Iowa has a poverty rate of 11.6. What does the model predict is Iowa's graduation rate?

Model Predictions



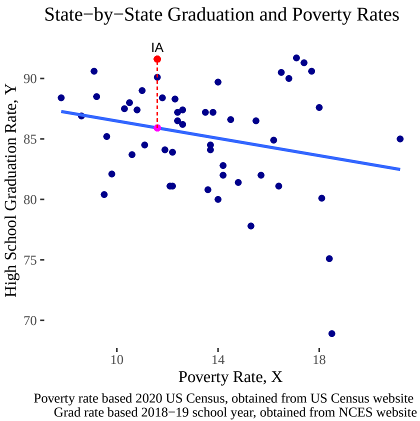
- Model:

$$\hat{Y} = 90 - 0.4 \cdot X$$

- Iowa has a poverty rate of 11.6. What does the model predict is Iowa's graduation rate?

$$\hat{Y} = 90 - 0.4 \cdot 11.6 = 85.36$$

Model Predictions



- Model:

$$\hat{Y} = 90 - 0.4 \cdot X$$

- Iowa has a poverty rate of 11.6. What does the model predict is Iowa's graduation rate?

$$\hat{Y} = 90 - 0.4 \cdot 11.6 = 85.36$$

But Iowa's actual graduation rate is 91.6

Residuals

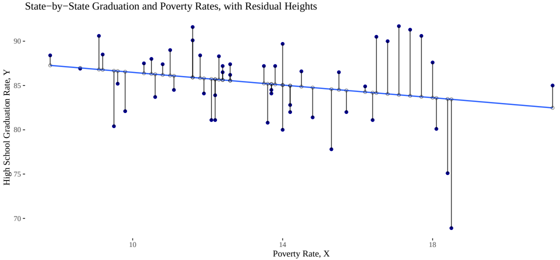
- **Residuals** are the leftover variation in the data after accounting for model fit.
- Each observation (X_i, Y_i) has its own residual e_i, which is the difference between the observed (Y_i) and predicted $(\hat{Y}_i)$ value:

$$e_i = Y_i - \hat{Y}_i$$

Residuals

- **Residuals** are the leftover variation in the data after accounting for model fit.
- Each observation (X_i, Y_i) has its own residual e_i, which is the difference between the observed (Y_i) and predicted $(\hat{Y}_i)$ value:

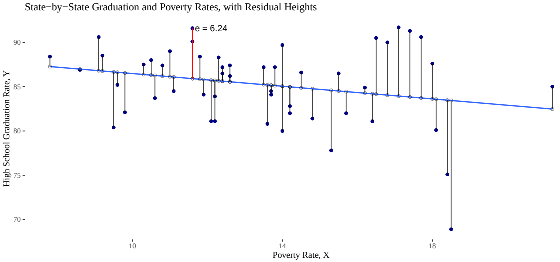
$$e_i = Y_i - \hat{Y}_i$$



Residuals

- **Residuals** are the leftover variation in the data after accounting for model fit.
- Each observation (X_i, Y_i) has its own residual e_i, which is the difference between the observed (Y_i) and predicted $(\hat{Y}_i)$ value:

$$e_i = Y_i - \hat{Y}_i$$

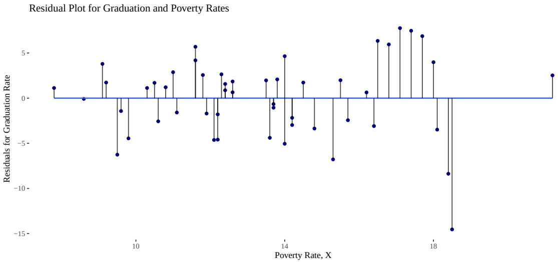


- Iowa's residual is

$$e = Y - \hat{Y} = 91.6 - 85.36 = 6.24$$

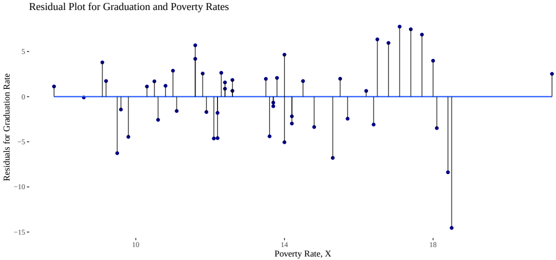
Residual Plot

- To visualize the degree of accuracy of a linear model, we use residual plots:



Residual Plot

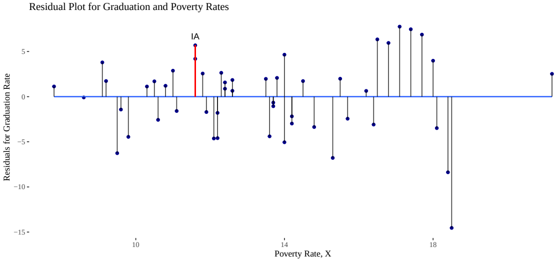
- To visualize the degree of accuracy of a linear model, we use residual plots:



- Points preserve original x-position, but with y-position equal to residual.

Residual Plot

- To visualize the degree of accuracy of a linear model, we use residual plots:



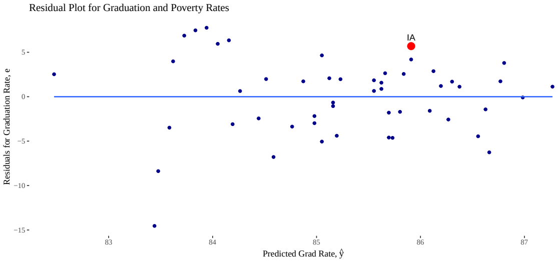
- Points preserve original x-position, but with y-position equal to residual.

Residual Plot

In many cases, it is more convenient to look at the residual plot of residuals vs **fitted values** (instead of vs X)

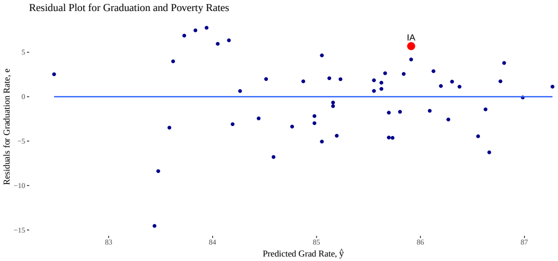
Residual Plot

In many cases, it is more convenient to look at the residual plot of residuals vs **fitted values** (instead of vs X)



Residual Plot

In many cases, it is more convenient to look at the residual plot of residuals vs **fitted values** (instead of vs X)



- This residual plot can still be used to determine accuracy of model, but can be used when we have more than 1 predictor.

Residual Sum of Squares

- Define the **Residual Sum of Squares** (RSS) as

$$\mathrm{RSS} = \sum_{i=1}^{n} (y_i - \hat{y}_i)^2 = e_1^2 + \cdots + e_n^2$$

Residual Sum of Squares

- Define the **Residual Sum of Squares** (RSS) as

$$\mathrm{RSS} = \sum_{i=1}^{n} (y_i - \hat{y}_i)^2 = e_1^2 + \cdots + e_n^2$$

  - Note that $\mathrm{RSS} = n \cdot \mathrm{MSE}$.

Residual Sum of Squares

- Define the **Residual Sum of Squares** (RSS) as

$$\mathrm{RSS} = \sum_{i=1}^{n} (y_i - \hat{y}_i)^2 = e_1^2 + \cdots + e_n^2$$

  - Note that $\mathrm{RSS} = n \cdot \mathrm{MSE}$.

- Using calculus or linear algebra, we can show that RSS is minimized when

$$\hat{\beta}_1 = \frac{\sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y})}{\sum_{i=1}^{n}(x_i - \bar{x})^2} \qquad \hat{\beta}_0 = \bar{y} - \hat{\beta}_1 \bar{x}$$

Residual Sum of Squares

- Define the **Residual Sum of Squares** (RSS) as

$$\mathrm{RSS} = \sum_{i=1}^{n} (y_i - \hat{y}_i)^2 = e_1^2 + \cdots + e_n^2$$

  - Note that $\mathrm{RSS} = n \cdot \mathrm{MSE}$.

- Using calculus or linear algebra, we can show that RSS is minimized when

$$\hat{\beta}_1 = \frac{\sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y})}{\sum_{i=1}^{n}(x_i - \bar{x})^2} \qquad \hat{\beta}_0 = \bar{y} - \hat{\beta}_1 \bar{x}$$

- Therefore, the least squares regression line has the lowest **training MSE** among all linear models.

Residual Sum of Squares

- Define the **Residual Sum of Squares** (RSS) as

$$\mathrm{RSS} = \sum_{i=1}^{n} (y_i - \hat{y}_i)^2 = e_1^2 + \cdots + e_n^2$$

  - Note that $\mathrm{RSS} = n \cdot \mathrm{MSE}$.

- Using calculus or linear algebra, we can show that RSS is minimized when

$$\hat{\beta}_1 = \frac{\sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y})}{\sum_{i=1}^{n}(x_i - \bar{x})^2} \qquad \hat{\beta}_0 = \bar{y} - \hat{\beta}_1 \bar{x}$$

- Therefore, the least squares regression line has the lowest **training MSE** among all linear models.
- Does this mean it has the lowest **test MSE** among linear models?

Residual Sum of Squares

- Define the **Residual Sum of Squares** (RSS) as

$$\mathrm{RSS} = \sum_{i=1}^{n} (y_i - \hat{y}_i)^2 = e_1^2 + \cdots + e_n^2$$

  - Note that $\mathrm{RSS} = n \cdot \mathrm{MSE}$.

- Using calculus or linear algebra, we can show that RSS is minimized when

$$\hat{\beta}_1 = \frac{\sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y})}{\sum_{i=1}^{n}(x_i - \bar{x})^2} \qquad \hat{\beta}_0 = \bar{y} - \hat{\beta}_1 \bar{x}$$

- Therefore, the least squares regression line has the lowest **training MSE** among all linear models.
- Does this mean it has the lowest **test MSE** among linear models?
  - No, as we will see later with *penalized regression* (Ch 6, ISLR)

Measuring Model Accuracy (Alphabet Soup)

The following (closely related) measures are used to assess accuracy of a linear model:

Measuring Model Accuracy (Alphabet Soup)

The following (closely related) measures are used to assess accuracy of a linear model:

- Residual Sum of Squares, Mean Squared Error and Root Mean Squared Error:

$$\text{RSS} = \sum_{i=1}^{n} (y_i - \hat{y}_i)^2 \qquad \text{training MSE} = \frac{\text{RSS}}{n} \qquad \text{RMSE} = \sqrt{\text{MSE}}$$

Measuring Model Accuracy (Alphabet Soup)

The following (closely related) measures are used to assess accuracy of a linear model:

- Residual Sum of Squares, Mean Squared Error and Root Mean Squared Error:

$$\text{RSS} = \sum_{i=1}^{n} (y_i - \hat{y}_i)^2 \qquad \text{training MSE} = \frac{\text{RSS}}{n} \qquad \text{RMSE} = \sqrt{\text{MSE}}$$

- Residual Standard Error (RSE or $\hat{\sigma}$)

$$\hat{\sigma} = \text{RSE} = \sqrt{\frac{1}{n-2}\text{RSS}} = \sqrt{\frac{n}{n-2}}\text{RMSE}$$

Measuring Model Accuracy (Alphabet Soup)

The following (closely related) measures are used to assess accuracy of a linear model:

- Residual Sum of Squares, Mean Squared Error and Root Mean Squared Error:

$$\mathrm{RSS} = \sum_{i=1}^{n} (y_i - \hat{y}_i)^2 \qquad \text{training MSE} = \frac{\mathrm{RSS}}{n} \qquad \mathrm{RMSE} = \sqrt{\mathrm{MSE}}$$

- Residual Standard Error (RSE or $\hat{\sigma}$)

$$\hat{\sigma} = \mathrm{RSE} = \sqrt{\frac{1}{n-2}\mathrm{RSS}} = \sqrt{\frac{n}{n-2}}\mathrm{RMSE}$$

  - RSE is an estimate of the standard deviation σ of model error ϵ
  - RSE measures the *typical* size of model errors

Measuring Model Accuracy (Alphabet Soup)

The following (closely related) measures are used to assess accuracy of a linear model:

- Residual Sum of Squares, Mean Squared Error and Root Mean Squared Error:

$$\mathrm{RSS} = \sum_{i=1}^{n}(y_i - \hat{y}_i)^2 \qquad \text{training MSE} = \frac{\mathrm{RSS}}{n} \qquad \mathrm{RMSE} = \sqrt{\mathrm{MSE}}$$

- Residual Standard Error (RSE or $\hat{\sigma}$)

$$\hat{\sigma} = \mathrm{RSE} = \sqrt{\frac{1}{n-2}\mathrm{RSS}} = \sqrt{\frac{n}{n-2}}\mathrm{RMSE}$$

  - RSE is an estimate of the standard deviation σ of model error ϵ
  - RSE measures the *typical* size of model errors

- The coefficient of determination R^2

$$R^2 = 1 - \frac{\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}{\sum_{i=1}^{n}(y_i - \bar{y})^2} = 1 - \frac{\mathrm{RSS}}{\mathrm{TSS}}$$

Measuring Model Accuracy (Alphabet Soup)

The following (closely related) measures are used to assess accuracy of a linear model:

- Residual Sum of Squares, Mean Squared Error and Root Mean Squared Error:

$$\mathrm{RSS} = \sum_{i=1}^{n}(y_i - \hat{y}_i)^2 \qquad \text{training MSE} = \frac{\mathrm{RSS}}{n} \qquad \mathrm{RMSE} = \sqrt{\mathrm{MSE}}$$

- Residual Standard Error (RSE or $\hat{\sigma}$)

$$\hat{\sigma} = \mathrm{RSE} = \sqrt{\frac{1}{n-2}\mathrm{RSS}} = \sqrt{\frac{n}{n-2}}\mathrm{RMSE}$$

  - RSE is an estimate of the standard deviation σ of model error ϵ
  - RSE measures the *typical* size of model errors

- The coefficient of determination R^2

$$R^2 = 1 - \frac{\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}{\sum_{i=1}^{n}(y_i - \bar{y})^2} = 1 - \frac{\mathrm{RSS}}{\mathrm{TSS}}$$

- R^2 is the proportion of variation in the response explained by the model.

Section 2

Inference for Linear Models

Statistical Inference

- **Goal**: Use *statistics* calculated from data to make estimates about unknown *parameters*

Statistical Inference

- **Goal**: Use *statistics* calculated from data to make estimates about unknown *parameters*
- **Parameters**: β_0, β_1

Statistical Inference

- **Goal**: Use *statistics* calculated from data to make estimates about unknown *parameters*
- **Parameters**: β_0, β_1
- **Statistics**: $\hat{\beta}_0$, $\hat{\beta}_1$

Statistical Inference

- **Goal**: Use *statistics* calculated from data to make estimates about unknown *parameters*
- **Parameters**: β_0, β_1
- **Statistics**: $\hat{\beta}_0$, $\hat{\beta}_1$
- **Tools**: confidence intervals, hypothesis tests

Statistical Inference

- **Goal**: Use *statistics* calculated from data to make estimates about unknown *parameters*
- **Parameters**: β_0, β_1
- **Statistics**: $\hat{\beta}_0$, $\hat{\beta}_1$
- **Tools**: confidence intervals, hypothesis tests
- **The Problems**: Our model will change if built using a different random sample. So in addition to estimates, we need to know about variability

The Confidence Interval

- Confidence Intervals give estimates **and** express an amount of uncertainty we have about those estimates

The Confidence Interval

- Confidence Intervals give estimates **and** express an amount of uncertainty we have about those estimates
- A *C*-level confidence interval for a parameter θ using the statistic $\hat{\theta}$ takes the form

$$\hat{\theta} \pm t_C^* \cdot \mathrm{SE}(\hat{\theta})$$

The Confidence Interval

- Confidence Intervals give estimates **and** express an amount of uncertainty we have about those estimates
- A C-level confidence interval for a parameter θ using the statistic $\hat{\theta}$ takes the form

$$\hat{\theta} \pm t_C^* \cdot \mathrm{SE}(\hat{\theta})$$

- The value t_C^* is the $1 - (1 - C)/2$ quantile for the sampling distribution of $\hat{\theta}$
  - i.e. if $\hat{\theta}$ is approximately Normally distributed and $C = .95$, then $t_C^* \approx 2$.

The Confidence Interval

- Confidence Intervals give estimates **and** express an amount of uncertainty we have about those estimates
- A C-level confidence interval for a parameter θ using the statistic $\hat{\theta}$ takes the form

$$\hat{\theta} \pm t_C^* \cdot \mathrm{SE}(\hat{\theta})$$

- The value t_C^* is the $1 - (1 - C)/2$ quantile for the sampling distribution of $\hat{\theta}$
  - i.e. if $\hat{\theta}$ is approximately Normally distributed and $C = .95$, then $t_C^* \approx 2$.
- The value $\mathrm{SE}(\hat{\theta})$ is the standard error of $\hat{\theta}$, or the standard deviation of the sampling distribution

Common Regression Assumptions

In order to use simple linear regression for inference, we require these assumptions:

Common Regression Assumptions

In order to use simple linear regression for inference, we require these assumptions:

1. Y is related to X by a simple linear regression model.

$$Y = \beta_0 + \beta_1 X + \epsilon$$

Common Regression Assumptions

In order to use simple linear regression for inference, we require these assumptions:

1. Y is related to X by a simple linear regression model.

$$Y = \beta_0 + \beta_1 X + \epsilon$$

2. The errors $e_1, e_2, \ldots, e_n$ are independent of one another.

Common Regression Assumptions

In order to use simple linear regression for inference, we require these assumptions:

1. Y is related to X by a simple linear regression model.

$$Y = \beta_0 + \beta_1 X + \epsilon$$

2. The errors $e_1, e_2, \ldots, e_n$ are independent of one another.
3. The errors have a common variance $\mathrm{Var}(\epsilon) = \sigma^2$.

Common Regression Assumptions

In order to use simple linear regression for inference, we require these assumptions:

1. Y is related to X by a simple linear regression model.

$$Y = \beta_0 + \beta_1 X + \epsilon$$

2. The errors $e_1, e_2, \ldots, e_n$ are independent of one another.
3. The errors have a common variance $\mathrm{Var}(\epsilon) = \sigma^2$.
4. The errors are normally distributed: $\epsilon \sim N(0, \sigma^2)$

Common Regression Assumptions

In order to use simple linear regression for inference, we require these assumptions:

1. Y is related to X by a simple linear regression model.

$$Y = \beta_0 + \beta_1 X + \epsilon$$

2. The errors $e_1, e_2, \ldots, e_n$ are independent of one another.
3. The errors have a common variance $\mathrm{Var}(\epsilon) = \sigma^2$.
4. The errors are normally distributed: $\epsilon \sim N(0, \sigma^2)$

If one or more of these conditions do not hold, our predictions may not be accurate and we should be skeptical of inferential claims.

The Sampling Distribution of $\hat{\beta}_1$

Assume the following true model:

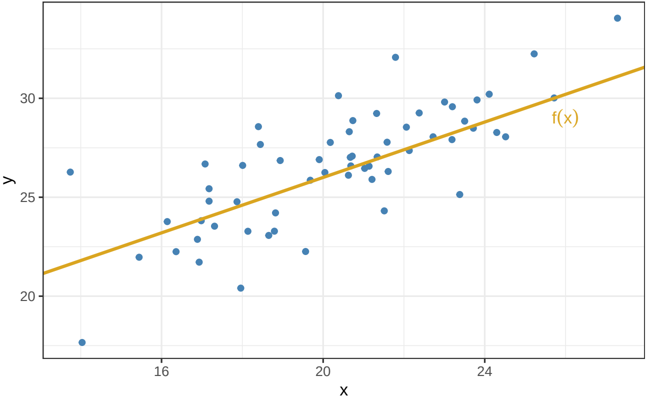
$$f(x) = 12 + 0.7x; \quad \epsilon \sim N(0, 4)$$

The Sampling Distribution of $\hat{\beta}_1$

Assume the following true model:

$$f(x) = 12 + 0.7x; \quad \epsilon \sim N(0, 4)$$

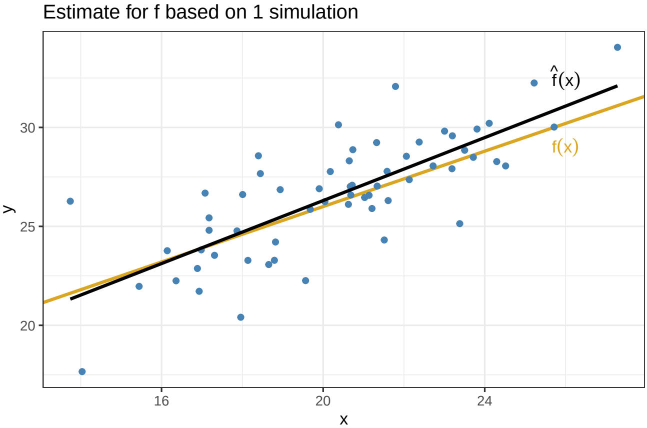
Simulated Data from true model



The Sampling Distribution of $\hat{\beta}_1$

Assume the following true model:

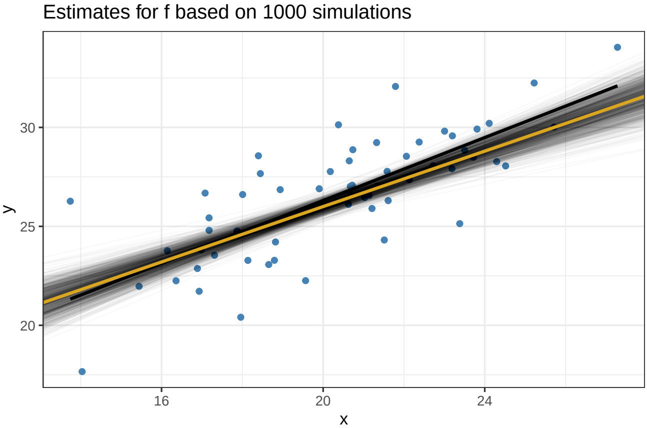
$$f(x) = 12 + 0.7x; \quad \epsilon \sim N(0, 4)$$



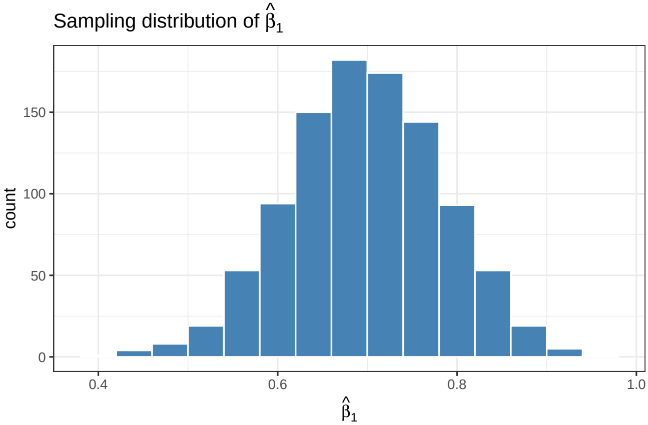
The Sampling Distribution of $\hat{\beta}_1$

Assume the following true model:

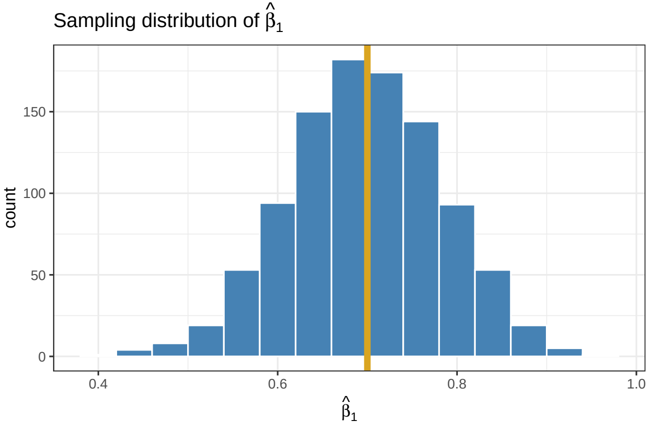
$$f(x) = 12 + 0.7x; \quad \epsilon \sim N(0, 4)$$



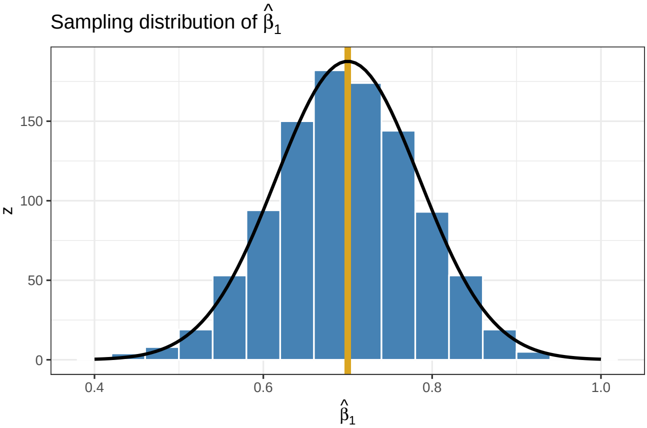
The Sampling Distribution of $\hat{\beta}_1$



The Sampling Distribution of $\hat{\beta}_1$



The Sampling Distribution of $\hat{\beta}_1$



The Sampling Distribution of $\hat{\beta}_1$

The Sampling Distribution has the following characteristics:

1. Centered at β_1, i.e. $E(\hat{\beta}_1) = \beta$.

The Sampling Distribution of $\hat{\beta}_1$

The Sampling Distribution has the following characteristics:

1. Centered at β_1, i.e. $E(\hat{\beta}_1) = \beta$.
2. $Var(\hat{\beta}_1) = \frac{\sigma^2}{S_{XX}}$.
   - where $S_{XX} = \sum_{i=1}^{n}(x_i - \bar{x})^2$

The Sampling Distribution of $\hat{\beta}_1$

The Sampling Distribution has the following characteristics:

1. Centered at β_1, i.e. $E(\hat{\beta}_1) = \beta$.
2. $Var(\hat{\beta}_1) = \frac{\sigma^2}{S_{XX}}$.
   - where $S_{XX} = \sum_{i=1}^{n}(x_i - \bar{x})^2$
3. $\hat{\beta}_1|X \sim N(\beta_1, \frac{\sigma^2}{S_{XX}})$.

Approximating the Sampling Dist. of $\hat{\beta}_1$

- Our best estimate of β_1 is $\hat{\beta}_1$ (since the expected value $\hat{\beta}_1$ is β_1)

Approximating the Sampling Dist. of $\hat{\beta}_1$

- Our best estimate of β_1 is $\hat{\beta}_1$ (since the expected value $\hat{\beta}_1$ is β_1)
- However, we have to estimate σ with the Residual Standard Error:

$$\hat{\sigma} = \mathrm{RSE} = \sqrt{\frac{\mathrm{RSS}}{n-2}}$$

Approximating the Sampling Dist. of $\hat{\beta}_1$

- Our best estimate of β_1 is $\hat{\beta}_1$ (since the expected value $\hat{\beta}_1$ is β_1)
- However, we have to estimate σ with the Residual Standard Error:

$$\hat{\sigma} = \mathrm{RSE} = \sqrt{\frac{\mathrm{RSS}}{n-2}}$$

- Thus, the distribution of $\frac{\hat{\beta}_1 - \beta_1}{\hat{\sigma}}$ isn't Normal...

Approximating the Sampling Dist. of $\hat{\beta}_1$

- Our best estimate of β_1 is $\hat{\beta}_1$ (since the expected value $\hat{\beta}_1$ is β_1)
- However, we have to estimate σ with the Residual Standard Error:

$$\hat{\sigma} = \mathrm{RSE} = \sqrt{\frac{\mathrm{RSS}}{n-2}}$$

- Thus, the distribution of $\frac{\hat{\beta}_1 - \beta_1}{\hat{\sigma}}$ isn't Normal...
- Instead, it is the t-distribution with $n-2$ degrees of freedom.

Approximating the Sampling Dist. of $\hat{\beta}_1$

- Our best estimate of β_1 is $\hat{\beta}_1$ (since the expected value $\hat{\beta}_1$ is β_1)
- However, we have to estimate σ with the Residual Standard Error:

$$\hat{\sigma} = \mathrm{RSE} = \sqrt{\frac{\mathrm{RSS}}{n-2}}$$

- Thus, the distribution of $\frac{\hat{\beta}_1 - \beta_1}{\hat{\sigma}}$ isn't Normal...
- Instead, it is the t-distribution with $n-2$ degrees of freedom.
- Our confidence interval for $\hat{\beta}_1$ is thus

$$\hat{\beta}_1 \pm t_{\alpha/2, n-2} \cdot SE(\hat{\beta}_1) \qquad \text{where } SE(\hat{\beta}_1) = \frac{\hat{\sigma}}{\sqrt{\sum_{i=1}^{n}(x_i - \bar{x})^2}}$$

Approximating the Sampling Dist. of $\hat{\beta}_1$

- Our best estimate of β_1 is $\hat{\beta}_1$ (since the expected value $\hat{\beta}_1$ is β_1)
- However, we have to estimate σ with the Residual Standard Error:

$$\hat{\sigma} = \mathrm{RSE} = \sqrt{\frac{\mathrm{RSS}}{n-2}}$$

- Thus, the distribution of $\frac{\hat{\beta}_1 - \beta_1}{\hat{\sigma}}$ isn't Normal...
- Instead, it is the t-distribution with $n-2$ degrees of freedom.
- Our confidence interval for $\hat{\beta}_1$ is thus

$$\hat{\beta}_1 \pm t_{\alpha/2,n-2} \cdot SE(\hat{\beta}_1) \qquad \text{where } SE(\hat{\beta}_1) = \frac{\hat{\sigma}}{\sqrt{\sum_{i=1}^{n}(x_i - \bar{x})^2}}$$

Interpretation We are *95% confident* that the true slope relating x and y lies between lower and upper bound of this interval.

Hypothesis test for $\hat{\beta}_1$

Suppose we are interested in testing the claim that the slope is zero.

$$H_0 : \beta_1^0 = 0 \qquad \text{vs} \quad H_A : \beta_1^0 \neq 0$$

Hypothesis test for $\hat{\beta}_1$

Suppose we are interested in testing the claim that the slope is zero.

$$H_0 : \beta_1^0 = 0 \qquad \text{vs} \quad H_A : \beta_1^0 \neq 0$$

- Consider the statistic t given by

$$t = \frac{\hat{\beta}_1}{SE(\hat{\beta}_1)}$$

- Then t will be t-distributed with $n-2$ degrees of freedom and $SE(\hat{\beta}_1)$ calculated the same as in the CI.

Hypothesis test for $\hat{\beta}_1$

Suppose we are interested in testing the claim that the slope is zero.

$$H_0 : \beta_1^0 = 0 \qquad \text{vs} \quad H_A : \beta_1^0 \neq 0$$

- Consider the statistic t given by

$$t = \frac{\hat{\beta}_1}{SE(\hat{\beta}_1)}$$

- Then t will be t-distributed with $n - 2$ degrees of freedom and $SE(\hat{\beta}_1)$ calculated the same as in the CI.
- The p-value for an observed test statistic t is the probability that a randomly chosen value from the t-dist is larger in absolute value than $|t|$.

Hypothesis test for $\hat{\beta}_1$

Suppose we are interested in testing the claim that the slope is zero.

$$H_0 : \beta_1^0 = 0 \qquad \text{vs} \quad H_A : \beta_1^0 \neq 0$$

- Consider the statistic t given by

$$t = \frac{\hat{\beta}_1}{SE(\hat{\beta}_1)}$$

- Then t will be t-distributed with $n-2$ degrees of freedom and $SE(\hat{\beta}_1)$ calculated the same as in the CI.
- The p-value for an observed test statistic t is the probability that a randomly chosen value from the t-dist is larger in absolute value than $|t|$.
- An observed t with p-value less than a desired significance level (often $\alpha = 0.05$) gives good evidence against the null-hypothesis.

Inference for other parameters in the linear model

- We can also perform inference for β_0, although it is often less interesting in practice (why?)

Inference for other parameters in the linear model

- We can also perform inference for β_0, although it is often less interesting in practice (why?)
  - We proceed as before, using a t distribution to estimate the sampling distribution of $\hat{\beta}_0$.
  - However, the SE of $\hat{\beta}_0$ is

$$\mathrm{SE}(\hat{\beta}_0) = \hat{\sigma}^2 \left(\frac{1}{n} + \frac{\bar{x}}{\sqrt{\sum_{i=1}^{n}(x_i - \bar{x})^2}} \right)$$

Inference for other parameters in the linear model

- We can also perform inference for β_0, although it is often less interesting in practice (why?)
  - We proceed as before, using a t distribution to estimate the sampling distribution of $\hat{\beta}_0$.
  - However, the SE of $\hat{\beta}_0$ is

$$\mathrm{SE}(\hat{\beta}_0) = \hat{\sigma}^2 \left(\frac{1}{n} + \frac{\bar{x}}{\sqrt{\sum_{i=1}^{n}(x_i - \bar{x})^2}} \right)$$

- Inference is even possible for combinations of β_0 and β_1 (i.e $\beta_0 + \beta_1 x$ for any fixed value of x)

Inference for other parameters in the linear model

- We can also perform inference for β_0, although it is often less interesting in practice (why?)
  - We proceed as before, using a t distribution to estimate the sampling distribution of $\hat{\beta}_0$.
  - However, the SE of $\hat{\beta}_0$ is

$$\mathrm{SE}(\hat{\beta}_0) = \hat{\sigma}^2 \left(\frac{1}{n} + \frac{\bar{x}}{\sqrt{\sum_{i=1}^{n}(x_i - \bar{x})^2}} \right)$$

- Inference is even possible for combinations of β_0 and β_1 (i.e $\beta_0 + \beta_1 x$ for any fixed value of x)
  - Why might we want to obtain a confidence interval for $\beta_0 + \beta_1 x$?

Inference for other parameters in the linear model

- We can also perform inference for β_0, although it is often less interesting in practice (why?)
  - We proceed as before, using a t distribution to estimate the sampling distribution of $\hat{\beta}_0$.
  - However, the SE of $\hat{\beta}_0$ is

$$\mathrm{SE}(\hat{\beta}_0) = \hat{\sigma}^2 \left(\frac{1}{n} + \frac{\bar{x}}{\sqrt{\sum_{i=1}^{n} (x_i - \bar{x})^2}} \right)$$

- Inference is even possible for combinations of β_0 and β_1 (i.e $\beta_0 + \beta_1 x$ for any fixed value of x)
  - Why might we want to obtain a confidence interval for $\beta_0 + \beta_1 x$?
  - The associated statistic is again t-distributed, although with more complicated SE.

Inference for other parameters in the linear model

- We can also perform inference for β_0, although it is often less interesting in practice (why?)
    - We proceed as before, using a t distribution to estimate the sampling distribution of $\hat{\beta}_0$.
    - However, the SE of $\hat{\beta}_0$ is

$$\mathrm{SE}(\hat{\beta}_0) = \hat{\sigma}^2 \left(\frac{1}{n} + \frac{\bar{x}}{\sqrt{\sum_{i=1}^{n}(x_i - \bar{x})^2}} \right)$$

- Inference is even possible for combinations of β_0 and β_1 (i.e $\beta_0 + \beta_1 x$ for any fixed value of x)
    - Why might we want to obtain a confidence interval for $\beta_0 + \beta_1 x$?
    - The associated statistic is again t-distributed, although with more complicated SE.
    - For details, see DeGroot and Schervish "Probability and Statistics" (or take STA 336)

Section 3

Linear Models in R

Creating Linear Models in R

Consider the `povery` data set, consisting of high school grad rate `Graduates` and its poverty rate `Poverty`:

Creating Linear Models in R

Consider the povery data set, consisting of high school grad rate `Graduates` and its poverty rate `Poverty`:

```
## # A tibble: 6 x 3
##   state      Graduates Poverty
##   <chr>          <dbl>   <dbl>
## 1 Alabama         91.7    17.1
## 2 Alaska          80.4     9.5
## 3 Arizona         77.8    15.3
## 4 Arkansas        87.6    18
## 5 California      84.5    13.7
## 6 Colorado        81.1    12.2
```

Creating Linear Models in R

Consider the povery data set, consisting of high school grad rate `Graduates` and its poverty rate `Poverty`:

```
## # A tibble: 6 x 3
##   state      Graduates Poverty
##   <chr>          <dbl>   <dbl>
## 1 Alabama         91.7    17.1
## 2 Alaska          80.4     9.5
## 3 Arizona         77.8    15.3
## 4 Arkansas        87.6    18
## 5 California      84.5    13.7
## 6 Colorado        81.1    12.2
```

- We fit a linear model using the `lm` function in R:

```
poverty_mod <- lm(Graduates ~ Poverty, data = poverty)
```

Summary of the Model

- When we use the `lm` function, R computes several values related to the linear model

Summary of the Model

- When we use the `lm` function, R computes several values related to the linear model
  - We can obtain a high-level summary of the model using `summary()`

Summary of the Model

- When we use the `lm` function, R computes several values related to the linear model
  - We can obtain a high-level summary of the model using `summary()`

```
summary(poverty_mod)
```

```
##
## Call:
## lm(formula = Graduates ~ Poverty, data = poverty)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -14.541  -2.774   0.876   2.543   7.758
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  90.0615     2.8347  31.772   <2e-16 ***
## Poverty      -0.3579     0.2056  -1.741    0.088 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 4.396 on 49 degrees of freedom
## Multiple R-squared:  0.05823,    Adjusted R-squared:  0.03901
## F-statistic:  3.03 on 1 and 49 DF,  p-value: 0.08802
```

Accessing Summary Statistics

- The summary table is itself an R object, with many attributes:

Accessing Summary Statistics

- The summary table is itself an R object, with many attributes:

```
mod_summary <- summary(poverty_mod)
names(mod_summary)
```

```
##  [1] "call"          "terms"         "residuals"     "coefficients"
##  [5] "aliased"       "sigma"         "df"            "r.squared"
##  [9] "adj.r.squared" "fstatistic"    "cov.unscaled"
```

Accessing Summary Statistics

- The summary table is itself an R object, with many attributes:

```
mod_summary <- summary(poverty_mod)
names(mod_summary)
```

```
##  [1] "call"          "terms"         "residuals"     "coefficients"
##  [5] "aliased"       "sigma"         "df"            "r.squared"
##  [9] "adj.r.squared" "fstatistic"    "cov.unscaled"
```

- To access these attributes, we can preface the name of the attribute with the summary table name and $:

Accessing Summary Statistics

- The summary table is itself an R object, with many attributes:

```
mod_summary <- summary(poverty_mod)
names(mod_summary)
```

```
##  [1] "call"          "terms"         "residuals"     "coefficients"
##  [5] "aliased"       "sigma"         "df"            "r.squared"
##  [9] "adj.r.squared" "fstatistic"    "cov.unscaled"
```

- To access these attributes, we can preface the name of the attribute with the summary table name and $:

```
mod_summary$r.squared
```

```
## [1] 0.05823356
```

Accessing Summary Statistics

- The summary table is itself an R object, with many attributes:

```
mod_summary <- summary(poverty_mod)
names(mod_summary)
```

```
##  [1] "call"          "terms"         "residuals"     "coefficients"
##  [5] "aliased"       "sigma"         "df"            "r.squared"
##  [9] "adj.r.squared" "fstatistic"    "cov.unscaled"
```

- To access these attributes, we can preface the name of the attribute with the summary table name and $:

```
mod_summary$r.squared
```

```
## [1] 0.05823356
```

```
mod_summary$sigma
```

```
## [1] 4.395734
```

Accessing Model Components

- When R creates a linear model, it saves many attributes in the model object

Accessing Model Components

- When R creates a linear model, it saves many attributes in the model object

```
names(poverty_mod)
```

```
##  [1] "coefficients"  "residuals"     "effects"       "rank"
##  [5] "fitted.values" "assign"        "qr"            "df.residual"
##  [9] "xlevels"       "call"          "terms"         "model"
```

Accessing Model Components

- When R creates a linear model, it saves many attributes in the model object

```
names(poverty_mod)
```

```
##  [1] "coefficients"  "residuals"     "effects"       "rank"
##  [5] "fitted.values" "assign"        "qr"            "df.residual"
##  [9] "xlevels"       "call"          "terms"         "model"
```

- To access these attributes, we can preface the name of the attribute with the model name and $.
- Two of the most useful attributes are `fitted.values` and `residuals`:

Accessing Model Components

- When R creates a linear model, it saves many attributes in the model object

```
names(poverty_mod)
```

```
##  [1] "coefficients"  "residuals"     "effects"       "rank"
##  [5] "fitted.values" "assign"        "qr"            "df.residual"
##  [9] "xlevels"       "call"          "terms"         "model"
```

- To access these attributes, we can preface the name of the attribute with the model name and $.
- Two of the most useful attributes are `fitted.values` and `residuals`:

```
poverty_mod$fitted.values
```

```
##        1        2        3        4        5        6
## 83.94205 86.66182 84.58621 83.61997 85.15879 85.69559
```

Accessing Model Components

- When R creates a linear model, it saves many attributes in the model object

```
names(poverty_mod)
```

```
##  [1] "coefficients"  "residuals"     "effects"       "rank"
##  [5] "fitted.values" "assign"        "qr"            "df.residual"
##  [9] "xlevels"       "call"          "terms"         "model"
```

- To access these attributes, we can preface the name of the attribute with the model name and $.
- Two of the most useful attributes are `fitted.values` and `residuals`:

```
poverty_mod$fitted.values
```

```
##        1        2        3        4        5        6
## 83.94205 86.66182 84.58621 83.61997 85.15879 85.69559
```

```
poverty_mod$residuals
```

```
##          1          2          3          4          5          6
##  7.7579486 -6.2618192 -6.7862069  3.9800264 -0.6587896 -4.5955859
```

Predictions from Linear Models

- The linear model object in R can be used to calculate predictions made by the model.

Predictions from Linear Models

- The linear model object in R can be used to calculate predictions made by the model.
  - Let's predict the graduation rate for states with poverty rates of 3, 10, and 15.

Predictions from Linear Models

- The linear model object in R can be used to calculate predictions made by the model.
    - Let's predict the graduation rate for states with poverty rates of 3, 10, and 15.
    - We first make a data frame storing our new data

```
new_states <- data.frame(Poverty = c(3, 10, 15))
```

```
##   Poverty
## 1       3
## 2      10
## 3      15
```

- The new data must contain a column with the same name as the predictor in the original data (`Poverty` in this case)

Predictions from Linear Models

- We can use the `predict` function to have R make predictions.

Predictions from Linear Models

- We can use the `predict` function to have R make predictions.
- The `predict` function takes two inputs: a model `object` and the `newdata` on which predictions will be made.

Predictions from Linear Models

- We can use the `predict` function to have R make predictions.
- The `predict` function takes two inputs: a model `object` and the `newdata` on which predictions will be made.

```
prediction <- predict(object = poverty_mod, newdata = new_states)
```

```
##        1        2        3
## 88.98794 86.48289 84.69357
```

Predictions from Linear Models

- We can use the `predict` function to have R make predictions.
- The `predict` function takes two inputs: a model `object` and the `newdata` on which predictions will be made.

```
prediction <- predict(object = poverty_mod, newdata = new_states)
```

```
##        1        2        3
## 88.98794 86.48289 84.69357
```

- We can append this to our `new_states` data frame:

```
new_states <- cbind(new_states, prediction)
```

```
##   Poverty prediction
## 1       3   88.98794
## 2      10   86.48289
## 3      15   84.69357
```

Predictions from Linear Models

- We can use the `predict` function to have R make predictions.
- The `predict` function takes two inputs: a model `object` and the `newdata` on which predictions will be made.

```
prediction <- predict(object = poverty_mod, newdata = new_states)
```

```
##        1        2        3
## 88.98794 86.48289 84.69357
```

- We can append this to our `new_states` data frame:

```
new_states <- cbind(new_states, prediction)
```

```
##   Poverty prediction
## 1       3   88.98794
## 2      10   86.48289
## 3      15   84.69357
```

- Let's now practice in R