Simple Regression

El Mex

Contents


```
library(tidyverse)
library(moderndive)
library(skimr)
library(gapminder)
```

1 Basics of regression

1.1 One numerical explanatory variable

1.1.1 Exploratory data analysis

Researchers at the University of Texas in Austin, Texas (UT Austin) tried to answer the following research question: what factors explain differences in instructor teaching evaluation scores? To this end, they collected instructor and course information on 463 courses

Let's use *simple linear regression* with two variables:

1. A numerical outcome variable `y` (the instructor's teaching score)
2. A single numerical explanatory variable `x` (the instructor's "beauty" score)

```
# explore the "evals" built-in data frame in "moderndive". Then save the variables in
# a new data frame called "evals_ch5"
glimpse(evals)
```

```
## Rows: 463
## Columns: 14
## $ ID          <int> 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16,...
## $ prof_ID     <int> 1, 1, 1, 1, 2, 2, 2, 3, 3, 4, 4, 4, 4, 4, 4, 4, 5, ...
## $ score       <dbl> 4.7, 4.1, 3.9, 4.8, 4.6, 4.3, 2.8, 4.1, 3.4, 4.5, 3.8,...
## $ age         <int> 36, 36, 36, 36, 59, 59, 59, 51, 51, 40, 40, 40, 40, 40...
## $ bty_avg     <dbl> 5.000, 5.000, 5.000, 5.000, 3.000, 3.000, 3.000, 3.333...
```

```
## $ gender       <fct> female, female, female, female, male, male, male, male...
## $ ethnicity    <fct> minority, minority, minority, minority, not minority, ...
## $ language     <fct> english, english, english, english, english, english, ...
## $ rank         <fct> tenure track, tenure track, tenure track, tenure track...
## $ pic_outfit   <fct> not formal, not formal, not formal, not formal, not fo...
## $ pic_color    <fct> color, color, color, color, color, color, color, color...
## $ cls_did_eval <int> 24, 86, 76, 77, 17, 35, 39, 55, 111, 40, 24, 24, 17, 1...
## $ cls_students <int> 43, 125, 125, 123, 20, 40, 44, 55, 195, 46, 27, 25, 20...
## $ cls_level    <fct> upper, upper, upper, upper, upper, upper, upper, upper...
```

```
# select columns ID, score, bty_avg and age
evals_ch5 <- evals %>%
  select(ID, score, bty_avg, age)

glimpse(evals_ch5)
```

```
## Rows: 463
## Columns: 4
## $ ID      <int> 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, ...
## $ score   <dbl> 4.7, 4.1, 3.9, 4.8, 4.6, 4.3, 2.8, 4.1, 3.4, 4.5, 3.8, 4.5,...
## $ bty_avg <dbl> 5.000, 5.000, 5.000, 5.000, 3.000, 3.000, 3.000, 3.333, 3.3...
## $ age     <int> 36, 36, 36, 36, 59, 59, 59, 51, 51, 40, 40, 40, 40, 40, 40,...
```

Summarize()and get the mean() & median()of score and bty_avg

```
evals_ch5 %>%
  summarize(mean_bty_avg = mean(bty_avg), mean_score = mean(score),
            median_bty_avg = median(bty_avg), median_score = median(score))
```

```
## # A tibble: 1 x 4
##   mean_bty_avg mean_score median_bty_avg median_score
##          <dbl>      <dbl>          <dbl>        <dbl>
## 1         4.42       4.17           4.33          4.3
```

What if we want other summary statistics as well, such as the standard deviation (a measure of spread), the minimum and maximum values, and various percentiles?

Typing out all these summary statistic functions in summarize() would be long and tedious. Instead, let's use the convenient skim() function from the skimr package. This function takes in a data frame, "skims" it, and returns commonly used summary statistics

```
evals_ch5 %>%
  select(score, bty_avg) %>%
  skim()
```

Table 1: Data summary

Name	Piped data
Number of rows	463
Number of columns	2
Column type frequency:	
numeric	2
Group variables	None

Variable type: numeric

skim_variable	n_missing	complete_rate	mean	sd	p0	p25	p50	p75	p100	hist
score	0	1	4.17	0.54	2.30	3.80	4.30	4.6	5.00	
bty_avg	0	1	4.42	1.53	1.67	3.17	4.33	5.5	8.17	

The `skim()` function only returns what are known as *univariate* summary statistics: functions that take a single variable and return some numerical summary of that variable

when the two variables are numerical, we can compute the *correlation coefficient*. A correlation coefficient is a quantitative expression of the *strength of the linear relationship between two numerical variables*. Its value ranges between -1 and 1

Use `get_correlation()` function in the `moderndive` package

```
# put the name of the outcome variable on the left-hand side of the ~ "tilde" sign,
# while putting the name of the explanatory variable on the right-hand side
evals_ch5 %>%
  get_correlation(formula = score ~ bty_avg)
```

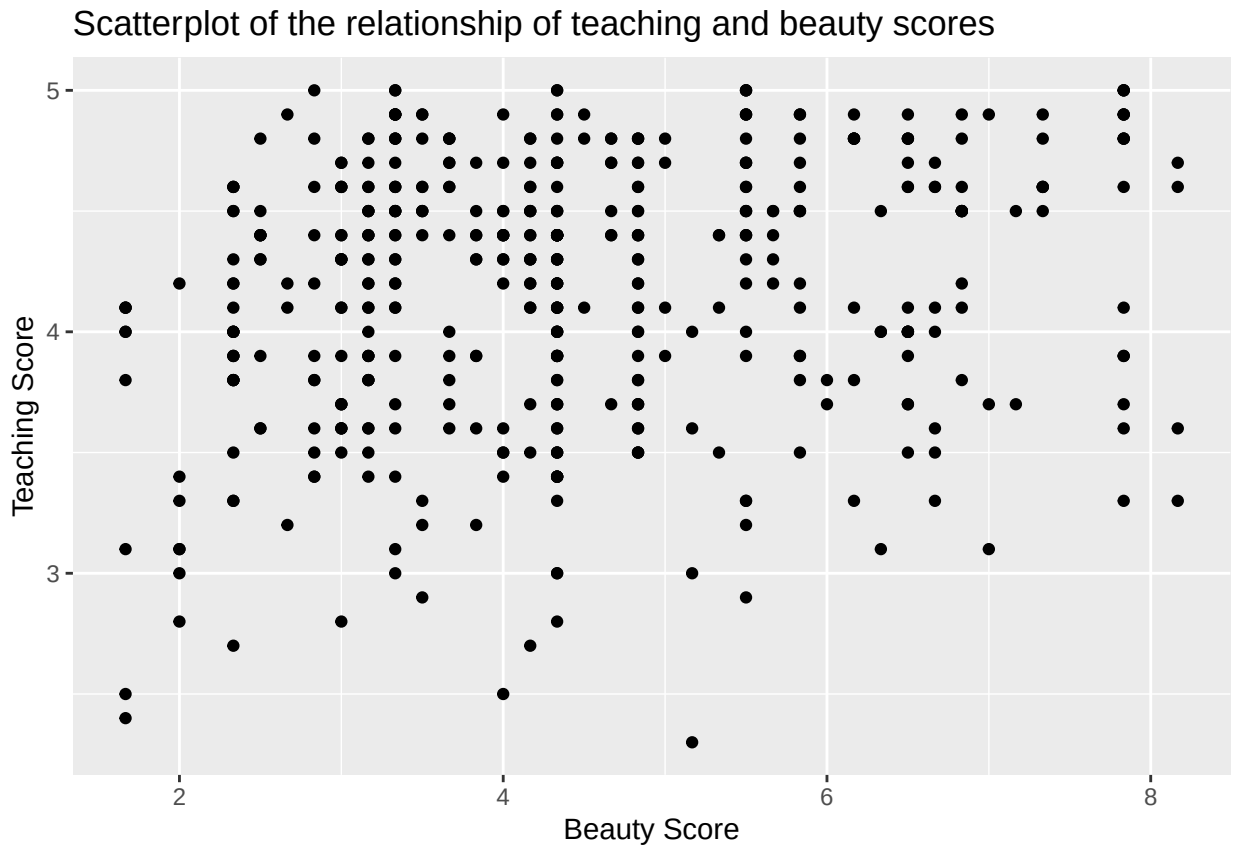
```
## # A tibble: 1 x 1
##     cor
##   <dbl>
## 1 0.187
```

An alternative way

```
evals_ch5 %>%
  summarize(correlation = cor(score, bty_avg))
```

Let's make a scatterplot by using `geom_point()`

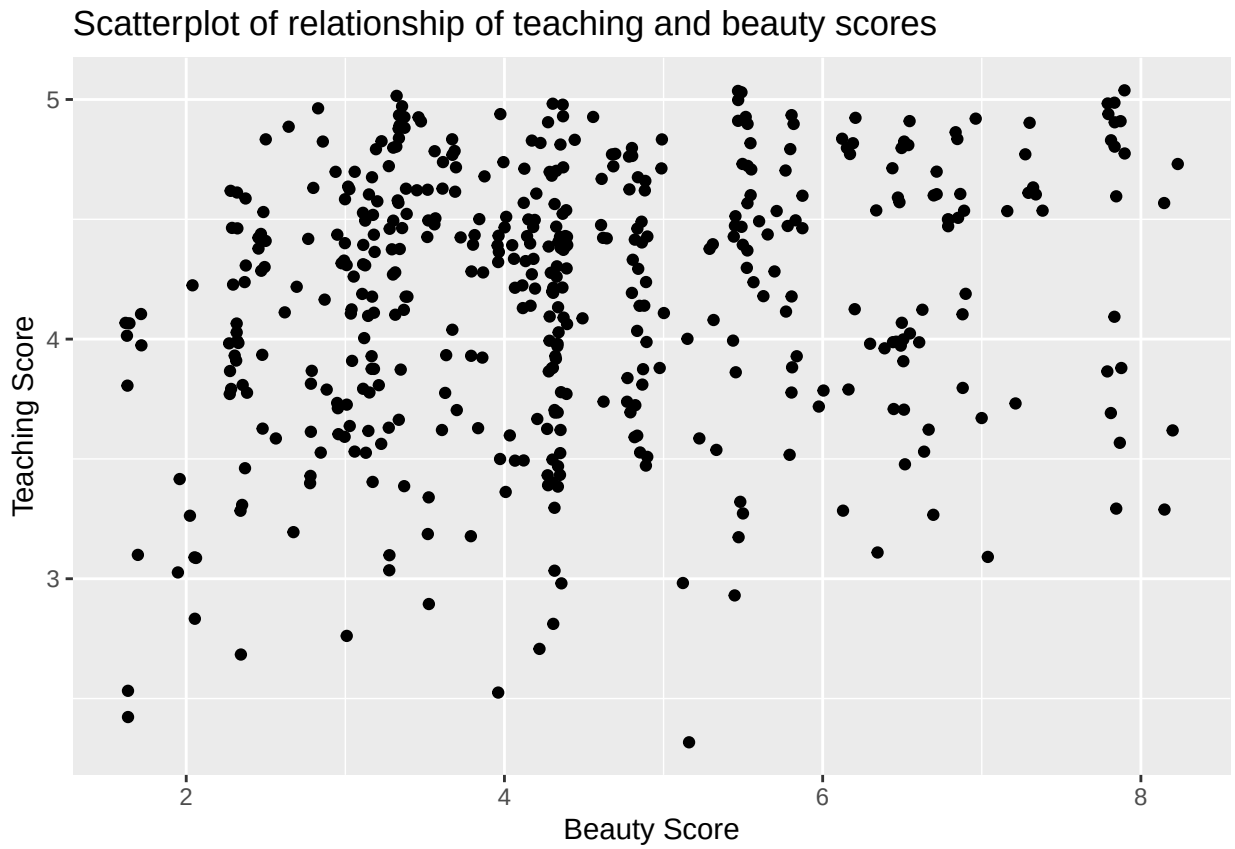
```
ggplot(evals_ch5, aes(x = bty_avg, y = score)) +
  geom_point() +
  labs(x = "Beauty Score",
       y = "Teaching Score",
       title = "Scatterplot of the relationship of teaching and beauty scores")
```



Observe that most "beauty" scores lie between 2 and 8, while most teaching scores lie between 3 and 5. Furthermore, while opinions may vary, it is our opinion that the relationship between teaching score and "beauty" score is "weakly positive." This is consistent with our earlier computed correlation coefficient of 0.187

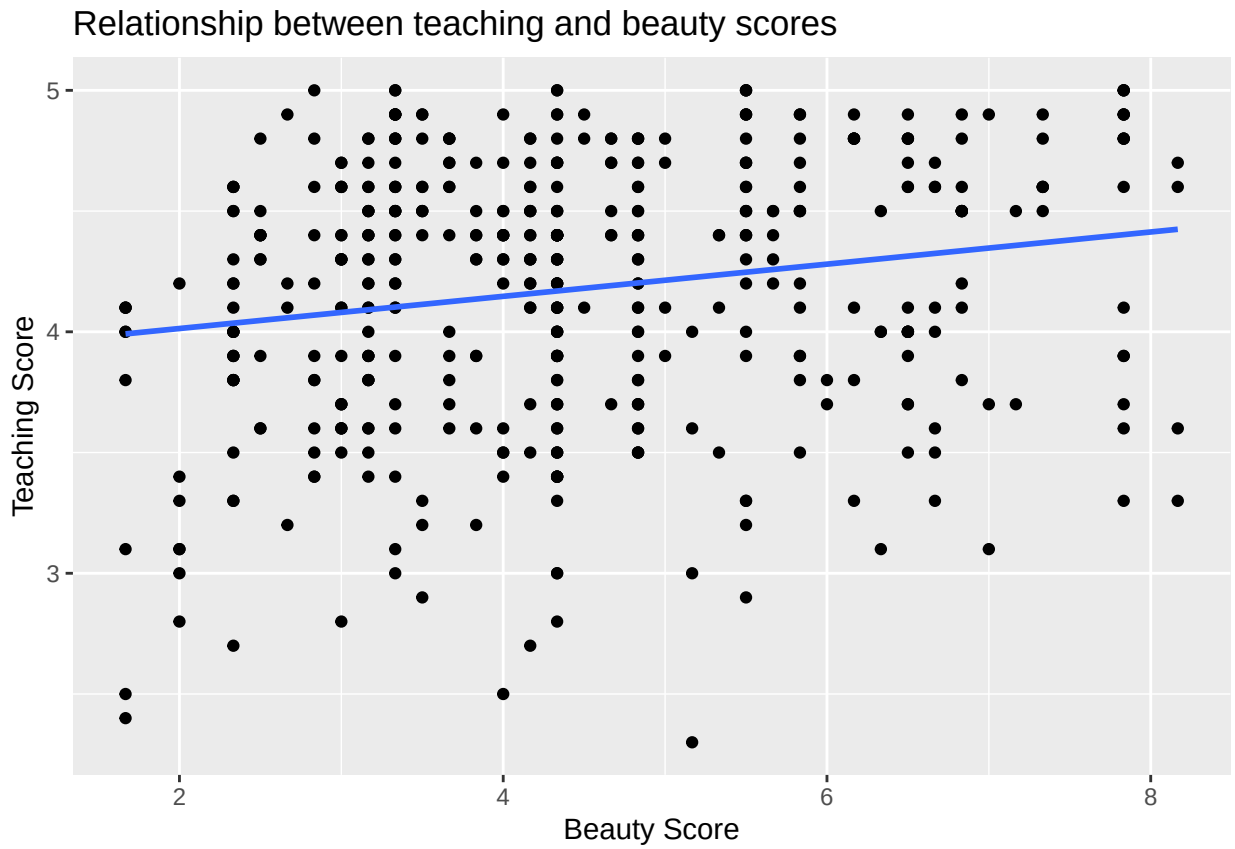
There appear to be six points in the top-right of this plot highlighted in the box. However, this is not actually the case, as this plot suffers from *overplotting*. Use `geom_jitter()` instead.

```
ggplot(evals_ch5, aes(x = bty_avg, y = score)) +
  geom_jitter() +
  labs(x = "Beauty Score", y = "Teaching Score",
       title = "Scatterplot of relationship of teaching and beauty scores")
```



For simplicity, keep using the unjitter scatterplot. Let's add a "best-fitting line" by using `geom_smooth(method = "lm", se = FALSE)`

```
# method = "lm" argument sets the line to be a "linear model." The se = FALSE argument suppresses
# standard error uncertainty bars
ggplot(evals_ch5, aes(x = bty_avg, y = score)) +
  geom_point() +
  labs(x = "Beauty Score", y = "Teaching Score",
       title = "Relationship between teaching and beauty scores") +
  geom_smooth(method = "lm", se = FALSE)
```



The *regression line* is a visual summary of the relationship between two numerical variables, in our case the outcome variable `score` and the explanatory variable `bty_avg`. The positive slope of the blue line is consistent with our earlier observed correlation coefficient of 0.187 suggesting that there is a positive relationship between these two variables: as instructors have higher "beauty" scores, so also do they receive higher teaching evaluations. We'll see later, however, that while the correlation coefficient and the slope of a regression line always have the same sign (positive or negative), they typically do not have the same value

1.1.2 Simple linear regression

The equation of the regression line is $\hat{y} = b_0 + b_1 * x$. The intercept coefficient is b_0, so b_0 is the value of $\hat{y}$ when $x = 0$. The slope coefficient for x is b_1, i.e., the increase in $\hat{y}$ for every increase of one in x

We know that the regression line has a positive slope b_1 corresponding to our explanatory x variable `bty_avg`. Why? Because as instructors tend to have higher `bty_avg` scores, so also do they tend to have higher teaching evaluation `scores`. However, what is the numerical value of the slope b_1? What about the intercept b_0?

We can obtain the values of the intercept b_0 and the slope for `bty_avg` b_1 by outputting a linear regression table. This is done in two steps:

1. We first "fit" the linear regression model using the `lm()` function and save it in `score_model`
2. We get the regression table by applying the `get_regression_table()` function from the `moderndive` package to `score_model`

```r
# fit regression model by using lm(y ~ x, data = data_frame_name)
score_model <- lm(score ~ bty_avg, data = evals_ch5)

# get regression table:
get_regression_table(score_model)
```

```
## # A tibble: 2 x 7
##   term      estimate std_error statistic p_value lower_ci upper_ci
##   <chr>        <dbl>     <dbl>     <dbl>   <dbl>    <dbl>    <dbl>
## 1 intercept    3.88      0.076     51.0        0    3.73     4.03
## 2 bty_avg      0.067     0.016      4.09       0    0.035    0.099
```

The intercept $b_0 = 3.88$ is the average teaching score $\hat{y} = \hat{score}$ for those courses where the instructor had a "beauty" score `bty_avg` of 0. While the intercept of the regression line has a mathematical interpretation, it has no practical interpretation here, since observing a `bty_avg` of 0 is impossible

Of greater interest is the slope `bty_avg` of 0.067 as this summarizes the relationship between the teaching and "beauty" score variables. Sign is positive, suggesting a positive relationship: teachers with higher "beauty" scores also tend to have higher teaching scores

Recall from earlier that the correlation coefficient is 0.187. They both have the same positive sign, but have a different value. Recall further that the correlation's interpretation is the "strength of linear association". The slope's interpretation is a little different:

> For every increase of 1 unit in `bty_avg`, there is an *associated* increase of, *on average*, 0.067 units of `score`

We only state that there is an associated increase and not necessarily a causal increase. Just because two variables are strongly associated, it doesn't necessarily mean that one causes the other

What the slope of 0.067 is saying is that across all possible courses, the average difference in teaching score between two instructors whose "beauty" scores differ by one is 0.067

1.1.3 Observed/fitted values and residuals

TABLE 5.3: Data for the 21st course out of 463

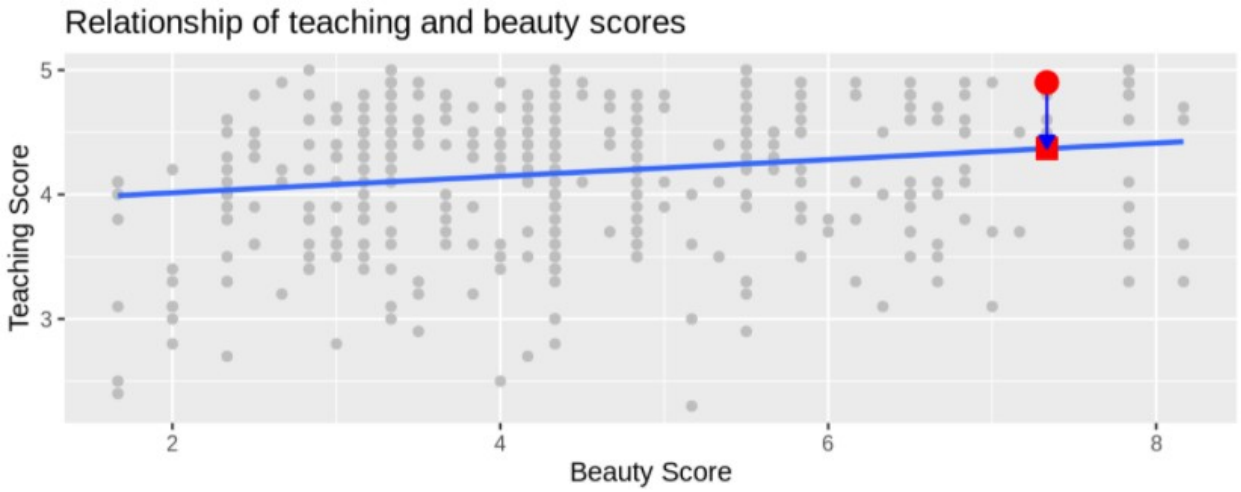
ID	score	bty_avg	age
21	4.9	7.33	31

Look at the table below to understand the equation:

- Circle: The *observed value* y = 4.9 is this course's instructor's actual teaching score.
- Square: The *fitted value* $\hat{y}$ is the value on the regression line for x = `bty_avg` = 7.333. This value is computed using the intercept and slope in the previous regression table:

$$\hat{y} = b_0 + b_1 \cdot x = 3.88 + 0.067 \cdot 7.333 = 4.369$$

- Arrow: The length of this arrow is the *residual* and is computed by subtracting the fitted value $\hat{y}$ from the observed value y. The residual can be thought of as a model's error or "lack of fit" for a particular observation. In the case of this course's instructor, it is $y - \hat{y}$ = 4.9 - 4.369 = 0.531.



Now say we want to compute both the fitted value $\hat{y} = b_0 + b_1 * x$ and the residual $y - \hat{y}$ for all 463 courses. Let's do this using a computer with the `get_regression_points()` function

```
# get results of only the 21st through 24th courses for brevity's sake
regression_points <- get_regression_points(score_model) %>%
  filter(ID == 21:24)

regression_points
```

```
## # A tibble: 4 x 5
##      ID score bty_avg score_hat residual
##   <int> <dbl>   <dbl>     <dbl>    <dbl>
## 1    21   4.9    7.33      4.37    0.531
## 2    22   4.6    7.33      4.37    0.231
## 3    23   4.5    7.33      4.37    0.131
## 4    24   4.4    5.5       4.25    0.153
```

1. The score column represents the observed outcome variable y. This is the y-position of the 463 black points
2. The bty_avg column represents the values of the explanatory variable x. This is the x-position of the 463 black points
3. The score_hat column represents the fitted values $\hat{y}$. This is the corresponding value on the regression line for the 463 x values
4. The residual column represents the residuals $y - \hat{y}$. This is the 463 vertical distances between the 463 black points and the regression line

1.2 One categorical explanatory variable

In this section, we'll explore differences in life expectancy in two ways:

1. Differences between continents: Are there significant differences in average life expectancy between the five populated continents of the world: Africa, the Americas, Asia, Europe, and Oceania?
2. Differences within continents: How does life expectancy vary within the world's five continents? For example, is the spread of life expectancy among the countries of Africa larger than the spread of life expectancy among the countries of Asia?

We'll use the `gapminder` data frame included in the `gapminder` package. This dataset has international development statistics such as life expectancy, GDP per capita, and population for 142 countries for 5-year intervals between 1952 and 2007. We'll use this data for basic regression again, but now using an explanatory variable x that is categorical:

1. A numerical outcome variable y (a country's life expectancy) and
2. A single categorical explanatory variable x (the continent that the country is a part of)

1.2.1 Exploratory data analysis

let's `filter()` for only those observations/rows corresponding to the year 2007 and `select()` these variables: country, lifeExp, continent and gdpPercap.

```
# assigned the value to a variable called "gapminder2007"
gapminder2007 <- gapminder %>%
  filter(year == 2007) %>%
  select(country, lifeExp, continent, gdpPercap)

glimpse(gapminder2007)
```

```
## Rows: 142
## Columns: 4
## $ country   <fct> Afghanistan, Albania, Algeria, Angola, Argentina, Austral...
## $ lifeExp   <dbl> 43.828, 76.423, 72.301, 42.731, 75.320, 81.235, 79.829, 7...
## $ continent <fct> Asia, Europe, Africa, Africa, Americas, Oceania, Europe, ...
## $ gdpPercap <dbl> 974.5803, 5937.0295, 6223.3675, 4797.2313, 12779.3796, 34...
```

```
head(gapminder2007, 10)
```

```
## # A tibble: 10 x 4
##    country     lifeExp continent gdpPercap
##    <fct>         <dbl> <fct>         <dbl>
##  1 Afghanistan    43.8 Asia           975.
##  2 Albania        76.4 Europe        5937.
##  3 Algeria        72.3 Africa        6223.
##  4 Angola         42.7 Africa        4797.
##  5 Argentina      75.3 Americas     12779.
##  6 Australia      81.2 Oceania      34435.
##  7 Austria        79.8 Europe       36126.
##  8 Bahrain        75.6 Asia         29796.
##  9 Bangladesh     64.1 Asia          1391.
## 10 Belgium        79.4 Europe       33693.
```

Let's look at a random sample of five out of the 142 countries

```
gapminder2007 %>%
  sample_n(size = 5)
```

```
## # A tibble: 5 x 4
##   country             lifeExp continent gdpPercap
##   <fct>                 <dbl> <fct>         <dbl>
## 1 Mongolia               66.8 Asia          3096.
## 2 Trinidad and Tobago    69.8 Americas     18009.
## 3 Germany                79.4 Europe       32170.
## 4 Myanmar                62.1 Asia           944
## 5 Portugal               78.1 Europe       20510.
```

Note that random sampling will likely produce a different subset of 5 rows for you than what's shown

Let's `select()` now the outcome and explanatory variables and apply the `skim()` function to see summary statistics

```
gapminder2007 %>%
  select(lifeExp, continent) %>%
  skim()
```

Table 3: Data summary

Name	Piped data
Number of rows	142
Number of columns	2
Column type frequency:	
factor	1
numeric	1
Group variables	None

Variable type: factor

skim_variable	n_missing	complete_rate	ordered	n_unique	top_counts
continent	0	1	FALSE	5	Afr: 52, Asi: 33, Eur: 30, Ame: 25

Variable type: numeric

skim_variable	n_missing	complete_rate	mean	sd	p0	p25	p50	p75	p100	hist
lifeExp	0	1	67.01	12.07	39.61	57.16	71.94	76.41	82.6	

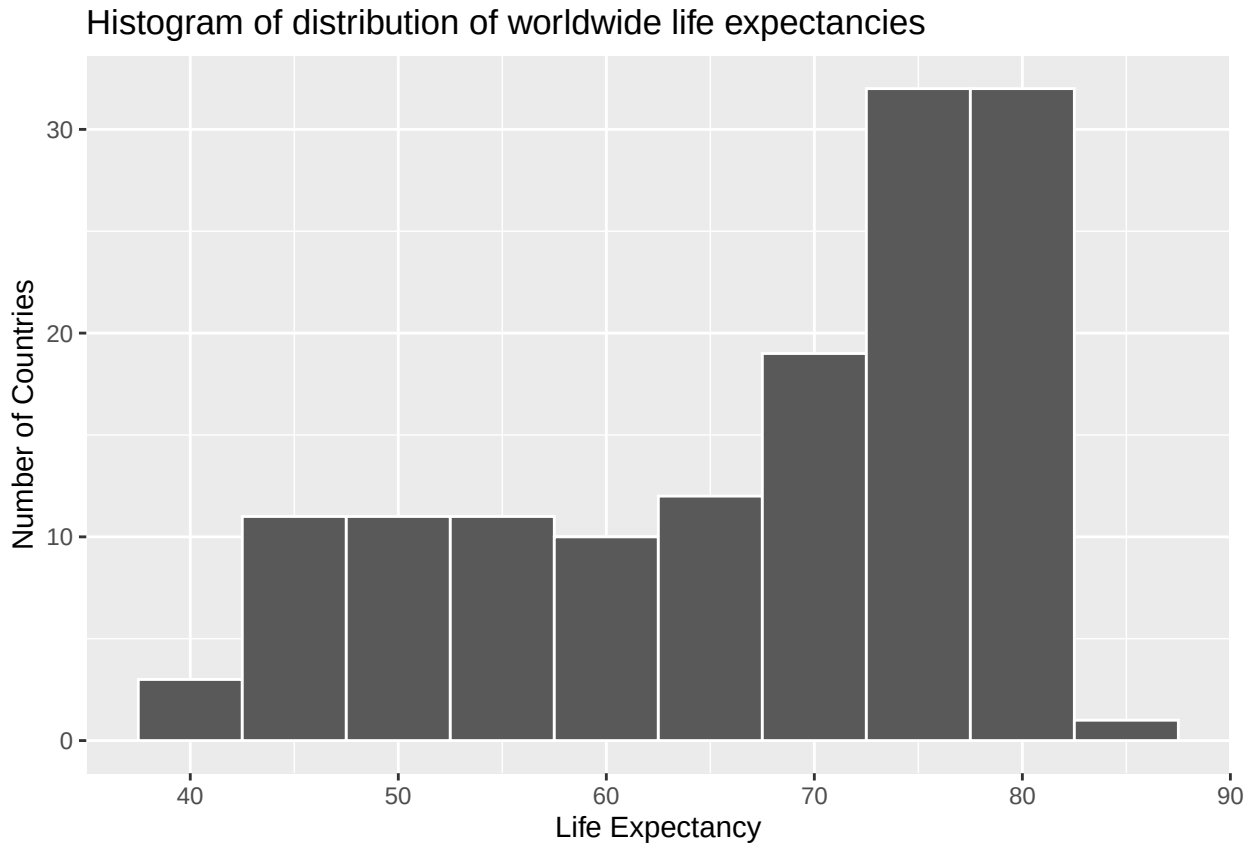
`skim()` reports summaries for categorical variables separately from the numerical variables

We observe that the global median life expectancy in 2007 was 71.94. Thus, half of the world's

countries (71 countries) had a life expectancy less than 71.94. The mean life expectancy of 67.01 is lower, however. Why is the mean life expectancy lower than the median?

Let's visualize the distribution of our outcome variable $y =$ `lifeExp`

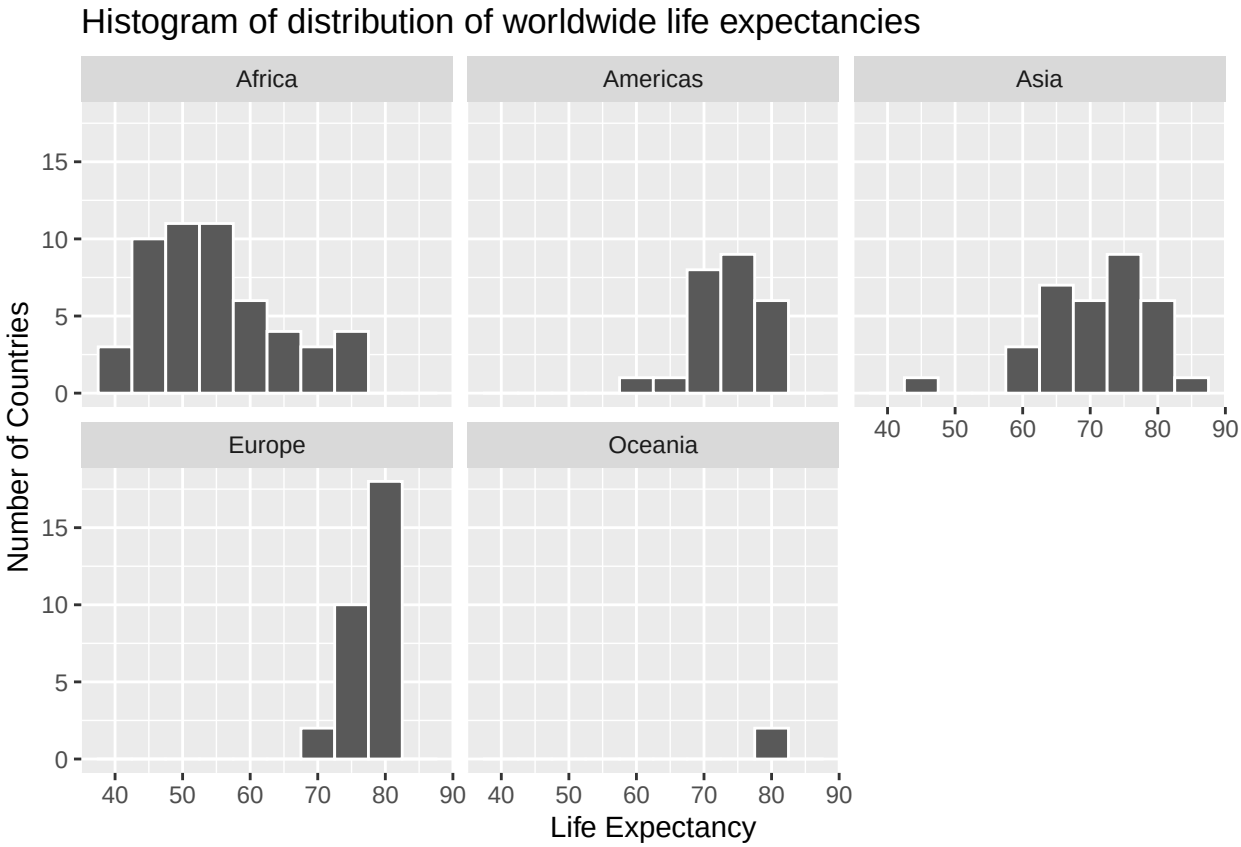
```
ggplot(gapminder2007, aes(x = lifeExp)) +
  geom_histogram(binwidth = 5, color = "white") +
  labs(x = "Life Expectancy",
       y = "Number of Countries",
       title = "Histogram of distribution of worldwide life expectancies")
```



We see that this data is left-skewed, also known as negatively skewed: there are a few countries with low life expectancy that are bringing down the mean life expectancy. However, the median is less sensitive to the effects of such outliers; hence, the median is greater than the mean in this case

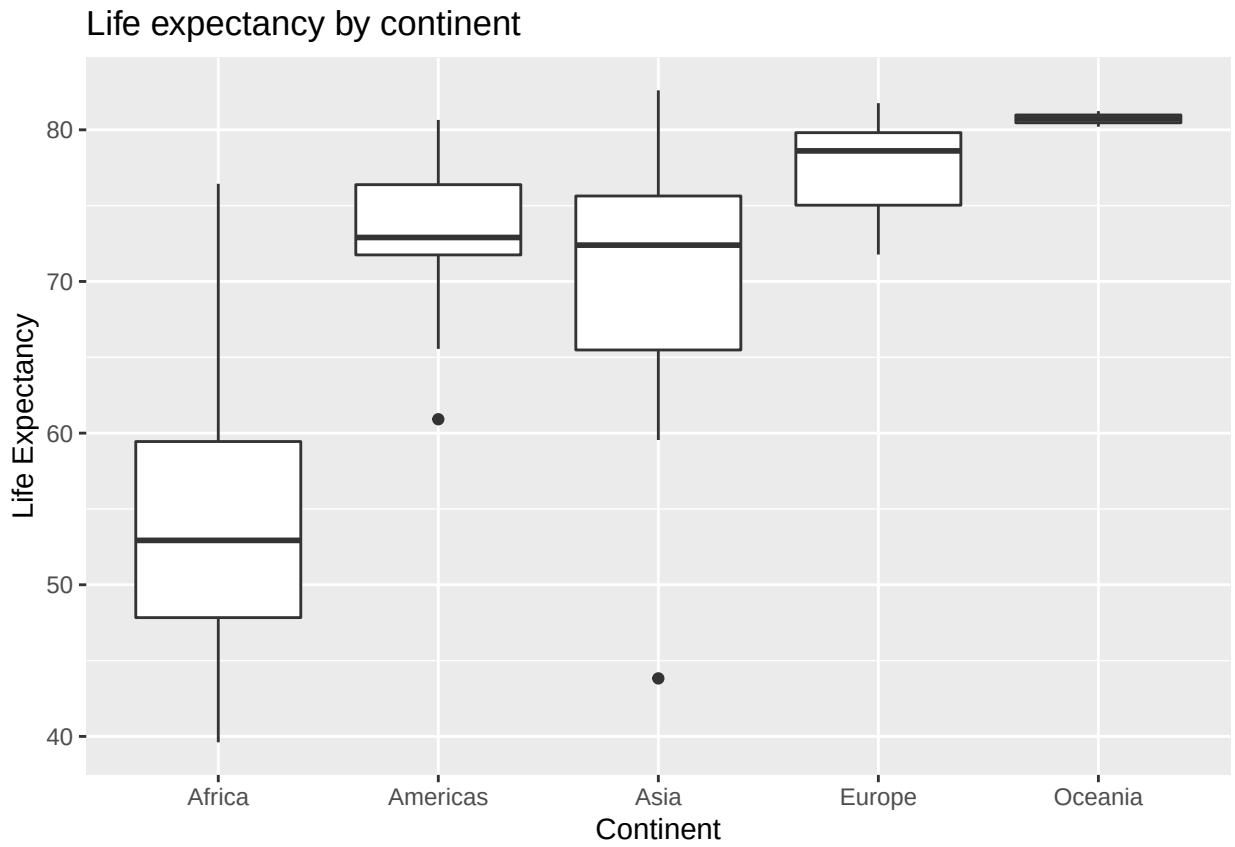
We also want to compare life expectancies both between continents and within continents. In other words, our visualizations need to incorporate some notion of the variable `continent`. We can do this easily with a faceted histogram

```
ggplot(gapminder2007, aes(x = lifeExp)) +
  geom_histogram(binwidth = 5, color = "white") +
  labs(x = "Life Expectancy",
       y = "Number of Countries",
       title = "Histogram of distribution of worldwide life expectancies") +
  facet_wrap(~ continent, nrow = 2)
```



An alternative method to visualize the distribution of a numerical variable split by a categorical variable is by using a side-by-side boxplot. We map the categorical variable continent to the x-axis and different life expectancies within each continent on the y-axis

```
ggplot(gapminder2007, aes(x = continent, y = lifeExp)) +
  geom_boxplot() +
  labs(x = "Continent",
       y = "Life Expectancy",
       title = "Life expectancy by continent")
```



We can quickly convince ourselves that Oceania has the highest median life expectancy by drawing an imaginary horizontal line at $y = 80$. Furthermore, as we observed in the faceted histogram, Africa and Asia have the largest variation in life expectancy as evidenced by their large interquartile ranges (the heights of the boxes)

Let's compute the median and mean life expectancy for each continent by using `group_by()` and `summarize()`

```
lifeExp_by_continent <- gapminder2007 %>%
  group_by(continent) %>%
  summarize(median = median(lifeExp),
            mean = mean(lifeExp))

lifeExp_by_continent
```

```
## # A tibble: 5 x 3
##   continent median  mean
##   <fct>      <dbl> <dbl>
## 1 Africa      52.9  54.8
## 2 Americas    72.9  73.6
## 3 Asia        72.4  70.7
## 4 Europe      78.6  77.6
## 5 Oceania     80.7  80.7
```

Using Africa's mean life expectancy of 54.8 as a baseline for comparison, let's start making comparisons

1. For the Americas, it is 73.6 - 54.8 = 18.8 years higher.
2. For Asia, it is 70.7 - 54.8 = 15.9 years higher.
3. For Europe, it is 77.6 - 54.8 = 22.8 years higher.
4. For Oceania, it is 80.7 - 54.8 = 25.9 years higher.

TABLE 5.7: Mean life expectancy by continent and relative differences from mean for Africa

continent	mean	Difference versus Africa
Africa	54.8	0.0
Americas	73.6	18.8
Asia	70.7	15.9
Europe	77.6	22.8
Oceania	80.7	25.9

1.2.2 Linear regression

Our model will not yield a "best-fitting" regression line but rather *offsets* relative to a baseline for comparison

1. We first "fit" the linear regression model using the `lm(y ~ x, data)` function and save it in `lifeExp_model`
2. We get the regression table by applying the `get_regression_table()` function from the `moderndive` package to `lifeExp_model`

```
lifeExp_model <- lm(lifeExp ~ continent, data = gapminder2007)
get_regression_table(lifeExp_model)
```

```
## # A tibble: 5 x 7
##   term              estimate std_error statistic p_value lower_ci upper_ci
##   <chr>                <dbl>     <dbl>     <dbl>   <dbl>    <dbl>    <dbl>
## 1 intercept             54.8      1.02     53.4        0     52.8     56.8
## 2 continentAmericas     18.8      1.8      10.4        0     15.2     22.4
## 3 continentAsia         15.9      1.65      9.68       0     12.7     19.2
## 4 continentEurope       22.8      1.70     13.5        0     19.5     26.2
## 5 continentOceania      25.9      5.33      4.86       0     15.4     36.4
```

1. `intercept` corresponds to the mean life expectancy of countries in Africa of 54.8 years
2. `continentAmericas` corresponds to countries in the Americas and the value +18.8 is the same difference in mean life expectancy relative to Africa we displayed in Table 5.7. In other words, the mean life expectancy of countries in the Americas is $54.8 + 18.8 = 73.6$. The same applies to the rest of the continents

The `estimate` column correspond to the "baseline for comparison" continent Africa (the intercept) as well as four "offsets" from this baseline for the remaining 4 continents: the Americas, Asia, Europe, and Oceania. Africa was chose as baseline for being first alphabetically

Let's now write the equation for our fitted values $\hat{y} = \widehat{\text{life exp}}$.

$$\begin{aligned}\hat{y} = \widehat{\text{life exp}} &= b_0 + b_{\text{Amer}} \cdot 1_{\text{Amer}}(x) + b_{\text{Asia}} \cdot 1_{\text{Asia}}(x) + \\ &\quad b_{\text{Euro}} \cdot 1_{\text{Euro}}(x) + b_{\text{Ocean}} \cdot 1_{\text{Ocean}}(x) \\ &= 54.8 + 18.8 \cdot 1_{\text{Amer}}(x) + 15.9 \cdot 1_{\text{Asia}}(x) + \\ &\quad 22.8 \cdot 1_{\text{Euro}}(x) + 25.9 \cdot 1_{\text{Ocean}}(x)\end{aligned}$$

Whoa! That looks daunting! Don't fret, however, as once you understand what all the elements mean, things simplify greatly. First, $1_A(x)$ is what's known in mathematics as an "indicator function." It returns only one of two possible values, 0 and 1, where

$$1_A(x) = \begin{cases} 1 & \text{if } x \text{ is in } A \\ 0 & \text{if otherwise} \end{cases}$$

In a statistical modeling context, this is also known as a *dummy variable*. In our case, let's consider the first such indicator variable $1_{\text{Amer}}(x)$. This indicator function returns 1 if a country is in the Americas, 0 otherwise:

$$1_{\text{Amer}}(x) = \begin{cases} 1 & \text{if country } x \text{ is in the Americas} \\ 0 & \text{otherwise} \end{cases}$$

Second, b_0 corresponds to the intercept as before; in this case, it's the mean life expectancy of all countries in Africa. Third, the b_{Amer}, b_{Asia}, b_{Euro}, and b_{Ocean} represent the 4 "offsets relative to the baseline for comparison" in the regression table output in Table 5.8: `continentAmericas`, `continentAsia`, `continentEurope`, and `continentOceania`.

Let's put this all together and compute the fitted value $\hat{y} = \widehat{\text{life exp}}$ for a country in Africa. Since the country is in Africa, all four indicator functions $1_{\text{Amer}}(x)$, $1_{\text{Asia}}(x)$, $1_{\text{Euro}}(x)$, and $1_{\text{Ocean}}(x)$ will equal 0, and thus:

$$\begin{aligned}\widehat{\text{life exp}} &= b_0 + b_{\text{Amer}} \cdot 1_{\text{Amer}}(x) + b_{\text{Asia}} \cdot 1_{\text{Asia}}(x) + \\ &\quad b_{\text{Euro}} \cdot 1_{\text{Euro}}(x) + b_{\text{Ocean}} \cdot 1_{\text{Ocean}}(x) \\ &= 54.8 + 18.8 \cdot 1_{\text{Amer}}(x) + 15.9 \cdot 1_{\text{Asia}}(x) + \\ &\quad 22.8 \cdot 1_{\text{Euro}}(x) + 25.9 \cdot 1_{\text{Ocean}}(x) \\ &= 54.8 + 18.8 \cdot 0 + 15.9 \cdot 0 + 22.8 \cdot 0 + 25.9 \cdot 0 \\ &= 54.8\end{aligned}$$

In other words, all that's left is the intercept b_0, corresponding to the average life expectancy of African countries of 54.8 years. Next, say we are considering a country in the Americas. In this case, only the indicator function $1_{\text{Amer}}(x)$ for the Americas will equal 1, while all the others will equal 0, and thus:

$$\begin{aligned}\widehat{\text{life exp}} &= 54.8 + 18.8 \cdot 1_{\text{Amer}}(x) + 15.9 \cdot 1_{\text{Asia}}(x) + 22.8 \cdot 1_{\text{Euro}}(x) + \\ &\quad 25.9 \cdot 1_{\text{Ocean}}(x) \\ &= 54.8 + 18.8 \cdot 1 + 15.9 \cdot 0 + 22.8 \cdot 0 + 25.9 \cdot 0 \\ &= 54.8 + 18.8 \\ &= 73.6\end{aligned}$$

which is the mean life expectancy for countries in the Americas of 73.6 years in Table 5.7. Note the "offset from the baseline for comparison" is +18.8 years.

Let's do one more. Say we are considering a country in Asia. In this case, only the indicator function $1_{\text{Asia}}(x)$ for Asia will equal 1, while all the others will equal 0, and thus:

$$\begin{aligned}\widehat{\text{life exp}} &= 54.8 + 18.8 \cdot 1_{\text{Amer}}(x) + 15.9 \cdot 1_{\text{Asia}}(x) + 22.8 \cdot 1_{\text{Euro}}(x) + \\ &\quad 25.9 \cdot 1_{\text{Ocean}}(x) \\ &= 54.8 + 18.8 \cdot 0 + 15.9 \cdot 1 + 22.8 \cdot 0 + 25.9 \cdot 0 \\ &= 54.8 + 15.9 \\ &= 70.7\end{aligned}$$

which is the mean life expectancy for Asian countries of 70.7 years in Table 5.7. The "offset from the baseline for comparison" here is +15.9 years.

Let's generalize this idea a bit. If we fit a linear regression model using a categorical explanatory variable x that has k possible categories, the regression table will return an intercept and $k-1$ "offsets." In our case, since there are $k = 5$ continents, the regression model returns an intercept corresponding to the baseline for comparison group of Africa and $k-1 = 4$ offsets corresponding to the Americas, Asia, Europe, and Oceania.

1.2.3 Observed/fitted values and residuals

1. Observe values y, or the observed value of the outcome variable
2. Fitted values $\hat{y}$, or the value on the regression line for a given x value
3. Residuals $y - \hat{y}$, or the error between the observed value and the fitted value

Use the `get_regression_points()` function and add an argument setting `ID = "country"`: this is telling the function to use the variable `country` in `gapminder2007` as an *identification variable* in the output

```
regression_points <- get_regression_points(lifeExp_model, ID = "country")

head(regression_points, 10)
```

```
## # A tibble: 10 x 5
##    country     lifeExp continent lifeExp_hat residual
##    <fct>         <dbl> <fct>           <dbl>    <dbl>
```

```
##  1 Afghanistan    43.8 Asia            70.7  -26.9
##  2 Albania        76.4 Europe          77.6   -1.23
##  3 Algeria        72.3 Africa          54.8   17.5
##  4 Angola         42.7 Africa          54.8  -12.1
##  5 Argentina      75.3 Americas        73.6    1.71
##  6 Australia      81.2 Oceania         80.7    0.516
##  7 Austria        79.8 Europe          77.6    2.18
##  8 Bahrain        75.6 Asia            70.7    4.91
##  9 Bangladesh     64.1 Asia            70.7   -6.67
## 10 Belgium        79.4 Europe          77.6    1.79
```

Observe in Table 5.9 that `lifeExp_hat` contains the fitted values $\hat{y} = \widehat{\text{lifeExp}}$. If you look closely, there are only 5 possible values for `lifeExp_hat`. These correspond to the five mean life expectancies for the 5 continents that we displayed in Table 5.7 and computed using the values in the `estimate` column of the regression table in Table 5.8.

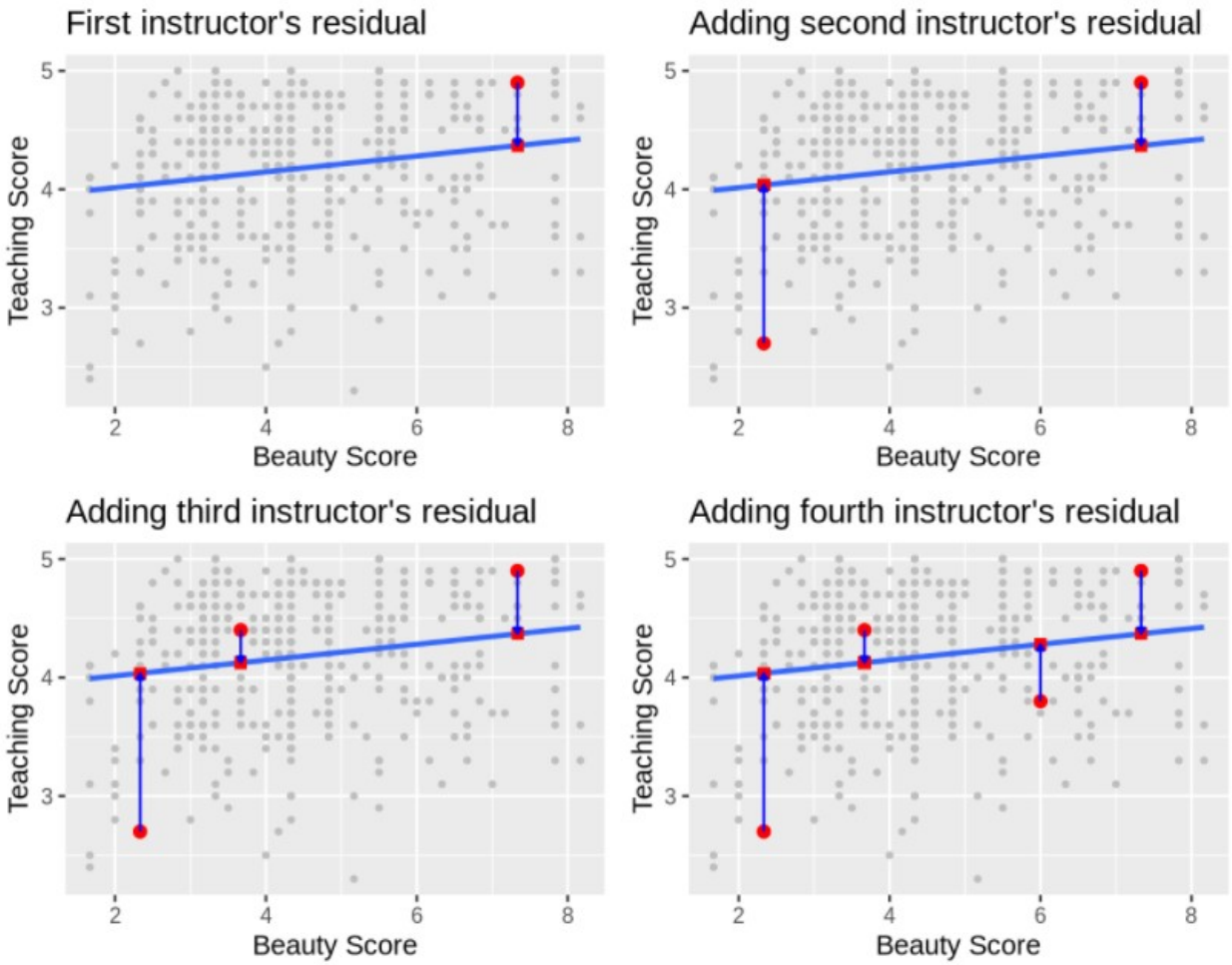
The `residual` column is simply $y - \hat{y}$ = `lifeExp - lifeExp_hat`. These values can be interpreted as the deviation of a country's life expectancy from its continent's average life expectancy. For example, look at the first row of Table 5.9 corresponding to Afghanistan. The residual of $y - \hat{y} = 43.8 - 70.7 = -26.9$ is telling us that Afghanistan's life expectancy is a whopping 26.9 years lower than the mean life expectancy of all Asian countries. This can in part be explained by the many years of war that country has suffered.

1.3 Related topics

1.3.1 Best-fitting line

What do we mean by "best"?

Recall that for an instructor with a beauty score of $x = 7.333$ we mark the observed value y with a circle, the fitted value $\hat{y}$ with a square, and the residual $y - \hat{y}$ with an arrow. We will add three more arbitrarily chosen course instructors:



The three other plots refer to:

1. A course whose instructor had a "beauty" score x = 2.333 and teaching score y = 2.7. The residual in this case is $2.7 - 4.036 = -1.336$, which we mark with a new blue arrow in the top-right plot.
2. A course whose instructor had a "beauty" score $x = 3.667$ and teaching score $y = 4.4$. The residual in this case is $4.4 - 4.125 = 0.2753$, which we mark with a new blue arrow in the bottom-left plot.
3. A course whose instructor had a "beauty" score $x = 6$ and teaching score $y = 3.8$. The residual in this case is $3.8 - 4.28 = -0.4802$, which we mark with a new blue arrow in the bottom-right plot.

Now say we repeated this process of computing residuals for all 463 courses' instructors, then we squared all the residuals, and then we summed them. We call this quantity the *sum of squared residuals*; it is a measure of the *lack of fit* of a model. Larger values of the sum of squared residuals indicate a bigger lack of fit. This corresponds to a worse fitting model.

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

```r
# fit regression model
score_model <- lm(score ~ bty_avg,
                  data = evals_ch5)

# get regression points
regression_points <- get_regression_points(score_model)

regression_points
```

```
## # A tibble: 463 x 5
##       ID score bty_avg score_hat residual
##    <int> <dbl>   <dbl>     <dbl>    <dbl>
##  1     1   4.7    5         4.21    0.486
##  2     2   4.1    5         4.21   -0.114
##  3     3   3.9    5         4.21   -0.314
##  4     4   4.8    5         4.21    0.586
##  5     5   4.6    3         4.08    0.52
##  6     6   4.3    3         4.08    0.22
##  7     7   2.8    3         4.08   -1.28
##  8     8   4.1    3.33      4.10   -0.002
##  9     9   3.4    3.33      4.10   -0.702
## 10    10   4.5    3.17      4.09    0.409
## # ... with 453 more rows
```

```r
# compute sum of squared residuals
regression_points %>%
  mutate(squared_residuals = residual^2) %>%
  summarize(sum_of_squared_residuals = sum(squared_residuals))
```

```
## # A tibble: 1 x 1
##   sum_of_squared_residuals
##                      <dbl>
## 1                     132.
```

Any other straight line drawn in the figure would yield a sum of squared residuals greater than 132.