STT 3850 : Week 5

Spring 2024

Appalachian State University

Section 1

Outline for the week

By the end of the week: Multiple Linear Regression

- Simple Linear Regression (One categorical explanatory variable)
- Multiple Regression (One numerical and one categorical explanatory variable)
- Interaction model
- Two numerical explanatory variables

Section 2

One categorical explanatory variable

One categorical explanatory variable

- It's an unfortunate truth that life expectancy is not the same across all countries in the world.
    - International development agencies are interested in studying these differences in life expectancy in the hopes of identifying where governments should allocate resources to address this problem.
- In this section, we'll explore differences in life expectancy in two ways:
    - 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?
    - 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?

One categorical explanatory variable

- To answer such questions, 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.
  - Recall we visualized some of this data earlier.
- We'll use this data for basic regression again, but now using an explanatory variable x that is **categorical**.
  - A numerical outcome variable y (a country's life expectancy) and
  - A single categorical explanatory variable x (the continent that the country is a part of).

Needed packages

Loading needed packages.

```
library(tidyverse)    # loading collection of packages
library(moderndive)   # datasets and regression functions
library(skimr)        # provides a simple-to-use functions
                      # for summary statistics
library(gapminder)    # datasets
```

Exploratory data analysis

- The data on the 142 countries can be found in the `gapminder` data frame included in the `gapminder` package.
    - However, to keep things simple, let's `filter()` for only those observations/rows corresponding to the year 2007.
    - Additionally, let's `select()` only the subset of the variables we'll consider in this chapter. We'll save this data in a new data frame called `gapminder2007`:

```
gapminder2007 <- gapminder %>%
  filter(year == 2007) %>%
  select(country, lifeExp, continent, gdpPercap)
glimpse(gapminder2007)
```

```
Rows: 142
Columns: 4
$ country   <fct> "Afghanistan", "Albania", "Algeria", "Angola", "A
$ lifeExp   <dbl> 43.828, 76.423, 72.301, 42.731, 75.320, 81.235, 7
$ continent <fct> Asia, Europe, Africa, Africa, Americas, Oceania, E
$ gdpPercap <dbl> 974.5803, 5937.0295, 6223.3675, 4797.2313, 12779.3
```

Exploratory data analysis

- A full description of all the variables included in `gapminder` can be found by reading the associated help file (run `?gapminder` in the console).
- However, let's fully describe the 4 variables we selected in `gapminder2007`:
  1. `country`: An identification variable of type character/text used to distinguish the 142 countries in the dataset.
  2. `lifeExp`: A numerical variable of that country's life expectancy at birth. This is the outcome variable y of interest.
  3. `continent`: A categorical variable with five levels. Here "levels" correspond to the possible categories: Africa, Asia, Americas, Europe, and Oceania. This is the explanatory variable x of interest.
  4. `gdpPercap`: A numerical variable of that country's GDP per capita in US inflation-adjusted dollars.

Exploratory data analysis

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 Cote d'Ivoire    48.3 Africa         1545.
2 Venezuela        73.7 Americas      11416.
3 Nigeria          46.9 Africa         2014.
4 South Africa     49.3 Africa         9270.
5 Lesotho          42.6 Africa         1569.
```

Exploratory data analysis

Let's compute the summary statistics using the `skim()` function

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

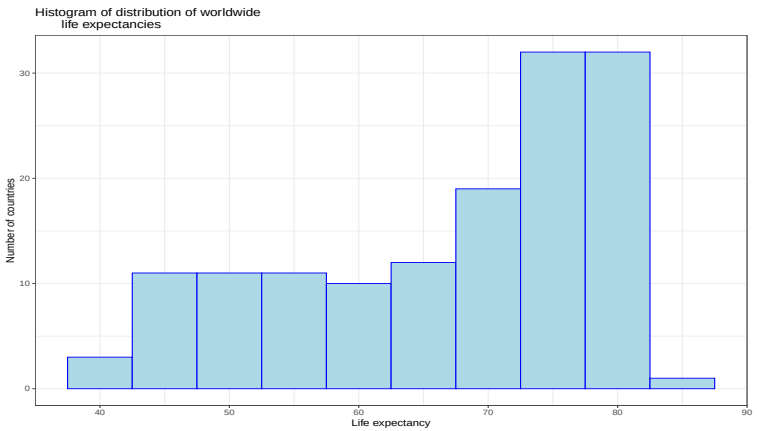
```
# Or using summary()
gapminder2007 %>%
  select(lifeExp, continent) %>%
  summary()
```

```
    lifeExp          continent
 Min.   :39.61   Africa  :52
 1st Qu.:57.16   Americas:25
 Median :71.94   Asia    :33
 Mean   :67.01   Europe  :30
 3rd Qu.:76.41   Oceania : 2
 Max.   :82.60
```

Exploratory data analysis

Why is the mean life expectancy lower than the median?

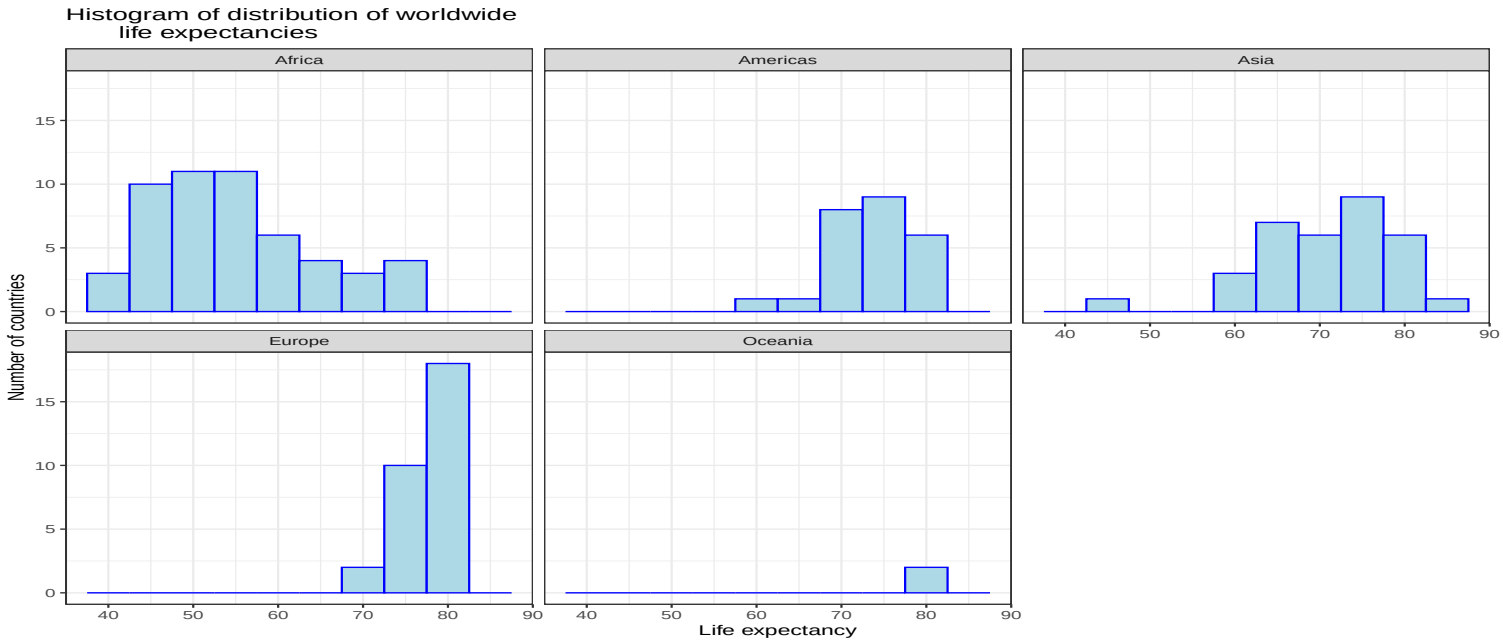
```
ggplot(gapminder2007, aes(x = lifeExp)) +
  geom_histogram(binwidth=5, color = "blue", fill = "lightblue") +
  labs(x = "Life expectancy", y = "Number of countries",
       title = "Histogram of distribution of worldwide
       life expectancies") +
  theme_bw()
```



We see that this data is skewed to the left $\rightarrow$ mean $<$ median.

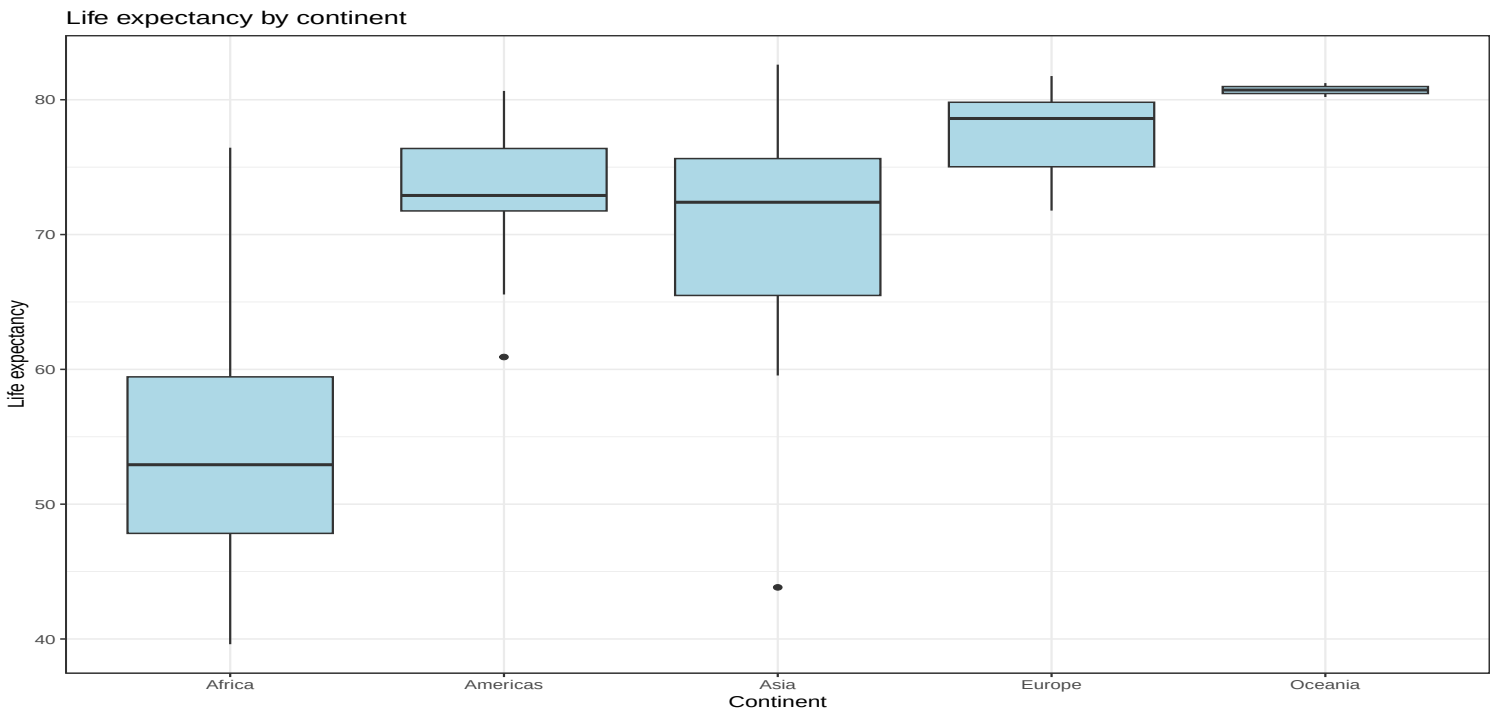
Exploratory data analysis

```
ggplot(gapminder2007, aes(x = lifeExp)) +
  geom_histogram(binwidth = 5, color = "blue",
                 fill = "lightblue") +
  labs(x = "Life expectancy",
       y = "Number of countries",
       title = "Histogram of distribution of worldwide
       life expectancies") +
  facet_wrap(vars(continent), nrow = 2) +
  theme_bw()
```



Exploratory data analysis

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



Exploratory data analysis

```
lifeExp_by_continent <- gapminder2007 %>%
  group_by(continent) %>%
  summarize(median = median(lifeExp),
            mean = mean(lifeExp)) %>%
  mutate(`Difference versus Africa` = mean - mean[1])
knitr::kable(lifeExp_by_continent)
```

continent	median	mean	Difference versus Africa
Africa	52.9265	54.80604	0.00000
Americas	72.8990	73.60812	18.80208
Asia	72.3960	70.72848	15.92245
Europe	78.6085	77.64860	22.84256
Oceania	80.7195	80.71950	25.91346

Linear regression

In our life expectancy example, we now instead have a **categorical** explanatory variable `continent`:

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

term	estimate	std_error	statistic	p_value	lower_ci	upper_ci
intercept	54.806	1.025	53.446	0	52.778	56.834
continent: Americas	18.802	1.800	10.448	0	15.243	22.361
continent: Asia	15.922	1.646	9.675	0	12.668	19.177
continent: Europe	22.843	1.695	13.474	0	19.490	26.195
continent: Oceania	25.913	5.328	4.863	0	15.377	36.450

Our model will not yield a "best-fitting" regression line like when the x is continuous, but rather offsets relative to a baseline for comparison.

Linear regression

Let's break the 5 estimates down one-by-one:

- `intercept` corresponds to the mean life expectancy of countries in Africa of 54.8 years.
- `continent: Americas` corresponds to countries in the Americas and the value $+18.8$ is the same difference in mean life expectancy relative to Africa we displayed earlier. In other words, the mean life expectancy of countries in the Americas is $54.8 + 18.8 = 73.6$.
- `continent: Asia` the mean life expectancy of countries in Asia is $54.8 + 15.9 = 70.7$.
- `continent: Europe` the mean life expectancy of countries in Europe is $54.8 + 22.8 = 77.6$.
- `continent: Oceania` the mean life expectancy of countries in Oceania is $54.8 + 25.9 = 80.7$.

Linear regression

We can change the baseline group to be another continent. In what follows the baseline is changed to be `Americas` instead of `Africa`.

```
gapminder2007$continent <- relevel(gapminder2007$continent, ref='Americas')
lifeExp_model1 <- lm(lifeExp ~ continent, data = gapminder2007)
knitr::kable(get_regression_table(lifeExp_model1))
```

term	estimate	std_error	statistic	p_value	lower_ci	upper_ci
intercept	73.608	1.479	49.772	0.000	70.684	76.533
continent: Africa	-18.802	1.800	-10.448	0.000	-22.361	-15.243
continent: Asia	-2.880	1.961	-1.469	0.144	-6.757	0.997
continent: Europe	4.040	2.002	2.018	0.046	0.081	8.000
continent: Oceania	7.111	5.434	1.309	0.193	-3.634	17.857

Linear regression

The equation for our fitted values for model (`lifeExp_model`) is written as:

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

where for example:

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

Linear regression

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 $1_{\text{Euro}}(x) = 1_{\text{Amer}}(x) = 1_{\text{Asia}}(x) = 1_{\text{Ocean}}(x) = 0$.

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

Linear regression

For a country in in the Americas, in this case, only the indicator function $1_{\text{Amer}}(x) = 1$.

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

In general, 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.

Observed/fitted values and residuals

```
regression_points <- get_regression_points(lifeExp_model,
                                           ID = "country")
knitr::kable(regression_points %>% head(n = 9))
```

country	lifeExp	continent	lifeExp_hat	residual
Afghanistan	43.828	Asia	70.728	-26.900
Albania	76.423	Europe	77.649	-1.226
Algeria	72.301	Africa	54.806	17.495
Angola	42.731	Africa	54.806	-12.075
Argentina	75.320	Americas	73.608	1.712
Australia	81.235	Oceania	80.720	0.516
Austria	79.829	Europe	77.649	2.180
Bahrain	75.635	Asia	70.728	4.907
Bangladesh	64.062	Asia	70.728	-6.666

Section 3

Multiple Regression

Multiple Regression

In the previous chapter, we introduced ideas related to modeling for **explanation.**

- In particular that the goal of modeling is to make explicit the relationship between some outcome variable y and some explanatory variable x.
- We focused on linear regression, where we only considered one explanatory x variable that is either numeric or categorical.
- Now, we we'll start considering models that include more than one explanatory variable x.
    - NOTE: the interpretation of the associated effect of any one explanatory variable must be made in conjunction with the other explanatory variables included in your model.

Needed packages

Let's load all the packages needed for this chapter.

```
library(tidyverse)
library(moderndive)
library(skimr)
library(ISLR)
```

One numerical and one categorical explanatory variable

Let's revisit the instructor evaluation data we introduced earlier.

- We studied the relationship between teaching evaluation scores as given by students and "beauty" scores.
  - The variable teaching score was the numerical outcome variable y, and the variable "beauty" score (`bty_avg`) was the numerical explanatory x variable.
- In this section, we are going to consider a different model
  - Our outcome variable will still be teaching score, but
  - we'll now include two different explanatory variables: `age` and `gender`.
  - Could it be that instructors who are older receive better teaching evaluations from students?
  - Or could it instead be that younger instructors receive better evaluations?
  - Are there differences in evaluations given by students for instructors of different genders?

Exploratory data analysis

Let's `select()` only the subset of the variables we'll consider in this chapter.

```
evals_ch6 <- evals %>%
  select(ID, score, age, gender)
```

Recall the three common steps in an exploratory data analysis:

1. Looking at the raw data values.
2. Computing summary statistics.
3. Creating data visualizations.

Exploratory data analysis

Let's first look at the raw data values.

```
glimpse(evals_ch6)
```

```
Rows: 463
Columns: 4
$ ID     <int> 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16
$ 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, 4
$ gender <fct> female, female, female, female, male, male, male, mal
```

```
# Or
evals_ch6 %>%
  sample_n(size = 2)
```

```
# A tibble: 2 x 4
     ID score   age gender
  <int> <dbl> <int> <fct>
1   434   2.8    62 male
2   208   4.4    62 male
```

Exploratory data analysis

```
evals_ch6 %>%
  select(score, age, gender) %>%
  skim()
```

```
# Or
evals_ch6 %>%
  select(score, age, gender) %>%
  summary()
```

```
     score             age            gender
 Min.   :2.300   Min.   :29.00   female:195
 1st Qu.:3.800   1st Qu.:42.00   male  :268
 Median :4.300   Median :48.00
 Mean   :4.175   Mean   :48.37
 3rd Qu.:4.600   3rd Qu.:57.00
 Max.   :5.000   Max.   :73.00
```

Exploratory data analysis

Let's compute the correlation coefficient between our two numerical variables: `score` and age:

```
evals_ch6 %>%
  summarize(r = cor(score, age))
```

```
# A tibble: 1 x 1
      r
   <dbl>
1 -0.107
```

```
# or using the get_correlation wrapper
# from moderndive
evals_ch6 %>%
  get_correlation(score ~ age)
```

```
# A tibble: 1 x 1
     cor
   <dbl>
1 -0.107
```

Exploratory data analysis

```
ggplot(evals_ch6, aes(x = age, y = score, color = gender)) +
  geom_point() +
  labs(x = "Age", y = "Teaching Score", color = "Gender",
       title = "Interaction Model") +
  geom_smooth(method = "lm", se = FALSE) +
  theme_bw() -> int_mod
int_mod
```

Exploratory data analysis

Interaction model

- Let's now quantify the relationship of our outcome variable y and the two explanatory variables using one type of multiple regression model known as an **interaction model**.
- Going back to our multiple regression model for teaching `score` using age and `gender` in the figure above, we generate the regression table using the same two-step approach.
    1. First, "fit" a model using the `lm()` (linear model) function of the form `y ~x1 + x2 + x1:x2` which is the same as `y ~ x1*x2` in R's modeling notation.
    2. Second, apply `get_regression_table()` or `summary()` to the linear model object created in 1.

Interaction model

```
# Fit regression model:
score_model_interaction <- lm(score ~ age + gender + age:gender,
                              data = evals_ch6)
# Get regression table:
knitr::kable(get_regression_table(score_model_interaction))
```

term	estimate	std_error	statistic	p_value	lower_ci	upper_ci
intercept	4.883	0.205	23.795	0.000	4.480	5.286
age	-0.018	0.004	-3.919	0.000	-0.026	-0.009
gender: male	-0.446	0.265	-1.681	0.094	-0.968	0.076
age:gendermale	0.014	0.006	2.446	0.015	0.003	0.024

Interaction model

Female instructors are the "baseline for comparison" group.

- The estimate for `intercept` is the intercept for only the `female` instructors.
- The estimate for age is the slope for age for only the female instructors.
- Thus, the red regression line in Figure 1 has an intercept of 4.883 and slope for age of -0.018.
- Note: The intercept has no practical interpretation since instructors can not have a **zero** age.

Interaction model

What about the intercept and slope for age of the male instructors in the blue line of Figure 1?

- The value for `gender: male` that appears in the Table (-0.446) is not the intercept for the male instructors but rather the offset in intercept for male instructors relative to female instructors.
  - The intercept for the male instructors is `intercept + gender:male` $= 4.883 + (-0.446) = 4.883 - 0.446 = 4.437$.
- Similarly, `age:gendermale` $= 0.014$ is not the slope for age for the male instructors, but rather the offset in slope for the male instructors.
  - Therefore, the slope for age for the male instructors is `age + age:gendermale` $= -0.018 + 0.014 = -0.004$.

Interaction model

Gender	Intercept	Slope for age
Female instructors	4.883	-0.018
Male instructors	4.437	-0.004

- Since the slope for age for the female instructors was -0.018, it means that on average, a female instructor who is a year older would have a teaching score that is 0.018 units **lower**.
- For the male instructors, the associated decrease in score is 0.004 units.
- While both slopes for age were negative, the slope for age for the female instructors is **larger** in magnitude.

Interaction model: Prediction

Let's now write the equation for our regression lines, which we can use to compute our fitted values

$$\begin{aligned}\hat{y} &= \widehat{\text{score}} = b_0 + b_{\text{age}} \cdot \text{age} + b_{\text{male}} \cdot 1_{\text{is male}}(x) + b_{\text{age:gender}} \cdot \text{age} \cdot 1_{\text{is male}}(x) \\ &= 4.883 - 0.018 \cdot \text{age} - 0.446 \cdot 1_{\text{is male}}(x) + 0.014 \cdot \text{age} \cdot 1_{\text{is male}}(x).\end{aligned}$$

where:

$$1_{\text{is male}}(x) = \begin{cases} 1 & \text{if instructor } x \text{ is male} \\ 0 & \text{otherwise} \end{cases}$$

Interaction model: Prediction

Let's put this all together and compute the fitted value $\hat{y} = \widehat{\text{score}}$ for female instructors.

- Since for female instructors $1_{\text{is male}}(x) = 0$.

$$\begin{aligned} \hat{y} = \widehat{\text{score}} &= 4.883 - 0.018 \cdot \text{age} - 0.446 \cdot 0 + 0.014 \cdot \text{age} \cdot 0. \\ &= 4.883 - 0.018 \cdot \text{age} \end{aligned}$$

- For male instructors $1_{\text{is male}}(x) = 1$.

$$\begin{aligned} \hat{y} = \widehat{\text{score}} &= 4.883 - 0.018 \cdot \text{age} - 0.446 \cdot 1 + 0.014 \cdot \text{age} \cdot 1. \\ &= (4.883 - 0.446) + (-0.018 + 0.014) \cdot \text{age} \\ &= 4.437 - 0.004 \cdot \text{age} \end{aligned}$$

Interaction model: Explanation

- The term $b_{\text{age:gender}}$ in the equation for the fitted value $\hat{y} = \widehat{\text{score}}$ is what's known in statistical modeling as an **interaction effect**.
- We say there is an interaction effect if the associated effect of one variable depends on the value of another variable.
    - Here, the associated effect of the variable age depends on the value of the other variable gender.
    - The difference in slopes for age of $+0.014$ of male instructors relative to female instructors shows this.

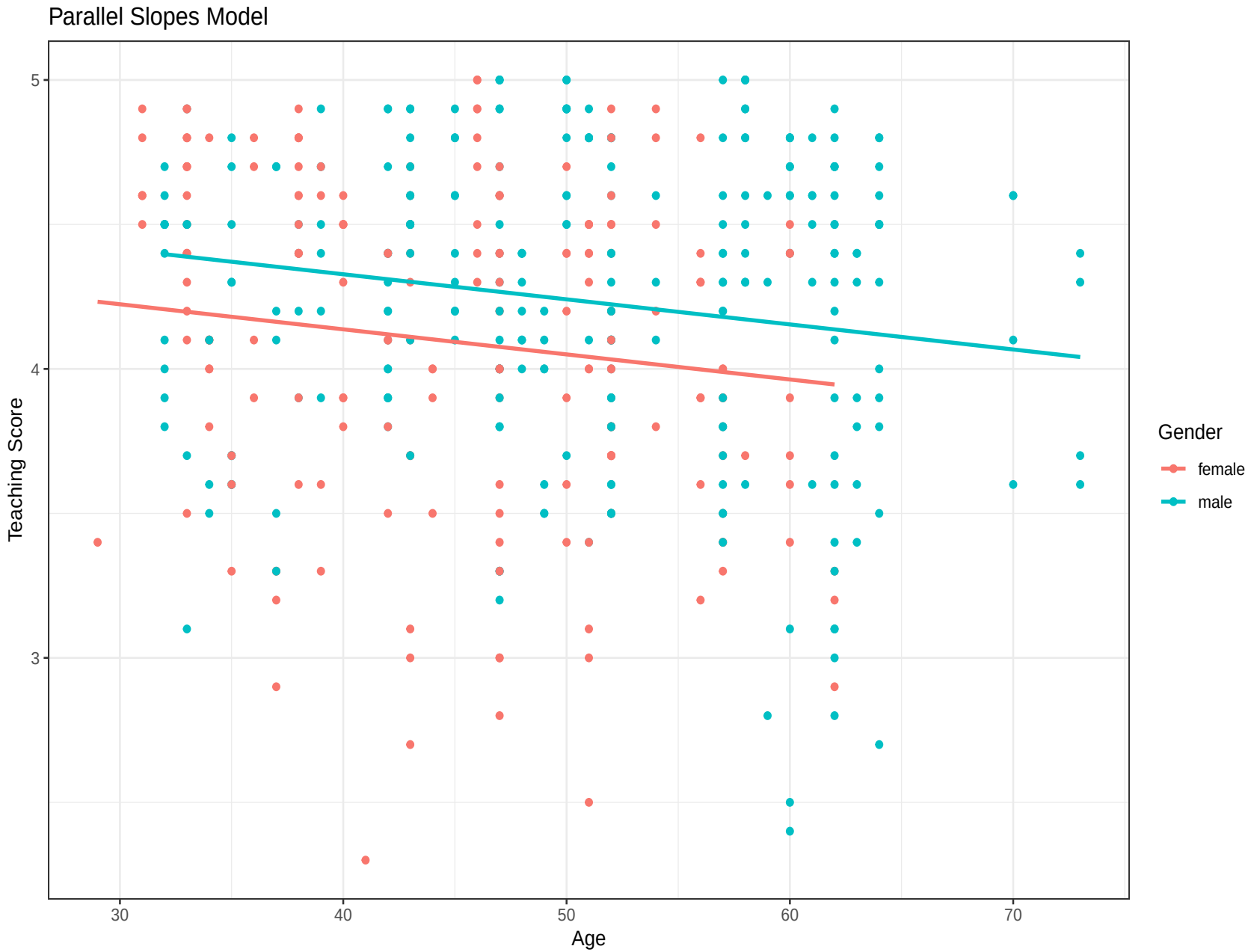
Parallel slopes model

With one numerical and one categorical explanatory variable, another type of model we can use is known as a **parallel slopes** model.

- Unlike interaction models, parallel slopes models still allow for different intercepts but force all lines to have the same slope.

```
ggplot(evals_ch6, aes(x = age, y = score, color = gender)) +
  geom_point() +
  labs(x = "Age", y = "Teaching Score", color = "Gender",
       title = "Parallel Slopes Model") +
  geom_parallel_slopes(se = FALSE) +
  theme_bw() -> ps_mod
ps_mod
```

Parallel slopes model



Parallel slopes model

```
# Fit regression model:
score_model_parallel_slopes <- lm(score ~ age + gender,
                                  data = evals_ch6)
# Get regression table:
knitr::kable(get_regression_table(score_model_parallel_slopes))
```

term	estimate	std_error	statistic	p_value	lower_ci	upper_ci
intercept	4.484	0.125	35.792	0.000	4.238	4.730
age	-0.009	0.003	-3.280	0.001	-0.014	-0.003
gender: male	0.191	0.052	3.632	0.000	0.087	0.294

Gender	Intercept	Slope for age
Female instructors	4.484	-0.009
Male instructors	4.675	-0.009

Parallel slopes model: Prediction

Let's now write the equation for our regression lines, which we can use to compute our fitted values

$$\begin{aligned}\hat{y} = \widehat{\text{score}} &= b_0 + b_{\text{age}} \cdot \text{age} + b_{\text{male}} \cdot 1_{\text{is male}}(x) \\ &= 4.484 - 0.009 \cdot \text{age} + 0.191 \cdot 1_{\text{is male}}(x).\end{aligned}$$

Parallel slopes model: Prediction

Let's put this all together and compute the fitted value $\hat{y} = \widehat{\text{score}}$ for female instructors.

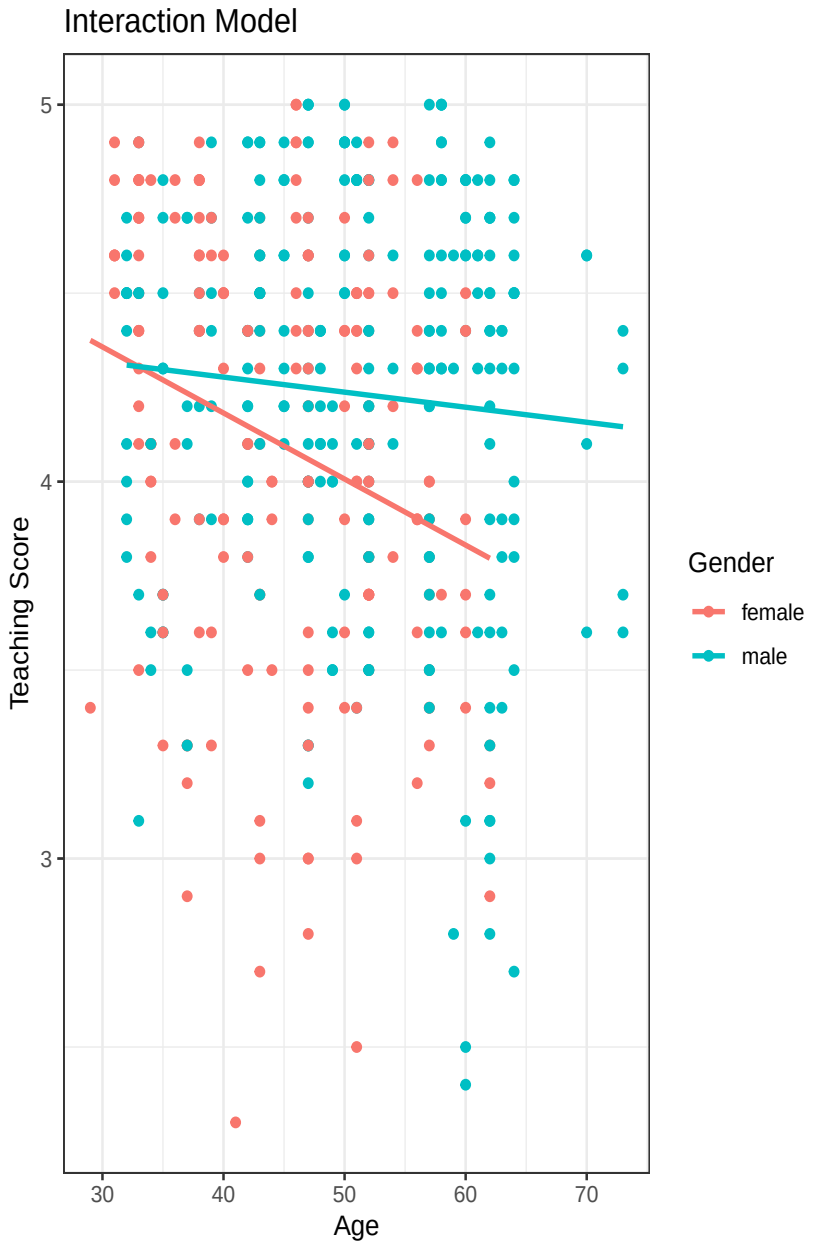
- Since for female instructors $1_{\text{is male}}(x) = 0$.

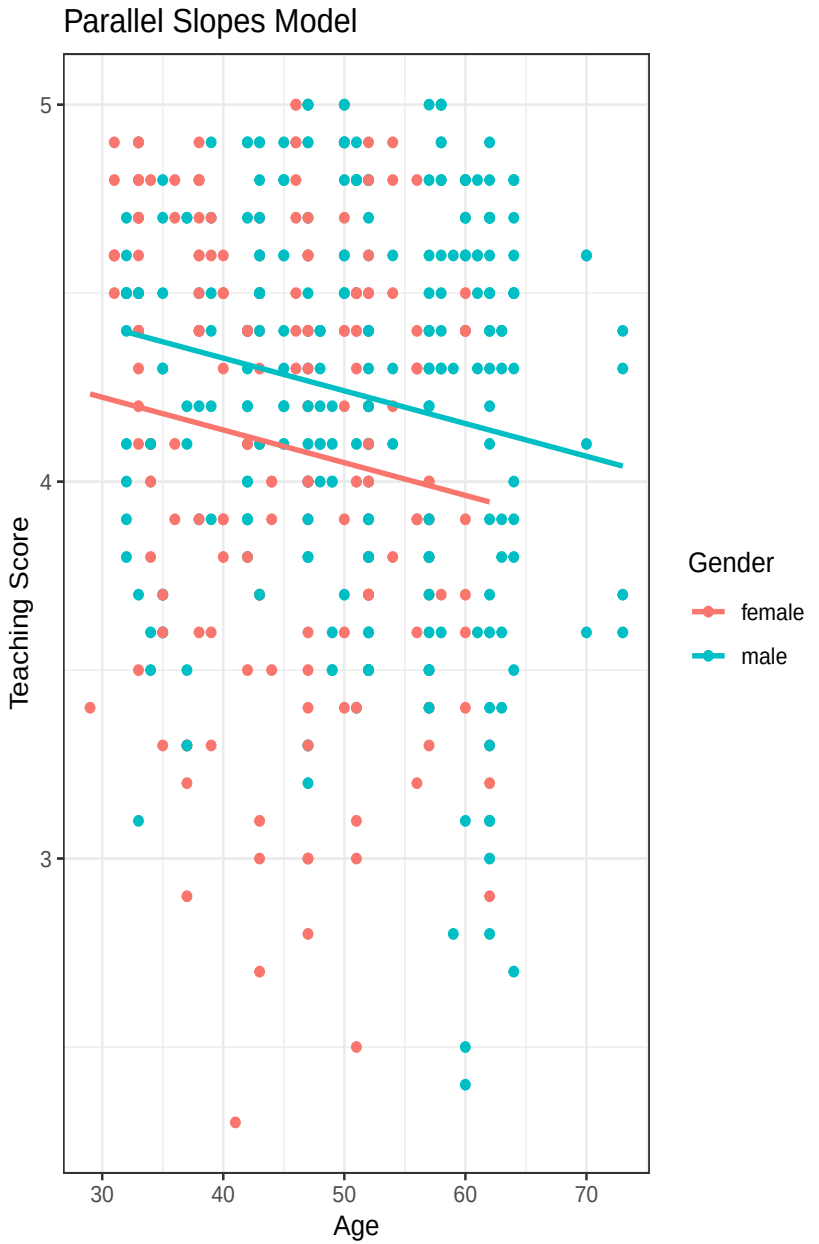
$$\begin{aligned} \hat{y} = \widehat{\text{score}} &= 4.484 - 0.009 \cdot \text{age} + 0.191 \cdot 0 \\ &= 4.484 - 0.009 \cdot \text{age} \end{aligned}$$

- For male instructors $1_{\text{is male}}(x) = 1$.

$$\begin{aligned} \hat{y} = \widehat{\text{score}} &= 4.484 - 0.009 \cdot \text{age} + 0.191 \cdot 1 \\ &= (4.484 + 0.191) - (0.009) \cdot \text{age} \\ &= 4.675 - 0.009 \cdot \text{age} \end{aligned}$$

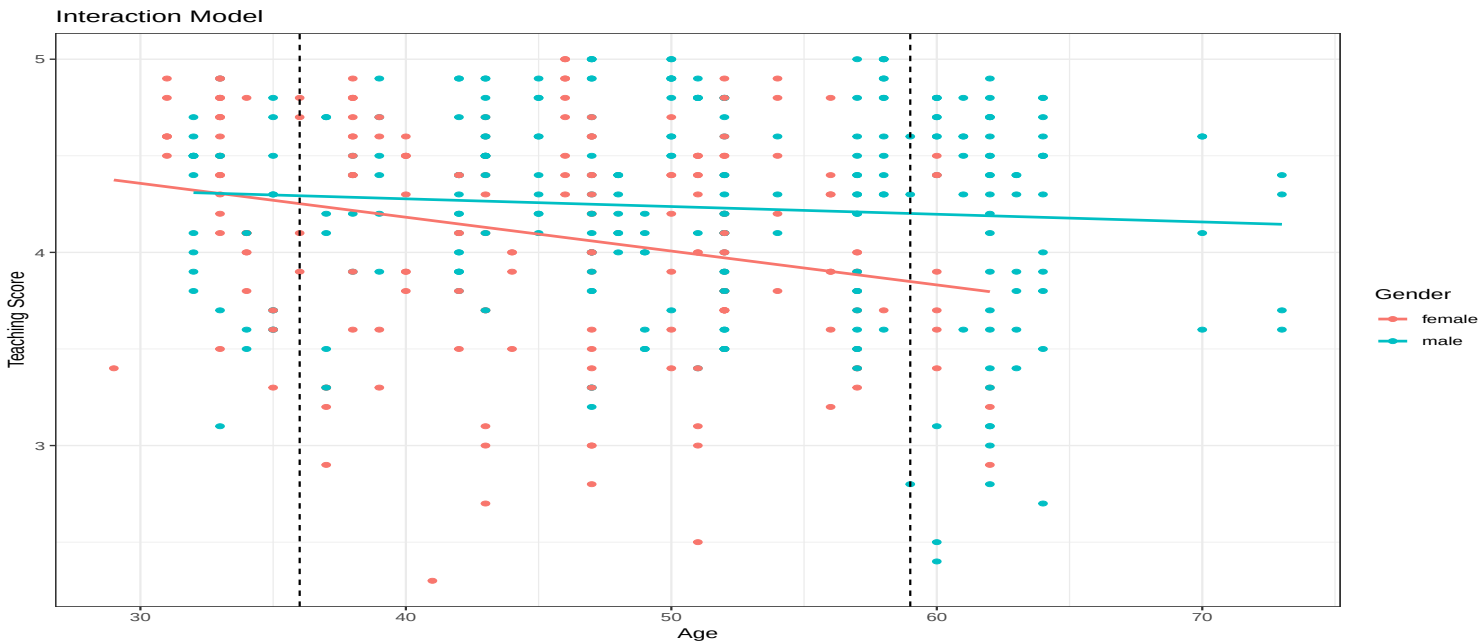
Interaction Model and Parallel Slopes Model





Observed/fitted values and residuals

- We'll compute the observed values, fitted values, and residuals for the interaction model which we saved in `score_model_interaction`.
  - Say, you have an instructor who identifies as female and is 36 years old. What fitted value $\hat{y} = \widehat{\text{score}}$ would our model yield?
  - Say, you have another instructor who identifies as male and is 59 years old. What would their fitted value $\hat{y}$ be?
- See if you can answer this question visually.



Observed/fitted values and residuals

- For female instructors, we have.

$$\begin{aligned}\hat{y} = \widehat{\text{score}} &= 4.883 - 0.018 \cdot \text{age} \\ &= 4.883 - 0.018 \cdot 36 = 4.24\end{aligned}$$

- For male instructors.

$$\begin{aligned}\hat{y} = \widehat{\text{score}} &= 4.437 - 0.004 \cdot \text{age} \\ &= 4.437 - 0.004 \cdot 59 = 4.20\end{aligned}$$

Observed/fitted values and residuals

Note: It is better to let R compute the values and round at the end.

```
predict(score_model_interaction,
        newdata = data.frame(age = 36, gender = "female"))
```

```
       1
4.252148
```

```
predict(score_model_interaction,
        newdata = data.frame(age = 59, gender = "male"))
```

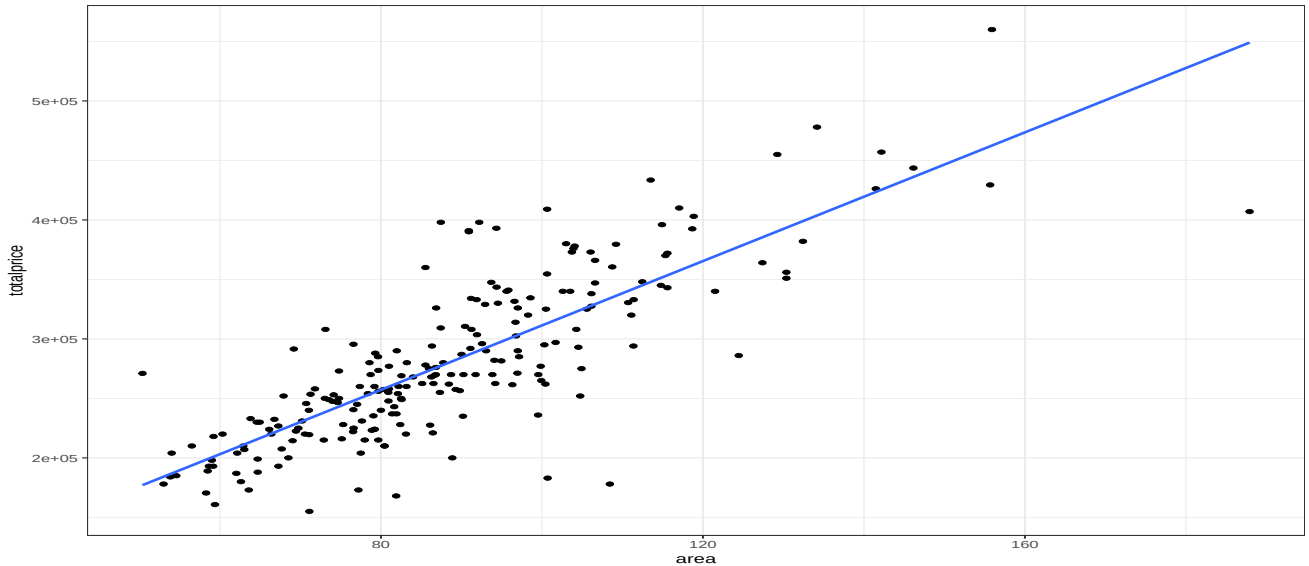
```
       1
4.201373
```

Example: One numerical and one categorical explanatory variable

Suppose a realtor wants to model the appraised price of an apartment as a function of the predictors living area (in m^2) and the presence or absence of elevators. Consider the data frame 'VIT2005', which contains data about apartments in Vitoria, Spain, including **totalprice**, **area**, and **elevator**, which are the appraised apartment value in Euros, living space in square meters, and the absence or presence of at least one elevator in the building, respectively. The realtor first wants to know if there is any relationship between appraised price (Y) and living area (x_1). Next, the realtor wants to know how adding a dummy variable for whether or not an elevator is present changes the relationship: Are the lines the same? Are the slopes the same? Are the intercepts the same?

Solution (is there a realationship between totalprice and area?):

```
library(PASWR2)
VIT2005 <- VIT2005 %>%
  mutate(elevator = factor(elevator, labels = c("No", "Yes")))
ggplot(data = VIT2005, aes(x = area, y = totalprice)) +
  geom_point() +
  theme_bw() +
  geom_smooth(method = "lm", se = FALSE)
```



Solution (is there a realationship between totalprice and area?):

```
mod_simple <- lm(totalprice ~ area, data = VIT2005)
summary(mod_simple)
```

```
Call:
lm(formula = totalprice ~ area, data = VIT2005)

Residuals:
    Min      1Q  Median      3Q     Max
-156126  -21564   -2155   19493  120674

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  40822.4    12170.1   3.354  0.00094 ***
area          2704.8      133.6  20.243  < 2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 40810 on 216 degrees of freedom
Multiple R-squared:  0.6548,    Adjusted R-squared:  0.6532
F-statistic: 409.8 on 1 and 216 DF,  p-value: < 2.2e-16
```

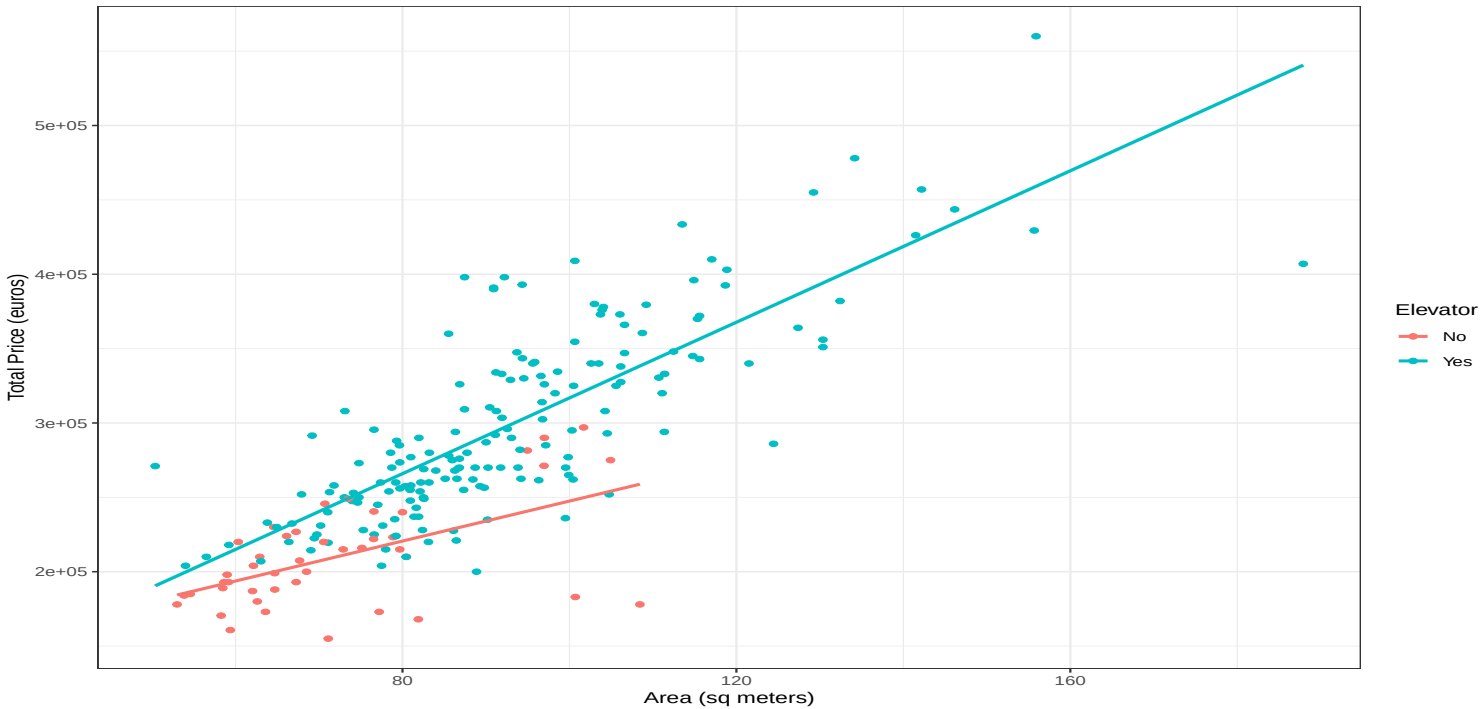
Solution (is there a realationship between totalprice and area?):

```
knitr::kable(get_regression_table(mod_simple))
```

term	estimate	std_error	statistic	p_value	lower_ci	upper_ci
intercept	40822.416	12170.092	3.354	0.001	16835.075	64809.757
area	2704.751	133.616	20.243	0.000	2441.393	2968.109

Solution (does adding a dummy variable (elevator) change the relationship?):

```
ggplot(VIT2005, aes(x = area, y = totalprice, color = elevator)) +
  geom_point() +
  labs(x = "Area (sq meters)", y = "Total Price (euros)",
       color = "Elevator") +
  geom_smooth(method = "lm", se = FALSE) +
  theme_bw()
```



Solution (does adding a dummy variable (elevator) change the relationship?):

```
mod_int <- lm(totalprice ~ area + elevator + area:elevator, data = VIT2005)
summary(mod_int)
```

```
Call:
lm(formula = totalprice ~ area + elevator + area:elevator, data = VIT2005)

Residuals:
    Min      1Q  Median      3Q     Max
-133610  -22216   -2423   20276  113159

Coefficients:
                 Estimate Std. Error t value Pr(>|t|)
(Intercept)      113114.0    28985.0   3.902 0.000128 ***
area               1343.7      392.2   3.426 0.000735 ***
elevatorYes      -50871.7    31990.6  -1.590 0.113264
area:elevatorYes   1202.0      417.4   2.880 0.004380 **
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 37610 on 214 degrees of freedom
Multiple R-squared:  0.7096,    Adjusted R-squared:  0.7055
F-statistic: 174.3 on 3 and 214 DF,  p-value: < 2.2e-16
```

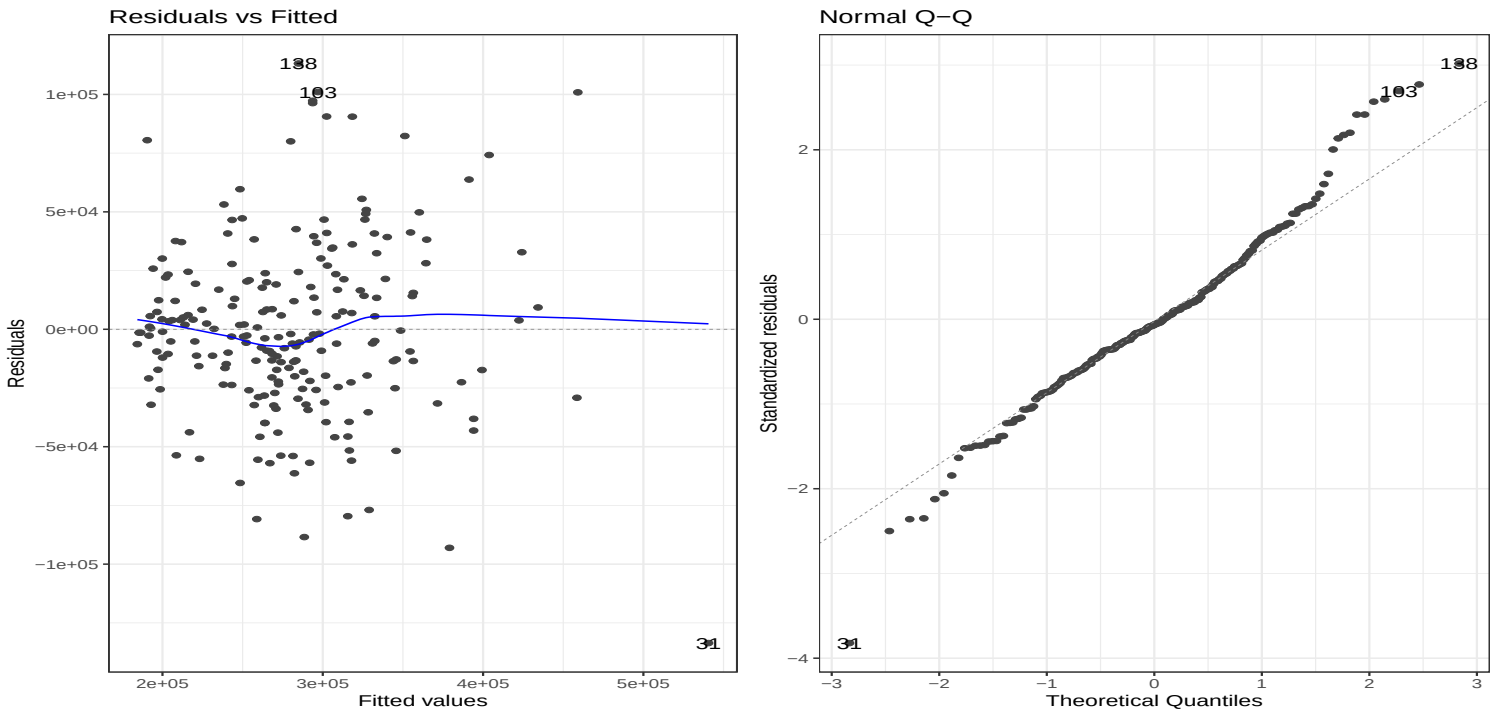
Solution (does adding a dummy variable (elevator) change the relationship?):

```
mod_ps <- lm(totalprice ~ area + elevator, data = VIT2005)
summary(mod_ps)
```

```
Call:
lm(formula = totalprice ~ area + elevator, data = VIT2005)

Residuals:
    Min      1Q  Median      3Q     Max
-120265  -20224   -2567   18281  112406

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  36173.6    11434.8   3.163  0.00178 **
area          2405.4      136.3  17.652  < 2e-16 ***
elevatorYes  39091.1     7022.8   5.566 7.71e-08 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 38240 on 215 degrees of freedom
Multiple R-squared:  0.6983,    Adjusted R-squared:  0.6955
F-statistic: 248.8 on 2 and 215 DF,  p-value: < 2.2e-16
```

Diagnostic Plots

```
library(ggfortify)
autoplot(mod_int, ncol = 2, nrow = 1, which = 1:2) +
  theme_bw()
```



Section 4

Two numerical explanatory variables

Two numerical explanatory variables

- Let's switch gears and consider multiple regression models where instead of one numerical and one categorical explanatory variable, we have two numerical explanatory variables.
- The `Credit` dataset we will use is from the `ISLR` package.
  - The outcome variable of interest is the credit card debt of 400 individuals.
  - Other variables like income, credit limit, credit rating, and age are included as well.
- Note that the `Credit` data is not based on real individuals' financial information, but rather is a simulated dataset used for educational purposes.

Exploratory data analysis

Use `select()` to create a subset of the variables we'll consider in this chapter.

```
library(ISLR)
credit_ch6 <- Credit %>%
  as_tibble() %>%
  select(ID, debt = Balance, credit_limit = Limit,
         income = Income, credit_rating = Rating, age = Age)
glimpse(credit_ch6)
```

```
Rows: 400
Columns: 6
$ ID            <int> 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14
$ debt          <int> 333, 903, 580, 964, 331, 1151, 203, 872, 279,
$ credit_limit  <int> 3606, 6645, 7075, 9504, 4897, 8047, 3388, 711
$ income        <dbl> 14.891, 106.025, 104.593, 148.924, 55.882, 80
$ credit_rating <int> 283, 483, 514, 681, 357, 569, 259, 512, 266, 4
$ age           <int> 34, 82, 71, 36, 68, 77, 37, 87, 66, 41, 30, 64
```

Exploratory data analysis

```
credit_ch6 %>%
  sample_n(size = 5)
```

```
# A tibble: 5 x 6
     ID  debt credit_limit income credit_rating   age
  <int> <int>        <int>  <dbl>         <int> <int>
1   101   298         3736   21.2           256    41
2   236   191         2923   10.5           232    25
3    83   503         4433   23.7           344    63
4   357   962         6090   34.5           442    36
5   115   271         3326   16.5           268    41
```

```
credit_ch6 %>%
  select(debt, credit_limit, income) %>%
  skim()
```

Exploratory data analysis

```
credit_ch6 %>%
  select(debt, credit_limit, income) %>%
  summary()
```

```
      debt          credit_limit       income      
 Min.   :   0.00   Min.   :  855   Min.   : 10.35  
 1st Qu.:  68.75   1st Qu.: 3088   1st Qu.: 21.01  
 Median : 459.50   Median : 4622   Median : 33.12  
 Mean   : 520.01   Mean   : 4736   Mean   : 45.22  
 3rd Qu.: 863.00   3rd Qu.: 5873   3rd Qu.: 57.47  
 Max.   :1999.00   Max.   :13913   Max.   :186.63  
```

Exploratory data analysis

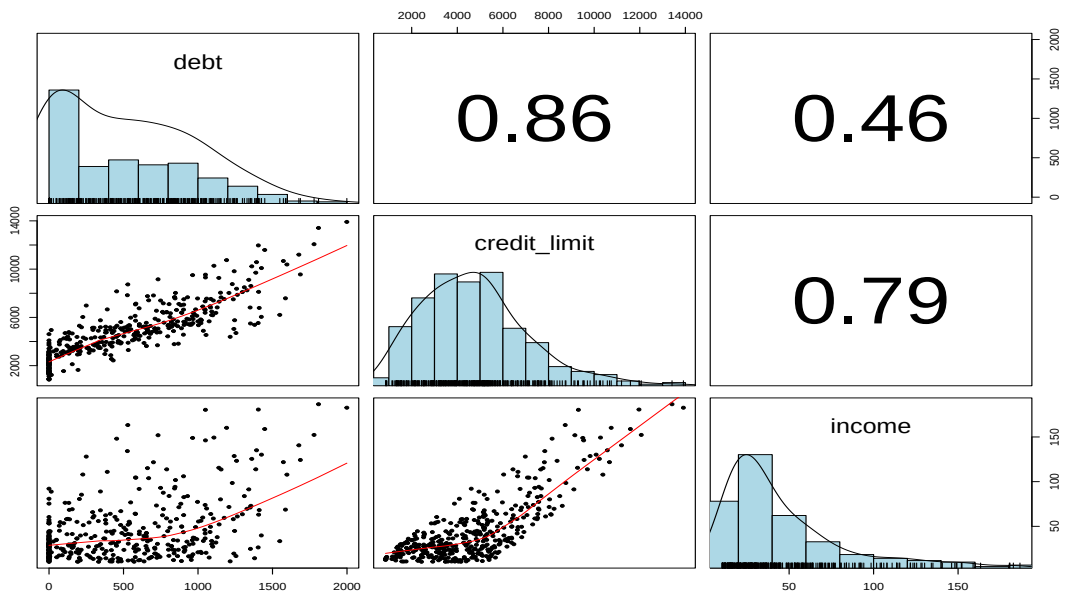
We can compute the correlation coefficient between the different possible pairs of these variables.

```
credit_ch6 %>%
  select(debt, credit_limit, income) %>%
  cor()
```

```
                  debt credit_limit    income
debt         1.0000000    0.8616973 0.4636565
credit_limit 0.8616973    1.0000000 0.7920883
income       0.4636565    0.7920883 1.0000000
```

Exploratory data analysis

```
library(psych)
pairs.panels(credit_ch6[, 2:4],  # select debt (2), credit_limit (3
             # income (4)
             method = "pearson", # correlation method
             hist.col = "lightblue",
             density = TRUE,  # show density plots
             ellipses = FALSE # show correlation ellipses
             )
```



Exploratory data analysis: Collinearity

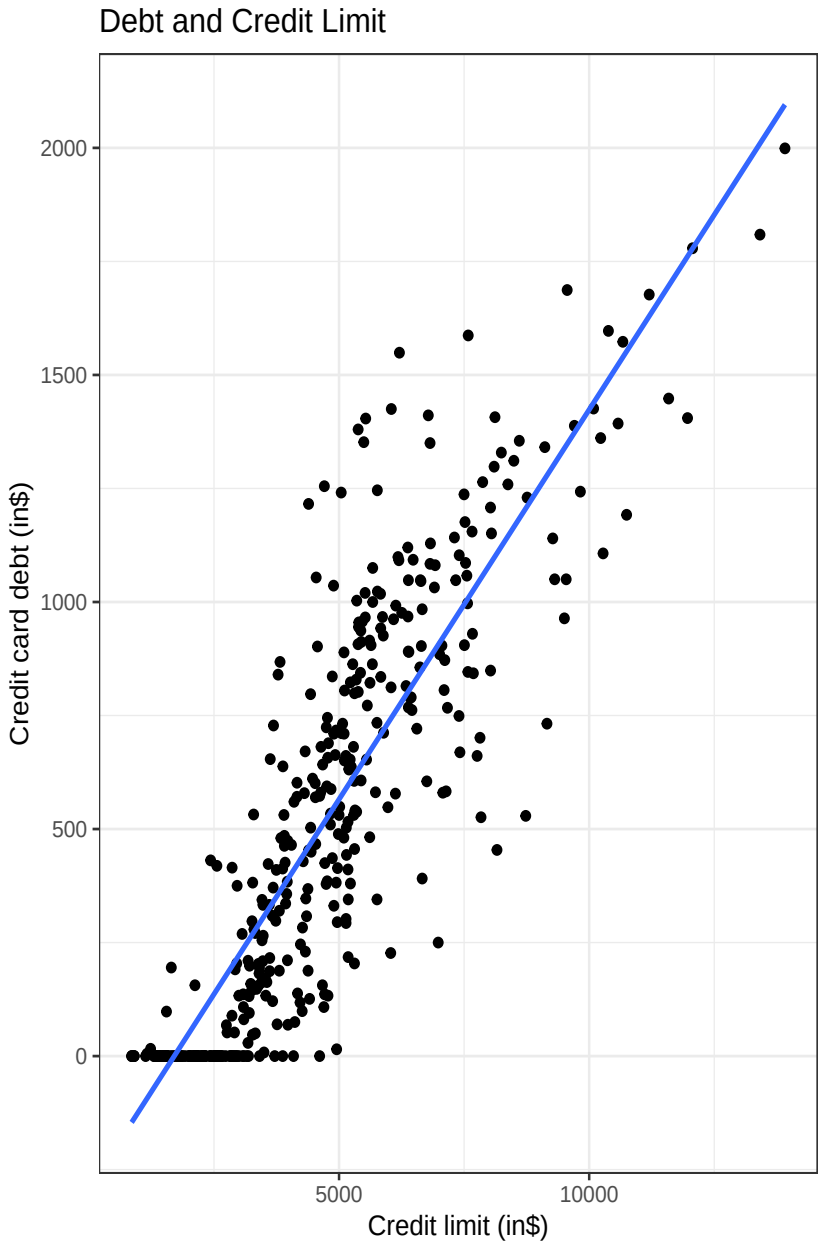
- We say there is a high degree of collinearity between the `credit_limit` and `income` explanatory variables.
- Collinearity (or multicollinearity) is a phenomenon where one explanatory variable in a multiple regression model is **highly correlated with another**.
- So in our case since `credit_limit` and `income` are highly correlated.
  - If we knew a persons' `credit_limit`, we could make a pretty good guess about their `income`.
  - Thus, these two variables provide somewhat redundant information.
- We will leave discussion on how to work with collinear explanatory variables for another course.

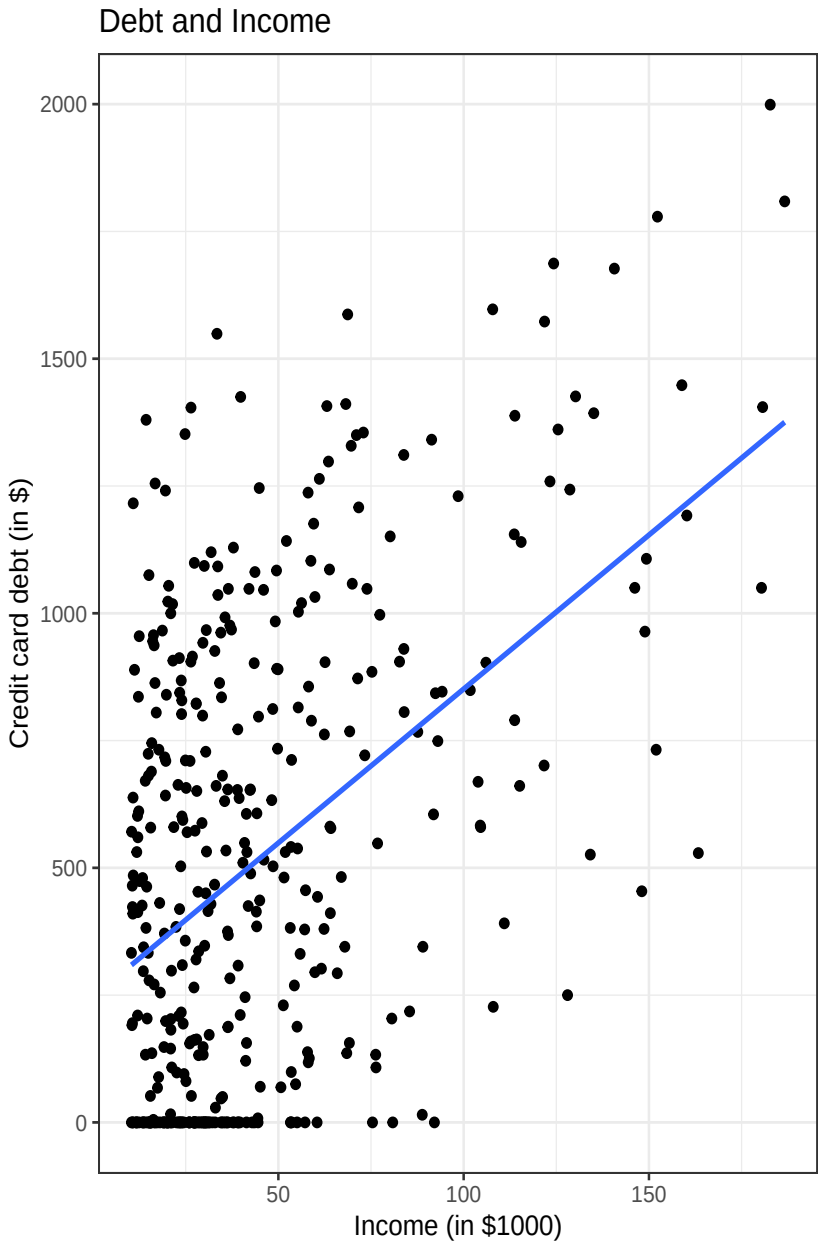
Exploratory data analysis: visualization

Let's visualize the relationship of the outcome variable with each of the two explanatory variables in two separate plots

```
ggplot(data = credit_ch6, aes(x = credit_limit, y = debt)) +
  geom_point() +
  labs(x= "Credit limit (in$)", y = "Credit card debt (in$)",
       title = "Debt and Credit Limit") +
  geom_smooth(method = "lm", se = FALSE) +
  theme_bw() -> p1
ggplot(data = credit_ch6, aes(x = income, y = debt)) +
  geom_point() +
  labs(x = "Income (in $1000)", y = "Credit card debt (in $)",
       title = "Debt and Income") +
  geom_smooth(method = "lm", se = FALSE) +
  theme_bw() -> p2
library(patchwork)
p1 + p2
```

Exploratory data analysis: visualization



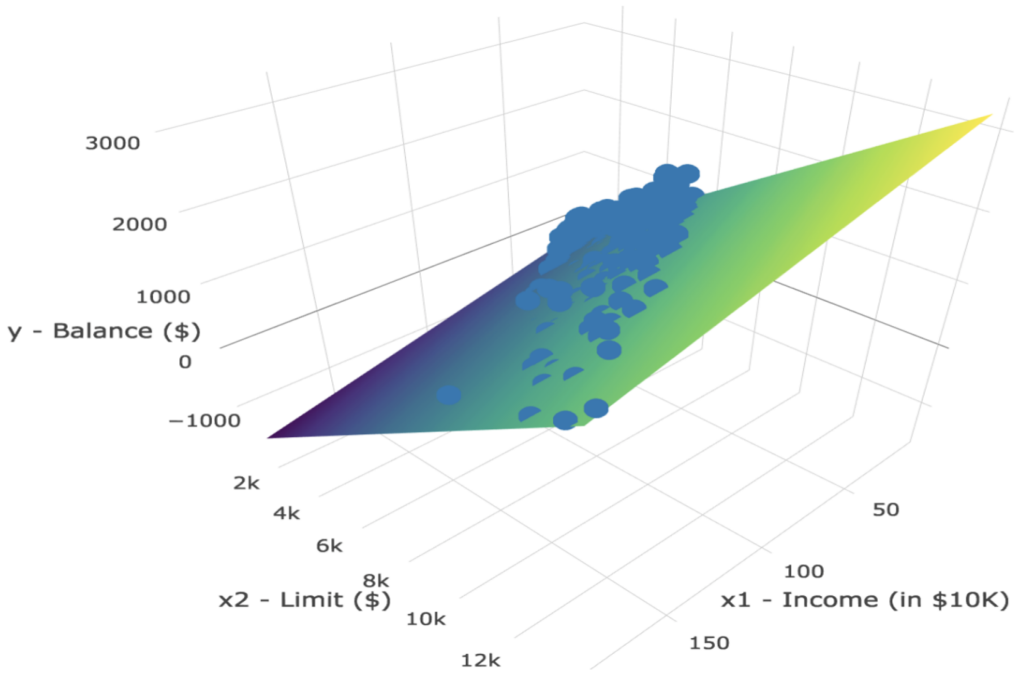


Exploratory data analysis: visualization

To visualize the joint relationship of all three variables simultaneously, we need a 3-dimensional (3D) scatterplot. The following code will create a 3-dimensional scatterplot.

```
library(plotly)
p <- plot_ly(data = credit_ch6, z = ~debt, x = ~credit_limit,
             y = ~income) %>% add_markers()
mod <- lm(debt ~ credit_limit + income, data = credit_ch6)
x <- seq(min(credit_ch6$credit_limit),
         max(credit_ch6$credit_limit), length = 70)
y <- seq(min(credit_ch6$income),
         max(credit_ch6$income), length = 70)
plane <- outer(x, y, function(a, b){coef(mod)[1] +
               coef(mod)[2]*a + coef(mod)[3]*b})
# draw the plane
p %>%
  add_surface(x = ~x, y = ~y, z = ~plane)
```

Exploratory data analysis: visualization



The regression plane is the "best-fitting" plane that similarly minimizes the sum of squared residuals.

Regression plane

```
# Fit regression model:
debt_model <- lm(debt ~ credit_limit + income,
                 data = credit_ch6)
# Get regression table:
knitr::kable(get_regression_table(debt_model))
```

term	estimate	std_error	statistic	p_value	lower_ci	upper_ci
intercept	-385.179	19.465	-19.789	0	-423.446	-346.912
credit_limit	0.264	0.006	44.955	0	0.253	0.276
income	-7.663	0.385	-19.901	0	-8.420	-6.906

Regression plane: Interpretation

- First, the intercept value is $-\$385.179$.
  - This intercept represents the credit card debt for an individual who has `credit_limit` of $\$0$ and income of $\$0$.
  - In our data, the intercept has no practical interpretation since no individuals had both `credit_limit` an `income` values of $\$0$.
  - Rather, the intercept is used to situate the regression plane in 3D space.
- Second, the `credit_limit` value is $\$0.264$.
  - Taking into account all the other explanatory variables in our model, for every increase of one dollar in `credit_limit`, there is an associated increase of on average $\$0.26$ in credit card debt.
  - Just as we earlier, we are cautious not to imply causality. We do this merely stating there was an associated increase.
- Third, income $= -\$7.66$.
  - Taking into account all other explanatory variables in our model, for every increase of one unit of income ($\$1000$ in actual income), there is an associated decrease of, on average, $\$7.66$ in credit card debt.

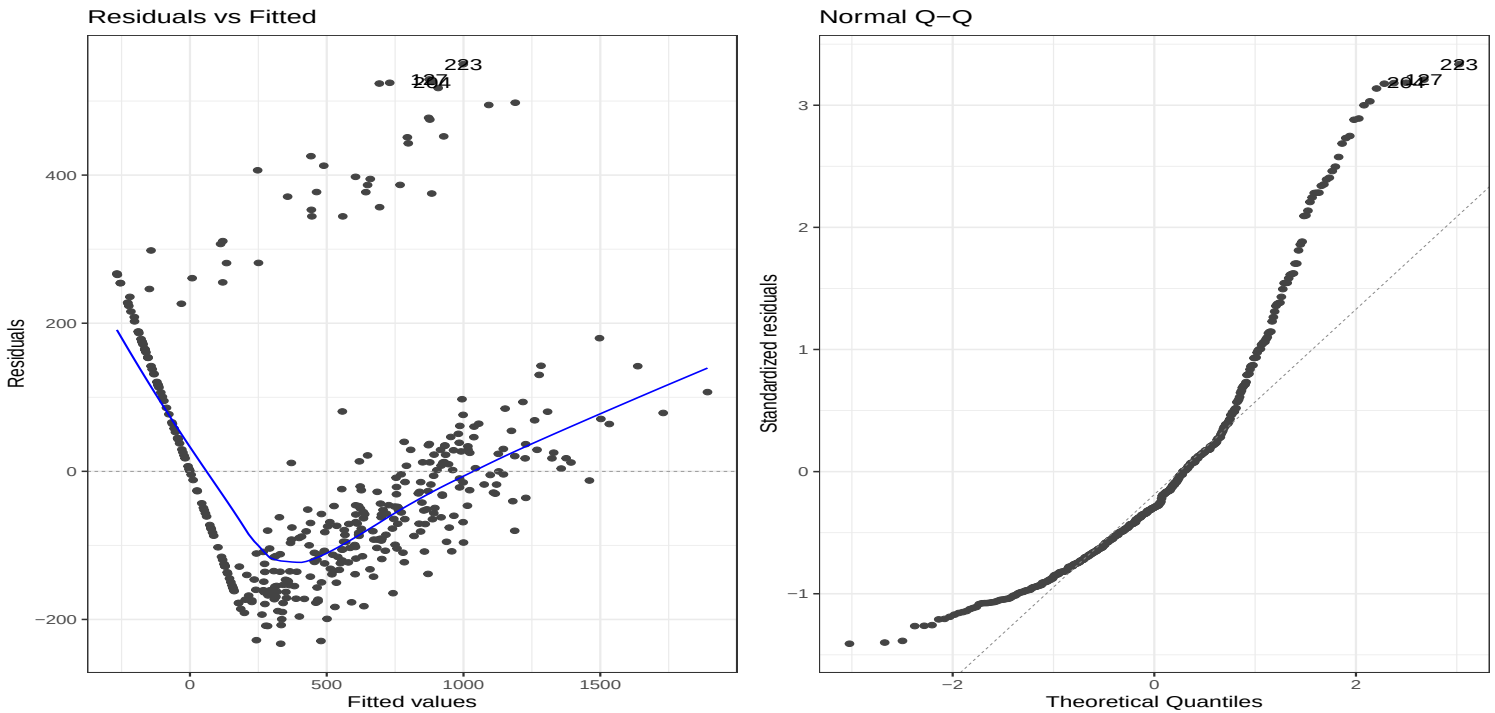
Regression plane: Prediction

Putting these results together, the equation of the regression plane that gives us fitted values $\hat{y} = \widehat{\text{debt}}$ is:

$$\begin{aligned}
\hat{y} &= b_0 + b_1 \cdot x_1 + b_2 \cdot x_2 \\
\widehat{\text{debt}} &= b_0 + b_{limit} \cdot \text{limit} + b_{income} \cdot \text{income} \\
&= -385.179 + 0.263 \cdot \text{limit} - 7.663 \cdot \text{income}
\end{aligned}$$

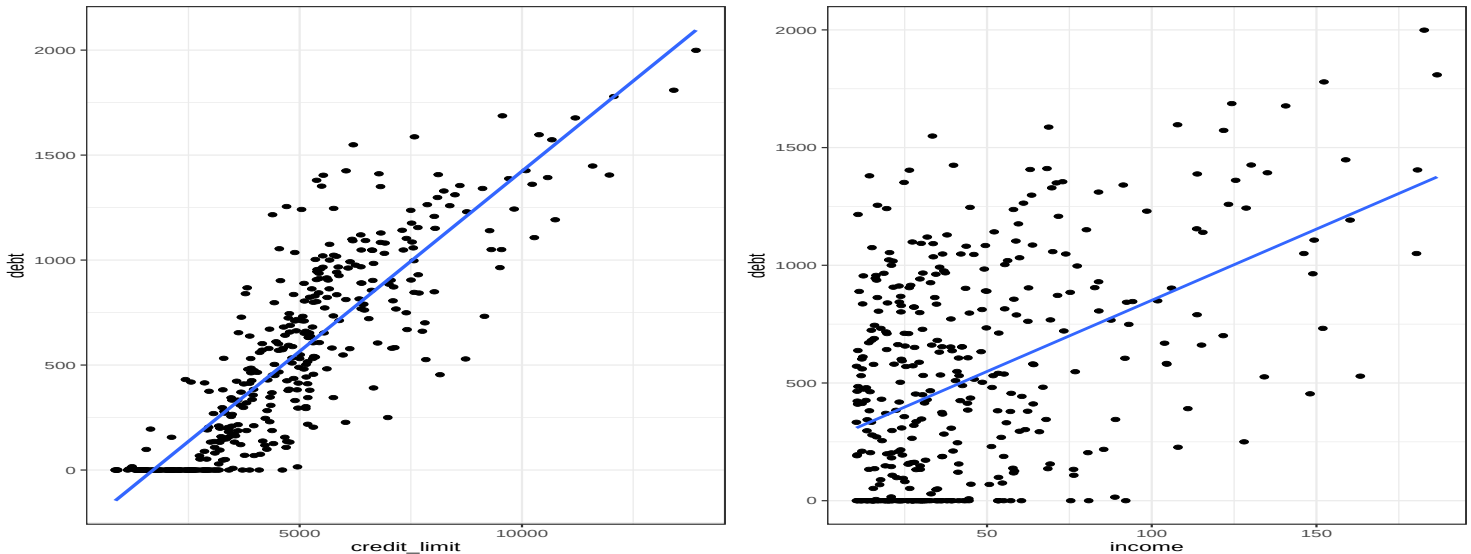
Diagnostic Plots

```
autoplot(debt_model, ncol = 2, nrow = 1, which = 1:2) +
  theme_bw()
```



Simpson's Paradox

```
library(ISLR)
credit_paradox <- Credit %>%
  select(ID, debt = Balance, credit_limit = Limit,
         credit_rating = Rating, income = Income, age = Age)
ggplot(data = credit_paradox, aes(x = credit_limit, y = debt)) +
  geom_point() +
  geom_smooth(method = "lm", se = FALSE) +
  theme_bw() -> p1
ggplot(data = credit_paradox, aes(x = income, y = debt)) +
  geom_point() +
  geom_smooth(method = "lm", se = FALSE) +
  theme_bw() -> p2
library(patchwork)
p1 + p2
```



Simpson's Paradox

```
mod <- lm(debt ~ credit_limit + income, data = credit_paradox)
summary(mod)$coef
```

```
                 Estimate   Std. Error   t value      Pr(>|t|)
(Intercept)  -385.1792604 19.464801525 -19.78850  3.878764e-61
credit_limit    0.2643216  0.005879729  44.95471 7.717386e-158
income         -7.6633230  0.385072058 -19.90101  1.260933e-61
```

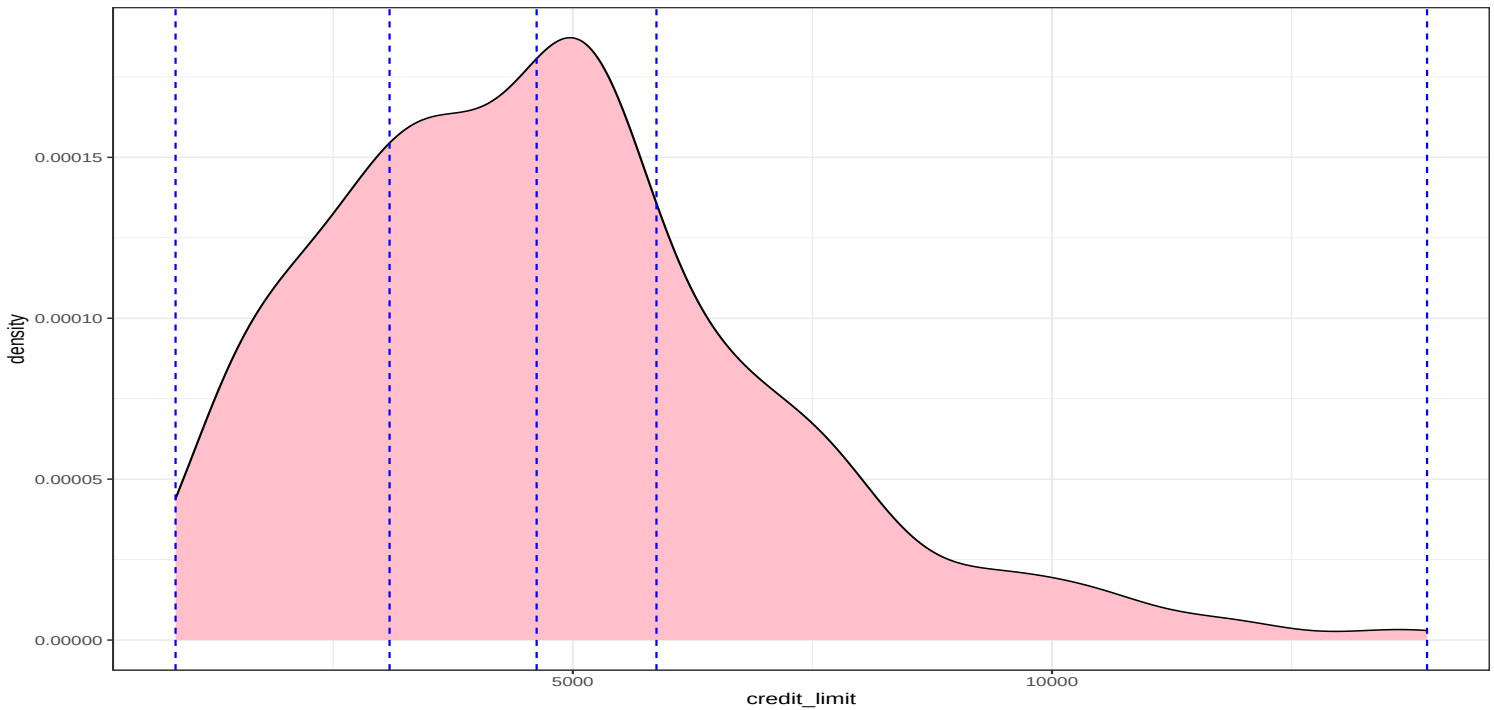
Simpson's Paradox

```
qs <- quantile(credit_paradox$credit_limit, probs = seq(0, 1, .25))
credit_paradox <- credit_paradox %>%
  mutate(credit_cats = cut(credit_limit, breaks = qs,
                           include.lowest = TRUE))
knitr::kable(head(credit_paradox))
```

ID	debt	credit_limit	credit_rating	income	age	credit_cats
1	333	3606	283	14.891	34	(3.09e+03,4.62e+03]
2	903	6645	483	106.025	82	(5.87e+03,1.39e+04]
3	580	7075	514	104.593	71	(5.87e+03,1.39e+04]
4	964	9504	681	148.924	36	(5.87e+03,1.39e+04]
5	331	4897	357	55.882	68	(4.62e+03,5.87e+03]
6	1151	8047	569	80.180	77	(5.87e+03,1.39e+04]

Simpson's Paradox

```
ggplot(data = credit_paradox, aes(x = credit_limit)) +
  geom_density(fill = "pink", color = "black") +
  geom_vline(xintercept = qs, color = "blue",
             linetype = "dashed") +
  theme_bw()
```



Simpson's Paradox

```
credit_paradox %>%
  group_by(credit_cats) %>%
  summarize(n())
```

```
# A tibble: 4 x 2
  credit_cats          `n()`
  <fct>                <int>
1 [855,3.09e+03]         100
2 (3.09e+03,4.62e+03]    100
3 (4.62e+03,5.87e+03]    100
4 (5.87e+03,1.39e+04]    100
```

Simpson's Paradox

```
p1 <- ggplot(data = credit_paradox, aes(x = income, y = debt))
  geom_point() +
  geom_smooth(method = "lm", se = FALSE) +
  theme_bw() +
  labs(y = "Credit card debt (in $)",
       x = "Income (in $1000)")
p2 <- ggplot(data = credit_paradox, aes(x = income, y = debt,
                                        color = credit_cats))
  geom_point() +
  geom_smooth(method = "lm", se = FALSE) +
  theme_bw() +
  labs(y = "Credit card debt (in $)",
       x = "Income (in $1000)",
       color = "Credit limit bracket")
p1 + p2
```

Simpson's Paradox