EDUC 784

Peter Halpin

2024-01-03

Table of contents

# Preface

These are the course notes for EDUC 784. Readings are assigned *before class.* Sections denoted with an asterisk (\*) are optional.

The notes contain **questions that are written in bold font**. The questions are also collected in a section called “Workbook” that appears towards the end of each chapter. During class time, we will discuss the Workbook questions, your answers, any additional question you have, etc. It is really important for you to do the readings, and write down your responses to the questions, before class. You won’t get much out of the lessons if you haven’t done this preparation.

Some chapters contain a section called “Exercises” that collects all of the R code from that chapter into a single overall workflow. **You don’t need to do the Exercises before class**, but you can if you want to. If a chapter doesn’t have an Exercises section, that means we will be working on an assignment together instead.

Image credit: Daniela Rodriguez-Mincey, Spring 2023

# 1. Review

This chapter is an exception to the overall format described in the Preface. It reviews some foundational material from EDUC 710 (Stat 1) that is useful for this course. There will be time to ask questions about the review material in the first class and second class, but there will not be time to review everything. So, if this review feels too short, you may also want to review your notes from EDUC 710.

Please read up to the Exercises in [Section 1.10](#sec-exercizes-1) before the first class. We will address any questions about this material in the first class and then begin the Exercises together.

## 1.1 Summation notation

Summation notation uses the symbol to stand-in for summation. For example, instead of writing

to represent the sum of the values of the variable in a sample of size , we can instead write:

The symbol means “add.” The symbol is called “Sigma” – it’s the capital Greek letter corresponding to the Latin letter “S”. The value is called the index, and is the starting value of the index and is the end value of the index. You can choose whatever start and end values you want to sum over. For example, if we just want to add the second and third values of , we write

When the start and end values are clear from context, we can use a shorthand notation that omits them. In the following, it is implicit that the sum is over all available values of (i.e., from to ):

## 1.2 Rules of summation

There are rules for manipulating summation notation that are useful for deriving results in statistics. You don’t need to do mathematical proofs or derivations in this class, but you will occasionally see some derivations in these notes (mainly in the optional sections).

Here are the rules:

**Rule 1: Sum of a constant (multiplication)**. Adding a constant to itself is equivalent to multiplication.

**Rule 2: Distributive property**. Multiplying each term of a sum by a constant is the same as multiplying the sum by a constant.

**Rule 3: Associative property**. It doesn’t matter what order we do addition in.

## 1.3 Sample statistics

Summation notation is useful for writing the formulas of statistics. The main statistics we use in the class are the mean, standard deviation, variance, covariance, and correlation. These are the building blocks for regression. Their symbols and formulas are presented below (using the shorthand summation notation). If you don’t remember their interpretation, you will need to go back to your Stat 1 notes.

* The mean
* The variance can be written as or sometimes using the symbol
* The standard deviation can be written or using the letter
* The covariance is a generalization of the variance to two variables, it describes how they co-vary:
* The correlation is the covariance divided by the product of the standard deviations of the variables. It takes on values between -1 and 1 and describes the strength and direction of the linear relationship between two variables.

For numerical examples see [Section 1.10.8](#sec-computing-stats-1).

## 1.4 Properties of sample statistics

The following are some useful properties of the sample statistics reviewed above. The properties tell us what happens to means, variances, covariances, and correlations when a variable is linearly transformed. We often linearly transform data (e.g., to compute percentages, proportions, z-scores, and in linear regression), so these properties turn out to be quite handy.

You can derive the properties using the rules of summation. For each property, the beginning of the derivation is shown. You should know the properties but completing the derivations is optional.

**Sum of deviations from the mean**. If we subtract the mean from each data point, we have a deviation (or deviation score): . Deviation scores sum to zero: .

* Derivation

**Mean of a linear transformation**. If with known constants and , then

* Derivation:

**Variance of a linear transformation**. If with known constants and , then

* Derivation

**Mean and variance of a z-score**. The z-score (or standardized score) is defined as . Standardized scores are useful because and .

* Derivation: use the rules for linear transformation with and .

**Covariance of linear transformations**. If and with known constants , , , , then

* Derivation

**Correlation of linear transformations**. If and with known constants , , , , then – i.e., the correlation is not affected by linear transformations.

* Derivation: use the rules for variances and covariances of linear transformation and the formula for correlation.

## 1.5 Bias and precision

In this section we consider two more important properties of sample statistics. These properties are defined in terms of *sampling distributions*. Recall that a sampling distribution arises from the following thought experiment:

1. Take a random sample of size from a population of interest.
2. Compute a statistic using the sample data. It can be any statistic, but let’s say the mean, , for concreteness.
3. Write down the value of the mean, and then return the sample to the population.

After doing these 3 steps many times, you will have many values the sample mean,

The distribution of these sample means is called a sampling distribution (i.e., the sampling distribution of the mean). A sampling distribution is just like any other distribution – so it has its own mean, and its own variance, etc. These quantities, when computed for a sampling distribution, have special names.

* **The expected value** of the mean, denoted , is the mean of the sampling distribution of the mean. That is a mouthful! That is why we say the “expected value” or “expectation” of a statistic rather than the mean of a statistic. It’s called the expected value because it’s the average value over many samples.
* **The standard error** of the mean, denoted , is the standard deviation of the sampling distribution of the mean. It describes the sample-to-sample variation of the mean around its expected value.

Now for the two additional properties of sample statistics:

* **Bias**: If the expected value of a statistic is equal to a population parameter, we say that the statistic is an unbiased estimate of that parameter. For example, the expected value of the sample mean is equal to the population mean (in symbols: , so we say that the sample mean is an unbiased estimate of the population mean.
* **Precision**: The inverse of the squared standard error (i.e., ) is called the precision of a statistic. So, the less a statistic varies from sample to sample, the more precise it is. That should hopefully make intuitive sense. The main thing to know about precision is that it is usually increasing in the sample size – i.e., we get more precise estimates by using larger samples. Again, this should feel intuitive.

Below is a figure that is often used to illustrate the ideas of bias and precision. The middle of the concentric circles represent the target parameter (like a bull’s eye) and the dots represent the sampling distribution of a statistic. You should be able to describe each panel in terms of the bias and precision of the statistic.

|  |
| --- |
| Figure 1.1: Bias and Precision |

## 1.6 t-tests

The -test is used to make an inference about the value of an unknown population parameter. The test compares the value of an unbiased estimate of the parameter to a hypothesized value of the parameter. It is assumed that the sampling distribution of the estimate is a normal distribution, so the -test applies to statistics like means and regression coefficients.

The conceptual formula for a -test is

When we conduct a -test, the basic rationale is as follows: If the “true” population parameter is equal to the hypothesized value, then the estimate should be close to the hypothesized value, and so should be close to zero.

In order to determine what values of are “close to zero”, we refer to its sampling distribution, which is called the -distribution. The -distribution tells what values of are typical if the hypothesis is true. Some examples of the -distribution are shown in the figure below. The x-axis denotes values of the statistic shown above, and is parameter called the “degrees of freedom” (more on this soon).

|  |
| --- |
| Figure 1.2: t-distribution (source: https://en.wikipedia.org/wiki/Student%27s\_t-distribution) |

You can see that the -distribution looks like a normal distribution centered a zero. So, when the hypothesis is true, the expected value of is zero. Informally, we could say that, if the hypothesis is true, values of greater than are pretty unlikely, and values greater than are very unlikely.

More formally, we can make an inference about whether the population parameter is equal to the hypothesized value by comparing the value of computed in our sample, denoted as , to a “critical value”, denoted as . The critical value is chosen so that the “significance level”, defined as , is sufficiently small. The significance level is chosen by the researcher. Often it is set to or

There are two equivalent ways of “formally” conducting a -test.

1. Compare the observed value of to the critical value. Specifically: if , reject the hypothesis.
2. Compare the significance level chosen by the researcher, , to the “p-value” of the test, computed as . Specifically: if , reject the hypothesis.

Informally, both of these just mean that the absolute value of should be pretty big (i.e., greater than ) before we reject the hypothesis.

A couple more things before moving on.

First, the hypothesized value of the population parameter is often zero, in which case it is called a null hypothesis. The null hypothesis usually translates into a research hypothesis of “no effect” or “no relationship.” So, if we reject the null hypothesis, we conclude that there was an effect.

Second, the t-distribution has a single parameter called its “degrees of freedom”, which is denoted as in [Figure 1.2](#fig-tdist). The degrees of freedom are an increasing function of the sample size, with larger samples leading to more degrees of freedom. When the degrees of freedom approach , the -distribution approaches a normal distribution. This means that the difference between a -test and a -test is pretty minor in large samples (say ).

## 1.7 Confidence intervals

A confidence interval uses the same equation as a -test, except we solve for the population parameter rather than the value of . Whereas a -test lets us make a guess about specific value of the parameter of interest (i.e., the null-hypothesized value), a confidence interval gives us a range of values that include the parameter of interest, with some degree of “confidence.”

Confidence intervals have the general formula:

We get the value of from the -distribution. In particular, if we want the interval to include the true population parameter of the time, then we choose to be the percentile of the -distribution. For example, if we set , we will have a confidence interval by choosing to be the -th percentile of the -distribution.

As mentioned, -tests and confidence intervals are closely related. In particular, if the confidence interval includes the value , this is the same as retaining the null hypothesis that the parameter is equal to . This should make intuitive sense. If the confidence interval includes , we are saying that it is a reasonable value of the population parameter, so we should not reject that value. This equivalence between tests and confidence intervals assumes we use the same level of for both of them.

In summary, if the confidence interval includes zero, we retain the null hypothesis at the stated level of . If the confidence interval does not include zero, we reject the null hypothesis at the stated level of .

## 1.8 F-tests

The -test is used to infer if two independent variances have the same expected value. This turns out to be useful when we analyze the variance of a variable into different sources (i.e., Analysis of Variance or ANOVA).

A variance can be defined as a sum-of-squares divided by its degrees of freedom. For example, the sample variance is just a sum-of-squared deviations from the sample mean (a sum of squares) divided by (its degrees of freedom).

The generic formula for an F-test is the ratio of two variances:

where denotes sums-of-squares and denotes degrees of freedom.

Just the like -test, the -test is called by the letter “F” because it has an -distribution when the null hypothesis is true (i.e., when the variances have the same expected value). The plot below shows some examples of - distributions. These distributions tell us the values of that are likely, if the null hypothesis is true.

|  |
| --- |
| Figure 1.3: F-distribution (source: https://en.wikipedia.org/wiki/F-distribution) |

The F distribution has two parameters, which are referred to as the “degrees of freedom in the numerator” and the “degrees of freedom in the denominator” (in the figure, d1 and d2, respectively). We always write the numerator first and then the denominator . So, the green line in the figure is “an -distribution on 10 and 1 degrees of freedom”, which means the in the numerator is 10 and the in the denominator is 1.

We use an -test the same way we use a -test – we set a significance level and use this level to determine how large the value of needs to be for us to reject the null hypothesis. The main difference is that is non-negative, because it is the ratio of squared numbers. We don’t usually compute confidence intervals for statistics with an -distribution.

## 1.9 APA reporting

It is important to write up the results of statistical analyses in a way that other people will understand. For this reason, there are conventions about how to report statistical results. In this class, we will mainly use tables and figures (formatted in R) rather than inline text. But sometimes reporting statistics using inline text unavoidable, in which case this course will use APA formatting. You don’t need to use APA in this class, but you should be familiar with some kind of conventions for reporting statistical results in your academic writing.

The examples below illustrate APA conventions. We haven’t covered the examples, they are just illustrative of the formatting (spacing, italics, number of decimal places, whether or not to use a leading zero before a decimal, etc). More details are available online (for example, [here](https://psych.uw.edu/storage/writing_center/stats.pdf)).

* Jointly, the two predictors explained about 22% of the variation in Academic Achievement, which was statistically significant at the .05 level ().
* After controlling for SES, a one unit of increase in Maternal Education was associated with units of increase in Academic Achievement ().
* After controlling for Maternal Education, a one unit of increase in SES was associated with units of increase in Academic Achievement. This was a statistically significant relationship ().

## 1.10 Exercises

This section will walk through some basics of programming with R. We will get started with this part of the review in the first class. You don’t need to do it before class.

If you are already familiar with R, please skim through the content and work on getting the NELS data loaded. If you are not familiar with R, or would like to brush up your R skills, you should work through this section.

### 1.10.1 General info about R

Some things to know about R before getting started:

* R is case sensitive. It matters if you useCAPS or lowercase in your code.
* Each new R command should begin on its own line.
* Unlike many other programming languages, R commands do **not** need to end with punctuation (e.g., ; or .).
* R uses the hash tag symbol (#) for comments. Comments are ignored by R but can be helpful for yourself and others to understand what your code does. An example is below.

# This is a comment. R doesn't read it.  
# Below is a code snippet. R will read it and return the result.   
2 + 2

[1] 4

* R’s working memory is cumulative. This means that you have to run code in order, one line after the next. It also means that any code you run is still hanging around in R’s memory until you clear it away using rm or the brush icon in R Studio - make sure to ask about how to do this in class if you aren’t sure.

### 1.10.2 The basics

As we have just seen, R can do basic math like a calculator. Some more examples are presented in the code snippets below. R’s main math symbols are

* + addition
* - subtraction or negative numbers
* \* multiplication
* / division (don’t use \)
* ^ or \*\* exponentiation

2 \* 2

[1] 4

# Remember pedmas? Make sure to use parentheses "()",   
# not brackets "[]" or braces "{}"  
(2 - 3) \* 4 / 5

[1] -0.8

# Exponentiation can be done two ways  
2^3

[1] 8

2\*\*3

[1] 8

# Square roots are "squirt". Again, make sure to use "()",   
# not brackets "[]" or braces "{}"  
sqrt(25)

[1] 5

# Logs and exponents, base e (2.718282....) by default   
log(100)

[1] 4.60517

exp(1)

[1] 2.718282

# We can override the default log by using the "base" option  
log(100, base = 2)

[1] 6.643856

# Special numbers...  
pi

[1] 3.141593

### 1.10.3 The help function

The help function is your best friend when using R. If we want more info on how to use an R function (like log), type:

help(log)

If you don’t exactly remember the name of the function, using ??log will open a more complete menu of options.

### 1.10.4 Logicals and strings

R can also work with logical symbols that evaluate to TRUE or FALSE. R’s main logical symbols are

* == is equal to
* != is not equal to
* > greater than
* < less than
* >= greater than or equal to
* <= less than or equal to

Here are some examples:

2 + 2 == 4

[1] TRUE

2 + 2 == 5

[1] FALSE

2 + 3 > 5

[1] FALSE

2 + 3 >= 5

[1] TRUE

The main thing to note is that the logical operators return TRUE or FALSE as their output, not numbers. There are also symbols for logical conjunction (&) and disjunction (|), but we won’t get to those until later.

In addition to numbers and logicals, R can work with text (also called “strings”). We wont use strings a lot but they are worth knowing about.

"This is a string in R. The quotation marks tell R the input is text."

[1] "This is a string in R. The quotation marks tell R the input is text."

### 1.10.5 Assignment (naming)

Often we want to save the result of a calculation so that we can use it later on. In R, this means we need to assign the result a name. Once we assign the result a name, we can use that name to refer to the result, without having to re-do the calculation that produced the result. For example:

x <- 2 + 2

Now we have given the result of 2 + 2 the name “x” using the assignment operator, <-.

Note that R no longer prints the result of the calculation to the console. If we want to see the result, we can type x

# To see the result a name refers to, just type the name  
x

[1] 4

We can also do assignment with the = operator.

y = 2 + 2  
y

[1] 4

It’s important to note that the = operator also gets used in other ways (e.g., to override default values in functions like log). Also, the math interpretation of “=” doesn’t really capture what is happening with assignment in computer code. In the above code, we are not saying that “2 + 2 equals y.” Instead, we are saying, “2 + 2 equals 4 and I want to refer to 4 later with the name ‘y’.”

Almost anything in R can be given a name and thereby saved in memory for later use. Assignment will become a lot more important when we name things like datasets, so that we can use the data for other things later on.

A few other side notes:

* Names cannot include spaces or start with numbers. If you want separate words in a name, consider using a period ., an underscore \_, or CamelCaseNotation.
* You can’t use the same name twice. If you use a name, and then later on re-assign that same name to a different result, the name will now only represent the new result. The old result will no longer have a name, it will be lost in the computer’s memory and will be cleaned up by R’s garbage collector. Because R’s memory is cumulative, it’s important to keep track of names to make sure you know what’s what.
* R has some names that are reserved for built-in stuff, like log and exp and pi. You can override those names, but R will give a warning. If you override the name, this means you can’t use the built-in functions until you delete that name (e.g., rm(x)).

### 1.10.6 Pop-quiz

1. In words, describe what the following R commands do.
   * x <- 7
   * x = 7
   * x == 7
   * 7 -> x
   * 7 > x

* Answers: Check the commands in R.

### 1.10.7 Vectors

In research settings, we often want to work with multiple numbers at once (surprise!). R has many data types or “objects” for doing this, for example, vectors, matrices, arrays, data.frames, and lists. We will start by looking at vectors.

Here is an example vector, containing the sequence of integers from 15 to 25.

# A vector containing the sequence of integers from 15 to 25  
y <- 15:25  
y

[1] 15 16 17 18 19 20 21 22 23 24 25

When we work with a vector of numbers, sometimes we only want to access a subset of them. To access elements of a vector we use the square bracket notation []. Here are some examples of how to index a vector with R:

# Print the first element of the vector y  
# Note: use brackets "[]" not parens"()"  
y[1]

[1] 15

# The first 3 elements  
y[1:3]

[1] 15 16 17

# The last 5  
y[6:11]

[1] 20 21 22 23 24 25

We can also access elements of a vector that satisfy a given logical condition.

# Print the elements of the vector y that are greater than the value 22  
y[y > 22]

[1] 23 24 25

This trick often comes in handy so its worth understanding how it works. First let’s look again at what y is, and what the logical statement y > 22 evaluates to:

# This is the vector y  
y

[1] 15 16 17 18 19 20 21 22 23 24 25

# This is the logical expression y > 22  
y > 22

[1] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE TRUE TRUE TRUE

We can see that y > 22 evaluates to TRUE or FALSE depending on whether the correspond number in the vector y is greater than 22. When we use the logical vector as an index – R will then return all the values for which y > 22 is TRUE.

In general, we can index a vector y with any logical vector of the same length as y. The result will return only the values for which the logical vector is TRUE.

### 1.10.8 Computing sample stats

The following are examples of statistical operations you can do with vectors of numbers. These examples follow closely to [Section 1.1](#sec-summation-1) to [Section 1.4](#sec-properties-1)

# Making a vector with the "c" command (combine)   
x <- c(10, 9, 15, 15, 20, 17)  
  
# Find out how long a vector is (i.e., the sample size)  
length(x)

[1] 6

# Add up the elements of a vector  
sum(x)

[1] 86

# Add up the elements of a subset of a vector  
sum(x[2:3])

[1] 24

# Check the distributive rule  
sum(x\*2) == sum(x) \* 2

[1] TRUE

# Check the associative rule  
y <- c(5, 11, 11, 19, 13, 15)  
sum(x) + sum(y) == sum(x + y)

[1] TRUE

# Compute the mean  
mean(x)

[1] 14.33333

# Compute the variance  
var(x)

[1] 17.46667

# Compute the standard deviation  
sd(x)

[1] 4.179314

# Compute the covariance  
cov(x, y)

[1] 10.66667

# Compute the correlation  
cor(x, y)

[1] 0.5457986

### 1.10.9 Working with datasets

Most of the time, we will be reading-in data from an external source. The easiest way to do this is if the data is in the .RData file format. Then we can just double-click the .Rdata file and Rstudio will open the file, or we can use the load command in the console – both do the same thing.

To get started, let’s load the NELS data. The data are a subsample of the 1988 National Educational Longitudinal Survey (NELS; see <https://nces.ed.gov/surveys/nels88/>).

This data and codebook are available on the Canvas site of the course under “Files/Data/NELS” and are linked in the “Module” for Week 1. You need to download the data onto your machine and then open the data file (e.g., by clicking it, or double-clicking, or whatever you do to open files on your computer). That will do the same thing as the following line of code

#This is what happens when you double-click NELS.RData  
load("NELS.RData")

The function dim reports the number of rows (500 persons) and columns (48 variables) for the data set.

dim(NELS)

[1] 500 48

If you want to look at the data in a spreadsheet, use the following command. It won’t render anything in this book, but you can see what it does in R. (You may need to install XQuartz from https://www.xquartz.org if you are using a Mac.)

View(NELS)

If you want to edit the data set using the spreadsheet, use edit(NELS). However, R’s spreadsheet editor is pretty wimpy, so if you want to edit data in spreadsheet format, use Excel or something.

Working with data is often made easier by “attaching”” the dataset. When a dataset it attached, this means that we can refer to the columns of the dataset by their names.

# Attach the data set  
attach(NELS)  
  
# Print the first 10 values of the NELS gender variable  
gender[1:10]

[1] Male Female Male Female Male Female Female Female Female Male   
Levels: Female Male

**Warning about attaching datasets.** Once you attach a dataset, all of the column names in that dataset enter R’s working memory. If the column names in your dataset were already used, the old names are overwritten. If you attach the same dataset more than once in the same session, R will print a warning telling you that the previously named objects have been “masked” – this won’t affect your analyses, but it can be irritating.

The basic point: we should only attach each dataset once per R session. Once you are done using a data set it is good practice to detach it:

detach(NELS)

### 1.10.10 Preview of next week

[Figure 1.4](#fig-nels-1) shows the relationship between Grade 8 Math Achievement (percent correct on a math test) and Socioeconomic Status (SES; a composite measure on a scale from 0-35). Once you have reproduced this figure, you are ready to start the next chapter.

# Load and attach the NELS data  
load("NELS.RData")  
attach(NELS)  
  
# Scatter plot  
plot(x = ses,   
 y = achmat08,   
 col = "#4B9CD3",   
 ylab = "Math Achievement (Grade 8)",   
 xlab = "SES")  
  
# Run a simple linear regression   
mod <- lm(achmat08 ~ ses)  
  
# Add the regression line to the plot  
abline(mod)  
  
# Detach the data set  
detach(NELS)

|  |
| --- |
| Figure 1.4: Math Achievement and SES (NELS88). |

# 2. Simple regression

The focus of this course is linear regression with multiple predictors (AKA *multiple regression*), but we start by reviewing regression with one predictor (AKA *simple regression*). Most of this material should be familiar from EDUC 710.

## 2.1 An example from NELS

Let’s begin by considering an example. [Figure 2.1](#fig-nels-2) shows the relationship between Grade 8 Math Achievement (percent correct on a math test) and Socioeconomic Status (SES; a composite measure on a scale from 0-35). The data are a subsample of the 1988 National Educational Longitudinal Survey (NELS; see <https://nces.ed.gov/surveys/nels88/>).

# Load and attach the NELS88 data  
load("NELS.RData")  
attach(NELS)  
  
# Scatter plot  
plot(x = ses,   
 y = achmat08,   
 col = "#4B9CD3",   
 ylab = "Math Achievement (Grade 8)",   
 xlab = "SES")  
  
# Run the regression model  
mod <- lm(achmat08 ~ ses)  
  
# Add the regression line to the plot  
abline(mod)

|  |
| --- |
| Figure 2.1: Math Achievement and SES (NELS88). |

The strength and direction of the linear relationship between the two variables is summarized by their correlation. In this sample, the value of the correlation is:

cor(achmat08, ses)

[1] 0.3182484

This is a moderate, positive correlation between Math Achievement and SES. This correlation means that eighth graders from more well-off families (higher SES) also tended to do better in math (higher Math Achievement).

The relationship between SES and academic achievement has been widely documented and discussed in education research (e.g., <https://www.apa.org/pi/ses/resources/publications/education>). **Please look over this web page and be prepared to share your thoughts/questions about the relationship between SES and academic achievement and its relevance for education research.**

## 2.2 The regression line

The section presents three interchangeable ways of writing the regression line in [Figure 2.1](#fig-nels-2). You should be familiar with all three ways of presenting regression equations and you are welcome to use whichever approach you like best in your writing for this class.

The regression line in [Figure 2.1](#fig-nels-2) can be represented mathematically as

where

* denotes Math Achievement
* denotes SES
* represents the regression intercept (the value of when )
* represents the regression slope (how much changes for each unit of increase in )

In this equation, the symbol represents the *predicted value* of Math Achievement for a given value of SES. In [Figure 2.1](#fig-nels-2), the predicted values are represented by the regression line. The *observed values* of Math Achievement are denoted as . In [Figure 2.1](#fig-nels-2), these values are represented by the points in the scatter plot.

Some more terminology: the variable is often referred to as the *outcome* or the *dependent variable.* The variable is often referred to as the *predictor*, *independent variable*, *explanatory variable*, or *covariate.* Different areas of research have different conventions about terminology for regression. We talk more about “big picture” interpretations of regression in [Chapter 3](#sec-chap-3).

The difference between an observed value and its predicted value is called a *residual*. Residuals are denoted as . The residuals for a subset of the data points in [Figure 2.1](#fig-nels-2) are shown in pink in [Figure 2.2](#fig-resid-2)

# Get predicted values from regression model  
yhat <- mod$fitted.values  
  
# select a subset of the data  
set.seed(10)  
index <- sample.int(500, 30)  
  
# plot again  
plot(x = ses[index],   
 y = achmat08[index],   
 ylab = "Math Achievement (Grade 8)",   
 xlab = "SES")  
  
abline(mod)  
  
# Add pink lines  
segments(x0 = ses[index],   
 y0 = yhat[index],   
 x1 = ses[index],   
 y1 = achmat08[index],   
 col = 6, lty = 3)  
  
# Overwrite dots to make it look at bit better  
points(x = ses[index],   
 y = achmat08[index],   
 col = "#4B9CD3",   
 pch = 16)

|  |
| --- |
| Figure 2.2: Residuals for a Subsample of the Example. |

Notice that by definition:

This leads to a second way of writing out a regression model:

The difference between [Equation 2.1](#eq-yhat) and [Equation 2.2](#eq-y) is that the former lets us talk about the predicted values (), whereas the latter lets us talk about the observed data points ().

A third way to write out the model is using the variable names (or abbreviations) in place of the more generic “X, Y” notation. For example,

This notation is useful when talking about a specific example, because we don’t have to remember what and stand for. But this notation is more clunky and doesn’t lend itself talking about regression in general or writing other mathematical expressions related to regression.

You should be familiar with all three ways of presenting regression equations ([Equation 2.1](#eq-yhat), [Equation 2.2](#eq-y), and [Equation 2.3](#eq-math)) and you are welcome to use whichever approach you like best in this class.

## 2.3 OLS

This section talks about how to estimate the regression intercept (denoted as in [Equation 2.1](#eq-yhat)) and the regression slope (denoted as in [Equation 2.1](#eq-yhat)). The intercept and slope are collectively referred to as the *parameters* of the regression line. They are also referred to as *regression coefficients.*

Our overall goal in this section is to “fit a line to the data” – i.e., we want to select the values of the regression coefficients that best represent our data. An intuitive way to approach this problem is by minimizing the residuals – i.e., minimizing the total amount of pink in [Figure 2.2](#fig-resid-2). We can operationalize this intuitive idea by minimizing the sum of squared residuals:

where indexes the respondents in the sample. When we estimate the regression coefficients by minimizing , this is called ordinary least squares (OLS) regression. OLS is very widely used and is the main focus of this course, although we will visit some other approaches in the second half of the course.

The values of the regression coefficients that minimize can be found using calculus (i.e., compute the derivatives of and set them to zero). This approach leads to the following equations for the regression coefficients:

(If you aren’t familiar with the symbols in these equations, check out the review materials in [Chapter 1](#sec-chap-1) for a refresher.)

The formulas in [Equation 2.4](#eq-reg-coeffs) tell us how to compute the regression coefficients using our sample data. However, on face value, these formulas don’t tell us much about how to interpret the regression coefficients. For interpreting the regression coefficients, it is more straightforward to refer to [Equation 2.1](#eq-yhat).

To clarify:

* To *interpret* the regression intercept, use [Equation 2.1](#eq-yhat): It is the value of when . Similarly, the regression slope is how much changes for a one-unit increase in .
* To *compute* the regression coefficients, use [Equation 2.4](#eq-reg-coeffs). These formulas are not very intuitive – they are just what we get when we fit a line to the data using OLS.

It is important to emphasize that the formulas in [Equation 2.4](#eq-reg-coeffs) do lead to some useful mathematical results about regression. [Section 2.9](#sec-properties-2), which is optional, derives some of the main results. If you want a deeper mathematical understanding of regression, make sure to check out this section. If you prefer to skip the math and just learn about the results as they become relevant, that is OK too.

### 2.3.1 Correlation and regression

Before moving on, it is worth noting something that we can learn from [Equation 2.4](#eq-reg-coeffs) without too much math: the regression slope is just a re-packaging of the correlation coefficient. In particular, if we assume that and are z-scores (i.e., they are standardized to have mean of zero and variance of one), then [Equation 2.4](#eq-reg-coeffs) reduces to:

There are two important things to note here.

First, the difference between correlation and simple regression depends on the scale of the variables. Otherwise stated, if we standardize both and , then regression is just correlation. In particular, if the correlation is equal to zero, then the regression slope is also equal to zero – these are just two equivalent ways of saying that the variables are not (linearly) related.

Second, this relationship between correlation and regression holds only for simple regression (i.e., one predictor). When we get to multiple regression, we will see that relationship between regression and correlation (and covariance) gets more complicated.

For the NELS example in [Figure 2.1](#fig-nels-2), the regression intercept and slope are, respectively:

coef(mod)

(Intercept) ses   
 48.6780338 0.4292604

**Please write down an interpretation of these numbers and be prepared to share your answers in class. How would your interpretation change if, rather than the value of the slope shown above, we had ?**

## 2.4 R-squared

In this section we introduce another statistic that is commonly used in regression, called “R-squared” (in symbols: ). First we will talk about its interpretation, then we will show how it is computed.

R-squared is the proportion of variance in the outcome variable that is associated with, or “explained by”, the predictor variable. In terms of the NELS example, the variance of the outcome can be interpreted in terms of individual differences in Math Achievement – i.e., how students deviate from, or vary around, the mean level of Math Achievement. R-squared tells us the extent to which these individual differences in Math Achievement are associated with, or explained by, individual differences in SES.

As mentioned, R-squared is a proportion. Because it is a proportion, it takes on values between and . If then a student’s SES doesn’t tell us anything about their Math Achievement – this is the same as saying the two variables aren’t correlated, or that there is no (linear) relationship between Math Achievement and SES. If , then all of the data points fall exactly on the regression line, and we can perfectly predict each student’s Math Achievement from their SES.

You might be asking – why do we need R-squared? We already have the regression coefficient (which is just a repackaging of the correlation), so why do we need yet another way of describing the relationship between Math Achievement and SES? This is very true for simple regression! However, when we move on to multiple regression, we will see that R-squared lets us talk about the relationship between the outcome and *all* of the predictors, or any subset of the predictors, whereas the regression coefficient only lets us talk about the relationship with one predictor at a time.

To see how R-squared is computed for the NELS example, let’s consider [Figure 2.3](#fig-rsquared-2). The horizontal grey line denotes the mean of Math Achievement. Recall that the variance of is computed using the sum-of-squared deviations from the mean. For each student, these deviations from the mean can be divided into two parts. The Figure shows these two parts for a single student, using black and pink dashed lines:

* The black dashed line represents the extent to which the student’s deviation from the mean level of Math Achievement is explained by the linear relationship between Math Achievement and SES.
* The pink dashed line is the regression residual, which was introduced in [Section 2.2](#sec-regression-line-2). This is the variation in Math Achievement that is “left over” after considering the linear relationship with SES.

|  |
| --- |
| Figure 2.3: The Idea Behind R-squared. |

The R-squared statistic averages the variation in Math Achievement associated with SES (i.e., the black dashed line) for all students in the sample, and then divides by the total variation in Math Achievement (i.e., black + pink).

The derivation of the R-squared statistic is not very complicated and provides some useful notation. To simplify the derivation, we can work the numerator of the variance, which is called the “total sum of squares:”

Next we add and subtract the predicted values (that old trick!):

The right-hand-side can be reduced to two other sums of squares using the rules of summation algebra (see [Section 1.2](#sec-rules-1) – the derivation is long but not complicated).

The first term on the right-hand-side is just the sum of squared residuals () from [Section 2.3](#sec-ols-2). The second term is called the sum of squared regression and denoted . Using this notation we can re-write the previous equation as

and the R-squared statistic is computed as

As discussed above, this quantity can be interpreted as the proportion of variance in that is explained by its linear relationship with .

For the NELS example, the R-squared statistic is:

summary(mod)$r.squared

[1] 0.1012821

**Please write down an interpretation of this number and be prepared to share your answer in class.** Hint: Instead of talking about proportions, it is often helpful to multiply by 100 and talk about percentages instead.

## 2.5 The population model

Up to this point we have discussed simple linear regression as a way of describing the relationship between two variables in a sample. The next step is to discuss statistical inference. Recall that statistical inference involves generalizing from a sample to the population from which the sample was drawn.

In the NELS example, the population of interest is U.S. eighth graders in 1988. We want to be able to draw conclusions about that population based on the sample of eighth graders that participated in NELS. In order to do that, we make some statistical assumptions about the population, which are collectively referred to as the *population model*.

The population model for simple linear regression is summarized in [Figure 2.4](#fig-pop-model). The three assumptions associated with this model are written below. We talk about how to check the plausibility of these assumptions in **?@sec-chapter-7**.

|  |
| --- |
| Figure 2.4: The Regression Population Model. |

The three assumptions:

1. Normality: The values of conditional on , denoted , are normally distributed. The figure shows these distributions for three values of . We can write this assumption formally as

* (This notation should be familiar from EDUC 710. In general, we write to denote that the variable has a normal distribution with mean and variance .)

1. Homoskedasticity: The conditional distributions have equal variances (also called “homogeneity of variance”, or simply “equal variances”).
2. Linearity: The means of the conditional distributions are a linear function of .

These three assumptions are summarized by writing

Sometimes it will be easier to state the assumptions in terms of the population residuals, . The residuals have distribution .

Sometimes it will also be easier to write the population regression line using expected values, , rather than . Both of these are interpreted the same way – they denote the mean of for a given value of .

An additional assumption is usually made about the data in the sample – that they were obtained as a simple random sample from the population. We will see some ways of dealing with other types of samples later on this course, but for now we can consider this a background assumption that applies to OLS regression.

From a mathematical perspective, these assumptions are important because they can be used to prove (a) that OLS regression provides unbiased estimates of the population regression coefficients and (b) that the OLS estimates are more precise than any other unbiased estimates of the population regression coefficients. There are other variations on these assumptions, which are sometimes called the Gauss-Markov assumptions [see https://en.wikipedia.org/wiki/Gauss%E2%80%93Markov\_theorem](https://en.wikipedia.org/wiki/Gauss%E2%80%93Markov_theorem).

From a practical perspective, these assumptions are important conditions that we should check when conducting data analyses. If the assumptions are violated – particularly the linearity assumption – then our statistical model may not be a good representation of the population. If the model is not a good representation of the population, then inferences based on the model may provide misleading conclusions about the population.

## 2.6 Clarifying notation

At this point we have used the mathematical symbols for regression (e.g., , ) in two different ways:

* In [Section 2.2](#sec-regression-line-2) they denoted sample statistics.
* In [Section 2.5](#sec-population-model-2) they denoted population parameters.

The population versus sample notation for regression is a bit of a hot mess, but the following conventions are used.

| Concept | Sample statistic | Population parameter |
| --- | --- | --- |
| regression line |  | or |
| slope |  |  |
| intercept |  |  |
| residual |  |  |
| variance explained |  |  |

The “hats” always denote sample quantities, and the Greek letters always denote population quantities, but there is some lack of consistency. For example, why not use instead of for the population slope? Well, is conventionally used to denote standardized regression coefficients in the *sample*, so its already taken (more on this in [Chapter 4](#sec-chap-4)).

If it is clear from context that we are talking about the sample rather than the population, then the hats are usually omitted from the statistics , , and . This doesn’t apply to , because the hat is required to distinguish the predicted values from the data points.

Another thing to note is that while is often called a predicted value, is not usually referred to this way. It is called the conditional mean function or the conditional expectation function. Using this language, we can say that regression is about estimating the conditional mean function.

**Please be prepared for a pop quiz on notation during class!**

## 2.7 Inference

This section reviews the main inferential procedures for regression. The formulas presented in this section are used to produce standard errors, t-tests, p-values, and confidence intervals for the regression coefficients, as well as an F-test for R-squared. It is very unlikely that you will ever need to compute these formulas by hand, so don’t worry about memorizing them.

However, it is important that you can interpret the numerical results in research settings. The interpretations of these procedures were reviewed in [Chapter 1](#sec-chap-1) and should be familiar from EDUC 710. This sections documents the formulas for simple regression and then asks you to interpret the results in the context of the NELS example.

It is worth noting that the regression intercept is often not of interest in simple regression. Recall that the intercept is the value of when . So, unless we have a hypothesis or research question about this particular value of (e.g., eighth graders with ), we won’t be interested in a test of the regression intercept. When we get into to multiple regression, we will see some situations where the intercept is of interest.

### 2.7.1 Inference for coefficients

When the population model is true, is an unbiased estimate of (in symbols: ). The standard error of is equal to (see (**fox-2016?**), Section 6.1):

Using these two results, we can compute t-tests and confidence intervals for the regression slope in the usual way.

**t-tests**

The null hypothesis can be tested against the alternative using the test statistic:

which has a t-distribution on degrees of freedom when the null hypothesis is true.

The test assumes that the population model is correct. The null hypothesized value of the parameter is usually chosen to be , in which case the test is interpreted in terms of the “statistical significance” of the regression slope.

**Confidence intervals**

For a given Type I Error rate, , the corresponding confidence interval is

where denotes the quantile of the -distribution with degrees of freedom. For example, if is chosen to be , the corresponding confidence interval uses , or the 2.5-th percentile of the t-distribution.

The standard error for the regression intercept, presented below, can be used to compute t-tests and confidence intervals for :

### 2.7.2 Inference for R-squared

The null hypothesis can be tested against the alternative using the F-test:

which has a F-distribution on and degrees of freedom when the null is true. The test assumes that the population model is true. Confidence intervals for R-squared are generally not reported.

### 2.7.3 The NELS example

For the NELS example, the standard errors, t-test, and p-values of the regression coefficients are shown in the table below (along with the OLS estimates). The F-test appears in the text below the table. Note that the output uses the terminology “multiple R-squared” to refer to R-squared.

summary(mod)

Call:  
lm(formula = achmat08 ~ ses)  
  
Residuals:  
 Min 1Q Median 3Q Max   
-20.5995 -6.5519 -0.1475 6.0226 27.6634   
  
Coefficients:  
 Estimate Std. Error t value Pr(>|t|)   
(Intercept) 48.6780 1.1282 43.147 < 2e-16 \*\*\*  
ses 0.4293 0.0573 7.492 3.13e-13 \*\*\*  
---  
Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
  
Residual standard error: 8.863 on 498 degrees of freedom  
Multiple R-squared: 0.1013, Adjusted R-squared: 0.09948   
F-statistic: 56.12 on 1 and 498 DF, p-value: 3.127e-13

The confidence intervals for the regression coefficients are:

confint(mod)

2.5 % 97.5 %  
(Intercept) 46.4614556 50.8946120  
ses 0.3166816 0.5418392

**Please write down your interpretation of the t-tests, confidence intervals, and F-test, and be prepared to share your answers in class.** Hint: state whether the tests are statistically significant and the corresponding conclusions are about the population parameters. For confidence intervals, report the range of values we are “confident” about. For practice, you might want to try using APA notation in your answer (or whatever style conventions are used in your field).

## 2.8 Power analysis\*

Statistical power is the probability of rejecting the null hypothesis, when it is indeed false. Rejecting the null hypothesis when it is false is sometimes called a “true positive”, meaning we have correctly inferred that a parameter of interest is not zero. Power analysis is useful for designing studies so that the statistical power / true positive rate is satisfactory.

In practice, statistical power comes down to having a large enough sample size. Consequently, power analysis is important when planning studies (e.g., in research grants proposals). In this class, we will not be planning any studies – rather, we will be working with secondary data analyses. In this context, power analysis is not very interesting, and so we do not mention it much. Nonetheless, power analysis is important for research and so we review the basics here.

Power analysis in regression is very similar to power analysis for the tests we studied last semester. There are five ingredients that go into a power analysis:

* The desired Type I Error rate, .
* The desired level of statistical power.
* The sample size, .
* The number of predictors.
* The effect size, which is Cohen’s f-squared statistic (AKA the signal to noise ratio):

In principal, we can plug-in values for any four of these ingredients and then solve for the fifth. But, as mentioned, power analysis is most useful when we solve for while planning a study. When solving for “prospectively,” the effect size should be based on reports of R-squared in past research. Power and are usually chosen to be .8 and .05, respectively.

The example below shows the sample size required to detect an effect size of . This effect size was based on the NELS example discussed above. Note that the values and denote the degrees of freedom in the numerator and denominator of the F-test of R-squared, respectively. The former provides information about the number of predictors in the model, the latter about sample size.

# Install package  
# install.packages("pwr")  
  
# Load library  
library(pwr)  
  
# Run power analysis  
pwr.f2.test(u = 1, f2 = .1 / (1 - .1), sig.level = .05, power = .8)

Multiple regression power calculation   
  
 u = 1  
 v = 70.61137  
 f2 = 0.1111111  
 sig.level = 0.05  
 power = 0.8

Rounding up, we would require 72 persons in the sample in order to have an 80% chance of detecting an effect size of with simple regression.

Another use of power analysis is to solve for the effect size. This can be useful when the sample size is constrained by external factors (e.g., budget). In this situation, we can use power analysis to address whether the sample size is sufficient to detect an effect that is “reasonable” (again, based on past research). In the NELS example, we have observations. The output below reports the smallest effect size we can detect with a power of and . This is sometimes called the “minimum detectable effect size” (MDES).

pwr.f2.test(u = 1, v = 498, sig.level = .05, power = .8)

Multiple regression power calculation   
  
 u = 1  
 v = 498  
 f2 = 0.01575443  
 sig.level = 0.05  
 power = 0.8

With a sample size of 500, and power of 80%, the MDES for simple regression is . Based on this calculation, we can conclude that this sample size is sufficient for applications of simple linear regression in which we expect to explain at least 3% of the variance in the outcome.

## 2.9 Properties of OLS\*

This section summarizes some properties of OLS regression that will be used later on. You can skip this section if you aren’t interested in the math behind regression – the results will be mentioned again when needed.

The derivations in this section follow from the three rules of summation reviewed [Section 1.2](#sec-rules-1) and make use of the properties of means, variances, and covariances already derived in [Section 1.4](#sec-properties-1). If you have any questions about the derivations, I would be happy to address them in class during open lab time.

### 2.9.1 Residuals

We start with two important implications of [Equation 2.4](#eq-reg-coeffs) for the OLS residuals. In particular, OLS residuals always have mean zero and are uncorrelated with the predictor variable. These properties generalize to multiple regression.

First we show that

From [Equation 2.4](#eq-reg-coeffs) we have

Solving for gives

Since is a linear transformation of , we know from [Section 1.4](#sec-properties-1) that

The previous two equations imply that . Consequently,

Next we show that

The derivation is:

The second last line uses the expression for the slope in [Equation 2.4](#eq-reg-coeffs).

### 2.9.2 Multiple correlation ()

Above we defined as a proportion of variance. This was a bit lazy. Instead, we can start with the definition of the multiple correlation

and from this definition derive the result, shown above, that is the proportion of variance in associated with .

Let’s start by showing that

Before deriving this result, note that [Equation 2.4](#eq-reg-coeffs) implies

and, using the the variance of a linear transformation ([Section 1.4](#sec-properties-1)), we have

These two results are used on the third and fourth lines of the following derivation, respectively.

Next we show that :

This derivation is nicer than the one in [Section 2.4](#sec-rsquared-2) because it obtains a result about using the definition of . However, this derivation does not show that the resulting ratio is a proportion, which requires a second step (which also uses [Equation 2.7](#eq-cov-y-yhat)):

Re-arranging gives

which shows that

One last detail concerns the relation between the multiple correlation and the regular correlation coefficient . Using the invariance of the correlation under linear transformation ([Section 1.4](#sec-properties-1)), we have

Consequently, in simple regression, – i.e., the proportion of variance explained by the predictor is just the squared Pearson product-moment correlation. When we add multiple predictors, this relationship between and no longer holds.

## 2.10 Workbook

This section collects the questions asked in this chapter. The lesson for this chapter will focus on discussing these questions and then working on the exercises in [Section 2.11](#sec-exercises-2). The lesson will **not** be a lecture that reviews all of the material in the chapter! So, if you haven’t written down / thought about the answers to these questions before class, the lesson will not be very useful for you. Please engage with each question by writing down one or more answers, asking clarifying questions about related material, posing follow up questions, etc.

[Section 2.1](#sec-example-2)

# Scatter plot  
plot(x = ses,   
 y = achmat08,   
 col = "#4B9CD3",   
 ylab = "Math Achievement (Grade 8)",   
 xlab = "SES")  
  
# Run the regression model  
mod <- lm(achmat08 ~ ses)  
  
# Add the regression line to the plot  
abline(mod)

|  |
| --- |
| Math Achievement and SES (NELS88). |

The strength and direction of the linear relationship between the two variables is summarized by their correlation. In this sample, the value of the correlation is:

cor(achmat08, ses)

[1] 0.3182484

This correlation means that eighth graders from more well-off families (higher SES) also tended to do better in Math (higher Math Achievement). This relationship between SES and academic achievement has been widely documented and discussed in education research (e.g., <https://www.apa.org/pi/ses/resources/publications/education>). Please look over this web page and be prepared to share your thoughts about this relationship.

[Section 2.3](#sec-ols-2)

For the NELS example, the regression intercept and slope are, respectively:

coef(mod)

(Intercept) ses   
 48.6780338 0.4292604

Please write down an interpretation of these numbers and be prepared to share your answers in class. How would your interpretation change if, rather than the value of the slope shown above, we had ?

[Section 2.4](#sec-rsquared-2)

For the NELS example, the R-squared statistic is:

summary(mod)$r.squared

[1] 0.1012821

Please write down an interpretation of this number and be prepared to share your answer in class. Hint: Instead of talking about proportions, it is often helpful to multiply by 100 and talk about percentages instead.

[Section 2.6](#sec-notation-2)

Please be prepared for a pop quiz on notation during class!

| Concept | Sample statistic | Population parameter |
| --- | --- | --- |
| regression line |  |  |
| slope |  |  |
| intercept |  |  |
| residual |  |  |
| variance explained |  |  |

[Section 2.7](#sec-inference-2)

For the NELS example, the standard errors, t-test, and p-values of the regression coefficients are shown in the table below (along with the OLS estimates). The F-test appears in the text below the table. Note that the output uses the terminology “multiple R-squared” to refer to R-squared.

summary(mod)

Call:  
lm(formula = achmat08 ~ ses)  
  
Residuals:  
 Min 1Q Median 3Q Max   
-20.5995 -6.5519 -0.1475 6.0226 27.6634   
  
Coefficients:  
 Estimate Std. Error t value Pr(>|t|)   
(Intercept) 48.6780 1.1282 43.147 < 2e-16 \*\*\*  
ses 0.4293 0.0573 7.492 3.13e-13 \*\*\*  
---  
Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
  
Residual standard error: 8.863 on 498 degrees of freedom  
Multiple R-squared: 0.1013, Adjusted R-squared: 0.09948   
F-statistic: 56.12 on 1 and 498 DF, p-value: 3.127e-13

The confidence intervals for the regression coefficients are:

confint(mod)

2.5 % 97.5 %  
(Intercept) 46.4614556 50.8946120  
ses 0.3166816 0.5418392

Please write down your interpretation of the t-tests, confidence intervals, and F-test, and be prepared to share your answers in class. Hint: state whether the tests are statistically significant and the corresponding conclusions are about the population parameters. For confidence intervals, report the range of values we are “confident” about. For practice, you might want to try using APA notation in your answer (or whatever style conventions are used in your field).

## 2.11 Exercises

These exercises collect all of the R input used in this chapter into a single step-by-step analysis. It explains how the R input works, and provides some additional exercises. We will go through this material in class together, so you don’t need to work on it before class (but you can if you want.)

Before staring this section, you may find it useful to scroll to the top of the page, click on the “</> Code” menu, and select “Show All Code.”

### 2.11.1 The lm function

The functionlm, short for “linear model”, is used to estimate linear regressions using OLS. It also provides a lot of useful output.

The main argument that the we provides to the lm function is a formula. For the simple regression of Y on X, a formula has the syntax:

Y ~ X

Here Y denotes the outcome variable and X is the predictor variable. The tilde ~ just means “equals”, but the equals sign = is already used to for other stuff in R, so ~ is used instead. We will see more complicated formulas as we go through the course. For more information on R’s formula syntax, see help(formula).

Let’s take a closer look using the following two variables from the NELS data.

* achmat08: eighth grade math achievement (percent correct on a math test)
* ses: a composite measure of socio-economic status, on a scale from 0-35

# Load the data. Note that you can click on the .RData file and RStudio will load it  
# load("NELS.RData") #Un-comment this line to run  
  
# Attach the data: will discuss this in class  
# attach(NELS) #Un-comment this line to run!  
  
# Scatter plot of math achievement against SES  
plot(x = ses, y = achmat08, col = "#4B9CD3")  
  
# Regress math achievement on SES; save output as "mod"  
mod <- lm(achmat08 ~ ses)  
  
# Add the regression line to the plot  
abline(mod)



# Print the regression coefficients  
coef(mod)

(Intercept) ses   
 48.6780338 0.4292604

Let’s do some quick calculations to check that the lm output corresponds the formulas for the slope and intercept in [Section 2.3](#sec-ols-2):

We won’t usually do this kind of “manual” calculation, but it is a good way consolidate knowledge presented in the readings with the output presented by R. It is also useful to refresh our memory about some useful R functions and how the R language works.

# Compute the slope as the covariance divided by the variance of X  
cov\_xy <- cov(achmat08, ses)  
var\_x <- var(ses)  
b <- cov\_xy / var\_x  
  
# Compare the "manual" calculation to the output from lm.   
b

[1] 0.4292604

# Compute the y-intercept using from the two means and the slope  
xbar <- mean(ses)  
ybar <- mean(achmat08)  
  
a <- ybar - b \* xbar  
  
# Compare the "manual" calculation to the output from lm.   
a

[1] 48.67803

Let’s also check our interpretation of the parameters. If the answers to these questions are not clear, please make sure to ask in class!

* What is the predicted value of achmat08 when ses is equal to zero?
* How much does the predicted value of achmat08 increase for each unit of increase in ses?

### 2.11.2 Variance explained

Another way to describe the relationship between the two variables is by considering the amount of variation in that is associated with (or explained by) its relationship with . Recall that one way to do this is via the “variance” decomposition

from which we can compute the proportion of variation in Y that is associated with the regression model:

The R-squared for the example is presented in the output below. You should be able to provide an interpretation of this number, so if it’s not clear make sure to ask in class!

# R-squared from the example  
summary(mod)$r.squared

[1] 0.1012821

As above, let’s compute “by hand” for our example.

# Compute the sums of squares  
ybar <- mean(achmat08)  
ss\_total <- sum((achmat08 - ybar)^2)  
ss\_reg <- sum((yhat - ybar)^2)  
ss\_res <- sum((achmat08 - yhat)^2)  
  
# Check that SS\_total = SS\_reg + SS\_res  
ss\_total

[1] 43526.91

ss\_reg + ss\_res

[1] 43526.91

# Compute R-squared (compare to value from lm)  
ss\_reg/ss\_total

[1] 0.1012821

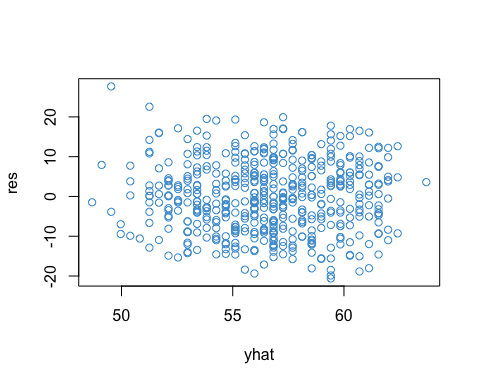
# Also check that R-squared is really equal to the square of the PPMC  
cor(achmat08, ses)^2

[1] 0.1012821

### 2.11.3 Predicted values and residuals

The lm function returns the predicted values and residuals and which we can access using the $ operator. These are useful for various reasons, especially model diagnostics, which we discuss later in the course. For now, lets just take a look at the residual vs fitted plot to illustrate the code.

yhat <- mod$fitted.values  
res <- mod$resid  
  
plot(yhat, res, col = "#4B9CD3")



Also note that the residuals values have mean zero and are uncorrelated with the predictor – this is always the case in OLS (See [Section 2.9](#sec-properties-2)})

mean(res)  
cor(yhat, res)

### 2.11.4 Inference

Next let’s address statistical inference, or how we can make conclusions about a population based on a sample from that population.

We can use the summary function to test the coefficients in our model.

summary(mod)

Call:  
lm(formula = achmat08 ~ ses)  
  
Residuals:  
 Min 1Q Median 3Q Max   
-20.5995 -6.5519 -0.1475 6.0226 27.6634   
  
Coefficients:  
 Estimate Std. Error t value Pr(>|t|)   
(Intercept) 48.6780 1.1282 43.147 < 2e-16 \*\*\*  
ses 0.4293 0.0573 7.492 3.13e-13 \*\*\*  
---  
Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
  
Residual standard error: 8.863 on 498 degrees of freedom  
Multiple R-squared: 0.1013, Adjusted R-squared: 0.09948   
F-statistic: 56.12 on 1 and 498 DF, p-value: 3.127e-13

In the table, the t-test and p-values are for the null hypothesis that the corresponding coefficient is zero in the population. We can see that the intercept and slope are both significantly different from zero at the .05 level. However, the test of the intercept is not very meaningful (why?).

The text below the table summarizes the output for R-squared, including its F-test, it’s degrees of freedom, and the p-value. (We will talk about adjusted R-square in [Chapter 4](#sec-chap-4))

We can also use the confint function to obtain confidence intervals for the regression coefficients. Use help to find out more about the confint function.

confint(mod)

2.5 % 97.5 %  
(Intercept) 46.4614556 50.8946120  
ses 0.3166816 0.5418392

Be sure to remember the correct interpretation of confidence intervals: *there is a 95% chance that the interval includes the true parameter value* (not: there is a 95% chance that the parameter falls in the interval). For example, there is a 95% chance that the interval [.32, .54] includes the true regression coefficient for SES.

### 2.11.5 Writing up results

We could write up the results from this analysis in APA format as follows. You should practice doing this kind of thing, because it is important to be able to write up the results of your analyses in a way that people in your area of research will understand.

In this analysis, we considered the relationship between Math Achievement in Grade 8 (percent correct on a math test) and SES (a composite on a scale from ). Regressing Math Achievement on SES, the relationship was positive and statistically significant at the level (, , , ). SES explained about of the variation in Math Achievement (, , ).

### 2.11.6 Additional exercises

If time permits, we will address these additional exercises in class.

These exercises replace achmat08 with

* achrdg08: eighth grade Reading Achievement (percent correct on a reading test)

Please answer the following questions using R.

* Plot achrdg08 against ses.
* What is the correlation between achrdg08 and ses? How does it compare to the correlation with Math and SES?
* How much variation in Reading is explained by SES? Is the proportion of variance explained significant at the .05 level?
* How much do predicted Reading scores increase for a one unit of increase in SES? Is this a statistically significant at the .05 level?
* What are your overall conclusions about the relationship between Academic Achievement and SES in the NELS data? Write up your results using APA formatting or whatever conventions are used in your area of research.

# 3. Two predictors

## 3.1 Interpretations

Multiple regression has three main interpretations:

* Prediction (focus on )
* Causation (focus on )
* Explanation (focus on )

By understanding these interpretations, we will have a better idea of how multiple regression is used in research. Each interpretation also provides a different perspective on the importance of using multiple predictor variables, rather than only a single predictor.

### 3.1.1 Prediction

Prediction was the original use of regression (<https://en.wikipedia.org/wiki/Regression_toward_the_mean#History>). In the context of simple regression, prediction means using observations of to make a guess about yet unobserved values of . Our guess is , and this is why is called the “predicted value” of .

When making predictions, we usually want some additional information about how precise the predictions are. In OLS regression, this information is provided by the standard error of prediction (**fox-2016?**):

This statistic quantifies our uncertainty when making predictions based on observations of that were not in our original sample. The prediction errors for the NELS example in [Chapter 2](#sec-chap-2) are represented in [Figure 3.1](#fig-pred-error-3) as a gray band around the regression line.

# Plotting library  
library(ggplot2)  
  
# Load data  
load("NELS.RData")  
  
# Run regression   
mod <- lm(achmat08 ~ ses, data = NELS)  
  
# Compute SE(Y-hat)  
n <- nrow(NELS)  
ms\_res <- var(mod$residuals) \* (n-1) / (n-2)  
d\_ses <- NELS$ses - mean(NELS$ses)   
se\_yhat <- sqrt(ms\_res \* (1 + 1/n + d\_ses^2 / sum(d\_ses^2)))  
  
# Plotting  
gg\_data <- data.frame(  
 achmat08 = NELS$achmat08,  
 ses = NELS$ses,  
 y\_hat = mod$fitted.values,  
 lwr = mod$fitted.values - 1.96 \* se\_yhat,  
 upr = mod$fitted.values + 1.96 \* se\_yhat)  
  
ggplot(gg\_data, aes(x = ses, y = achmat08))+  
 geom\_point(color='#3B9CD3', size = 2) +  
 geom\_line(aes(x = ses, y = y\_hat), color = "grey35") +  
 geom\_ribbon(aes(ymin=lwr,ymax=upr),alpha=0.3) +   
 ylab("Math Achievement (Grade 8)") +  
 xlab("SES") +  
 theme\_bw()

|  |
| --- |
| Figure 3.1: Prediction Error for NELS Example. |

We can see in the figure that the error band is quite wide. So, we might wonder how to make our predictions more precise. On way to do this is by including more predictors in the regression model – i.e., multiple regression.

To see why including more predictors improves the precision of predictions, note that the standard error of prediction shown in [Equation 3.1](#eq-se-pred) increases with , which is the variation in the outcome that is *not* explained by the predictor (see [Section 2.4](#sec-rsquared-2)). In most situations, is the largest contributor the prediction error. As we will see below, one way to reduce is by adding more predictors to the model.

#### 3.1.1.1 More about prediction

Regression got its name from a statistical property of predicted scores called “regression toward the mean.” To explain this property, let’s assume and are z-scores (i.e., both variables have and ). Recall that this implies that and , so the regression equation reduces to

Since , the absolute value of the must be less than or equal to that of . And, since both variables have , this implies that is closer to the mean of than is to the mean of . This is sometimes called regression toward the mean.

Although prediction was the original use of regression, many research problems do not involve prediction. For instance, there are no students in the NELS data for whom we need to predict Math Achievement – all of the test scores are already in the data! However, there has been a resurgence of interest in prediction in recent years, especially in machine learning. Although the methods used in machine learning are often more complicated than OLS regression, the basic problem is the same. Because the models are more complicated, theoretical results like [Equation 3.1](#eq-se-pred) are more difficult to obtain. Consequently, machine learning uses data-driven procedures like cross-validation to evaluate model predictions. As one example, we could evaluate the accuracy and precision of out-of-sample predictions by splitting our data into two samples, fitting the model in one sample (the “training data”), and then making predictions in the other sample (the “test data”). [Equation 3.1](#eq-se-pred) is a theoretical result saves us the trouble of doing this with OLS. Machine learning has also introduced some new techniques for choosing which predictors to include in a model (“variable selection” methods like the lasso). We will touch on these topics later in the course when we get to model building.

### 3.1.2 Causation

A causal interpretation of regression means that that changing by one unit will change by units. This is interpreted as a claim about the expected value of “in real life”, not simply a claim about the mechanics of the regression line. In terms of our example, a causal interpretation would state that improving students’ SES by one unit will, on average, cause Math Achievement to increase by about half a percentage point.

The gold standard for inferring causality is to randomly assign people to different treatment conditions. In a regression context, treatment is represented by the independent variable, or the variable. While randomized experiments are possible in some settings, there are many types of variables that we cannot feasibly randomly assign (e.g., SES).

The concept of an omitted variable is used to describe what happens when we can’t (or don’t) randomly assign people to treatment conditions. An omitted variable is any variable that is correlated with both and . In our example, this would be any variable correlated with both Math Achievement and SES (e.g., School Quality). When we use random assignment, we ensure that is uncorrelated with *all* pre-treatment variables – i.e., randomization ensure that there are no omitted variables. However, when we don’t use random assignment, our results may be subject to *omitted variable bias*.

The overall idea of omitted variable bias is the same as “correlation causation”. The take-home message is summarized in the following points, which are stated in terms of the our NELS example.

* Any variable that is correlated with Math Achievement and with SES is called an omitted variable. One example is School Quality. This is an omitted variable because we did not include it as a predictor in our simple regression model.
* The problem is not just that we have an incomplete picture of how School Quality is related to Math Achievement.
* Omitted variable bias means that the predictor variable that *was included in the model* ends up having the wrong regression coefficient. Otherwise stated, the regression coefficient of SES is biased because we did not consider School Quality.
* In order to mitigate omitted variable bias, we want to include plausible omitted variables in our regression models – i.e., multiple regression.

#### 3.1.2.1 Omitted variable bias\*

Omitted variable bias is nicely explained by Gelman and Hill (**gelman-2007?**), and a modified version of their discussion is provided below. We start by assuming a “true” regression model with two predictors. In the context of our example, this means that there is one other variable, in addition to SES, that is important for predicting Math Achievement. Of course, there are many predictors of Math Achievement (see Section [Section 2.1](#sec-example-2)), but we only need two to explain the problem of omitted variable bias.

Write the “true” model as:

where is SES and is any other variable that is correlated with both and (e.g., School Quality).

Next, imagine that instead of using the model in [Equation 3.2](#eq-2parm), we analyze the data using the model with just SES, leading to the usual simple regression:

The problem of omitted variable bias is that – i.e., the regression coefficient in the true model is not the same as the regression coefficient in the model with only one predictor. This is perhaps surprising – leaving out School Quality gives us the wrong regression coefficient for SES!

To see why, start by writing as a function of .

Next we use [Equation 3.4](#eq-X2) to substitute for in [Equation 3.2](#eq-2parm),

Notice that in the last line, is predicted using only , so it is equivalent to [Equation 3.3](#eq-1parm). Based on this comparison, we can write

* $a^\* = \color{orange}{a + \alpha}$
* $b^\*\_1 = \color{green}{b\_1 + b\_2\beta}$

The equation for is what we are most interested in. It shows that the regression parameter in our one-parameter model () is not equal to the “true” regression parameter using both predictors ().

This is what omitted variable bias means – leaving out in Equation [Equation 3.3](#eq-1parm) gives us the wrong regression parameter for . This is one of the main motivations for including more than one predictor variable in a regression model – i.e., to avoid omitted variable bias.

Notice that there two special situations in which omitted variable bias is not a problem:

* When the two predictors are not related – i.e., .
* When the second predictor is not related to – i.e., .

### 3.1.3 Explanation

Many uses of regression fall somewhere between prediction and causation. We want to do more than just predict outcomes of interest, but we often don’t have a basis for making the strong assumptions required for a causal interpretation of regression coefficients. This grey area between prediction and causation can be referred to as explanation.

In terms of our example, we might want to explain why eighth graders differ in their Math Achievement. There are large number of potential reasons for individual difference in Math Achievement, such as

* Student factors
  + attendance
  + past academic performance in Math
  + past academic performance in other subjects (Question: why include this?)
  + …
* School factors
  + their ELA teacher
  + the school they attend
  + their peers
  + …
* Home factors
  + SES
  + maternal education
  + paternal education
  + parental expectations
  + …

When the goal of an analysis is explanation, it usual to focus on the proportion of variation in the outcome variable that is explained by the predictors, i.e., R-squared (see [Section 2.4](#sec-rsquared-2)). Later in the course we will see how to systematically study the variance explained by individual predictors, or blocks of several predictors (e.g., student factors).

Note that even a long list of predictors such as that above leaves out potential omitted variables. While the addition of more predictors can help us explain more of the variation in Math Achievement, it is rarely the case that we can claim that all relevant variables have been included in the model.

## 3.2 An example from ECLS

In the remainder of this chapter we will consider a new example from the 1998 Early Childhood Longitudinal Study (ECLS; <https://nces.ed.gov/ecls/>). Below is a description of the data from the official NCES codebook (page 1-1 of <https://nces.ed.gov/ecls/data/ECLSK_K8_Manual_part1.pdf>):

*The ECLS-K focuses on children’s early school experiences beginning with kindergarten and ending with eighth grade. It is a multisource, multimethod study that includes interviews with parents, the collection of data from principals and teachers, and student records abstracts, as well as direct child assessments. In the eighth-grade data collection, a student paper-and-pencil questionnaire was added. The ECLS-K was developed under the sponsorship of the U.S. Department of Education, Institute of Education Sciences, National Center for Education Statistics (NCES). Westat conducted this study with assistance provided by Educational Testing Service (ETS) in Princeton, New Jersey.*

*The ECLS-K followed a nationally representative cohort of children from kindergarten into middle school. The base-year data were collected in the fall and spring of the 1998–99 school year when the sampled children were in kindergarten. A total of 21,260 kindergartners throughout the nation participated*.

The subset of the ECLS-K data used in this class was obtained from the link below.

<http://routledgetextbooks.com/textbooks/_author/ware-9780415996006/data.php>

The codebook for this subset of data is available on our course website. In this chapter, we will be using a even smaller subset of cases from the example data set (the ECLS250.RData data)

We focus on the following three variables.

* Math Achievement in the first semester of Kindergarten. This variable can be interpreted as the number of questions (out of 60) answered correctly on a math test. Don’t worry – the respondents in this study did not have to write a 60-question math test in the first semester of K! Students only answered a few of the questions and their scores were re-scaled to be out a total of 60 questions afterwards.
* Socioecomonic Status (SES), which is a composite of household factors (e.g., parental education, household income) ranging from 30-72.
* Approaches to Learning (ATL), which is a teacher-reported measure of behaviors that affect the ease with which children can benefit from the learning environment. It includes six items that rate the child’s attentiveness, task persistence, eagerness to learn, learning independence, flexibility, and organization. The items have 4 response categories (1-4), with higher values representing more positive responses, and ATL is scored as an unweighted average the six items.

### 3.2.1 Correlation matrices

As was the case in simple regression, the correlation coefficient is a building block of multiple regression. So, we will start by examining the correlations in our example. We also introduce a new way of presenting correlations, the correlation matrix. The notation developed in this section will appear throughout the rest of the chapter.

In the scatter plots below, the panels are arranged in matrix format. The variables named in the diagonal panels correspond to the vertical () axis in that row and the horizontal () axis in that column. For example, Math is in the first diagonal, so it is the variable on vertical axis in the first row and the horizontal axis in the first column. This can be a bit confusing at first, so take a moment to make sure you know which variable is on which axis in each plot. Also notice that plots below the diagonal are just mirror image of the plots above the diagonal.

load("ECLS250.RData")  
attach(ecls)  
example\_data <- data.frame(c1rmscal, wksesl, t1learn)  
names(example\_data) <- c("Math", "SES", "ATL")  
pairs(example\_data , col = "#4B9CD3")

|  |
| --- |
| Figure 3.2: ECLS Example Data. |

The format of [Figure 3.2](#fig-pairs-3) is the same as that of the correlation matrix among the variables, which is shown below.

cor(example\_data)

Math SES ATL  
Math 1.0000000 0.4384619 0.3977048  
SES 0.4384619 1.0000000 0.2877015  
ATL 0.3977048 0.2877015 1.0000000

Again, notice that the entries below the diagonal are mirrored by the entries above the diagonal. We can see that SES and ATL have similar correlations with Math Achievement (0.4385 and 0.3977, respectively), and are also moderately correlated with each other (0.2877).

In order to represent the correlation matrix among a single outcome variable () and two predictors ( and ) we will use the following notation:

In this notation, is the correlation between and . Note that each correlation coefficient (“”) has two subscripts that tell us which two variables are being correlated. For the outcome variable we use the subscript , and for the two predictors we use the subscripts and . The order of the predictors doesn’t matter but we use the subscripts to keep track of which is which. In our example, is SES and is ATL.

As with the numerical examples, the values below the diagonal mirror the values above the diagonal. So, we really just need the three correlations shown in the matrix below.

The three correlations are interpreted as follows:

* - the correlation between the outcome () and the first predictor ().
* - the correlation between the outcome () and the second predictor ().
* - the correlation between the two predictors.

**If you have questions about how scatter plots and correlations can be presented in matrix format, please write them down now and share them class.**

## 3.3 The two-predictor model

In the ECLS example, we can think of Kindergarteners’ Math Achievement as the outcome variable, with SES and Approaches to Learning as potential predictors / explanatory variables. The multiple regression model for this example can be written as

where

* denotes the predicted Math Achievement
* SES and ATL (it doesn’t matter which predictor we denote as or )
* and are the regression slopes
* The intercept is denoted by (rather than ).

Just like simple regression, the residual for [Equation 3.5](#eq-yhat-3) is defined as and the model can be equivalently written as . Also, remember that you can write out the model using the variable names in place of and if that helps keep track of all the notation. For example,

As mentioned in [Chapter 2](#sec-chap-2), feel free to use whatever notation works best for you.

You might be wondering, what is the added value of multiple regression compared to the correlation co-efficients reported in the previous section? Well, correlations only consider two-variables-at-a-time. Multiple regression let’s us further consider how the predictors work together to explain variation in the outcome, and to consider the relationship between each predictor and the outcome while holding the other predictors constant. In the context of our example, multiple regression let’s us address the following questions:

* How much of variation in Math Achievement do both predictors explain together?
* What is the relationship between Math Achievement and ATL if we hold SES constant?
* Similarly, what is the relationship between Math Achievement and SES if we hold ATL constant?

Notice that this is different from simple regression – simple regression was just a repackaging of correlation, but multiple regression is something new.

## 3.4 OLS with two predictors

We can estimate the parameters of the two-predictor regression model in [Equation 3.5](#eq-yhat-3) model using same approach as for simple regression. We do this by choosing the values of that minimize

Solving the minimization problem (setting derivatives to zero) leads to the following equations for the regression coefficients. Remember, the subscript denotes the first predictor and the subscript denotes the second predictor – see [Section 3.2](#sec-ecls-3) for notation. Also note that represents standard deviations.

As promised, these equations are more complicated than for simple regression :) The next section addresses the interpretation of the regression coefficients.

## 3.5 Interpreting the coefficients

An important part of using multiple regression is getting the correct interpretation of the regression coefficients. The basic interpretation is that the slope for SES represents how much predicted Math Achievement changes for a one unit increase of SES, *while holding ATL constant.* (The same interpretation holds when switching the predictors.) The important difference with simple regression is the “holding the other predictor constant” part, so let’s dig into it.

### 3.5.1 “Holding the other predictor constant”

Let’s start with the regression model for the predicted values:

If we increase by one unit and hold constant, we get new predicted value (denoted with an asterisk):

The difference between and is how much the predicted value changes for a one unit increase in SES, while holding ATL constant:

In words: the multiple regression coefficient tells us how the much the predicted value changes for a one unit increase in the predictor, while holding the other predictor(s) constant.

This why we interpret the regression coefficients in multiple regression differently than simple regression. In simple regression, the slope is just a re-scaled version of the correlation. In multiple regression, the slope of each predictor is interpreted in terms of the “effect” of that predictor, while holding the other predictor(s) constant. This is sometimes referred to as “ceteris paribus,” which is Latin for “with other conditions remaining the same.” So, we could say that multiple regression is a statistical way of making ceteris paribus arguments.

Note that the interpretation of the regression intercept is basically the same as for simple regression: it is the value of when and (i.e., still not very interesting).

### 3.5.2 “Controlling for the other predictor”

Another interpretation of the regression coefficients is in terms of the equations for and presented in [Section 3.4](#sec-ols-3). For example, the equation for is

This is the same equation as from [Section 3.4](#sec-ols-3), but the correlation between the predictors is shown in red. Note that if the predictors are uncorrelated (i.e., $\color{red}{r^2\_{12}}$ = 0) then

which is just the regression coefficient from simple regression ([Section 2.3](#sec-ols-2)).

In general, the formulas for the regression slopes in the two-predictor model are more complicated because they “control for” or “account for” the relationship between the predictors. In simple regression, we only had one predictor, so we didn’t need to account for how the predictors were related to each other.

The equations for the regression coefficients show that, if the predictors are uncorrelated, then doing a multiple regression is just the same thing as doing simple regression multiple times. However, most of the time our predictors will be correlated, and multiple regression “controls for” the relationship between the predictors when examining the relationship between each predictor and the outcome.

### 3.5.3 The ECLS example

Below, the R output from the ECLS example is reported. **Please provide a written explanation of the regression coefficients for SES and ATL. If you have questions about how to interpret the coefficients, please also note them now and be prepared to share them in class. Note that this question is not asking about whether the coefficients are significant – it is asking what the numbers in the first column of the Coefficients table mean.**

# Run the regression model and print output  
mod1 <- lm(c1rmscal ~ wksesl + t1learn)  
summary(mod1)

Call:  
lm(formula = c1rmscal ~ wksesl + t1learn)  
  
Residuals:  
 Min 1Q Median 3Q Max   
-14.101 -4.034 -1.075 3.289 29.543   
  
Coefficients:  
 Estimate Std. Error t value Pr(>|t|)   
(Intercept) -6.05016 2.90027 -2.086 0.038 \*   
wksesl 0.35125 0.05633 6.235 1.94e-09 \*\*\*  
t1learn 3.52125 0.67390 5.225 3.70e-07 \*\*\*  
---  
Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
  
Residual standard error: 6.164 on 247 degrees of freedom  
Multiple R-squared: 0.2726, Adjusted R-squared: 0.2668   
F-statistic: 46.29 on 2 and 247 DF, p-value: < 2.2e-16

### 3.5.4 Standardized coefficients

One question that arises in the interpretation of the example is the relative contribution of the two predictors to Kindergartener’s Math Achievement. In particular, the regression coefficient for ATL is 10 times larger than the regression coefficient for SES – does this mean that ATL is 10 times more important than SES?

The short answer is, “no.” ATL is on a scale of 1-4 whereas SES ranges from 30-72. In order to make the regression coefficients more comparable, we can standardize the variables so that they have the same variance. Many researchers go a step further and standardize all of the variables to be z-scores with M = 0 and SD = 1. The resulting regression coefficients are often called -coefficients or -weights ( is pronounced “beta”).

The -weights are related to the regular regression coefficients from [Section 3.4](#sec-ols-3):

A similar expression holds for .

Note that the lm function in R does not provide an option to report standardized output. So, if you want to get the -coefficients in R, it’s easiest to just standardized the variables first and then do the regression with the standardized variables.

Regardless of how you compute them, the interpretation of the -coefficients is in terms of the standard deviation units of both the variable and the variable – e.g., increasing by one standard deviation changes by standard deviations (holding the other predictors constant).

# Unlike other software, R doesn't have a convenience functions for beta coefficients.   
z\_example\_data <- as.data.frame(scale(example\_data))  
z\_mod <- lm(Math ~ SES + ATL, data = z\_example\_data)  
summary(z\_mod)

Call:  
lm(formula = Math ~ SES + ATL, data = z\_example\_data)  
  
Residuals:  
 Min 1Q Median 3Q Max   
-1.9590 -0.5604 -0.1493 0.4569 4.1043   
  
Coefficients:  
 Estimate Std. Error t value Pr(>|t|)   
(Intercept) 1.337e-15 5.416e-02 0.000 1   
SES 3.533e-01 5.666e-02 6.235 1.94e-09 \*\*\*  
ATL 2.961e-01 5.666e-02 5.225 3.70e-07 \*\*\*  
---  
Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
  
Residual standard error: 0.8563 on 247 degrees of freedom  
Multiple R-squared: 0.2726, Adjusted R-squared: 0.2668   
F-statistic: 46.29 on 2 and 247 DF, p-value: < 2.2e-16

We should be careful when using beta-coefficients to “ease” the comparison of predictors. In the context of our example, we might wonder whether the overall cost of raising a child’s Approaches to Learning by 1 SD is comparable to the overall cost of raising their family’s SES by 1 SD. In general, putting variables on the same scale is only a superficial way of making comparisons among their regression coefficients.

**Please write down an interpretation of the of beta (standardized) regression coefficients in the above output. Your interpretation should include reference to the fact that the variables have been standardized. Based on this analysis, do you think one predictor is more important than the other? Why or why not? Please be prepared to share your interpretations / questions in class!**

## 3.6 (Multiple) R-squared

R-squared in multiple regression has the same general formula and interpretation as in simple regression. The formula is

and it is interpreted as the proportion of variance in the outcome variable that is “associated with” or “explained by” its linear relationship with the predictor variables.

As discussed below, we can also say a bit more about R-squared in multiple regression.

### 3.6.1 Relation with simple regression

Like the regression coefficients in [Section 3.4](#sec-ols-3), the equation for R-squared can also be written in terms of the correlations among the three variables:

If the correlation between the predictors is zero, then this equation simplifies to

In words: When the predictors are uncorrelated, their total contribution to variance explained is just the sum of their individual contributions.

However, when the predictors are correlated, either positively or negatively, it can be show that

In other words: correlated predictors jointly explain less variance than if we added the contributions of each predictor considered separately. Intuitively, this is because correlated predictors share some variation with each other. If we considered the predictors one at a time, we double-count their shared variation.

The interpretation of R-squared for one versus two predictors can be explained in terms of the following Venn diagram.

|  |
| --- |
| Figure 3.3: Shared Variance Among , , and . |

In the diagram, the circles represent the variance of each variable and the overlap between circles represents their shared variance (i.e., the R-squared for each pair of variables). When we conduct a multiple regression, the variance in the outcome explained by both predictors is equal to the sum of the areas A + B + C. If we instead conduct two simple regressions and then add up the R-squared values, we would double count the area labelled “B”.

The Venn diagram in [Figure 3.3](#fig-venn-diagram) is also useful for understanding other aspects of multiple regression. **In the lesson we will discuss the following questions. Please write down your answers now so you are prepared to contribute to the discussion:**

* **Which area represents the correlation between the predictors?**
* **Which areas represent the regression coefficients from multiple regression?**
* **Which areas represent the regression coefficients from simple regression?**

### 3.6.2 Adjusted R-squared

The sample R-squared is an upwardly biased estimate of the population R-squared. The adjusted R-squared corrects this bias. This section explains the main ideas.

The bias of R-squared is illustrated in the figure below. In the example, we are considering simple regression (one predictor), and we assume that the population correlation between the predictor and the outcome is zero (i.e., ).

|  |
| --- |
| Figure 3.4: Sampling Distribution of and when $ ho = 0$. |

In the left panel, we can see that “un-squared” correlation, , has a sampling distribution that is centered at the true value . This means that is an unbiased estimate of .

But in the right panel, we can see that the sampling distribution of the squared correlation, , must have a mean greater than zero. This is because all of the sample-to-sample deviations in left panel are now positive (because they have been squared). Since the average value of is greater than 0, is an upwardly biased estimate of .

The adjusted R-squared corrects this bias. The formula for the adjustment is:

where is the number of predictors in the model.

The formula contains two main terms, the proportion of residual variance, , and the adjustment factor (the ratio of to ). We can understand how the adjustment works by considering these two terms.

First, it can be seen that the adjustment factor is larger when the number of predictors, , is large relative to the sample size, . So, roughly speaking, the adjustment will be more severe when there are a lot of predictors in the model relative to the sample size.

Second, it can also be seen that the adjustment proportional to . This means that the adjustment is more severe if the model explains less variance in the outcome. For example, if and the adjustment factor is , then adjusted . In this case the adjustment is a decrease of 1% of variance explained. But if we start off explaining less variance, say and use the same adjustment factor, then adjusted . Now the adjustment is a decrease of 9% variance explained, even though we didn’t change the adjustment factor.

In summary, the overall interpretation of adusted R-squared is as follows: the adjustment will be larger when there are lots of predictors in the model but they don’t explain much variance in the outcome. This situation is sometimes called “overfitting” the data, so we can think of adjusted R-squared as a correction for overfitting.

There is no established standard for when you should reported R-squared or adjusted R-squared. I recommend that you report both whenever they would would lead to different substantive conclusions. We can discuss this more in class.

### 3.6.3 The ECLS example

As shown in [Section 3.5.4](#sec-beta-3), the R-squared for the ECLS example is equal to .2726 and the adjusted R-squared is equal to .2668. **Please write down your interpretation of these value and be prepared to share your answer in class.**

## 3.7 Inference

There isn’t really any thing new that about inference with multiple regression, except the formula for the standard errors (see (**fox2016?**) chap.6). We present the formulas for an abribrary number of predictors, denoted .

### 3.7.1 Inference for the coefficients

In multiple regression

In this formula, denotes the number of predictors and is the R-squared that results from regressing predictor on the other predictors (without the variable).

Notice that the first part of the standard error (before the “”) is the same as simple regression (see [Section 2.7](#sec-inference-2)). The last part, which includes , is different and we talk about it more below.

The standard errors can be used to construct t-tests and confidence intervals using the same approach as for simple regression (see [Section 2.7](#sec-inference-2)). The degrees of freedom for the t-distribution is . This formula for the degrees of freedom applies to simple regression too, where .

### 3.7.2 Precision of

We can use [Equation 3.6](#eq-se-3) to understand the factors that influence the size of the standard errors of the regression coefficients. Recall that standard errors describe the sample-to-sample variability of a statistic. If there is a lot sample-to-sample variability, the statistic is said to be imprecise. [Equation 3.6](#eq-se-3) shows us what factors make more or less precise.

* The standard errors *decrease* with
  + The sample size,
  + The proportion of variance in the outcome explained by the predictors,
* The standard errors *increase* with
  + The number of predictors,
  + The proportion of variance in the predictor that is explained by the other predictors,

So, large sample sizes and a large proportion of variance explained lead to precise estimates of the regression coefficients. On the other hand, including many predictors that are highly correlated with each other leads to less precision. In particular, the situation where approaches the value of is called *multicollinearity*. We will talk about multicollinearity in more detail in [Chapter 5](#sec-chap-5).

### 3.7.3 Inference for R-squared

The R-squared statistic in multiple regression tells us how much variation in the outcome is explained by all of the predictors together. If the predictors do not explain any variation, then the population R-squared is equal to zero.

Notice that implies (in the population). So, testing the significance of R-squared is equivalent to testing whether any of the regression parameters are non-zero. When we addressed ANOVA last semester, we called this the omnibus hypothesis. But in regression analysis, it is usually just referred to as a test of R-squared.

The null hypothesis can be tested using the statistic

which has an F-distribution on and degrees of freedom when the null hypothesis is true.

### 3.7.4 The ECLS example

The R output for the ECLS example is presented (again) below. **Please write down your conclusions about the statistical significance of the predictors and the R-squared statistic, and be prepared to share your answer in class. Please also write down the factors that affect the precision of the regression coefficients. This would be a good opportunity to practice APA formatting.**

summary(mod1)

Call:  
lm(formula = c1rmscal ~ wksesl + t1learn)  
  
Residuals:  
 Min 1Q Median 3Q Max   
-14.101 -4.034 -1.075 3.289 29.543   
  
Coefficients:  
 Estimate Std. Error t value Pr(>|t|)   
(Intercept) -6.05016 2.90027 -2.086 0.038 \*   
wksesl 0.35125 0.05633 6.235 1.94e-09 \*\*\*  
t1learn 3.52125 0.67390 5.225 3.70e-07 \*\*\*  
---  
Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
  
Residual standard error: 6.164 on 247 degrees of freedom  
Multiple R-squared: 0.2726, Adjusted R-squared: 0.2668   
F-statistic: 46.29 on 2 and 247 DF, p-value: < 2.2e-16

## 3.8 Workbook

This section collects the questions asked in this chapter. The lesson for this chapter will focus on discussing these questions and then working on the exercises in [Section 3.9](#sec-exercises-3). The lesson will **not** be a lecture that reviews all of the material in the chapter! So, if you haven’t written down / thought about the answers to these questions before class, the lesson will not be very useful for you. Please engage with each question by writing down one or more answers, asking clarifying questions about related material, posing follow up questions, etc.

[Section 3.2](#sec-ecls-3)

If you have questions about the interpretation of a correlation matrix (below) or pairwise plots (see [Section 3.2](#sec-ecls-3)), please write them down now and share them class.

Numerical output for the ECLS example:

cor(example\_data)

Math SES ATL  
Math 1.0000000 0.4384619 0.3977048  
SES 0.4384619 1.0000000 0.2877015  
ATL 0.3977048 0.2877015 1.0000000

Mathematical notation for formulas

[Section 3.5](#sec-interpretation-3)

Below, the R output from the ECLS example is reported. Please provide a written explanation of the regression coefficients for SES and ATL. If you have questions about how to interpret the coefficients, please also note them now and be prepared to share them in class. Note that this question is not asking about whether the coefficients are significant – it is asking what the numbers in the first column of the Coefficients table mean.

# Run the regression model and print output  
mod1 <- lm(c1rmscal ~ wksesl + t1learn)  
summary(mod1)

Call:  
lm(formula = c1rmscal ~ wksesl + t1learn)  
  
Residuals:  
 Min 1Q Median 3Q Max   
-14.101 -4.034 -1.075 3.289 29.543   
  
Coefficients:  
 Estimate Std. Error t value Pr(>|t|)   
(Intercept) -6.05016 2.90027 -2.086 0.038 \*   
wksesl 0.35125 0.05633 6.235 1.94e-09 \*\*\*  
t1learn 3.52125 0.67390 5.225 3.70e-07 \*\*\*  
---  
Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
  
Residual standard error: 6.164 on 247 degrees of freedom  
Multiple R-squared: 0.2726, Adjusted R-squared: 0.2668   
F-statistic: 46.29 on 2 and 247 DF, p-value: < 2.2e-16

[Section 3.5.4](#sec-beta-3)

Please write down an interpretation of the of beta (standardized) regression coefficients in the output below. Your interpretation should include reference to the fact that the variables have been standardized. Based on this analysis, do you think one predictor is more important than the other? Why or why not?

# Unlike other software, R doesn't have a convenience functions for beta coefficients.   
z\_example\_data <- as.data.frame(scale(example\_data))  
z\_mod <- lm(Math ~ SES + ATL, data = z\_example\_data)  
summary(z\_mod)

Call:  
lm(formula = Math ~ SES + ATL, data = z\_example\_data)  
  
Residuals:  
 Min 1Q Median 3Q Max   
-1.9590 -0.5604 -0.1493 0.4569 4.1043   
  
Coefficients:  
 Estimate Std. Error t value Pr(>|t|)   
(Intercept) 1.337e-15 5.416e-02 0.000 1   
SES 3.533e-01 5.666e-02 6.235 1.94e-09 \*\*\*  
ATL 2.961e-01 5.666e-02 5.225 3.70e-07 \*\*\*  
---  
Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
  
Residual standard error: 0.8563 on 247 degrees of freedom  
Multiple R-squared: 0.2726, Adjusted R-squared: 0.2668   
F-statistic: 46.29 on 2 and 247 DF, p-value: < 2.2e-16

[Section 3.6](#sec-rsquared-3)

The Venn diagram below is useful for understanding multiple regression. In the lesson we will discuss the following questions. Please write down your answers now so you are prepared to contribute to the discussion:

* Which area represents the correlation between the predictors?
* Which areas represent the R-squared from multiple regression?
* Which areas represent the R-squared from simple regression?
* Which areas represent the regression coefficients from multiple regression?
* Which areas represent the regression coefficients from simple regression?

|  |
| --- |
| Shared Variance Among , , and . |

* Last question: The R-squared for the ECLS example is equal to .2726 and the adjusted R-squared is equal to .2668. Please write down your interpretation of these value and be prepared to share your answer in class.

[Section 3.7](#sec-inference-3)

The R output for the ECLS example is presented (again) below. Please write down your conclusions about the statistical significance of the predictors and the R-squared statistic, and be prepared to share your answer in class. This would be a good opportunity to practice APA formatting. Please also write down the factors that negatively affect the precision of the regression coefficients and address whether you think they are problematic for the example.

summary(mod1)

Call:  
lm(formula = c1rmscal ~ wksesl + t1learn)  
  
Residuals:  
 Min 1Q Median 3Q Max   
-14.101 -4.034 -1.075 3.289 29.543   
  
Coefficients:  
 Estimate Std. Error t value Pr(>|t|)   
(Intercept) -6.05016 2.90027 -2.086 0.038 \*   
wksesl 0.35125 0.05633 6.235 1.94e-09 \*\*\*  
t1learn 3.52125 0.67390 5.225 3.70e-07 \*\*\*  
---  
Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
  
Residual standard error: 6.164 on 247 degrees of freedom  
Multiple R-squared: 0.2726, Adjusted R-squared: 0.2668   
F-statistic: 46.29 on 2 and 247 DF, p-value: < 2.2e-16

## 3.9 Exercises

These exercises collect all of the R input used in this chapter into a single step-by-step analysis. It explains how the R input works, and provides some additional exercises. We will go through this material in class together, so you don’t need to work on it before class (but you can if you want.)

Before staring this section, you may find it useful to scroll to the top of the page, click on the “</> Code” menu, and select “Show All Code.”

### 3.9.1 The ECLS250 data

Let’s start by getting our example data loaded into R.

Make sure to download the file ECLS250.RData from Canvas and then double click the file to open it

load("ECLS250.RData") # load new example  
attach(ecls) # attach   
  
# knitr and kable are just used to print nicely -- you can just use head(ecls[, 1:5])   
knitr::kable(head(ecls[, 1:5]))

| caseid | gender | race | c1rrscal | c1rrttsco |
| --- | --- | --- | --- | --- |
| 960 | 2 | 1 | 28 | 58 |
| 113 | 1 | 8 | 14 | 39 |
| 1828 | 1 | 1 | 22 | 50 |
| 1693 | 1 | 1 | 21 | 50 |
| 643 | 2 | 1 | 14 | 39 |
| 772 | 1 | 1 | 21 | 49 |

The naming conventions for these data are bit challenging.

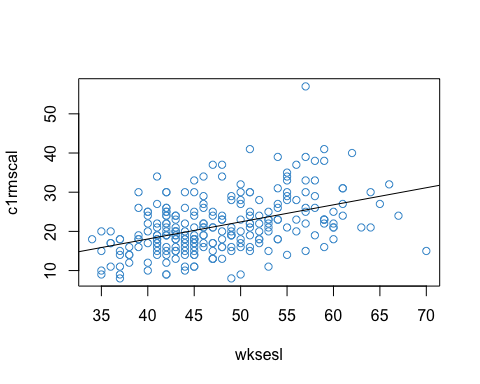
* Variable names begin with c, p, or t depending on whether the respondent was the child, parent, or teacher. Variables that start with wk were created by the ECLS using other data sources available in during the kindergarten year of the study.
* The time points (1-4 denoting fall and spring of K and Gr 1) appear as the second character.
* The rest of the name describes the variable.

The variables we will use for this illustration are:

* c1rmscal: Child’s score on a math assessment, in first semester of Kindergarten . The scores can be interpreted as number of correct responses out of a total of approximately 60 math exam questions.
* wksesl: An SES composite of household factors (e.g., parental education, household income) ranging from 30-72.
* t1learn: Approaches to Learning Scale (ATLS), teacher reported in first semester of kindergarten. This scale measures behaviors that affect the ease with which children can benefit from the learning environment. It includes six items that rate the child’s attentiveness, task persistence, eagerness to learn, learning independence, flexibility, and organization. The items have 4 response categories (1-4), so that higher values represent more positive responses, and the scale is an unweighted average the six items.

To get started lets produce the simple regression of Math with SES. This is another look at the relationship between Academic Achievement and SES that we discussed in Chapter [Chapter 2](#sec-chap-2)). If you do not feel comfortable running this analysis or interpreting the output, take another look at [Section 2.11](#sec-exercises-2).

plot(x = wksesl,   
 y = c1rmscal,   
 col = "#4B9CD3")  
  
mod <- lm(c1rmscal ~ wksesl)  
abline(mod)



summary(mod)

Call:  
lm(formula = c1rmscal ~ wksesl)  
  
Residuals:  
 Min 1Q Median 3Q Max   
-16.1314 -4.3549 -0.8486 3.6775 31.5358   
  
Coefficients:  
 Estimate Std. Error t value Pr(>|t|)   
(Intercept) 0.61595 2.73925 0.225 0.822   
wksesl 0.43594 0.05674 7.683 3.61e-13 \*\*\*  
---  
Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
  
Residual standard error: 6.482 on 248 degrees of freedom  
Multiple R-squared: 0.1922, Adjusted R-squared: 0.189   
F-statistic: 59.03 on 1 and 248 DF, p-value: 3.612e-13

cor(wksesl, c1rmscal)

[1] 0.4384619

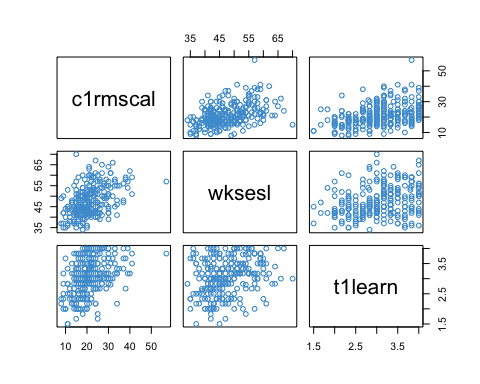
### 3.9.2 Multiple regression with lm

First, let’s tale a look at the “zero-order” relationship among the three variables. This type of descriptive, two-way analysis is a good way to get familiar with your data before getting into multiple regression. We can see that the variables are all moderately correlated and their relationships appear reasonably linear.

# Use cbind to create a data.frame with just the 3 variables we want to examine  
data <- cbind(c1rmscal, wksesl, t1learn)  
  
# Correlations  
cor(data)

c1rmscal wksesl t1learn  
c1rmscal 1.0000000 0.4384619 0.3977048  
wksesl 0.4384619 1.0000000 0.2877015  
t1learn 0.3977048 0.2877015 1.0000000

# Scatterplots  
pairs(data, col = "#4B9CD3")



In terms of input, multiple regression with lm is similar to simple regression. The only difference is the model formula. To include more predictors in a formula, just include them on the right hand side, separated by at + sign.

* e.g, Y ~ Χ1 + Χ2

For our example, let’s consider the regression of math achievement on SES and Approaches to Learning. We’ll save our result as mod1 which is short for “model one”.

mod1 <- lm(c1rmscal ~ wksesl + t1learn)  
summary(mod1)

Call:  
lm(formula = c1rmscal ~ wksesl + t1learn)  
  
Residuals:  
 Min 1Q Median 3Q Max   
-14.101 -4.034 -1.075 3.289 29.543   
  
Coefficients:  
 Estimate Std. Error t value Pr(>|t|)   
(Intercept) -6.05016 2.90027 -2.086 0.038 \*   
wksesl 0.35125 0.05633 6.235 1.94e-09 \*\*\*  
t1learn 3.52125 0.67390 5.225 3.70e-07 \*\*\*  
---  
Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
  
Residual standard error: 6.164 on 247 degrees of freedom  
Multiple R-squared: 0.2726, Adjusted R-squared: 0.2668   
F-statistic: 46.29 on 2 and 247 DF, p-value: < 2.2e-16

We can see from the output that regression coefficient for t1learn is about 3.5. This means that, as the predictor increases by a single unit, children’s predicted math scores increase by 3.5 points (out of 60), after controlling for the SES. You should be able to provide a similar interpretation of the regression coefficient for wksesl. Together, both predictors accounted for about 27% of the variation in students’ math scores. In education, this would be considered a pretty strong relationship.

We will talk about the statistical tests later on. For now let’s consider the relationship with simple regression.

### 3.9.3 Relations between simple and multiple regression

First let’s consider how the two simple regression compare to the multiple regression with two variables. Here is the relevant output:

# Compare the multiple regression output to the simple regressions  
mod2a <- lm(c1rmscal ~ wksesl)  
summary(mod2a)

Call:  
lm(formula = c1rmscal ~ wksesl)  
  
Residuals:  
 Min 1Q Median 3Q Max   
-16.1314 -4.3549 -0.8486 3.6775 31.5358   
  
Coefficients:  
 Estimate Std. Error t value Pr(>|t|)   
(Intercept) 0.61595 2.73925 0.225 0.822   
wksesl 0.43594 0.05674 7.683 3.61e-13 \*\*\*  
---  
Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
  
Residual standard error: 6.482 on 248 degrees of freedom  
Multiple R-squared: 0.1922, Adjusted R-squared: 0.189   
F-statistic: 59.03 on 1 and 248 DF, p-value: 3.612e-13

mod2b <- lm(c1rmscal ~ t1learn)  
summary(mod2b)

Call:  
lm(formula = c1rmscal ~ t1learn)  
  
Residuals:  
 Min 1Q Median 3Q Max   
-14.399 -4.211 -0.997 3.770 31.844   
  
Coefficients:  
 Estimate Std. Error t value Pr(>|t|)   
(Intercept) 7.0394 2.1485 3.276 0.0012 \*\*   
t1learn 4.7301 0.6929 6.826 6.66e-11 \*\*\*  
---  
Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
  
Residual standard error: 6.618 on 248 degrees of freedom  
Multiple R-squared: 0.1582, Adjusted R-squared: 0.1548   
F-statistic: 46.6 on 1 and 248 DF, p-value: 6.665e-11

The important things to note here are

* The regression coefficients from the simple models ( and ) are larger than the regression coefficients from the two-predictor model. Can you explain why? (Hint: see Section [Section 3.5](#sec-interpretation-3).
* The R-squared values in the two simple models (.192 + .158 = .350) add up to more than the R-squared in the two-predictor model (.273). Again, take a moment to think about why before reading on. (Hint: see Section [Section 3.6](#sec-rsquared-3).)

### 3.9.4 Inference with 2 predictors

Let’s move on now to consider the statistical tests and confidence intervals provided with the lm summary output.

For regression with more than one predictor, both the t-tests and F-tests have a very similar construction and interpretation as with simple regression. The main differences are the formulas, not so much the interpretations of the procedures. Some differences:

* The degrees of freedom for both tests now involve , the number of predictors.
* The standard error of the b-weight is more complicated, because it involves the inter-correlation among the predictors.

We can see for mod1 that both b-weights are significant at the .05 level, and so is the R-square. As mentioned previously, it is not usual to interpret or report results on the regression intercept unless you have a special reason to do so (e.g., see the next chapter).

# Revisting the output of mod1  
summary(mod1)

Call:  
lm(formula = c1rmscal ~ wksesl + t1learn)  
  
Residuals:  
 Min 1Q Median 3Q Max   
-14.101 -4.034 -1.075 3.289 29.543   
  
Coefficients:  
 Estimate Std. Error t value Pr(>|t|)   
(Intercept) -6.05016 2.90027 -2.086 0.038 \*   
wksesl 0.35125 0.05633 6.235 1.94e-09 \*\*\*  
t1learn 3.52125 0.67390 5.225 3.70e-07 \*\*\*  
---  
Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
  
Residual standard error: 6.164 on 247 degrees of freedom  
Multiple R-squared: 0.2726, Adjusted R-squared: 0.2668   
F-statistic: 46.29 on 2 and 247 DF, p-value: < 2.2e-16

### 3.9.5 APA reporting of results

This section shows how we might write out the results of our regression using APA format. When we have a regression model with many predictors, or are comparing among different models, it is more usual to put all the relevant statistics in a table rather than writing them out one by one. We will see how to do that later on in the course. For more info on APA format, see the APA publications manual: [(https://www.apastyle.org/manual)](https://www.apastyle.org/manual).

* The regression of Math Achievement on SES was positive and statistically significant at the .05 level ().
* The regression of Math Achievement on Approaches to Learning was also positive and statistically significant at the .05 level ().
* Together both predictors accounted for about 27% of the variation in Math Achievement (, ), which was also statistically significant at the .05 level ().

# 4. Categorical predictors

So far we have considered examples in which we regress a continuous outcome variable (e.g., Math Achievement) on one or more continuous predictors (SES, Approaches to Learning). In this chapter we consider how regression can be used with categorical predictors.

In an experimental context, the canonical example of a categorical predictor is treatment status (e.g., 1 = treatment group, 0 = control group). Examples of other categorical predictors commonly used in education research include:

* Geographical region / school district (Orange, CH-C, Wake, …)
* Type of school (public, private, charter, religious)
* Which classroom, teacher, or school a student was assigned to
* Gender (if recorded as categorical)
* Race / ethnicity (if recorded as categorical)
* Free / reduced price lunch status
* English language learner status
* Individualized learning plan status
* …

In regression, we deal with these types of predictors using an approach called *contrast coding* (also called *effect coding* or *dummy coding*). In particular, we will address three types of contrast coding:

* The special case of a single binary predictor
* Reference-group coding (called “treatment” contrasts in R)
* Deviation coding (called “sum” contrasts in R)

There are other types of contrast coding out there. We do not attempt an exhaustive review. However, the techniques we use in this chapter can be used with other types of contrast coding.

Along the way we will see that regression with a categorical predictor encompasses the independent samples t-tests of means and one-way ANOVA, both of which we discussed last semester. In the next chapter we will address how to combine categorical and continuous predictors in the same model.

## 4.1 Focus on interpretation

Categorical predictors can be challenging to understand because, depending on the contrast coding used, the model results can appear quite different.

For example, the two models below use the same data and the same variables (Math Achievement regressed on Urbanicity), but their regression coefficients have different values. Why? Because the models used different contrast coding for Urbanicity. In the first model, Urban and Suburban students are compared to Rural students. In the second model, Rural and Suburban students are compared to the unweighted average across all three groups. We talk in more detail about these interpretations in the following sections. For now, the main thing to notice is just that “different contrast coding = different results”.

load("NELS.RData")  
attach(NELS)  
  
# run model with default contrast (treatment / dummy coding)  
egA <- lm(achrdg08 ~ urban)  
  
# change to sum / deviation contrasts and run again  
contrasts(urban) <- contr.sum(n = 3)  
colnames(contrasts(urban)) <- c("Rural", "Suburban")  
egB <- lm(achrdg08 ~ urban)  
  
# print  
summary(egA)

Call:  
lm(formula = achrdg08 ~ urban)  
  
Residuals:  
 Min 1Q Median 3Q Max   
-22.3002 -6.1620 0.2098 6.7948 15.5573   
  
Coefficients:  
 Estimate Std. Error t value Pr(>|t|)   
(Intercept) 54.9927 0.6885 79.876 < 2e-16 \*\*\*  
urbanSuburban 0.6675 0.9117 0.732 0.46439   
urbanUrban 3.1275 1.0480 2.984 0.00298 \*\*   
---  
Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
  
Residual standard error: 8.763 on 497 degrees of freedom  
Multiple R-squared: 0.01904, Adjusted R-squared: 0.0151   
F-statistic: 4.824 on 2 and 497 DF, p-value: 0.008411

summary(egB)

Call:  
lm(formula = achrdg08 ~ urban)  
  
Residuals:  
 Min 1Q Median 3Q Max   
-22.3002 -6.1620 0.2098 6.7948 15.5573   
  
Coefficients:  
 Estimate Std. Error t value Pr(>|t|)   
(Intercept) 56.2577 0.4021 139.897 <2e-16 \*\*\*  
urbanRural -1.2650 0.5654 -2.237 0.0257 \*   
urbanSuburban -0.5975 0.5299 -1.128 0.2600   
---  
Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
  
Residual standard error: 8.763 on 497 degrees of freedom  
Multiple R-squared: 0.01904, Adjusted R-squared: 0.0151   
F-statistic: 4.824 on 2 and 497 DF, p-value: 0.008411

# Reset to original contrasts  
contrasts(urban) <- contr.treatment(n = 3)  
colnames(contrasts(urban)) <- c("Suburban", "Urban")

Note that the lm output doesn’t explicitly tell us what kind of coding was used. So, we need to know what is going on “under the hood” in order to interpret the output correctly. Also note that the while the *Coefficients* tables of the two models are different, they both explain the same amount of variance in the outcome. So, we could say that both models “fit” the data equally well, but they provide different interpretations of the relationship between Math Achievement and Urbanicity.

The choice of which contrast to use is up to the researcher. This choice should reflect the research questions you want to address. For example, if we wanted to know how Urban and Suburban students differ from Rural students, that is what the first output shows us. If we wanted to know how Suburban and Rural students differ from the overall average, that is what the second output shows us. As we will discuss below, we can’t make all possible comparisons among groups in a single model, so we have to make some compromises when choosing contrasts. Understanding how to align our research questions with the choice of contrasts is a big part of what this chapter is about!

## 4.2 Data and social constructs

Before getting into the math, let’s consider some conceptual points.

First, some terminology. *Binary* means that a variable can take on only two values: 1 and 0. If a variable takes on two values but these are represented with other numbers (e.g., 1 and 2) or with non-numeric values (“male”, “female”), it is called *dichotomous* rather than binary. Otherwise stated, a binary variable is a dichotomous variable whose values are 1 and 0. The term *categorical* is used to describe variables with two or more categories – it includes binary and dichotomous variables, as well as variables with more categories.

Note that encoding a variable as dichotomous does not imply that the underlying social construct is dichotomous. For example, we can encode educational attainment as a dichotomous variable indicating whether or not a person has graduated high school. This does not imply that educational attainment has only two values “irl”, or even that educational attainment is best conceptualized in terms of years of formal education. Nonetheless, for many outcomes of interest it can be meaningful to consider whether individuals have completed high school (e.g., <https://www.ssa.gov/policy/docs/research-summaries/education-earnings.html>).

In general, the way that a variable is encoded in a dataset is not a statement about reality – it reflects a choice made by researchers about how to represent social constructs. When conducting quantitative research, we are often faced with less-than-ideal encodings of social constructs, especially demographic variables. For example, both NELS and ECLS conceptualize gender as dichotomous and use a limited set of mutually exclusive categories for race. These representations are not well aligned with current literature on gender and racial identity. Some recent perspectives on how these issues play into quantitative research are available here: <https://www.sree.org/critical-perspectives>.

Despite the conceptual issues inherent in categorizing social constructs, I would argue that categorical variables often have utility, *especially* in the study of social inequality. Here is an example of why I think gender qua “male/female” is a flawed but important consideration in global education: <https://www.unicef.org/education/girls-education>.

**Please take a moment to write down your thoughts on the tensions that arise when conceptualizing social constructs such as gender or race as categorical, and I will invite you to share you thoughts in class.**

## 4.3 Binary predictors

Let’s start our interpretation of categorical predictors with the simplest case: a single binary predictor.

[Figure 4.1](#fig-reading-on-gender-4) uses the NELS data to illustrate the regression of Reading Achievement in Grade 8 (achrdg08) on a binary encoding of Gender (female = 0, male = 1). There isn’t a lot going on the plot! However, we can see the conditional distributions of Reading Achievement for each value of Gender, and the means of the two groups are indicated.

# Don't read this unless you really like working on graphics :)   
binary\_gender <- (gender == "Male") \* 1   
plot(binary\_gender, achrdg08,   
 col = "#4B9CD3",   
 xlab = "binary gender (Female = 0, Male = 1)",  
 ylab = "Reading (Grade 8)")  
  
means <- tapply(achrdg08, binary\_gender, mean)  
labels <- c(expression(bar(Y)[0]), expression(bar(Y)[1]))  
  
text(x = c(.1, .9), y = means, labels = labels, cex = 1.5)   
text(x = c(0, 1), y = means, labels = c("\_", "\_"), cex = 2)

|  |
| --- |
| Figure 4.1: Reading Achievement on Binary Gender. |

We can represent the relationship between Reading Achievement and Gender using the same simple regression equation from [Section 2.2](#sec-regression-line-2)

and we can still interpret the regression slope in terms of a one-unit increase in . However, since can only take on two values (0 or 1), there are also other interpretations of the regression coefficients. In this section we are interested in the these “more specific” interpretations of regression coefficients when is binary.

The general strategy for approaching the interpretation of regression coefficients with categorical predictors has two steps:

* *Step 1.* Plug the values for into the regression equation to get the predicted values for each group
* *Step 2.* Solve for the coefficients in terms the predicted values.

Looking at [Equation 4.1](#eq-binary-1) we can see that intercept () is equal to the predicted value of Reading Achievement for Females, and [Equation 4.2](#eq-binary-2) shows that the regression slope () is equal to the difference between predicted Reading Achievement for Males and Females.

There is one last detail that is important for interpreting these equations: for a single categorical predictor, the predicted values for each category are just the group means on the variable (see [Section 4.7](#sec-categorical-results-4), which is optional). So, using the notation of [Figure 4.1](#fig-reading-on-gender-4), we can re-write [Equation 4.1](#eq-binary-1) and [Equation 4.2](#eq-binary-2) as

We will use the equivalence between predicted values and group means throughout this chapter. However, it is important to note this equivalence holds only when there is single categorical predictor in the model, and no other predictors. Additional predictors are discussed in [Section 5.2](#sec-binary-continuous-5)

For the example data, regression coefficients are:

# convert "Female / Male" coding to binary  
gender <- NELS$gender  
binary\_gender <- (gender == "Male")\*1  
mod\_binary <- lm(achrdg08 ~ binary\_gender)  
coef(mod\_binary)

(Intercept) binary\_gender   
 56.4678022 -0.9223396

**Please take a moment and write down how these two numbers are related to** [**Figure 4.1**](#fig-reading-on-gender-4)**. In particular, what is equal to, what is equal to, and what is their difference equal to?**

### 4.3.1 Relation with t-tests of means

Simple regression with a binary predictor is equivalent to conducting an independent samples t-test in which the variable (Gender) is the grouping variable and the variable (Reading Achievement) is the outcome. The following output illustrates this.

For the regression model (same as above):

summary(mod\_binary)

Call:  
lm(formula = achrdg08 ~ binary\_gender)  
  
Residuals:  
 Min 1Q Median 3Q Max   
-20.7278 -6.1472 0.3784 6.9765 15.0045   
  
Coefficients:  
 Estimate Std. Error t value Pr(>|t|)   
(Intercept) 56.4678 0.5342 105.703 <2e-16 \*\*\*  
binary\_gender -0.9223 0.7928 -1.163 0.245   
---  
Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
  
Residual standard error: 8.827 on 498 degrees of freedom  
Multiple R-squared: 0.00271, Adjusted R-squared: 0.0007076   
F-statistic: 1.353 on 1 and 498 DF, p-value: 0.2452

For the independent samples t-test (with homogeneity of variance assumed):

t.test(achrdg08 ~ binary\_gender, var.equal = T)

Two Sample t-test  
  
data: achrdg08 by binary\_gender  
t = 1.1633, df = 498, p-value = 0.2452  
alternative hypothesis: true difference in means between group 0 and group 1 is not equal to 0  
95 percent confidence interval:  
 -0.6353793 2.4800586  
sample estimates:  
mean in group 0 mean in group 1   
 56.46780 55.54546

We will go through the comparison between these two outputs in class together. **If you have any questions about the relation between these two sets of output, please note them now and be prepared ask them in class.**

### 4.3.2 Summary

When doing regression with a binary predictor:

* The regression intercept is equal to the mean of the group coded “0”.
* The regression slope is equal to the mean of the group coded “0” subtracted from the mean of the groups coded “1”.
* Testing is equivalent to testing the mean difference
  + i.e., regression with a binary variable is the same as a t-test of means for independent samples

## 4.4 Reference-group coding

Now that we know about regression with a binary predictor, let’s consider how to extend this approach to contrast coding of categorical predictors with categories.

The basic idea is to represent the categories of the predictor in terms of “dummy variables.” Binary coding of a dichotomous predictor is one example of this: We represented a categorical variable with categories using binary predictor.

For predictors with categories, the most common approach to contrast coding is called *reference-group* coding. In R, the approach is called *treatment contrasts* and is the default coding for categorical predictors.

It is called reference-group coding because:

* The researcher chooses a reference-group.
* The intercept is interpreted as the mean of the reference-group.
* The regression slopes are interpreted as the mean differences between the other groups and the reference-group.

Note that reference-group coding is a generalization of binary coding. In the example from [Section 4.3](#sec-binary-predictors-4):

* Females were the reference-group.
* The intercept was equal to the mean Reading Achievement for females.
* The regression coefficient was equal to the mean difference between males and females.

The rest of this section considers how to generalize this approach to more than 2 groups.

### 4.4.1 A hypothetical example

[Figure 4.2](#fig-martital-status1) presents a toy example. The data show the Age and marital status (Mstatus) of 16 hypothetical individuals. Marital status is encoded as

* Single
* Married
* Divorced

knitr::include\_graphics("files/images/marital\_status1.png")

|  |
| --- |
| Figure 4.2: Toy Martital Status Example. |

These 3 categories are represented by two binary variables, denoted and .

* is a binary variable that is equal to 1 when Mstatus is “married”, and equal to 0 otherwise.
* is a binary variable that is equal to 1 when Mstatus is “divorced”, and equal to 0 otherwise.

The binary variables are often called *dummies* or *indicators*. For example, is a dummy or indicator for married respondents.

In reference-group coding, the group that does not have a dummy variable is the reference group. It is also the group that is coded zero on all of the included dummies. In this example, “Single” is the reference group.

We can more compactly write down the contrast coding in the example using a *contrast matrix* .

This notation is an abbreviated version of the data matrix in [Figure 4.2](#fig-martital-status1). It summarizes how the two indicator variables correspond to the different levels of the categorical variable. As we will see in the [Section 4.9](#sec-exercises-4), contrast matrices are useful for programming in R.

### 4.4.2 Interpreting the regression coefficients

Regressing Age on the dummies we have:

In order to interpret the regression coefficients we can use the same two steps as in [Section 4.3](#sec-binary-predictors-4)

* *Step 1.* Plug the values for the variables into the regression equation to get the predicted values for each group
* *Step 2.* Solve for the model parameters in terms the predicted values.

**Using the above equations, please write down an interpretation of the regression parameters for the hypothetical example. (Note: this question is not asking for a numerical answer, it is just asking you to put the above equations into words.)**

### 4.4.3 More than 3 categories

[Figure 4.3](#fig-martital-status2) extends the example by adding another category for Mstatus (“widowed”).

knitr::include\_graphics("files/images/marital\_status2.png")

|  |
| --- |
| Figure 4.3: Toy Martital Status Example, Part 2. |

Here we are interested in the model

**Please work through the following questions and be prepared to share your answers in class**

* **How should be coded so that “single” is the reference-group?**
* **Write down the contrast matrix for the example.**
* **Using the two-step approach illustrated above, write out the interpretation of the regression coefficients.**
* **Let’s imagine that we included an additional dummy variable for “Single” in the model. In this case we would have dummies in the model rather than dummies. Please take a moment to write down what you think will happen (Hint: something goes wrong)**

### 4.4.4 Summary

In reference-group coding with a single categorical variable:

* The reference is group is chosen by the analyst – it is the group that is coded zero on all dummies, or the one that has its indicator omitted from the model.
* The intercept is interpreted as the mean of the reference group.
* The regression coefficients of the dummy variables are interpreted as the difference between the mean of the indicated group and the mean of the reference group.

## 4.5 Deviation coding

In some research scenarios, there is a clear reference group. For example, in an experiment, all treatment groups are can be compared to the control group. But in other cases, it is less clear what the reference group should be. In both of the examples we have considered so far (gender and marital status), the choice of reference group was arbitrary.

When there is not a clear reference group, it can be preferable to use other types of contrast coding than reference-group coding. One such approach is called *deviation coding*. In R, this is called *sum-to-zero constrasts*, or *sum* contrasts for short.

The main difference between deviation coding and reference-group coding is the interpretation of the intercept. In deviation coding, it is no longer the mean of the reference-group, but instead represents the mean of all groups’ predicted values. When the groups have equal sample size, the deviation-coded intercept is also equal to overall mean on .

More specifically, in deviation coding:

* The intercept is equal to the mean of the predicted values for each category i.e.,

* When the groups have equal sample sizes (), the intercept is also equal to the overall mean on the outcome variable:

The regression slopes compare the each group to the intercept. Consequently, when the groups have equal sample sizes, the regression slopes are interpreted as the deviation of each group mean from the overall mean, which is why it is called deviation coding.

When the groups have unequal sample size, the situation is a bit more complicated. In particular, we have to weight the predicted values in [Equation 4.5](#eq-unweighted-mean) by the group sample sizes. This is addressed in [Section 4.5.4](#sec-extra1-4) (optional). To clarify that intercept in deviation coding is not always equal to , we refer to it as an “unweighted mean” of group means / predicted scores.

It is important to note that there are still only regression slopes So, one group gets left out of the analysis, and the researcher has to chose which one. This is a shortcoming of deviation coding, which is addressed in [Section 4.5.5](#sec-extra2-4) (optional).

### 4.5.1 A hypothetical example

The International Development and Early Learning Assessment (IDELA) is an assessment designed to measure young children’s development in literacy, numeracy, social-emotional, and motor domains, in international settings. [Figure 4.4](#fig-idela1) shows the countries in which the IDELA had been used as of 2017 (for more info, see <https://www.savethechildren.net/sites/default/files/libraries/GS_0.pdf>).

knitr::include\_graphics("files/images/idela\_map.png")

|  |
| --- |
| Figure 4.4: IDELA Worldwide Usage, 2017. |

If our goal was to compare countries’ IDELA scores, agreeing on which country should serve as the reference group would politically fraught. Therefore, it would be preferable to avoid the problem of choosing a reference group altogether. In particular, deviation coding let’s us compare each country’s mean IDELA score to the (unweighted) mean over all of the countries.

[Figure 4.5](#fig-idela2) presents a toy example. The data show the IDELA scores and Country for 16 hypothetical individuals. The countries considered in this example are

* Ethiopia
* Vietnam
* Boliva

These 3 countries are represented by two binary variables, denoted and .

* is a indicator for Ethiopia
* is a indicator for Vietnam

knitr::include\_graphics("files/images/idela1.png")

|  |
| --- |
| Figure 4.5: Toy IDELA Example. |

Note that the dummy variables are different than for the case of reference-group coding discussed in [Section 4.4](#sec-reference-group-coding-4). In deviation coding, the dummies always take on values . The same group must receive the code for all dummies. The group with the value is the group that gets left out of the analysis.

**Please take a moment to write down the contrast matrix for the IDELA example.**

### 4.5.2 Interpreting the regression coefficients

Regressing IDELA on the dummies we have:

In order to interpret the regression coefficients, we proceed using the same two steps as in [Section 4.3](#sec-binary-predictors-4) and [Section 4.4](#sec-reference-group-coding-4).

* *Step 1.* Plug the values for into the regression equation to get the predicted values for each group.
* *Step 2.* Solve for the model parameters in terms of the predicted values.

At this point, we want to use the first two lines to substitute in for and in the last line:

In the last line we see that is equal to the (unweighted) mean of the predicted values.

**Using the above equations, please write down an interpretation of the regression parameters for the hypothetical example. (Note: this question is not asking for a numerical answer, it is just asking you to put the above equations into words.)**

### 4.5.3 Summary

In deviation coding with a single categorical variable:

* The intercept is interpreted as the unweighted mean of the groups’ means, which is equal to the overall mean on the variable when the groups have equal sample sizes.
* The regression slopes on the dummy variables are interpreted as the difference between the mean of the indicated group and the unweighted mean of the groups.
* There are still only regression coefficients, so one group gets left out (see extra material for how to get around this).

### 4.5.4 Extra: Deviation coding with unequal sample sizes\*

When groups have unequal sample size, the unweighted mean of the group means is not equial the overall mean of the variable. This is not always a problem. In particular, in the IDELA example, it is reasonable that each country should receive equal weight, even if the size of the samples in each country differed.

However, if you want to compare each groups’ mean to the overall mean on , deviation coding can be adjusted by replacing the dummy-coded value with the ratio of indicated group’s sample size to the omitted group’s sample size. An example for 3 groups is shown below.

You can use the 2-step procedure to show that this coding, called *weighted deviation coding*, results in

Replacing with you can also show that , using the rules of summation algebra.

The punchline: unlike the case for the deviation coding, the intercept in *weighted deviation coding* is always equal to the overall mean on the outcome variance, regardless of the sample sizes of the groups. In R, you can use the package wec for weighted effect coding.

### 4.5.5 Extra: Deviation coding with all groups included\*

Another shortcoming of deviation coding is that it requires leaving one group out of the model. As a work around, one can instead use the following approach. Note that this approach will affect the statistical significance of R-squared, so you should only use it if you aren’t interested in reporting R-squared (or are willing to do some more fiddling around with R).

* Step 1: Center the variable so that .
* Step 2: Compute binary dummy variables (reference-group coding) for all groups,
* Step 3: Regress on the all dummy variables, without the intercept in the model:

Using the two step approach, it is easy to show that the regression coefficients are just the means of the indicated group. Since the overall mean of is zero (because of Step 1), the group means can be interpreted as deviations from the overall mean on .

Using the lm function in R, you can remove the intercept from the model using -1 in the formula syntax:

Y ~ -1 + X1 + ...

As noted, this will affect the statistical significance of R-squared, as reported in summary(lm). So, you shouldn’t use this approach when you want to report R-squared. The next two paragraphs explain why, but the explanation requires some material we won’t discuss until [Chapter 6](#sec-chap-6).

The F-test of R-squared can be computed by comparing two different models, a reference model with no predictors and a focal model with the predictors. The reference model used by lm depends on whether the intercept is included in the focal model. If the focal model has an intercept, the reference model also has an intercept. In the focal model omits the intercept, the reference also omits the intercept.

In the present context, the focal model omits the intercept and use dummies This model has the same R-squared as if we had used the intercept and dummies. However, the F-statistic for the test of R-squared (as reported by summary(lm)) is different for the two approaches, because of the choice of reference model. You can change the reference model, but it takes a bit of fiddling around. If you are interested, just ask in class.

## 4.6 Relation with ANOVA

As mentioned in [Chapter 3](#sec-chap-3), the F-test of R-squared is equivalent to simultaneously testing whether all of the regression slopes are equal to zero. In symbols:

Testing

is equivalent to testing

We have see above that the regression slopes the regression coefficient of dummy variables can be interpreted in terms of group-mean differences. For example, under reference-group coding, we can re-write the null hypothesis above as:

In this notation, we let denote the reference group. If we add $\_C $ to each equality in the null hypothesis, we have

Note that this is the same null hypothesis as one-way ANOVA – i.e., all the group means are equal. Consequently, the F-test of R-squared using a categorical predictor is equivalent to the F-test of the omnibus hypothesis in one-way ANOVA (assuming homoskedasticity). In [Section 4.9](#sec-exercises-4), we show how the two approaches produce the same numerical values.

Although regression with a categorical predictor is mathematically equivalent to one-way ANOVA, you may have noticed that the two approahces are not used in the same way. In ANOVA, we first conduct an F-test of the omnibus hypothesis, and, if the test is significant, we follow-up by comparing the groups using procedures that control for familywise error rate. In regression, we just report the F-test along with the test of the individual contrasts, without any discussion of familywise error rate control. Which approach is correct? Depends who you ask….

## 4.7 More about categorical data\*

Check back later for:

* mean and variance of binary variables
* covariance between binary variables and between binary and continuous
* regression coefficient when X is binary

## 4.8 Workbook

This section collects the questions asked in this chapter. The lesson for this chapter will focus on discussing these questions and then working on the exercises in [Section 4.9](#sec-exercises-4). The lesson will **not** be a lecture that reviews all of the material in the chapter! So, if you haven’t written down / thought about the answers to these questions before class, the lesson will not be very useful for you. Please engage with each question by writing down one or more answers, asking clarifying questions about related material, posing follow up questions, etc.

[Section 4.2](#sec-social-constructs-4)

Please take a moment to write down your thoughts on the tensions that arise when conceptualizing social constructs such as gender or race as categorical, and I will invite you to share you thoughts in class.

[Section 4.3](#sec-binary-predictors-4)

Please take a moment to write down how the regression output is related to the Figure. In particular, what is equal to, what is equal to, and what is their difference equal to?

# Don't read this unless you really like working on graphics :)   
binary\_gender <- (gender == "Male") \* 1   
plot(binary\_gender, achrdg08,   
 col = "#4B9CD3",   
 xlab = "binary gender (Female = 0, Male = 1)",  
 ylab = "Reading (Grade 8)")  
  
means <- tapply(achrdg08, binary\_gender, mean)  
labels <- c(expression(bar(Y)[0]), expression(bar(Y)[1]))  
  
text(x = c(.1, .9), y = means, labels = labels, cex = 1.5)   
text(x = c(0, 1), y = means, labels = c("\_", "\_"), cex = 2)

|  |
| --- |
|  |

# convert "Female / Male" coding to binary  
gender <- NELS$gender  
binary\_gender <- (gender == "Male")\*1  
mod\_binary <- lm(achrdg08 ~ binary\_gender)  
coef(mod\_binary)

(Intercept) binary\_gender   
 56.4678022 -0.9223396

[Section 4.3.1](#sec-t-test-4)

We will go through the comparison between these two outputs in class together. If you have any questions about the relation between these two sets of output, please note them now and be prepared ask them in class.

Regression with a binary predictor:

summary(mod\_binary)

Call:  
lm(formula = achrdg08 ~ binary\_gender)  
  
Residuals:  
 Min 1Q Median 3Q Max   
-20.7278 -6.1472 0.3784 6.9765 15.0045   
  
Coefficients:  
 Estimate Std. Error t value Pr(>|t|)   
(Intercept) 56.4678 0.5342 105.703 <2e-16 \*\*\*  
binary\_gender -0.9223 0.7928 -1.163 0.245   
---  
Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
  
Residual standard error: 8.827 on 498 degrees of freedom  
Multiple R-squared: 0.00271, Adjusted R-squared: 0.0007076   
F-statistic: 1.353 on 1 and 498 DF, p-value: 0.2452

Independent samples t-test (with homogeneity of variance assumed):

t.test(achrdg08 ~ binary\_gender, var.equal = T)

Two Sample t-test  
  
data: achrdg08 by binary\_gender  
t = 1.1633, df = 498, p-value = 0.2452  
alternative hypothesis: true difference in means between group 0 and group 1 is not equal to 0  
95 percent confidence interval:  
 -0.6353793 2.4800586  
sample estimates:  
mean in group 0 mean in group 1   
 56.46780 55.54546

[Section 4.4.2](#sec-reference-group-interpretation-4)

Regressing Age on the dummies we have:

In order to interpret the regression coefficients we can use the same two steps as in [Section 4.3](#sec-binary-predictors-4)

* *Step 1.* Plug the values for the variables into the regression equation to get the predicted values for each group
* *Step 2.* Solve for the model parameters in terms the predicted values.

Using the above equations, please write down an interpretation of the regression parameters for the hypothetical example. (Note: this question is not asking for a numerical answer, it is just asking you to put the above equations into words.)

[Section 4.4.3](#sec-reference-group-more-than-three-4)

The Mstatus example with 4 categories:

knitr::include\_graphics("files/images/marital\_status2.png")

|  |
| --- |
|  |

Here we are interested in the model:

Please work through the following questions and be prepared to share your answers in class

* How should be coded so that “single” is the reference-group?
* Write down the contrast matrix for the example.
* Using the two-step approach illustrated above, write out the interpretation of the regression coefficients.
* Let’s imagine that we included an additional dummy variable for “Single” in the model. In this case we would have dummies in the model rather than dummies. Please take a moment to write down what you think will happen (Hint: something goes wrong).

[Section 4.5.1](#sec-idela-4)

Please take a moment to write down the contrast matrix for the IDELA example:

[Section 4.5.2](#sec-deviation-interpretation-4)

Regressing IDELA on the dummies we have:

In order to interpret the regression coefficients, we proceed using the same two steps as in [Section 4.3](#sec-binary-predictors-4) and [Section 4.4](#sec-reference-group-coding-4).

* *Step 1.* Plug the values for into the regression equation to get the predicted values for each group.
* *Step 2.* Solve for the model parameters in terms of the predicted values.

At this point, we want to use the first two lines to substitute in for and in the last line, which leads to

In the last line we see that is equal to the (unweighted) mean of the predicted values.

Using the above equations, please write down an interpretation of the regression parameters for the hypothetical example. (Note: this question is not asking for a numerical answer, it is just asking you to put the above equations into words.)

## 4.9 Exercises

These exercises provide an overview of contrast coding with categorical predictors in R. Two preliminary topics are also discussed: linear regression with a binary predictor, and the factor data class in R. We will go through this material in class together, so you don’t need to work on it before class (but you can if you want.)

Before staring this section, you may find it useful to scroll to the top of the page, click on the “</> Code” menu, and select “Show All Code.”

### 4.9.1 A single binary predictor

Regression with a single binary predictor is equivalent to the independent samples t-test of group means. Let’s illustrate this using the NELS dataset with Reading achievement in grade 8 (achrdg08) as the outcome and gender as the predictor. Although gender need not be conceptualized as dichotomous or even categorical, the variable gender reported in NELS data is dichotomous, with values “Female” and “Male”.

To start with, let’s recode gender to binary\_gender by setting Female = 0 and Male = 1. Below we use a little trick for doing this. First we create a logical vector (gender == "Male") and then coerce the logical vector to binary by multiplying it by 1. There are other ways to do this but this approach is pretty easy.

# load("NELS.RData")  
# attach(NELS)  
  
# Note that gender is non-numeric -- it is a factor with levels "Female" and "Male"  
head(gender)

[1] Male Female Male Female Male Female  
Levels: Female Male

class(gender)

[1] "factor"

# A trick to create a binary indicator for males   
binary\_gender <- (gender == "Male") \* 1   
  
# Check that the two variables are telling us the same thing  
table(gender, binary\_gender)

binary\_gender  
gender 0 1  
 Female 273 0  
 Male 0 227

Next we regress Reading Achievement (achrdg08) on binary\_gender and compare this to an independent samples t-test using the same two variables.

# Regression with a binary variable  
mod1 <- lm(achrdg08 ~ binary\_gender)  
summary(mod1)

Call:  
lm(formula = achrdg08 ~ binary\_gender)  
  
Residuals:  
 Min 1Q Median 3Q Max   
-20.7278 -6.1472 0.3784 6.9765 15.0045   
  
Coefficients:  
 Estimate Std. Error t value Pr(>|t|)   
(Intercept) 56.4678 0.5342 105.703 <2e-16 \*\*\*  
binary\_gender -0.9223 0.7928 -1.163 0.245   
---  
Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
  
Residual standard error: 8.827 on 498 degrees of freedom  
Multiple R-squared: 0.00271, Adjusted R-squared: 0.0007076   
F-statistic: 1.353 on 1 and 498 DF, p-value: 0.2452

# Compare to the output of t-test  
t.test(achrdg08 ~ binary\_gender, var.equal = T)

Two Sample t-test  
  
data: achrdg08 by binary\_gender  
t = 1.1633, df = 498, p-value = 0.2452  
alternative hypothesis: true difference in means between group 0 and group 1 is not equal to 0  
95 percent confidence interval:  
 -0.6353793 2.4800586  
sample estimates:  
mean in group 0 mean in group 1   
 56.46780 55.54546

Note that:

* The intercept in mod1 is equal the mean of the group coded 0 (females) in the t-test:
* The b-weight in mod1 is equal to difference between the means:
* The t-test of the b-weight, and its p-value, are equivalent to the t-test of the mean difference (except for the sign):

In summary, a t-test of a b-weight of a binary predictor is equal equivalent to an independent samples t-test of means.

### 4.9.2 Reference-group coding

As discussed in [Section 4.4](#sec-reference-group-coding-4), the basic idea of contrast coding is to replace a categorical variable with categories with “dummy” variables. In reference-group coding, the dummy variables are binary, and the resulting interpretation is:

* The intercept is interpreted as the mean of the reference group. The reference is group is chosen by the analyst – it is the group that is coded zero on all dummies, or the group whose dummy variable is “left out” of the dummy variables.
* The regression coefficients of the dummy variables are interpreted as the difference between the mean of the indicated group and the mean of the reference group

Reference-group coding is the default contrast coding in R. However, in order for contrast coding to be implemented, our categorical predictor needs to be represented in R as a special data type called a factor.

### 4.9.3 Factors in R

Factors are the way that R deals with categorical data. If you want to know if your variable is a factor or not, you can use the functions class or is.factor. Let’s illustrate this with the urban variable from NELS.

# Two ways of checking what type a variable is   
class(urban)

[1] "factor"

is.factor(urban)

[1] TRUE

# Find out the levels of a factor using "levels"  
levels(urban)

[1] "Rural" "Suburban" "Urban"

# Find out what contrast coding R is using for a factor "contrasts"  
contrasts(urban)

Suburban Urban  
Rural 0 0  
Suburban 1 0  
Urban 0 1

In the above code, we see that urban is a factor with 3 levels and the default reference-group contrasts are set up so that Rural is the reference group. Below we will show how to change the contrasts for a variable.

If we are working with a variable that is not a factor, but we want R to treat it as a factor, we can use the factor command. Let’s illustrate this by turning binary\_gender into a factor.

# Change a numeric variable into a factor  
class(binary\_gender)

[1] "numeric"

factor\_gender <- factor(binary\_gender)  
class(factor\_gender)

[1] "factor"

levels(factor\_gender)

[1] "0" "1"

We can also use the levels function to tell R what labels we want it to use for our factor.

# Change the levels of factor\_gender to "F" and "M"  
levels(factor\_gender) <- c("Female", "Male")  
levels(factor\_gender)

[1] "Female" "Male"

### 4.9.4 Back to reference-group coding

OK, back to reference coding. Let’s see what lm does when we regress achrdg08 on urban.

mod2 <- lm(achrdg08 ~ urban)  
summary(mod2)

Call:  
lm(formula = achrdg08 ~ urban)  
  
Residuals:  
 Min 1Q Median 3Q Max   
-22.3002 -6.1620 0.2098 6.7948 15.5573   
  
Coefficients:  
 Estimate Std. Error t value Pr(>|t|)   
(Intercept) 54.9927 0.6885 79.876 < 2e-16 \*\*\*  
urbanSuburban 0.6675 0.9117 0.732 0.46439   
urbanUrban 3.1275 1.0480 2.984 0.00298 \*\*   
---  
Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
  
Residual standard error: 8.763 on 497 degrees of freedom  
Multiple R-squared: 0.01904, Adjusted R-squared: 0.0151   
F-statistic: 4.824 on 2 and 497 DF, p-value: 0.008411

In the output, we see that two regression coefficients are reported, one for Suburban and one for Urban. As discussed in [Section 4.4](#sec-reference-group-coding-4), these coefficients are the mean difference between the indicated group and the reference group (Rural).

We can see that Urban students scores significantly higher than Rural students (3.125 percentage points), but there was no significant difference between Rural and Suburban students.

The intercept is the mean of the reference group (Rural) – about 55% on the reading test.

Note the R-squared – Urbanicity accounts for about 2% of the variation reading achievement. As usual, the F-test of R-squared has degrees of freedom and , but now (the number of predictors) is equal to – the number of categories minus one.

### 4.9.5 Changing the reference group

What if we wanted to use a group other than Rural as the reference group? We can chose a different reference group using the cont.treatment function. This function takes two arguments

* n tells R how many levels there
* base tells R which level should be the reference group

# The current reference group is Rural  
contrasts(urban)

Suburban Urban  
Rural 0 0  
Suburban 1 0  
Urban 0 1

# Chance the reference group to the Urban (i.e., the last level)  
contrasts(urban) <- contr.treatment(n = 3, base = 3)  
contrasts(urban)

1 2  
Rural 1 0  
Suburban 0 1  
Urban 0 0

Note that when we first ran contrasts(urban), the column names were names of the levels. But after changing the reference group, the column names are just the numbers 1 and 2. To help interpret the lm output, it is helpful to name the contrast levels appropriately

# Naming our new contrasts  
colnames(contrasts(urban)) <- c("Rural", "Suburban")  
contrasts(urban)

Rural Suburban  
Rural 1 0  
Suburban 0 1  
Urban 0 0

Now we are ready to run our regression again, this time using a different reference group.

mod3 <- lm(achrdg08 ~ urban)  
summary(mod3)

Call:  
lm(formula = achrdg08 ~ urban)  
  
Residuals:  
 Min 1Q Median 3Q Max   
-22.3002 -6.1620 0.2098 6.7948 15.5573   
  
Coefficients:  
 Estimate Std. Error t value Pr(>|t|)   
(Intercept) 58.1202 0.7901 73.559 < 2e-16 \*\*\*  
urbanRural -3.1275 1.0480 -2.984 0.00298 \*\*   
urbanSuburban -2.4600 0.9907 -2.483 0.01335 \*   
---  
Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
  
Residual standard error: 8.763 on 497 degrees of freedom  
Multiple R-squared: 0.01904, Adjusted R-squared: 0.0151   
F-statistic: 4.824 on 2 and 497 DF, p-value: 0.008411

Compared to the output from mod2, note that

* The intercept now represents the mean reading scores of the Urban group, because this is the new reference group.
* The regression coefficients now represent the mean difference between the indicated group with the new reference group.
* The R-square and F-test stay the same – in other words, the total amount of variation explained by the variable Urban does not change, just because we changed the reference group.

### 4.9.6 Deviation coding

It is possible to change R’s default contrast coding to one of the other built-in contrasts (see help(contrasts) for more information on the built-in contrasts).

For instance, to change to deviation coding, we use R’s contr.sum function and tell it how many levels there are for the factor (n). In deviation coding, the intercept is equal to the unweighted mean of the predicted values, and the regression coefficients are difference between the indicated group and the unweighted mean.

contrasts(urban) <- contr.sum(n = 3)  
contrasts(urban)

[,1] [,2]  
Rural 1 0  
Suburban 0 1  
Urban -1 -1

# As above, it is helpful to name the contrasts using "colnames"  
colnames(contrasts(urban)) <- c("Rural", "Suburban")

Now we are all set to use deviation coding with lm.

mod4 <- lm(achrdg08 ~ urban)  
summary(mod4)

Call:  
lm(formula = achrdg08 ~ urban)  
  
Residuals:  
 Min 1Q Median 3Q Max   
-22.3002 -6.1620 0.2098 6.7948 15.5573   
  
Coefficients:  
 Estimate Std. Error t value Pr(>|t|)   
(Intercept) 56.2577 0.4021 139.897 <2e-16 \*\*\*  
urbanRural -1.2650 0.5654 -2.237 0.0257 \*   
urbanSuburban -0.5975 0.5299 -1.128 0.2600   
---  
Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
  
Residual standard error: 8.763 on 497 degrees of freedom  
Multiple R-squared: 0.01904, Adjusted R-squared: 0.0151   
F-statistic: 4.824 on 2 and 497 DF, p-value: 0.008411

Note the following things about the output:

* The regression coefficients compare each group’s mean to the unweighted mean of the groups. Rural and Suburban students are below the unweighted mean, but the difference is only significant for Rural.
* Although the output looks similar to that of mod3, the coefficients are all different and they all have different interpretations. The lm output doesn’t tell us this, we have to know what is going on under the hood.
* The R-square does not change from mod2 – again, the type of contrast coding used doesn’t affect how much variation is explained by the predictor.

### 4.9.7 Extra: Relation to ANOVA

As our next exercise, let’s compare the output of lm and the output of aov – R’s module for Analysis of Variance. If regression and ANOVA are really doing the same thing, we should be able to illustrate it with these two modules. Note that if you check out help(aov), it explicitly states that aov uses the lm function, so, finding that the two approaches give similar output shouldn’t be a big surprise!

# Run our model as an ANOVA  
aov1 <- aov(achrdg08 ~ urban)  
  
# Compare the output with lm  
summary(aov1)

Df Sum Sq Mean Sq F value Pr(>F)   
urban 2 741 370.5 4.824 0.00841 \*\*  
Residuals 497 38163 76.8   
---  
Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

summary(mod2)

Call:  
lm(formula = achrdg08 ~ urban)  
  
Residuals:  
 Min 1Q Median 3Q Max   
-22.3002 -6.1620 0.2098 6.7948 15.5573   
  
Coefficients:  
 Estimate Std. Error t value Pr(>|t|)   
(Intercept) 54.9927 0.6885 79.876 < 2e-16 \*\*\*  
urbanSuburban 0.6675 0.9117 0.732 0.46439   
urbanUrban 3.1275 1.0480 2.984 0.00298 \*\*   
---  
Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
  
Residual standard error: 8.763 on 497 degrees of freedom  
Multiple R-squared: 0.01904, Adjusted R-squared: 0.0151   
F-statistic: 4.824 on 2 and 497 DF, p-value: 0.008411

If we compare the output from aov to the F-test reported by lm we see that the F-test, degrees of freedom, and p-value are all identical. If we compute eta-squared from aov, we also find that it is equal to the R-squared value from lm.

# Compute eta-squared from aov output  
741 / (741 + 38163)

[1] 0.01904688

In short, ANOVA and regression are doing the same thing: R-squared is the same as eta-squared and the ANOVA omnibus F-test is the same as the F-test of R-squared.

### 4.9.8 Extra: Weighted deviation coding with all groups

This section presents a “hack” for addressing the two main issues with deviation coding noted in [Section 4.5](#sec-deviation-coding-4). This is a hack in the sense that we are working around R’s usual procedures. The result of this hack is to provide deviation coding in which comparisons are made to the grand mean on the outcome for all categories, not just categories. As a side effect, the F-test for the R-squared statistic reported in summary(lm) no longer correct, so we show how to fix that too.

First, note that if if we omit the intercept term, all of the reference-group coded dummies can be included in the model and the regression coefficients now correspond to means of each group. We omit the intercept using -1 in the model formula.

# Omit the intercept using -1  
mod5 <- lm(achrdg08 ~ -1 + urban)  
summary(mod5)

Call:  
lm(formula = achrdg08 ~ -1 + urban)  
  
Residuals:  
 Min 1Q Median 3Q Max   
-22.3002 -6.1620 0.2098 6.7948 15.5573   
  
Coefficients:  
 Estimate Std. Error t value Pr(>|t|)   
urbanRural 54.9927 0.6885 79.88 <2e-16 \*\*\*  
urbanSuburban 55.6602 0.5976 93.14 <2e-16 \*\*\*  
urbanUrban 58.1202 0.7901 73.56 <2e-16 \*\*\*  
---  
Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
  
Residual standard error: 8.763 on 497 degrees of freedom  
Multiple R-squared: 0.9763, Adjusted R-squared: 0.9761   
F-statistic: 6822 on 3 and 497 DF, p-value: < 2.2e-16

Note the following things about the output:

* The coefficients are no **longer interpreted as mean differences**, just the raw means.
* The R-square and F-test **do** change from mod2. When the intercept is omitted, R-squared can no longer interpretated as a proportion of variance unless the variable variable is centered. When the intercept is omitted, it also changes the degrees of freedom in the F-test, because there are now 3 predictors instead of 2. In general, when the intercept is omitted, R-square and its F-test do not have the usual interpretation, so we need to be careful.

By itself, the “group-means” appoach to contrast coding is not very interesting – we don’t usually want to test whether the group means are different from zero. However, this approach can be used to provide an alternative to deviation coding if we center the variable before using the above approach. After centering, the grand mean of is zero, and so the group means represent deviations from the grand mean (i.e., deviations from zero) and the tests of the regression coefficients are tests of whether the group means are different from the grand mean.

# Center the outcome variable  
dev\_achrdg08 <- achrdg08 - mean(achrdg08)  
  
# Run the regression with group mean coding  
mod6 <- lm(dev\_achrdg08 ~ -1 + urban)  
summary(mod6)

Call:  
lm(formula = dev\_achrdg08 ~ -1 + urban)  
  
Residuals:  
 Min 1Q Median 3Q Max   
-22.3002 -6.1620 0.2098 6.7948 15.5573   
  
Coefficients:  
 Estimate Std. Error t value Pr(>|t|)   
urbanRural -1.0564 0.6885 -1.534 0.12556   
urbanSuburban -0.3889 0.5976 -0.651 0.51554   
urbanUrban 2.0711 0.7901 2.621 0.00903 \*\*  
---  
Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
  
Residual standard error: 8.763 on 497 degrees of freedom  
Multiple R-squared: 0.01904, Adjusted R-squared: 0.01312   
F-statistic: 3.216 on 3 and 497 DF, p-value: 0.02264

Note the following things about the output:

* The regression coefficients compare each group’s mean to the overall mean on the outcome. Urban students are significantly above average, Rural and Suburban students are below average but the difference is not significant.
* The R-square is the same as mod2 because the variable is centered, but the F test is not the same as from mod2. The next two paragraphs explain why, but the explanation requires some material we won’t discuss until [Chapter 6](#sec-chap-6).

The F-test of R-squared can be computed by comparing two different models, a reference model with no predictors and a focal model with the predictors. The reference model used by lm depends on whether the intercept is included in the focal model. If the focal model has an intercept, the reference model also has an intercept. In the focal model omits the intercept, the reference also omits the intercept.

In the present context, the focal model omits the intercept and use dummies This model has the same R-squared as if we had used the intercept and dummies. However, the F-statistic for the test of R-squared (as reported by summary(lm)) is different for the two approaches, because of the choice of reference model. You can change the reference model, but it takes a bit of fiddling around. If you are interested, just ask in class.

What all of this mean is that we can compute the “correct” F-test of R-squared by comparing mod6 to a model with just the intercept. The following code shows how to do this.

# A model with only the intercept  
mod7 <- lm(dev\_achrdg08 ~ 1)  
  
# Compute the F-test of comparing mod6 to mod7  
anova(mod6, mod7)

Analysis of Variance Table  
  
Model 1: dev\_achrdg08 ~ -1 + urban  
Model 2: dev\_achrdg08 ~ 1  
 Res.Df RSS Df Sum of Sq F Pr(>F)   
1 497 38163   
2 499 38904 -2 -740.91 4.8244 0.008411 \*\*  
---  
Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

This is the same F-test reported for mod2, mod3 and mod4 above.

# 5. Interactions

In statistics, the term *interaction* means that the relationship between two variables depends on a third variable. In the context of regression, we are usually interested in the situation where the relationship between the outcome and a predictor depends on another predictor . This situation is also referred to as *moderation* or sometimes as *effect heterogeneity*.

Interactions are a big-picture idea with a lot conceptual power, especially when describing topics related to social inequality or “gaps”. Some examples of interactions are:

* The relationship between wages and years of education depends on gender. This has been called the gender pay gap and it is considered a pretty important issue for gender equality (e.g., <https://en.wikipedia.org/wiki/Gender_pay_gap>).
* The relationship between reading achievement and age depends on race. This has been interpreted in terms of racial inequality in educational outcomes (e.g., <https://cepa.stanford.edu/educational-opportunity-monitoring-project/achievement-gaps/race/>).
* The effect of COVID-19 school shutdowns on academic achievement depended on SES. This has been interpreted in terms of social inequality in access to educational resources outside of schools (e.g., <https://www.mckinsey.com/industries/education/our-insights/covid-19-and-student-learning-in-the-united-states-the-hurt-could-last-a-lifetime>).

I hope these examples convince you that some of the big issues facing education and society at large are actually about interactions – how the relationship between two variables depends on a third variable. In this chapter we are going to talk about how you can use regression to conduct research on these types of topics.

Some terminology: When we talk about statistical interactions, we often leave the outcome variable implicit and focus on the predictors. For example, the gender pay gap can be described as an interaction between gender and years of education. The outcome variable (wages) is implicit in this description. Throughout this chapter we are exclusively interested in interactions between two predictors at a time, which are called *two-way interactions*. There are actually three variables involved, because a two-way interaction also requires an outcome variable. It is possible to consider “higher-order” interactions (e.g., interactions among three predictors or *three-way interactions*) but we aren’t going to do that here. Sometimes we will use the terminology *main effect* to describe the relationship between each individual predictor and the outcome variable, as distinct from the interactions among the predictors. Up to now, we have be working with main effects only.

We start by considering what happens when both categorical and continuous predictors are used together in a multiple regression model. We use this combination of predictors as bridge from the previous chapter and as a way of digging into the math behind interactions. Later sections will consider what happens when we have interactions between two continuous predictors, or two categorical predictors.

It might be helpful to mention that this chapter gets pretty complicated and is very long (sorry!). It is definitely the hardest material we have covered so far, because we are drawing on everything we have done up to now and adding even more stuff into the mix. So, if these readings feel daunting at moments, that is to be expected. You might consider taking a break and doing the readings in smaller chunks. Also keep in mind that we are going to discuss all of this in class together, so just get what you can from these notes, provide some initial responses to the Workbook questions, and press on. This is where regression starts to get really interesting – you got this!

## 5.1 An example from NELS

For the first few sections of this chapter, we will focus on the gender gap in Math Achievement as our example (e.g., <https://www.nctm.org/Publications/TCM-blog/Blog/Current-Research-on-Gender-Differences-in-Math/>). The t-test reported below uses the NELS data to illustrate the gender gap in Math Achievement in 12th grade. The output shows that, on average, males scored about 3.18 percentage points higher than females on a Grade 12 Math test. This gap isn’t very big. However, it tends to grow rather than get smaller as students progress to higher grades, and it has implications for gender equality in STEM education and STEM professions.

load("NELS.RData")  
attach(NELS)  
t.test(achmat12 ~ gender, var.equal = T)

Two Sample t-test  
  
data: achmat12 by gender  
t = -4.5529, df = 498, p-value = 6.658e-06  
alternative hypothesis: true difference in means between group Female and group Male is not equal to 0  
95 percent confidence interval:  
 -4.527002 -1.797687  
sample estimates:  
mean in group Female mean in group Male   
 55.47092 58.63326

In this chapter, our goal is to use linear regression to better understand the gender gap in Math Achievement. To help us do this, we will also consider a third variable, Reading Achievement. The plot below shows the relationship between Math Achievement and Reading Achievement estimated just for males (Blue), and the same relationship estimated just for females (Black).

# Create indicators for females and males  
females <- gender == "Female"  
males <- gender == "Male"  
  
# Regress math on reading, for each group separately  
mod1 <- lm(achmat12[females] ~ achrdg12[females])  
mod2 <- lm(achmat12[males] ~ achrdg12[males])  
  
# Plot reading and math for females  
plot(achrdg12[females],   
 achmat12[females],   
 xlab = "Reading",   
 ylab = "Math")  
  
abline(mod1, lwd = 2)  
  
# Add again for males  
points(achrdg12[males],   
 achmat12[males],   
 col = "#4B9CD3",   
 pch = 2)  
  
abline(mod2, col = "#4B9CD3", lwd = 2)  
  
# Add a legend  
legend(x = "topleft",   
 legend = levels(gender),   
 pch = c(1, 2),   
 col = c(1, "#4B9CD3"))

|  |
| --- |
| Figure 5.1: Math Achievement, Reading Achievement, and Gender. |

**Please take a minute to think about what this plot is telling us about the relationships among the three variables. In particular:**

* **Does the gender gap in Math Achievement change as a function of Reading Achievement?**
* **Is the relationship between Math Achievement and Reading Achievement the the same for males and females?**
* **What do the results mean for gender equality in Math and STEM education?**

Note that in [Figure 5.1](#fig-math-reading-1), we estimated two separate simple regression models, one just for males and one just for females. In the next few sections, we will work our way towards a single multiple regression model that can be used to represent the relationships among these three variables.

## 5.2 Binary + continuous

Let’s start by considering what happens when we include both Gender and Reading Achievement as predictors of Math Achievement in our usual multiple regression equation:

where

* is Math Achievement in grade 12
* is Reading Achievement in grade 12
* is Gender (binary, with female = 0 and male = 1)

Note that this model does **not** include an interaction between the two predictors – we are first going to consider what is “missing” from the usual regression model, and then use this to motivate inclusion of another predictor that represents the interaction. To get an initial sense of what is missing, the model in [Equation 5.1](#eq-yhat-5a) is plotted in [Figure 5.2](#fig-math-reading-2) using the NELS data – can you spot the difference with [Figure 5.1](#fig-math-reading-1)?

# Run the regression  
mod3 <- lm(achmat12 ~ achrdg12 + gender, data = NELS)  
  
a\_females <- coef(mod3)[1]  
b\_females <- coef(mod3)[2]  
  
# Get the slope and intercept for males  
a\_males <- a\_females + coef(mod3)[3]  
b\_males <- b\_females   
  
# Plot reading and math for females  
plot(achrdg12[females],   
 achmat12[females],   
 xlab = "Reading",   
 ylab = "Math")  
  
abline(a\_females, b\_females, lwd = 2)  
  
# Add points and line for males  
points(achrdg12[males],   
 achmat12[males],   
 col = "#4B9CD3",   
 pch = 2)  
  
abline(a\_males, b\_males, col = "#4B9CD3", lwd = 2)  
  
# Add a legend  
legend(x = "topleft",   
 legend = levels(gender),   
 pch = c(1, 2),   
 col = c(1, "#4B9CD3"))

|  |
| --- |
| Figure 5.2: Math Achievement, Reading Achievement, and Gender (No Interaction). |

In order to interpret our multiple regression model, we can use the same overall approach as we used to interpret categorical predictors in [Chapter 4](#sec-chap-4). If we plug-in values for the categorical predictor, we get:

The equations for and are referred to as *simple trends* or *simple slopes*. These describe the regression of Math on Reading, simply for females, or simply for males. The difference between the two simple regression equations is the predicted gender gap in Math Achievement.

Based on these equations we can interpret regression coefficients as follows

* The regression intercept, , is the intercept of the simple trend for the group coded “0” (i.e., the intercept of the regression of Math on Reading, simply for females; see [Equation 5.2](#eq-females1-5)).
* The regression slope for the continuous predictor, , is the slope of both of the simple trends (see [Equation 5.2](#eq-females1-5) and [Equation 5.3](#eq-males1-5))
* The regression slope for the binary predictor, , is the difference between the intercepts of the simple trends (i.e., we add to to get the intercept of the regression of Math on Reading, simply for males; see [Equation 5.3](#eq-males1-5)).
* In this model, is the also the predicted gender gap in Math Achievement (see [Equation 5.4](#eq-gap1-5)).

The regression coefficients for the example data are shown below. **Please use these numbers to provide an interpretation of the simple trends and the gender gap in Math Achievement for the NELS example. (Don’t worry about statistical significance, just focus on the meaning of the coefficients reported in the “Estimate” column.)**

summary(mod3)

Call:  
lm(formula = achmat12 ~ achrdg12 + gender, data = NELS)  
  
Residuals:  
 Min 1Q Median 3Q Max   
-19.2448 -3.6075 0.3968 3.9836 15.5606   
  
Coefficients:  
 Estimate Std. Error t value Pr(>|t|)   
(Intercept) 19.98122 1.86278 10.727 < 2e-16 \*\*\*  
achrdg12 0.63551 0.03275 19.404 < 2e-16 \*\*\*  
genderMale 3.50166 0.52473 6.673 6.69e-11 \*\*\*  
---  
Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
  
Residual standard error: 5.839 on 497 degrees of freedom  
Multiple R-squared: 0.4538, Adjusted R-squared: 0.4516   
F-statistic: 206.4 on 2 and 497 DF, p-value: < 2.2e-16

Here is an example to help you get started:

* Simply for females, the regression of Math Achievement on Reading Achievement has an intercept equal to 19.98 and a slope equal to 0.64. The intercept tells us that a female student with Reading Achievement score of 0% is expected to have a Math Achievement score of 19.98% (the units of the two tests are percent correct). Since the lowest value of Reading Achievement in our example is about 35%, the intercept is not very meaningful for these data. The regression slope tells us that, for females, a 1 unit increase in Reading Achievement is associated with a .64 unit increase in Math Achievement.
* Simply for males, ….
* The gender gap in Math Achievement was equal to …

### 5.2.1 Marginal means

Before moving on to consider interactions, let’s revisit a topic from the previous chapter. In [Chapter 4](#sec-chap-4), we noted that the regression coefficients for categorical predictors (dummy variables) can be interpreted in terms of the group means of the outcome variable. However, when additional predictors are included in the regression model, this interpretation no longer holds. This section explains why.

To see what the issue is, let’s compare the output of the t-test in [Section 5.1](#sec-example-5) with the regression output shown above. In the t-test, the mean Math Achievement for females was 55.47, and for males it was 58.63. The mean difference was

However, the regression coefficient on Gender in the multiple regression model above is equal to . Thus, unlike [Chapter 4](#sec-chap-4), the regression coefficient on Gender is no longer equal to the group-mean difference in Math Achievement. But why?

Remember that in the multiple regression model, the regression coefficient on Gender is interpreted as the relationship between Math Achievement and Gender, *holding Reading Achievement constant*. So, the regression coefficient on Gender is still interpreted as a mean difference, but now it is a *predicted* mean difference that represents the gender gap in Math Achievement after controlling for Reading Achievement. The t-test doesn’t control for Reading.

In order to emphasize the distinction between “raw” group means computed from the data and the predicted group means obtained from a multiple regression model, the latter are referred to as *marginal means*, or sometimes as *adjusted means* or *least squares means*. I think they should be called *predicted means*, but, alas.

### 5.2.2 Summary

In a regression model with one continuous predictor and one binary predictor (and no interaction):

* The model results in two regression lines, one for each value of the binary predictor. These are called the simple trends.
* The simple trends are parallel but can have different intercepts; the difference between the intercepts is equal to regression coefficient of the binary variable.
* The difference between the simple trends is often called a “gap”, and the gap is also equal to the regression coefficient of the binary variable.
* It is important to note that the predicted group means for the binary variable are no longer equal to the “raw” group means computed directly from the data, because the predicted group means control for the correlation between the predictors. The predicted group means are called marginal means to emphasize this distinction.

Keep in mind that this summary applies to the multiple regression model **without an interaction**. In the next section we improve our model by adding an interaction.

## 5.3 Binary + continuous + interaction

In this section, we discuss what was missing from the multiple regression model in the previous section: The interaction between Gender and Reading.

Mathematically, an interaction is just the product between two variables. [Equation 5.5](#eq-yhat-5b) shows how to include this product in our multiple regression model – we just take the product of the two predictors and add it into the model as a third predictor:

In terms of computation, we would literally take the product of our two predictors and save it as a new variable in our data set, then add the new variable in to the regression model. In practice, R will do all of this for us behind the scenes, so we don’t actually need to “hard code” new variables.

For the NELS example, the regression model with the interaction is depicted in [Figure 5.3](#fig-math-reading-3). Note the the simple trends are no longer parallel and the regression lines agree exactly with what we had in [Section 5.1](#sec-example-5). So, as promised, we have now arrived at a single multiple regression model that captures the relationships among Math Achievement, Reading Achievement, and Gender.

# Interaction via hard coding  
genderXachrdg12 <- (as.numeric(gender) - 1) \* achrdg12  
mod4 <- lm(achmat12 ~ achrdg12 + gender + genderXachrdg12)  
  
# Get the coefficients for females  
a\_females <- coef(mod4)[1]  
b\_females <- coef(mod4)[2]  
  
# Get the coefficients for males  
a\_males <- a\_females + coef(mod4)[3]  
b\_males <- b\_females + coef(mod4)[4]  
  
# Plot reading and math for females  
plot(achrdg12[females],   
 achmat12[females],   
 xlab = "Reading",   
 ylab = "Math")  
  
abline(a\_females, b\_females, lwd = 2)  
  
# Add points and line for males  
points(achrdg12[males],   
 achmat12[males],  
 col = "#4B9CD3",   
 pch = 2)  
  
abline(a\_males, b\_males, col = "#4B9CD3", lwd = 2)  
  
# Add a legend  
legend(x = "topleft",  
 legend = levels(gender),   
 pch = c(1, 2),   
 col = c(1, "#4B9CD3"))

|  |
| --- |
| Figure 5.3: Math Achievement, Reading Achievement, and Gender (No Interaction). |

To see how to interpret the coefficients in this model, let’s work through the model equations using our two-step procedure. As before, we first plug-in values for the categorical predictor, then we use the resulting equations solve for the simple trends and the gender gap in Math.

These equations are summarized graphically below in [Figure 5.4](#fig-interactions).

|  |
| --- |
| Figure 5.4: Interaction between a categorical and a continuous predictor. Image credit: Daniela Rodriguez-Mincey, Spring 2023 |

Based on the equations and figure, we can interpret regression coefficients as follows. Some of these interpretations are the same as in the previous section, but some are different.

* The regression intercept, , is the intercept of the simple trend for the group coded “0” (i.e., the intercept of the regression of Math on Reading, simply for females; see [Equation 5.6](#eq-females2-5)). This is the **same** interpretation as for the model without the interaction discussed in [Section 5.2](#sec-binary-continuous-5).
* The regression slope for the continuous predictor, , is the slope of the simple trend for the group coded “0” (females; see [Equation 5.6](#eq-females2-5)). This is **different** from the interpretation of the model in [Section 5.2](#sec-binary-continuous-5) – in that model, was the slope of both simple trends, not just the trend for females.
* The regression slope for the binary predictor, , is the difference between the intercepts of the simple trends (i.e., we add to to get the intercept of the regression of Math on Reading, simply for males; see [Equation 5.7](#eq-males2-5)). This is the **same** interpretation as for the model without the interaction discussed in [Section 5.2](#sec-binary-continuous-5).
* The regression slope for the interaction term (or simply, the interaction), , is the difference between the slopes of the simple trends (i.e., we add to to get the slope of the regression of Math on Reading, simply for males; see [Equation 5.7](#eq-males2-5)). This is **different** from the interpretation of the model in [Section 5.2](#sec-binary-continuous-5) – in that model, was the slope of both simple trends.
* The difference between the predicted values (i.e., the predicted gender gap in Math Achievement) is no longer constant, but is instead a function of (see [Equation 5.8](#eq-gap2-5)). In particular, the predicted gender gap in Math changes by units for each unit of increase in Reading. This is **different** from the interpretation of the model in [Section 5.2](#sec-binary-continuous-5) – in that model, the predicted gender gap was constant over Reading and equal to .

This last point is especially important in the context of our example. The gender gap in Math Achievement is a function of Reading Achievement. This is the mathematical meaning behind the concept of an interaction – the relationship between two variables (Math and Gender) is changing as a function of a third variable (Reading).

### 5.3.1 Choosing the moderator

Interpreting interactions can feel a bit unwieldy at first. This section introduces some additional terminology that helps better align the mathematical results with the kinds of research scenarios we considered in the introduction to this chapter.

First, note that it is equally valid to say

* the relationship between Math and Gender depends on Reading, or
* the relationship between Math and Reading depends on Gender.

In other words, it is equally valid to interpret our interaction in terms of the gender gap (i.e., the relationship between Math and Gender) or in terms of the simple trends (the relationship between Math and Reading).

Although both interpretations are equally valid, in most research settings we will be more interested in one of them rather than the other. For example, our research interest in [Section 5.1](#sec-example-5) was about the gender gap in Math Achievement. So, we can simplify our lives by focusing the gender gap (i.e., the relationship between Math and Gender). For the purpose of our example, the simple trends are an equivalent but less interesting way of interpreting the interaction. The point is: we don’t have to report both the simple trends and the gender gap – we usually just choose one.

In the two bullet points above, whichever variable appears in the “depends on” clause is called the *moderator*, and the other two variables are called the focal variables. The researcher chooses which variable to treat as the moderator when interpreting an interaction. The overall idea here is to “break down” the interaction in the way that is most compatible with your research question(s).

Since our focus is the gender gap in Math (i.e., the relationship is between Math and Gender), Reading is our moderator. In particular, we might interpret the interaction as follows

* The predicted gender gap in Math changes by units for each unit of increase in Reading.

By contrast, if we were mainly interested in the relationship between Math and Reading (i.e., the simple trends), then we could treat Gender as the moderator. For example, we might say:

* For females, predicted Math Achievement changed by units for each unit of increase in Reading, whereas for males, the predicted change was () units for each unit of increase in Reading.

The simple trends might feel less intuitive than the gender gap, but the two interpretations are mathematically equivalent. It is just a matter of whether you want to interpret the interaction with reference to the gender gap, or with reference to the simple trends. When writing up your research, you don’t need to do both. But I am going to make you do both in the next section for practice.

### 5.3.2 Back to the example

The regression coefficients for the example data are shown below. **Please use these numbers to provide an interpretation of the interaction between Gender and Reading. For practice, please attempt the interpret the interaction in terms of (a) the gender gap in Math, with Reading as the moderator; and (b) the relationship between Math and Reading, with Gender as the moderator. Don’t worry about statistical significance, just focus on the interpreting the coefficients reported in the “Estimate” column.**

summary(mod4)

Call:  
lm(formula = achmat12 ~ achrdg12 + gender + genderXachrdg12)  
  
Residuals:  
 Min 1Q Median 3Q Max   
-19.0582 -3.7864 0.5014 4.0775 16.2889   
  
Coefficients:  
 Estimate Std. Error t value Pr(>|t|)   
(Intercept) 14.80310 2.64922 5.588 3.8e-08 \*\*\*  
achrdg12 0.72824 0.04702 15.487 < 2e-16 \*\*\*  
genderMale 13.39328 3.65828 3.661 0.000278 \*\*\*  
genderXachrdg12 -0.17794 0.06514 -2.732 0.006524 \*\*   
---  
Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
  
Residual standard error: 5.801 on 496 degrees of freedom  
Multiple R-squared: 0.4619, Adjusted R-squared: 0.4586   
F-statistic: 141.9 on 3 and 496 DF, p-value: < 2.2e-16

Some potential answers are hidden in the “Code” tab below, but don’t peak until you have tried it for yourself!

# Gender gap  
# General: The gender gap in Math is smaller for students who are also strong in Reading  
# Specific: The gender gap in Math Achievement decreases by .18 percentage points for each percentage point increase Reading Achievement  
  
# Simple slopes  
# General: The relationship between Math and Reading is stronger (i.e. has a larger slope) for females than for males  
# Specific:  
# For females, Math scores are predicted to increase by .72 percentage points for each percentage point increase in Reading Achievement  
# For males, Math scores are predicted to increase by .55 percentage points for each percentage point increase in Reading Achievement

### 5.3.3 Centering the continuous predictor

You may have noticed that the regression coefficient on Gender was wildly different in regression models with and without the interaction. In the model without the interaction ([Section 5.2](#sec-binary-continuous-5)) the coefficient on Gender was 3.50, and in the model with the interaction (above), it was 13.40. So in one model, the “effect” of being male was a 3.5 percentage point gain on a Math test, but in the other model, it was a 13.40 percentage point gain. Why this huge difference in the “effect” of Gender?

The answer can be seen in the equation for the gender gap. In the model **without the interaction**, the gender gap was constant and equal to the regression coefficient on Gender (denoted as in the model):

But in the regression model **with the interaction**, the gender gap was a linear function of Reading and the regression coefficient on Gender is the intercept of that linear relationship.

This equation tell us that, in the model with the interaction, is the gender gap for students who score 0% on the Reading test. Since the lowest score on Reading was around 35%, the intercept in this equation (i.e., , the regression coefficient on Gender) is not very meaningful.

In previous sections, we have ignored the regression intercept when it was not meaningful. But, ignoring the regression slopes for predictor variables can get confusing, and, in general, it is nice for the regression coefficients to be interpretable (otherwise, why are we doing this!).

One way to address this situation is to center Reading Achievement so that it has a mean of zero. To do this, we compute the deviation score

is the mean-centered version of (Reading Achievement). If we regress Math Achievement on rather than we end up with the following equation for the gender gap in Math:

Since when , the regression coefficient on Gender () is now interpretable as the gender gap in Math Achievement, for students with average Reading Achievement. This is a much more interpretable number than the coefficient in the original interaction model!

Using the example data, this approach yields the following regression coefficients (the \_dev notation means the variable was centered):

# compute the deviation scores for reading  
reading\_dev <- achrdg12 - mean(achrdg12, na.rm = T)   
  
# Run the interaction model as above  
genderXreading\_dev <- (as.numeric(gender) - 1) \* reading\_dev  
  
mod5 <- lm(achmat12 ~ reading\_dev + gender + genderXreading\_dev)  
summary(mod5)

Call:  
lm(formula = achmat12 ~ reading\_dev + gender + genderXreading\_dev)  
  
Residuals:  
 Min 1Q Median 3Q Max   
-19.0582 -3.7864 0.5014 4.0775 16.2889   
  
Coefficients:  
 Estimate Std. Error t value Pr(>|t|)   
(Intercept) 55.29439 0.35127 157.411 < 2e-16 \*\*\*  
reading\_dev 0.72824 0.04702 15.487 < 2e-16 \*\*\*  
genderMale 3.49930 0.52135 6.712 5.26e-11 \*\*\*  
genderXreading\_dev -0.17794 0.06514 -2.732 0.00652 \*\*   
---  
Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
  
Residual standard error: 5.801 on 496 degrees of freedom  
Multiple R-squared: 0.4619, Adjusted R-squared: 0.4586   
F-statistic: 141.9 on 3 and 496 DF, p-value: < 2.2e-16

The regression coefficient for Gender is now pretty close to what it was in the multiple regression model without the interaction, but the interpretation is different (i.e., it is now the predicted gender gap in Math for students with an average level of Reading, rather than the predicted gender gap in Math for all students).

Notice that the intercept in the model above has also changed compared to the previous model in which Reading was not centered. This is should make sense based on what you already know about the regression intercept.

Also note that centering Reading did not affect the regression coefficient for Reading or the interaction. So, centering makes the regression slope on Gender more interpretable, but it doesn’t affect our overall interpretation of the simple trends or the gender gap. To learn more about how centering works, see [Section 5.5.5](#sec-how-centering-works-5) (which is optional).

**Please write down your interpretation of the intercept and the regression coefficient on Gender in the above regression output, and be prepared to share your answer in class**.

### 5.3.4 Summary

The interaction between two variables is just their product. When this product is added as a predictor in a multiple regression model with one continuous and one binary predictor:

* The model again results in two regression lines (simple trends), one for each value of the binary predictor.
* However, the simple trends can now have different intercepts *and different slopes*. The difference in slopes is equal to the regression coefficient on the interaction term. In other words, the simple trends have different slopes because of the interaction.
* The difference (“gap”) between the regression lines changes as a linear function of the continuous predictor, and the slope of this linear function is again equal to the regression coefficient on the interaction term.
* The last two points are equivalent ways of stating the central idea behind a (two-way) interaction: the relationship between two variables changes as a function of a third variable.
* When interpreting an interaction, the researcher chooses which pair of variables will be the “focal relationship” and which variable will be the moderator.
  + In our example, we focused on the gender gap in Math (i.e., the relationship between Math and Gender) and Reading was the moderator.
  + But, if we were more interested in the relationship between Math and Reading, would could have focused on the simple trends and treated Gender as the moderator.
  + It is usual to only report one way of these two ways of “breaking down” the interaction – the one that is most relevant to your research question.
* Centering the continuous predictor can be helpful for ensuring that the regression coefficient on the binary predictor remains interpretable in the presence of an interaction.

## 5.4 Following-up an interaction

The procedures discussed in this section are used typically used *after* we have concluded that there is a statistically significant between two predictors (i.e., after we have examined the summary(lm) output in the previous section). When we follow-up an interaction, the goal is to gain a better understanding of *how* the focal relationship depends on the moderator. Basically, the summary(lm) output from the previous section tells us whether or not the interaction is significant, and, if it is significant, the procedures discussed in this section let us describe the interaction in more detail.

The overall idea is illustrated [Figure 5.5](#fig-visreg-1). Compared to the plots we have seen previously in this chapter, [Figure 5.5](#fig-visreg-1) now includes confidence bands for the simple trends. The confidence bands show the values of Reading for which the gender gap in Math is statistically significant. In particular, it appears that the gap is not significant for students at the highest levels of Reading Achievement.

Note that this information was not available from the summary(lm) output in the previous section. The summary(lm) output told us that the interaction term was statistically significant, which means that the gender gap in Math changes as a function of reading. The plot provides more information about *how* the gender gap depends on Reading – it is statistical significant for students at lower levels of Reading Achievement, but appears to shrink and eventually disappear as Reading Achievement increases.

# Install the package if you haven't already done so  
# install.packages("visreg")  
# Load the package into memory  
library(visreg)  
  
# Run the regression model using R's syntax for interactions "\*"  
mod6 <- lm(achmat12 ~ gender\*achrdg12, data= NELS)  
  
# One line of code to plot trends with confidence bands :)   
visreg(mod6, xvar = "achrdg12", by = "gender", overlay = TRUE)

|  |
| --- |
| Figure 5.5: Example of a plot using the visreg package. |

In this section, we discuss how to translate the confidence bands in the plot into statistical tests that can provide more specific information about the how the gender gap in Math depends on Reading. Making these kinds of inferences about interactions is one of the main advantages of using multiple regression rather than just fitting the simple trends separately as we did in [Section 5.1](#sec-example-5). Another advantage is that we can now easily produce nice plots like [Figure 5.5](#fig-visreg-1) :)

Before moving on, it is important to emphasize that, if the interaction term in the summary(lm) output is **not** significant, we don’t use the procedures discussed in this section. This is because a non-significant interaction means that we have inferred the focal relationship *does not* depend on the moderator. Consequently, there is no need to describe this dependence in more detail, which is what the procedures in this section are for.

### 5.4.1 Marginal effects

Since we are interested in the gender gap in Math Achievement, we will start by considering the values of Reading for which the gap is statistically significant. Note that this information was not available from the standard summary(lm) output shown in [Section 5.3](#sec-binary-continuous-interaction-5) – the output showed that the regression coefficient representing the interaction was statistically significant, but it didn’t tell us for the values of Reading for which the gender gap was statistically significant. We can answer this type of question using the marginal “effects” discussed in this section.

There are three main types of marginal effects. For linear models, they are all basically the same, and we will only use one of them in this section. However, it is important that you can distinguish among them, especially when we get into non-linear models (e.g., logistic regression; see [Chapter 10](#sec-chap-10)). In general, when you report marginal effects in your research, you should be able to tell your reader which approach you used so they understand what you did and how to interpret the results.

To explain the three approaches, first let’s write the gender gap in Math Achievement using a slightly more compact notation (“” for difference):

The three types of marginal effects are:

* Marginal effects at the mean (MEM): Report the gap at the mean value of
* Average marginal effect (AVE): Average the effect over values of :
* Marginal effects at representative values (MERV): Report the marginal effect for a range of “interesting values” chosen by the researcher (denoted by \*, , etc.)

MEM and AVE are equivalent in linear models, but are different for nonlinear models (see [Chapter 10](#sec-chap-10)). The MERV approach also overlaps with MEM and AVE, because usually we choose the mean (or median) as one of the representative values of the predictor.

In this section we focus on MERV, because it is widely used. One common choice for the “interesting values” is the quartiles of , which are reported below for the example data. The output shows the gender gap in Math Achievement for 5 values of Reading Achievement. The values of Reading Achievement are its 5 quartiles. You can think of the output as a tabular summary of [Figure 5.5](#fig-visreg-1).

# Install the emmeans package if you haven't already done so  
# install.packages("emmeans")  
  
# Load the package into memory  
library(emmeans)  
  
# Fit the model using R's formula syntax for interaction '\*'  
mod6 <- lm(achmat12 ~ gender\*achrdg12, data = NELS)  
  
# Use the emmeans function to get the gender means on math, broken down by reading  
gap <- emmeans(mod6,   
 specs = "gender",  
 by = "achrdg12",   
 cov.reduce = quantile)  
  
# Test whether the differences are significant  
contrast(gap, method = "pairwise")

achrdg12 = 31.8:  
 contrast estimate SE df t.ratio p.value  
 Female - Male -7.74 1.637 496 -4.728 <.0001  
  
achrdg12 = 51.3:  
 contrast estimate SE df t.ratio p.value  
 Female - Male -4.27 0.593 496 -7.207 <.0001  
  
achrdg12 = 57.0:  
 contrast estimate SE df t.ratio p.value  
 Female - Male -3.25 0.529 496 -6.138 <.0001  
  
achrdg12 = 61.7:  
 contrast estimate SE df t.ratio p.value  
 Female - Male -2.41 0.658 496 -3.659 0.0003  
  
achrdg12 = 68.1:  
 contrast estimate SE df t.ratio p.value  
 Female - Male -1.28 0.967 496 -1.321 0.1872

**Please use the output to make a conclusion about the levels of Reading Achievement for which the gender gap was significant. Please be prepared to share your answer in class!**

Note that the computations going on “under the hood” when testing marginal effects can get pretty complicated. You can read more details here: <https://cran.r-project.org/web/packages/emmeans/vignettes/basics.html>

### 5.4.2 Simple trends

In this section we test whether the slope of each of the simple trends is significantly different from zero. Like the marginal effects in the previous section, this is information was not entirely available in the summary(lm) output discussed in [Section 5.3](#sec-binary-continuous-interaction-5). From the summary(lm) output we learned the following two points about the simple trends:

* The regression coefficient on Reading told us about the relationship between Math and Reading, simply for the group designated as zero on the binary predictor (simply for females).
* The regression coefficient on the interaction term told us whether the simple trends differ for the two groups (e.g., whether the simple trend for males differs from the simple trend for females).

Note that what is missing, or implicit, is a test of whether the simple trend for males is different from zero. This test is of less relevance to our example (since we are treating Reading as the moderator), but we consider it for illustrative purposes.

The tests of the simple trends for the example data are reported below. As stated, these aren’t super interesting in the context of our example, but you should **check your understanding of simple trends by writing down an interpretation of the output below**.

# The regression coefficients on reading, broken down by gender  
simple\_slopes <- emtrends(mod6, var = "achrdg12", specs = "gender")  
test(simple\_slopes)

gender achrdg12.trend SE df t.ratio p.value  
 Female 0.728 0.0470 496 15.487 <.0001  
 Male 0.550 0.0451 496 12.208 <.0001

### 5.4.3 Summary

When making inferences about an interaction:

* If the interaction isn’t significant in the summary(lm) output, we stop there. But if the interaction is significant, we may want to report more information about how the focal relationship depends on the moderator.
* When the focal predictor is categorical, we can follow-up a significant interaction by taking a closer look at the statistical significance of the marginal effects (e.g, how the gender gap in Math changes as a function of Reading)
* When the focal predictor is continuous, we can follow-up a significant interaction by taking a closer look at the statistical significance of the simple trends / simple slopes.

## 5.5 Two continuous predictors

At this point, we have covered the main ideas behind two-way interactions. In this section and the next, we apply these ideas to different combinations of predictors variables. In this section we address interactions between two continuous predictors. In the next section we address two categorical predictors. In both cases, the regression equation and overall interpretation is the same as the previous sections – e.g., the relationship between and changes as a function of . However, there are also some special details that crop up in these different settings.

In this section we will address:

* The importance of centering the two continuous predictors. Centering helps us interpret the “main effects” (i.e., the regression coefficients on the individual predictors).
* How to follow-up a significant interaction using simple trends. As was the case for a categorical and a continuous predictor, this helps us interpret the interaction in more detail.

First, we introduce an new example.

### 5.5.1 Another NELS example

To illustrate an interaction between two continuous predictors, let’s replace Gender with SES in our previous analysis. Apologies that this new example is mainly for convenience and doesn’t represent a great research question about, e.g., why the relationships between Math and Reading might change as a function of SES.

The overall approach with SES as the moderator is illustrated in [Figure 5.6](#fig-readingXses-5) below. It presents the simple trends for Math and Reading at three values of SES. The overall situation should hopefully feel pretty familiar from the previous sections. The displayed values of SES (9, 19, and 28) are its 10th, 50th, and 90th percentiles, which is the default choice for the software we are using for plotting. For visual clarity, the confidence bands are not shown.

#Interaction without centering   
mod7 <- lm(achmat12 ~ achrdg12\*ses, data = NELS)  
  
# Note that band = F removes the confidence intervals  
visreg(mod7,   
 xvar = "achrdg12",   
 by = "ses",   
 overlay = TRUE,   
 band = F)

|  |
| --- |
| Figure 5.6: Math (achmat), Reading (achrdg), and SES |

### 5.5.2 Centering the predictors

The take home message of this section is that you should center continuous predictors when their interaction is included in the model. There are two reasons:

* Just like [Section 5.3](#sec-binary-continuous-interaction-5), centering makes it easier to interpret the “main effects” (i.e., the regression coefficients on the individual predictors).
* Centering can make the estimates of the main effects more precise. This is because centering can reduce the correlation between the individual predictors and the interaction term. Remember from [Section 3.7.1](#sec-inference-for-coeffecients-3) that highly correlated predictors lead to less precise estimates of the regression coefficients. We discuss this problem in more detail in **?@sec-multicollinearity-6**, but for now we just discuss how centering helps us avoid the problem.

First lets consider how centering facilitates interpretation. Begin by noting that the coefficients and in the regression model

can be interpreted in terms of the following simple trends:

The first equation shows us that is the slope of relationship between and , when is equal to zero – in our example, the relationship between Math and Reading when SES is equal to zero. Similarly, the second equation shows us that is the slope of relationship between and when is equal to zero – in our example, the relationship between Math and SES when Reading is equal to zero.

In general, the value of zero may not be meaningful for continuous predictors. But, when the predictor is *centered* (i.e., a deviation score), the value of zero is always the mean of the original variable. For example, if we centered SES, then would represent the relationship between Math and Reading for students with average SES. Similar considerations apply if we treat Reading as the moderator instead of SES. Note that this is just the same trick as [Section 5.3](#sec-binary-continuous-interaction-5), but this time both predictors are continuous and so both can be centered.

The second main reason for centering is a bit more technical. It has to do with reducing the correlation between the predictors and their interaction. In general, the interaction term will be correlated with both predictors if (a) the predictors themselves are correlated and (b) both predictors take on strictly positive (or strictly negative) values. Highly correlated predictors lead to redundant information the model, so we want to avoid this situation (this is technically called *multicollinearity* and we discuss it in more detail in a **?@sec-multicollinearity-6**).

To see how centering can reduce the correlation between the predictors and their interaction, let’s take a look at [Figure 5.7](#fig-centering-5). The left hand panel shows the relationship between SES and its interaction with Reading. We can see that they are highly correlated. This is because (a) SES and Reading are themselves correlated, and (b) both SES and Reading take on strictly positive values. As mentioned above, the interaction term will be correlated with both predictors whenever these two conditions hold. (The figure shows the correlation just for SES and the interaction, but the same situation holds for Reading.)

# Correlation without centering  
r <- cor(ses, achrdg12\*ses)  
  
# Plot  
par(mfrow = c(1, 2))  
title <- paste0("correlation = ", round(r, 3))  
plot(ses, achrdg12\*ses,   
 col = "#4B9CD3",   
 main = title,   
 xlab = "SES",   
 ylab = "SES X Reading")  
  
# Correlation with centering  
achrdg12\_dev <- achrdg12 - mean(achrdg12)  
ses\_dev <- ses - mean(ses)  
r <- cor(ses\_dev, achrdg12\_dev\*ses\_dev)  
  
# Plot  
title <- paste0("correlation = ", round(r, 3))  
plot(ses\_dev,   
 achrdg12\_dev\*ses\_dev,   
 col = "#4B9CD3",   
 main = title,   
 xlab = "SES Centered",   
 ylab = "SES Centered X Reading Centered")

|  |
| --- |
| Figure 5.7: Correlation Between SES and SES X Reading, With and Without Centering |

We can see in the right hand panel of [Figure 5.7](#fig-centering-5) how centering the two predictors “breaks” the linear relationship between SES and its interaction with Reading. After centering, the relationship between SES and its interaction is now highly non-linear, and the correlation is approximately zero. Again, the same is true for the relationship between Reading and the interaction, but the figure only shows the situation for SES. The upshot of all this is that centering reduces multicollinearity between the “main effects” of the predictors and their interaction.

Below we show the output for two regression models. Both models regress Math on Reading, SES, and their interaction. The first model does not center the predictors, but the second model does (the \_dev notation denotes the centered predictors).

The main difference between the models is that SES is a significant predictor in the centered model but not in the “un-centered” model. This is because the main effect of SES has a different meaning in the centered model, and it is also less correlated with the interaction. This is also true for Reading, but the differences between the two sets of results are less pronounced for Reading.

# Without centering  
mod7 <- lm(achmat12 ~ achrdg12\*ses)  
summary(mod7)

Call:  
lm(formula = achmat12 ~ achrdg12 \* ses)  
  
Residuals:  
 Min 1Q Median 3Q Max   
-17.1336 -3.8944 0.7278 4.1301 15.0153   
  
Coefficients:  
 Estimate Std. Error t value Pr(>|t|)   
(Intercept) 25.849200 5.389797 4.796 2.14e-06 \*\*\*  
achrdg12 0.511601 0.099427 5.145 3.85e-07 \*\*\*  
ses -0.100114 0.291214 -0.344 0.731   
achrdg12:ses 0.004270 0.005196 0.822 0.412   
---  
Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
  
Residual standard error: 6.031 on 496 degrees of freedom  
Multiple R-squared: 0.4184, Adjusted R-squared: 0.4149   
F-statistic: 118.9 on 3 and 496 DF, p-value: < 2.2e-16

# With centering  
mod8 <- lm(achmat12 ~ achrdg12\_dev\*ses\_dev)  
summary(mod8)

Call:  
lm(formula = achmat12 ~ achrdg12\_dev \* ses\_dev)  
  
Residuals:  
 Min 1Q Median 3Q Max   
-17.1336 -3.8944 0.7278 4.1301 15.0153   
  
Coefficients:  
 Estimate Std. Error t value Pr(>|t|)   
(Intercept) 56.826225 0.286906 198.065 <2e-16 \*\*\*  
achrdg12\_dev 0.590313 0.036103 16.351 <2e-16 \*\*\*  
ses\_dev 0.137305 0.041485 3.310 0.001 \*\*   
achrdg12\_dev:ses\_dev 0.004270 0.005196 0.822 0.412   
---  
Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
  
Residual standard error: 6.031 on 496 degrees of freedom  
Multiple R-squared: 0.4184, Adjusted R-squared: 0.4149   
F-statistic: 118.9 on 3 and 496 DF, p-value: < 2.2e-16

**Please provide an interpretation of all four regression coefficients in the centered model. Your interpretations should make reference to the situation where one or both predictors are equal to zero (see** [**Equation 5.9**](#eq-simple-5) **above) and should also mentioned the interpretation of the value of zero for the centered variables.**

Note that the interaction and the R-squared are the same in the centered and uncentered models. This is discussed in more detail in the extra material at the end of this section ([Section 5.6](#sec-two-categorical-predictors-5)), but it is sufficient to note that centering only affects the interpretation of the main effects (and the intercept, of course).

### 5.5.3 Simple trends

Centering helps us interpret the main effects of the individual predictors, but we haven’t yet discussed how to interpret the interaction term. As shown in [Figure 5.6](#fig-readingXses-5), the overall situation is not that different than with a binary predictor.

The usual way to follow up a significant interaction between two continuous is using the MERV approach discussed in [Section 5.4](#sec-inference-for-interactions-5). Using this approach, we consider the focal relationship for some “interesting values” of the moderator. As with MERV, the choice of values of the moderator is up to the researcher, but some usual choices are

* The quartiles of the moderator
* M 1 SD of the moderator
* A selection of percentiles of the moderator (The visreg plot in [Figure 5.6](#fig-readingXses-5) uses the 10th, 50th, and 90th)

These are all doing very similar things, so choosing among them usually isn’t super important.

Although the interaction between Reading and SES was not significant in our example model, let’s break down the interaction using SES as the moderator, just to see how this approach works. The output below presents the simple slopes for the three values of SES shown in [Figure 5.6](#fig-readingXses-5) (i.e., the 10th, 50th, and 90th percentiles). We can see in the output that the simple slopes are all different from zero. (And the non-significant interaction in the summary(lm) output tells us that the slopes are not statistically different from one another.)

# Break down interaction with SES as moderator  
simple\_slopes <-emtrends(mod7,   
 var = "achrdg12",   
 specs = "ses",   
 at = list(ses = c(9, 19, 28)))  
test(simple\_slopes)

ses achrdg12.trend SE df t.ratio p.value  
 9 0.550 0.0583 496 9.429 <.0001  
 19 0.593 0.0365 496 16.251 <.0001  
 28 0.631 0.0639 496 9.878 <.0001

### 5.5.4 Summary

When regressing an outcome on two continuous predictors and their interaction, the overall interpretation of the model is same as discussed in [Section 5.3](#sec-binary-continuous-interaction-5), but:

* It is useful to center both predictors, to facilitate the interpretation of the main effects (i.e., regression coefficients on the individual predictors), and to improve the precision of the main effects (i.e., reduce multicollinearity).
* When following up a significant interaction, the usual approach is to report the simple trends for the focal variables at a selection of values of the moderator (e.g., a selection of percentiles). The example illustrated how to do this even though the interaction was not significant, but you shouldn’t follow up a non-significant interaction.

### 5.5.5 Extra: How centering works\*

It might seem that centering both predictors is a bit dubious – how can we just change the predictors in the model? This sections shows that using the centered or the un-centered predictors doesn’t make a difference in term of what predictors are in the model, it just changes the interpretation of the main effects (and intercept).

Using for the centered variables, simple algebra shows:

The second line of the equation shows that we are not changing what we regress on – i.e., the predictors are still and . We are changing the interpretation of the main effects (and intercept), but this is exactly the purpose of this approach. Also note centering does not change the regression coefficient for the interaction at all. So, we get main effects (and intercept) that can be more easily interpreted, and the same interaction

## 5.6 Two categorical predictors

This section addresses interactions between two categorical predictors. Up until now, we have looked at interactions only for categorical predictors that are dichotomous. In this section, we address an example in which one of the categorical predictors has more than two levels. This requires combining what we learned about contrast coding ([Chapter 4](#sec-chap-4)) with what we have learned about interactions. One nice aspect of interactions among categorical predictors is that we usually don’t need to use procedures like marginal effects to follow up significant interactions, so long as we make good use of contrast coding.

In experimental (as opposed to observational) settings, interactions among categorical predictors fall under the much larger topic of ANOVA and experimental design. The analysis we look at in this section is a two-way between-subjects ANOVA, meaning that there are two categorical predictors considered, as well as their interaction, and both predictors are cross-sectional. ANOVA is a big topic and is not the focus of this course. However, we will discuss how to summarize the results of our analysis in an ANOVA table, and consider how this differs from the standard regression approach.

### 5.6.1 An example from ECLS

For this topic we will switch over to the ECLS data and examine how SES and Pre-K attendance interact to predict Math Achievement at the beginning of Kindergarten. The variables we will examine are

* Math Achievement at the beginning of K (c1rmscal). This is the number of correct questions on a test with approximately 70 items.
* Whether the child attended Pre-K (p1center). This is a binary variable that indicates pre-K attendance.
* SES, coded as quintiles (wksesq5). We will denote this variable as SES, but keep in mind it is quintiles in this example (e.g., SES = 1 are the respondents with SES between the minimum and the first quintile).

Coding SES as quintiles allows us to consider it as a categorical predictor with 5 levels. This is a widely-used practice, because SES often has non-linear relationships with outcome variables of interest, and these relationships can be more easily captured by treating SES as a categorical variable. This approach to SES is also convenient for our illustration of interactions between categorical predictors.

In this analysis, our focus will be whether the “effect” of Pre-K on Math Achievement depends on (i.e., is moderated by) the child’s SES. Please note that I will use the term “effect” in this section to simplify language, but we know that Pre-K attendance was not randomly assigned in ECLS, so please keep in mind that this terminology is not strictly correct.

The relationship among the three variables is summarized in the visreg plot below. We can see that the effect of Pre-K on Math Achievement appears to differ as a function of SES – i.e., it appears that there is an interaction between Pre-K and SES. Our goal in this section is to produce an analysis corresponding to the figure.

#   
load("ECLS2577.Rdata")  
ecls$prek <- factor(2 - ecls$p1center)  
ecls$wksesq5 <- factor(ecls$wksesq5)  
mod <- lm(c1rmscal ~ prek\*wksesq5, data = ecls)  
visreg::visreg(mod, xvar = "wksesq5", by = "prek",   
 partial = F, rug = F, overlay = T,   
 strip.names = T, xlab = "SES",   
 ylab = "Math Achievement in K")

|  |
| --- |
| Figure 5.8: Math Achievement, Pre-K Attendence, and SES |

**Before moving on, please take a moment to write down your interpretation of** [**Figure 5.8**](#fig-prek-by-ses-5)**, focussing on how it illustrates an interaction between SES and Pre-K. Additionally, please describe how the figure would be different if there was no interaction between Pre-K and SES**.

### 5.6.2 The “no-interaction” model

As in [Section 5.2](#sec-binary-continuous-5), we will start with a model that includes only the main effects of SES and Pre-K. Seeing where that model “goes wrong” is a good way of understanding the interaction between the two predictors.

In order to represent a model with multiple categorical predictors, it is helpful to change our notation from the usual and to the more informative “variable names” notation:

In this notation, the predictor variables are indicators (binary dummies). The variable is just the indicator for Pre-K attendance, as defined above. The variable is an indicator for the j-th quintile of SES.

Both predictors use reference-group coding, as discussed in [Chapter 4](#sec-chap-4). For , reference-group coding is implied because it is a binary indicator. For , reference-group coding is accomplished by omitting the binary dummy for the first quintile (i.e., the first quintile is the reference group).

We can interpret the coefficients in this model using the same two-step procedure described in [Chapter 4](#sec-chap-4). Since there are many terms in the model, things are going to start getting messy quickly, so brace yourself for some long equations (but simple math!).

The main points about the interpretation of this model are as follows.

* The intercept is the predicted value of Math Achievement for students in the first SES quintile who did not attend Pre-K. This corresponds to the blue line in the first column of [Figure 5.8](#fig-prek-by-ses-5).
* The effect of Pre-K attendance for students in the first SES quintile is equal to . This corresponds to the difference between the red and blue lines in the first column of [Figure 5.8](#fig-prek-by-ses-5).
* Because the model in [Equation 5.10](#eq-prek-ses-5) does not include an interaction, we already know that it implies that the effect of Pre-K is constant over levels of SES. Below we show that effect of Pre-K for SES = 2 is the same as the effect for SES = 1. The same approach can be used to show the effect is constant over all levels of SES. Note that while the model assumes the effect of Pre-K is constant over levels of SES, this is actually inconsistent with what is shown in [Figure 5.8](#fig-prek-by-ses-5). We will improve on this model by adding an interaction in the following section.

This equation says that the difference between the red and blue lines in the second column of [Figure 5.8](#fig-prek-by-ses-5) is the same as the difference in the first column – i.e., they both equal . This is what it means for there to be no interaction between two categorical predictors.

If you want more practice with this, you can show that [Equation 5.10](#eq-prek-ses-5) implies the effect of Pre-K is constant over all levels of SES. Additionally, you can use the 2-step approach to show that the effect of SES is constant over levels of Pre-K attendance.

### 5.6.3 Adding the interaction(s)

We have just seen that [Equation 5.10](#eq-prek-ses-5) implies that the effect of Pre-K is constant over levels of SES, and vise versa. In order to address our research question about whether the relationship between Pre-K attendance and Math Achievement depends on children’s SES, we will need to add something to the model – an interaction (surprise!).

We know that interactions are just products (multiplication) of predictor variables. Since SES is represented as 4 dummies, this means we need 4 products to represent the interaction of Pre-K with SES. The resulting model can be written:

As you can see, we have a lot of predictors in this model! Although we are only considering two distinct “conceptual” predictors, we have 9 coefficients in our regression model (+ the intercept).

Again, there are a few main things to notice:

* The interpretation of the intercept has not changed. It still corresponds to the blue line in the first column of [Figure 5.8](#fig-prek-by-ses-5).
* The regression coefficient on is still the “effect” of Pre-K for students in the first SES quintile (i.e., the difference between the red and blue line in the first column of [Figure 5.8](#fig-prek-by-ses-5)). This is because all the variables are equal to zero for students in the first SES quintile, and so all of the interaction terms in [Equation 5.11](#eq-prek-ses-5b) are equal to zero.
* The effect of Pre-K is no longer constant over levels of SES. Again we will focus on SES = 2, but the same approach works for the other levels of SES.

The last line shows that the “effect” of Pre-K for students in the second SES quintile is . This is not the same as the effect for students in the first quintile, which was just . In other words, the difference between the red and blue lines in the first column of [Figure 5.8](#fig-prek-by-ses-5) (i.e., ) is not equal to the difference in the second column (i.e., ) unless the interaction is equal to zero (i.e., ).

The same approach shows that the effect of Pre-K at each level of SES results in a similar equation:

This pattern makes it clear that, to isolate the interactions (i.e., through ), we need to subtract off – i.e., we need to subtract off the effect of Pre-K for students in the first SES quintile. In anology with reference group coding for single predictor (see [Section 4.4](#sec-reference-group-coding-4)), we can think of the “reference effect” or baseline to which the interaction terms are compared.

For example

* The interaction between Pre-K and the second SES quintile is the effect Pre-K has on Math Achievement for students in the second SES quintile, *as compared to the effect in the first SES quintile.*
* The interaction between Pre-K and the third SES quintile is the effect Pre-K has on Math Achievement for students in the 3rd SES quintile, *as compared to the effect in the first SES quintile.*
* etc. etc.

Mathematically, the interaction terms are represented as “differences-in-differences”. For example,

This looks quite complicated but it is just an extension of reference-group coding. This equation is saying that the “reference effect” or “baseline” for interpreting the interaction () is the effect of Pre-K in the first SES quintile (i.e., ). As noted above, all of the interaction terms have the same reference effect.

### 5.6.4 Back to the example

That last section was a lot to take in, so let’s put some numbers on the page to check our understanding. The output below shows the summary for a model that regresses Math Achievement on Pre-K, SES, and their interaction. **Please write down an interpretation of magnitude, direction, and statistical significance of each regression coefficient in this output (including the intercept), and be prepared to share your answers in class.** Remember that wksesq5 is the variable code for the SES quintiles – the digit that follows the variable code indicates the level of variable. It may be helpful to refer to [Figure 5.8](#fig-prek-by-ses-5) when interpreting the coefficients.

mod <- lm(c1rmscal ~ prek\*wksesq5, data = ecls)  
summary(mod)

Call:  
lm(formula = c1rmscal ~ prek \* wksesq5, data = ecls)  
  
Residuals:  
 Min 1Q Median 3Q Max   
-16.7682 -4.7682 -0.9755 3.9545 31.2318   
  
Coefficients:  
 Estimate Std. Error t value Pr(>|t|)   
(Intercept) 16.0455 0.7355 21.816 < 2e-16 \*\*\*  
prek1 -0.3735 0.9601 -0.389 0.69732   
wksesq52 2.2931 0.9570 2.396 0.01663 \*   
wksesq53 2.9300 0.9127 3.210 0.00134 \*\*   
wksesq54 4.6310 0.9439 4.906 9.87e-07 \*\*\*  
wksesq55 7.2990 1.0343 7.057 2.19e-12 \*\*\*  
prek1:wksesq52 1.0638 1.2118 0.878 0.38012   
prek1:wksesq53 2.1087 1.1544 1.827 0.06785 .   
prek1:wksesq54 1.6715 1.1684 1.431 0.15267   
prek1:wksesq55 2.7972 1.2340 2.267 0.02349 \*   
---  
Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
  
Residual standard error: 6.9 on 2567 degrees of freedom  
Multiple R-squared: 0.1621, Adjusted R-squared: 0.1592   
F-statistic: 55.19 on 9 and 2567 DF, p-value: < 2.2e-16

### 5.6.5 The ANOVA approach

The output in the previous section is detailed enough that it is not usually required to follow-up a significant interaction among categorical predictors using marginal effects. However, the summary output still omits some information we might be interested in. For example, the Pre-K indicator in the above output tells us the effect of Pre-K, but only for children in the first SES quintile. We might also want to know about the overall effect of Pre-K across levels of SES – i.e., is there a significant difference in Math Achievement for students who attended Pre-K, after controlling for their level of SES? Similarly, what is the overall main effect of SES?

One way to summarize the main effects of Pre-K and SES, as well as their interaction, by asking how much variance they explain after controlling for the other predictors in the model. This is the ANOVA approach we discussed last semester, but now applied to two categorical predictors.

The ANOVA table for our example is below, and it is followed by the R-squared coefficients for each predictor, which are called “eta-squared” () in the context of ANOVA. These R-squared (eta-squared) coefficients tell us what proportion of the variance in Math Achievement is attributable to the main effects and the interaction.

# ANOVA Table  
anova(mod)

Analysis of Variance Table  
  
Response: c1rmscal  
 Df Sum Sq Mean Sq F value Pr(>F)   
prek 1 3434 3434.3 72.1437 <2e-16 \*\*\*  
wksesq5 4 19914 4978.4 104.5805 <2e-16 \*\*\*  
prek:wksesq5 4 299 74.7 1.5701 0.1795   
Residuals 2567 122198 47.6   
---  
Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

# R-squared (eta-squared)  
#install.packages("effectsize")  
effectsize::eta\_squared(mod, partial = F)

# Effect Size for ANOVA (Type I)  
  
Parameter | Eta2 | 95% CI  
--------------------------------------  
prek | 0.02 | [0.01, 1.00]  
wksesq5 | 0.14 | [0.12, 1.00]  
prek:wksesq5 | 2.05e-03 | [0.00, 1.00]  
  
- One-sided CIs: upper bound fixed at [1.00].

**Please write down your interpretation of the ANOVA table and R-squared (eta-squared) coefficients and be prepared to share you thoughts in class.** Note that the ANOVA output leads to different conclusions than the regression output above. We will discuss the discrepancies between the ANOVA and regression output in class.

Let’s end this discussion of ANOVA with two qualifications.

First, I should clarify that the term “main effect” has a somewhat different meaning in ANOVA as compared to regression. In the regression examples, we talked about the effect of a predictor, *conditional on* the other predictor being zero. In ANOVA stuff above, we instead talked about the average or overall effect of a predictor, while holding the other predictor constant. These two interpretations are related but not the same, and in the ANOVA literature, “main effect” usually means the average or overall effect.

Second, some people claim that it is bad practice to interpret main effects *qua* average effects in the presence of an interaction. The basic argument is that we shouldn’t report the average effect when the “real message” of the interaction is that the effect changes as a function of the other predictor. I think that main effects and interactions aren’t really incompatible concepts, especially if we are talking about conditional main effects rather than average main effects. But, you should be aware that this topic is debated and you are free to make up your own mind (as always!).

## 5.7 Workbook

This section collects the questions asked in this chapter. The lessons for this chapter will focus on discussing these questions and then working on the exercises in [Section 5.8](#sec-exercises-5). The lesson will **not** be a lecture that reviews all of the material in the chapter! So, if you haven’t written down / thought about the answers to these questions before class, the lesson will not be very useful for you. Please engage with each question by writing down one or more answers, asking clarifying questions about related material, posing follow up questions, etc.

[Section 5.1](#sec-example-5)

* What does the following plot is telling us about the relationships among the three variables. In particular:
  + Does the gender gap in Math Achievement change as a function of Reading Achievement?
  + Is the relationship between Math Achievement and Reading Achievement the the same for males and females?
  + What do the results mean for gender equality in Math and STEM education?

mod1 <- lm(achmat12[females] ~ achrdg12[females])  
mod2 <- lm(achmat12[males] ~ achrdg12[males])  
  
# Plot reading and math for females  
plot(achrdg12[females],   
 achmat12[females],   
 xlab = "Reading",   
 ylab = "Math")  
  
abline(mod1, lwd = 2)  
  
# Add points and line for males  
points(achrdg12[males],   
 achmat12[males],   
 col = "#4B9CD3", pch = 2)  
  
abline(mod2, col = "#4B9CD3", lwd = 2)  
  
# Add a legend  
legend(x = "topleft",   
 legend = levels(gender),  
 pch = c(1, 2),   
 col = c(1, "#4B9CD3"))

|  |
| --- |
|  |

[Section 5.2](#sec-binary-continuous-5)

* No interaction model: The regression coefficients for the example data are shown below. Please use these numbers to provide an interpretation of the simple trends and the gender gap in Math Achievement for the NELS example. Don’t worry about statistical significance, just focus on the meaning of the coefficients.

mod3 <- lm(achmat12 ~ achrdg12 + gender, data = NELS)  
summary(mod3)

Call:  
lm(formula = achmat12 ~ achrdg12 + gender, data = NELS)  
  
Residuals:  
 Min 1Q Median 3Q Max   
-19.2448 -3.6075 0.3968 3.9836 15.5606   
  
Coefficients:  
 Estimate Std. Error t value Pr(>|t|)   
(Intercept) 19.98122 1.86278 10.727 < 2e-16 \*\*\*  
achrdg12 0.63551 0.03275 19.404 < 2e-16 \*\*\*  
genderMale 3.50166 0.52473 6.673 6.69e-11 \*\*\*  
---  
Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
  
Residual standard error: 5.839 on 497 degrees of freedom  
Multiple R-squared: 0.4538, Adjusted R-squared: 0.4516   
F-statistic: 206.4 on 2 and 497 DF, p-value: < 2.2e-16

Here is an example to help you get started:

* Simply for females, the regression of Math Achievement on Reading Achievement has an intercept equal to 19.98 and a slope equal to 0.64. The intercept tells us that a female student with Reading Achievement score of 0% is expected to have a Math Achievement score of 19.98% (the units of the two tests are percent correct). Since the lowest value of Reading Achievement in our example is about 35%, the intercept is not very meaningful for these data. The regression slope tells us that, for females, a 1 unit increase in Reading Achievement is associated with a .64 unit increase in Math Achievement.
* Simply for males, ….
* The gender gap in Math Achievement was equal to …

[Section 5.3](#sec-binary-continuous-interaction-5)

* Interaction model: The regression coefficients for the example data are shown below. Please use these numbers to provide an interpretation of the interaction between Gender and Reading. Don’t worry about statistical significance, just focus on the meaning of the coefficients.

# hard code the interaction term  
genderXachrdg12 <- (as.numeric(gender) - 1) \* achrdg12  
  
# Rund the model with the interaction included  
mod4 <- lm(achmat12 ~ achrdg12 + gender + genderXachrdg12)  
summary(mod4)

Call:  
lm(formula = achmat12 ~ achrdg12 + gender + genderXachrdg12)  
  
Residuals:  
 Min 1Q Median 3Q Max   
-19.0582 -3.7864 0.5014 4.0775 16.2889   
  
Coefficients:  
 Estimate Std. Error t value Pr(>|t|)   
(Intercept) 14.80310 2.64922 5.588 3.8e-08 \*\*\*  
achrdg12 0.72824 0.04702 15.487 < 2e-16 \*\*\*  
genderMale 13.39328 3.65828 3.661 0.000278 \*\*\*  
genderXachrdg12 -0.17794 0.06514 -2.732 0.006524 \*\*   
---  
Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
  
Residual standard error: 5.801 on 496 degrees of freedom  
Multiple R-squared: 0.4619, Adjusted R-squared: 0.4586   
F-statistic: 141.9 on 3 and 496 DF, p-value: < 2.2e-16

* Interaction model using the centered continuous predictor: Please write down your interpretation of the intercept and the regression coefficient for Gender in the regression output below.

# compute the deviation scores for reading  
reading\_dev <- achrdg12 - mean(achrdg12, na.rm = T)   
  
# Run the interaction model as above  
genderXreading\_dev <- (as.numeric(gender) - 1) \* reading\_dev  
  
mod5 <- lm(achmat12 ~ reading\_dev + gender + genderXreading\_dev)  
summary(mod5)

Call:  
lm(formula = achmat12 ~ reading\_dev + gender + genderXreading\_dev)  
  
Residuals:  
 Min 1Q Median 3Q Max   
-19.0582 -3.7864 0.5014 4.0775 16.2889   
  
Coefficients:  
 Estimate Std. Error t value Pr(>|t|)   
(Intercept) 55.29439 0.35127 157.411 < 2e-16 \*\*\*  
reading\_dev 0.72824 0.04702 15.487 < 2e-16 \*\*\*  
genderMale 3.49930 0.52135 6.712 5.26e-11 \*\*\*  
genderXreading\_dev -0.17794 0.06514 -2.732 0.00652 \*\*   
---  
Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
  
Residual standard error: 5.801 on 496 degrees of freedom  
Multiple R-squared: 0.4619, Adjusted R-squared: 0.4586   
F-statistic: 141.9 on 3 and 496 DF, p-value: < 2.2e-16

[Section 5.4](#sec-inference-for-interactions-5)

* The output shows the gender gap in Math Achievement for 5 values of Reading Achievement. The values of Reading Achievement are its 5 quartiles. You can think of the output as a tabular summary of [Figure 5.5](#fig-visreg-1). Please use the output to make a conclusion about the levels of Reading Achievement for which the gender gap was significant. Please be prepared to share your answer in class

# Install the emmeans package if you haven't already done so  
# install.packages("emmeans")  
  
# Load the package into memory  
library(emmeans)  
  
# Fit the model using R's formula syntax for interaction '\*'  
mod6 <- lm(achmat12 ~ gender\*achrdg12, data = NELS)  
  
# Use the emmeans function to get the gender means on math, broken down by reading  
gap <- emmeans(mod6,   
 specs = "gender",  
 by = "achrdg12",   
 cov.reduce = quantile)  
  
# Test whether the differences are significant  
contrast(gap, method = "pairwise")

achrdg12 = 31.8:  
 contrast estimate SE df t.ratio p.value  
 Female - Male -7.74 1.637 496 -4.728 <.0001  
  
achrdg12 = 51.3:  
 contrast estimate SE df t.ratio p.value  
 Female - Male -4.27 0.593 496 -7.207 <.0001  
  
achrdg12 = 57.0:  
 contrast estimate SE df t.ratio p.value  
 Female - Male -3.25 0.529 496 -6.138 <.0001  
  
achrdg12 = 61.7:  
 contrast estimate SE df t.ratio p.value  
 Female - Male -2.41 0.658 496 -3.659 0.0003  
  
achrdg12 = 68.1:  
 contrast estimate SE df t.ratio p.value  
 Female - Male -1.28 0.967 496 -1.321 0.1872

* The test of the slopes of the simple trends for the example are reported below. As previously stated, these aren’t super interesting in the context of our example, but you should check your understanding of simple trends by writing down an interpretation of the output below.

# The regression coefficients on reading, broken down by gender  
simple\_slopes <- emtrends(mod6, var = "achrdg12", specs = "gender")  
test(simple\_slopes)

gender achrdg12.trend SE df t.ratio p.value  
 Female 0.728 0.0470 496 15.487 <.0001  
 Male 0.550 0.0451 496 12.208 <.0001

[Section 5.5](#sec-two-continuous-predictors-5)

* Please provide an interpretation of all four regression coefficients in the centered model. Your interpretations should make reference to the situation where one or both predictors are equal to zero (see [Equation 5.9](#eq-simple-5) above) and should also mentioned the interpretation of the value of zero for the centered variables.

# With centering  
mod8 <- lm(achmat12 ~ achrdg12\_dev\*ses\_dev)  
summary(mod8)

Call:  
lm(formula = achmat12 ~ achrdg12\_dev \* ses\_dev)  
  
Residuals:  
 Min 1Q Median 3Q Max   
-17.1336 -3.8944 0.7278 4.1301 15.0153   
  
Coefficients:  
 Estimate Std. Error t value Pr(>|t|)   
(Intercept) 56.826225 0.286906 198.065 <2e-16 \*\*\*  
achrdg12\_dev 0.590313 0.036103 16.351 <2e-16 \*\*\*  
ses\_dev 0.137305 0.041485 3.310 0.001 \*\*   
achrdg12\_dev:ses\_dev 0.004270 0.005196 0.822 0.412   
---  
Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
  
Residual standard error: 6.031 on 496 degrees of freedom  
Multiple R-squared: 0.4184, Adjusted R-squared: 0.4149   
F-statistic: 118.9 on 3 and 496 DF, p-value: < 2.2e-16

[Section 5.6](#sec-two-categorical-predictors-5)

* Please take a moment to write down your interpretation of the figure below, focussing on how it illustrates an interaction between SES and Pre-K. Additionally, please describe how the figure would be different if there was no interaction between Pre-K and SES.

#   
load("ECLS2577.Rdata")  
ecls$prek <- factor(2 - ecls$p1center)  
ecls$wksesq5 <- factor(ecls$wksesq5)  
mod <- lm(c1rmscal ~ prek\*wksesq5, data = ecls)  
visreg::visreg(mod, xvar = "wksesq5", by = "prek",   
 partial = F, rug = F, overlay = T,   
 strip.names = T, xlab = "SES",   
 ylab = "Math Achievement in K")

|  |
| --- |
|  |

* The output below shows the summary for a model that regresses Math Achievement on Pre-K, SES, and their interaction. Please write down an interpretation of magnitude, direction, and statistical significance of each regression coefficient in this output (including the intercept), and be prepared to share your answers in class. Remember that wksesq5 is the variable code for the SES quintiles – the digit that follows the variable code indicates the level of variable. It may be helpful to refer to the previous figure in your interpretations.

mod <- lm(c1rmscal ~ prek\*wksesq5, data = ecls)  
summary(mod)

Call:  
lm(formula = c1rmscal ~ prek \* wksesq5, data = ecls)  
  
Residuals:  
 Min 1Q Median 3Q Max   
-16.7682 -4.7682 -0.9755 3.9545 31.2318   
  
Coefficients:  
 Estimate Std. Error t value Pr(>|t|)   
(Intercept) 16.0455 0.7355 21.816 < 2e-16 \*\*\*  
prek1 -0.3735 0.9601 -0.389 0.69732   
wksesq52 2.2931 0.9570 2.396 0.01663 \*   
wksesq53 2.9300 0.9127 3.210 0.00134 \*\*   
wksesq54 4.6310 0.9439 4.906 9.87e-07 \*\*\*  
wksesq55 7.2990 1.0343 7.057 2.19e-12 \*\*\*  
prek1:wksesq52 1.0638 1.2118 0.878 0.38012   
prek1:wksesq53 2.1087 1.1544 1.827 0.06785 .   
prek1:wksesq54 1.6715 1.1684 1.431 0.15267   
prek1:wksesq55 2.7972 1.2340 2.267 0.02349 \*   
---  
Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
  
Residual standard error: 6.9 on 2567 degrees of freedom  
Multiple R-squared: 0.1621, Adjusted R-squared: 0.1592   
F-statistic: 55.19 on 9 and 2567 DF, p-value: < 2.2e-16

* Please write down your interpretation of the ANOVA table and R-squared (eta-squared) coefficients below and be prepared to share you thoughts in class. Note that the ANOVA output leads to different conclusions than the regression output above. We will discuss the discrepancies between the ANOVA and regression output in class.

# ANOVA Table  
anova(mod)

Analysis of Variance Table  
  
Response: c1rmscal  
 Df Sum Sq Mean Sq F value Pr(>F)   
prek 1 3434 3434.3 72.1437 <2e-16 \*\*\*  
wksesq5 4 19914 4978.4 104.5805 <2e-16 \*\*\*  
prek:wksesq5 4 299 74.7 1.5701 0.1795   
Residuals 2567 122198 47.6   
---  
Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

# R-squared (eta-squared)  
#install.packages("effectsize")  
effectsize::eta\_squared(mod, partial = F)

# Effect Size for ANOVA (Type I)  
  
Parameter | Eta2 | 95% CI  
--------------------------------------  
prek | 0.02 | [0.01, 1.00]  
wksesq5 | 0.14 | [0.12, 1.00]  
prek:wksesq5 | 2.05e-03 | [0.00, 1.00]  
  
- One-sided CIs: upper bound fixed at [1.00].

## 5.8 Exercises

These exercises provide an overview of how to compute interactions using the lm function, how to center continuous predictors, and how to follow-up significant interactions with the emmeans package. We will go through this material in class together, so you don’t need to work on it before class (but you can if you want.)

Before staring this section, you may find it useful to scroll to the top of the page, click on the “</> Code” menu, and select “Show All Code.”

### 5.8.1 Binary + continuous + interaction

There are multiple ways of implementing interactions in R.

* We can “hard code” new variables into our data (e.g., the product of a binary gender variable and reading)
* We can use R’s formula notation for single term interactions (:)
* We can use R’s formula notation for factorial interactions (\*)

The following code illustrates the three approaches and shows that they all producing the same output. In general, the \* syntax is the easiest to use, so we will stick with that one going forward. The variables used in the example are from the NELS data:

* achmat12 is Mat Achievement (percent correct on a math test) in grade 12.
* achrdg12 is Reading Achievement (percent correct on a reading test) in grade 12.
* gender is dichotomous encoding of gender with values Male and Female (it is not a binary variable, but a factor, as discussed in [Section 4.9](#sec-exercises-4).

# Interaction via hard coding  
genderXreading <- (as.numeric(gender) - 1) \* achrdg12  
mod1 <- lm(achmat12 ~ achrdg12 + gender + genderXreading)  
summary(mod1)

Call:  
lm(formula = achmat12 ~ achrdg12 + gender + genderXreading)  
  
Residuals:  
 Min 1Q Median 3Q Max   
-19.0582 -3.7864 0.5014 4.0775 16.2889   
  
Coefficients:  
 Estimate Std. Error t value Pr(>|t|)   
(Intercept) 14.80310 2.64922 5.588 3.8e-08 \*\*\*  
achrdg12 0.72824 0.04702 15.487 < 2e-16 \*\*\*  
genderMale 13.39328 3.65828 3.661 0.000278 \*\*\*  
genderXreading -0.17794 0.06514 -2.732 0.006524 \*\*   
---  
Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
  
Residual standard error: 5.801 on 496 degrees of freedom  
Multiple R-squared: 0.4619, Adjusted R-squared: 0.4586   
F-statistic: 141.9 on 3 and 496 DF, p-value: < 2.2e-16

# Interaction via `:` operator  
mod2 <- lm(achmat12 ~ achrdg12 + gender + achrdg12:gender)  
summary(mod2)

Call:  
lm(formula = achmat12 ~ achrdg12 + gender + achrdg12:gender)  
  
Residuals:  
 Min 1Q Median 3Q Max   
-19.0582 -3.7864 0.5014 4.0775 16.2889   
  
Coefficients:  
 Estimate Std. Error t value Pr(>|t|)   
(Intercept) 14.80310 2.64922 5.588 3.8e-08 \*\*\*  
achrdg12 0.72824 0.04702 15.487 < 2e-16 \*\*\*  
genderMale 13.39328 3.65828 3.661 0.000278 \*\*\*  
achrdg12:genderMale -0.17794 0.06514 -2.732 0.006524 \*\*   
---  
Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
  
Residual standard error: 5.801 on 496 degrees of freedom  
Multiple R-squared: 0.4619, Adjusted R-squared: 0.4586   
F-statistic: 141.9 on 3 and 496 DF, p-value: < 2.2e-16

# Interaction via `\*` operator  
mod3 <- lm(achmat12 ~ achrdg12\*gender)  
summary(mod3)

Call:  
lm(formula = achmat12 ~ achrdg12 \* gender)  
  
Residuals:  
 Min 1Q Median 3Q Max   
-19.0582 -3.7864 0.5014 4.0775 16.2889   
  
Coefficients:  
 Estimate Std. Error t value Pr(>|t|)   
(Intercept) 14.80310 2.64922 5.588 3.8e-08 \*\*\*  
achrdg12 0.72824 0.04702 15.487 < 2e-16 \*\*\*  
genderMale 13.39328 3.65828 3.661 0.000278 \*\*\*  
achrdg12:genderMale -0.17794 0.06514 -2.732 0.006524 \*\*   
---  
Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
  
Residual standard error: 5.801 on 496 degrees of freedom  
Multiple R-squared: 0.4619, Adjusted R-squared: 0.4586   
F-statistic: 141.9 on 3 and 496 DF, p-value: < 2.2e-16

Before moving on, check your interpretation of the coefficients in the models. In particular, what does the regression coefficient on the interaction term mean?

### 5.8.2 Centering continuous predictors

As noted in [Section 5.3](#sec-binary-continuous-interaction-5), the regression coefficient on Gender is not very interpretable when there is an interaction in the model. In the above output, the coefficient on gender tells us the gender gap in Math Achievement when achrdg12 = 0. We can fix this issue by re-scaling achrdg12 so that zero has a meaningful value. One widely used approach is to center achrdg12 at its mean. When a variable is centered at its mean it is called a *deviation score.*

Let’s see what happens to our regression output when we use deviation scores for achrdg12 instead of the “raw” score

# Re-run the model with reading centered at its mean  
achrdg12\_dev <- achrdg12 - mean(achrdg12)  
mod4 <- lm(achmat12 ~ achrdg12\_dev\*gender)  
summary(mod4)

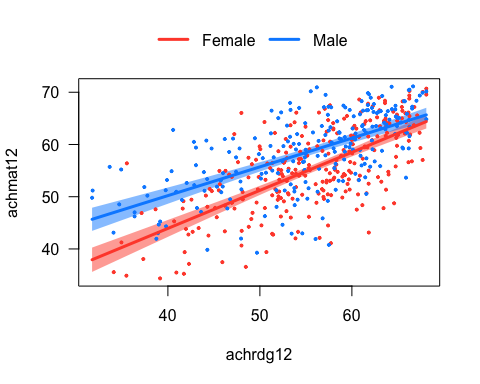
Call:  
lm(formula = achmat12 ~ achrdg12\_dev \* gender)  
  
Residuals:  
 Min 1Q Median 3Q Max   
-19.0582 -3.7864 0.5014 4.0775 16.2889   
  
Coefficients:  
 Estimate Std. Error t value Pr(>|t|)   
(Intercept) 55.29439 0.35127 157.411 < 2e-16 \*\*\*  
achrdg12\_dev 0.72824 0.04702 15.487 < 2e-16 \*\*\*  
genderMale 3.49930 0.52135 6.712 5.26e-11 \*\*\*  
achrdg12\_dev:genderMale -0.17794 0.06514 -2.732 0.00652 \*\*   
---  
Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
  
Residual standard error: 5.801 on 496 degrees of freedom  
Multiple R-squared: 0.4619, Adjusted R-squared: 0.4586   
F-statistic: 141.9 on 3 and 496 DF, p-value: < 2.2e-16

Note that the intercept and the regression coefficient on Gender have changed values compared to mod3. What is the interpretation of these coefficients in the new model?

### 5.8.3 Breaking down a significant interaction

Next, let’s plot our model with the interaction term. One advantage of having everything in a single model is that we can level-up our plotting! The following code uses the visreg package. Note that the error bands in the plot are produced using the standard errors from emmeans, which is discussed below. If you want to know more about how visreg works, type help(visreg).

# Install the package if you haven't already done so  
# install.packages("visreg")  
  
# Load the package into memory  
library(visreg)  
  
visreg(mod3, xvar = "achrdg12", by = "gender", overlay = TRUE)



If the interaction is significant, then we usually want to report a bit more information about how the focal relationship changes as a function of the moderators. There are two main ways to do this:

* Marginal effects (aka marginal means, least squares means, adjusted means): This approach is used when the focal predictor is categorical and we want to compare means across the categories, conditional on levels of the moderator.
* Simple trends (aka simple slopes): This approach is used when the focal predictor is continuous and we want to examine the slopes of the simple trends, conditional on the moderator.

Usually, the researcher will chose one or the other approach, whichever is best suited to address the research questions of interest. Our example was motivated by consideration of the gender gap in STEM (i.e., the relationship between a STEM and a categorical predictor), so the marginal effects approach is better suited. We will also illustrate simple trends, just to show how that approach works.

### 5.8.4 Marginal effects

Let’s breakdown the interaction by asking how the relationship between Math and Gender (i.e., the gender achievement gap in Math) changes as a function of Reading. This can be done using emmeans package, and the main function in that pacakge is also called emmeans.

The three main arguments for the emmeans function:

* object – the output of lm. This is the first argument
* specs – which factors in the model we want the means of (i.e., the focal predictor)
* by – which predictor(s) we want to use to breakdown the means (i.e., the moderator(s))

We can use emmeans to compute the marginal effect at the mean (MEM) as follows:

# Install the package if you haven't already done so  
 # install.packages("emmeans")  
  
# Load the package into memory  
library(emmeans)  
  
# Use the emmeans function to get the gender means on math, broken down by reading  
gap <- emmeans(mod3, specs = "gender", by = "achrdg12")  
summary(gap)

achrdg12 = 55.6:  
 gender emmean SE df lower.CL upper.CL  
 Female 55.3 0.351 496 54.6 56.0  
 Male 58.8 0.385 496 58.0 59.6  
  
Confidence level used: 0.95

# Test whether the difference is significant  
contrast(gap, method = "pairwise")

achrdg12 = 55.6:  
 contrast estimate SE df t.ratio p.value  
 Female - Male -3.5 0.521 496 -6.712 <.0001

In the above output, we only get one Gender difference in Math, and that is computed for the value of achrdg12 = 55.6, which is the mean value of Reading. As noted, this is called the marginal effect at the mean (MEM).

It is often more helpful to report Gender difference for multiple different values of achrdg12, which is called MERV (marginal effects at representative values). While there are many ways to chose the representative values, one convenient approach approach is to use the quartiles of achrdg12. This is accomplished using the cov.reduce argument of emmeans as follows.

# Use the the covarate reduce option of emmeans with the quantile function  
gap\_quartiles <- emmeans(mod3, specs = "gender", by = "achrdg12", cov.reduce = quantile)  
summary(gap\_quartiles)

achrdg12 = 31.8:  
 gender emmean SE df lower.CL upper.CL  
 Female 37.9 1.186 496 35.6 40.3  
 Male 45.7 1.129 496 43.5 47.9  
  
achrdg12 = 51.3:  
 gender emmean SE df lower.CL upper.CL  
 Female 52.1 0.412 496 51.3 52.9  
 Male 56.4 0.426 496 55.6 57.2  
  
achrdg12 = 57.0:  
 gender emmean SE df lower.CL upper.CL  
 Female 56.3 0.355 496 55.6 57.0  
 Male 59.6 0.393 496 58.8 60.3  
  
achrdg12 = 61.7:  
 gender emmean SE df lower.CL upper.CL  
 Female 59.8 0.447 496 58.9 60.6  
 Male 62.2 0.482 496 61.2 63.1  
  
achrdg12 = 68.1:  
 gender emmean SE df lower.CL upper.CL  
 Female 64.4 0.674 496 63.1 65.7  
 Male 65.7 0.693 496 64.3 67.0  
  
Confidence level used: 0.95

# Test whether the gender difference in math achievement is significant at each quartile of reading achievement  
contrast(gap\_quartiles, method = "pairwise")

achrdg12 = 31.8:  
 contrast estimate SE df t.ratio p.value  
 Female - Male -7.74 1.637 496 -4.728 <.0001  
  
achrdg12 = 51.3:  
 contrast estimate SE df t.ratio p.value  
 Female - Male -4.27 0.593 496 -7.207 <.0001  
  
achrdg12 = 57.0:  
 contrast estimate SE df t.ratio p.value  
 Female - Male -3.25 0.529 496 -6.138 <.0001  
  
achrdg12 = 61.7:  
 contrast estimate SE df t.ratio p.value  
 Female - Male -2.41 0.658 496 -3.659 0.0003  
  
achrdg12 = 68.1:  
 contrast estimate SE df t.ratio p.value  
 Female - Male -1.28 0.967 496 -1.321 0.1872

At this point, you should be able to summarize your conclusions about the gender gap in Math and how it depends on Reading.

### 5.8.5 Simple trends

Next we will show how to use emtrends to test the conditional or “simple” slopes of Math on Reading, given Gender. As mentioned, this approach is not very well suited to the example, but we are going through it here just to illustrate how to do this type of analysis.

The three main arguments for emtrends are

* object – the output of lm. This is the first argument
* var – which continuous predictor in the model we want the slopes of
* specs – which factor predictor(s) in the model to break the trend down by

Let’s see how it works.

# Use the emtrends function to get the regression coefficients on reading, broken down by gender  
simple\_slopes <- emtrends(mod3, var = "achrdg12", specs = "gender")  
summary(simple\_slopes)

gender achrdg12.trend SE df lower.CL upper.CL  
 Female 0.728 0.0470 496 0.636 0.821  
 Male 0.550 0.0451 496 0.462 0.639  
  
Confidence level used: 0.95

test(simple\_slopes)

gender achrdg12.trend SE df t.ratio p.value  
 Female 0.728 0.0470 496 15.487 <.0001  
 Male 0.550 0.0451 496 12.208 <.0001

The foregoing analysis tells us how the relationship between reading and math changes as a function of gender, and, in particular, whether the simple slopes are significant for males and females. Recall that the simple slope for females (the group coded zero) is just the regression coefficient on reading in the original lm output. So, the only new thing this output gives us is the simple slope for males.

### 5.8.6 Two continuous predictors

Interactions with continuous predictors are basically the same as for continuous and categorical. One main issue is that we should always center the predictors, not only to facilitate interpretation of the regression coefficients, but also to reduce the correlation between the main effects and the interaction.

For an example, let’s replace gender with SES from our previous analysis. Apologies that this new example is mainly for convenience and doesn’t represent a great research question about, e.g., about why the relationships between math and reading might change as a function of SES!

Here we will focus on how centering affects the results of a regression with interactions among continuous predictors.

# Without centering  
mod5 <- lm(achmat12 ~ achrdg12\*ses)  
summary(mod1)

Call:  
lm(formula = achmat12 ~ achrdg12 + gender + genderXreading)  
  
Residuals:  
 Min 1Q Median 3Q Max   
-19.0582 -3.7864 0.5014 4.0775 16.2889   
  
Coefficients:  
 Estimate Std. Error t value Pr(>|t|)   
(Intercept) 14.80310 2.64922 5.588 3.8e-08 \*\*\*  
achrdg12 0.72824 0.04702 15.487 < 2e-16 \*\*\*  
genderMale 13.39328 3.65828 3.661 0.000278 \*\*\*  
genderXreading -0.17794 0.06514 -2.732 0.006524 \*\*   
---  
Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
  
Residual standard error: 5.801 on 496 degrees of freedom  
Multiple R-squared: 0.4619, Adjusted R-squared: 0.4586   
F-statistic: 141.9 on 3 and 496 DF, p-value: < 2.2e-16

# With centering  
achrdg12\_dev <- achrdg12 - mean(achrdg12)  
ses\_dev <- ses - mean(ses)  
mod6 <- lm(achmat12 ~ achrdg12\_dev\*ses\_dev)  
summary(mod2)

Call:  
lm(formula = achmat12 ~ achrdg12 + gender + achrdg12:gender)  
  
Residuals:  
 Min 1Q Median 3Q Max   
-19.0582 -3.7864 0.5014 4.0775 16.2889   
  
Coefficients:  
 Estimate Std. Error t value Pr(>|t|)   
(Intercept) 14.80310 2.64922 5.588 3.8e-08 \*\*\*  
achrdg12 0.72824 0.04702 15.487 < 2e-16 \*\*\*  
genderMale 13.39328 3.65828 3.661 0.000278 \*\*\*  
achrdg12:genderMale -0.17794 0.06514 -2.732 0.006524 \*\*   
---  
Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
  
Residual standard error: 5.801 on 496 degrees of freedom  
Multiple R-squared: 0.4619, Adjusted R-squared: 0.4586   
F-statistic: 141.9 on 3 and 496 DF, p-value: < 2.2e-16

We can see that, while both models account for the same overall variation in math, SES is significant in the centered model. This has to do both with changing the interpretation of the coefficient (it now represents the relationship between math and reading for students with average reading) and because it is no longer so highly redundant with the interaction term.

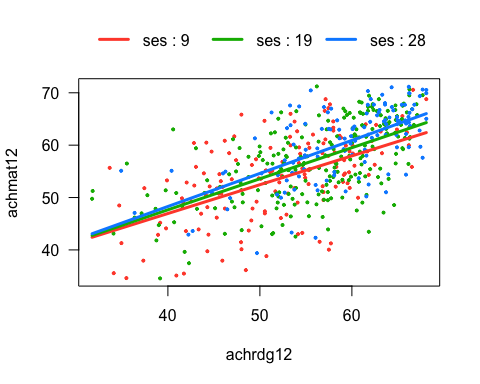
Although the interaction with SES was not significant in either model, let’s break down the interaction with emtrends just to see how it works. This time we will use the at option rather than the ’cov.reduceoption to break down the interaction. The values 9, 19, and 28 are the 10th, 50th, and 90th percentile of SES, which is the same approachvisreguses (You can overwrite the defaults using thebreaksargument -- seehelp(visreg)`).

# Break down interaction with SES as moderator  
simple\_slopes <-emtrends(mod5, var = "achrdg12", specs = "ses", at = list(ses = c(9, 19, 28)))  
summary(simple\_slopes)

ses achrdg12.trend SE df lower.CL upper.CL  
 9 0.550 0.0583 496 0.435 0.665  
 19 0.593 0.0365 496 0.521 0.664  
 28 0.631 0.0639 496 0.506 0.757  
  
Confidence level used: 0.95

Finally let’s summarize our (non significant) interaction with a nice plot.

# Note that band = F removes the confidence intervals  
visreg(mod5, xvar = "achrdg12", by = "ses", overlay = TRUE, band = F)



### 5.8.7 Two categorical predictors

For this topic we will switch over to the ECLS data and examine how SES and Pre-K attendance interact to predict Math Achievement at the beginning of Kindergarten. The variables we will examine are

* Math Achievement at the beginning of K (c1rmscal). This is the number of correct questions on a test with approximately 70 items.
* Whether the child attended Pre-K (p1center). This is a binary variable that indicates pre-K attendance.
* SES, coded as quintiles (wksesq5). We will denote this variable as SES, but keep in mind it is quintiles in this example (e.g., SES = 1 are the respondents with SES between the minimum and the first quintile).

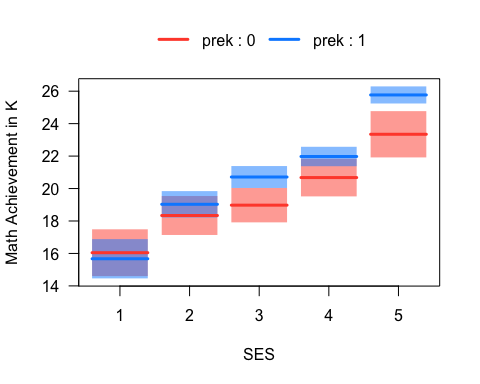
The regression model is as follows. Note that both variables need to be converted to factors in R, so that R will treat them as categorical variables. Also recall that in R the default contrast coding for categorical predictors is reference-group coding.

load("ECLS2577.Rdata")  
ecls$prek <- factor(2 - ecls$p1center)  
ecls$wksesq5 <- factor(ecls$wksesq5)  
mod <- lm(c1rmscal ~ prek\*wksesq5, data = ecls)  
summary(mod)

Call:  
lm(formula = c1rmscal ~ prek \* wksesq5, data = ecls)  
  
Residuals:  
 Min 1Q Median 3Q Max   
-16.7682 -4.7682 -0.9755 3.9545 31.2318   
  
Coefficients:  
 Estimate Std. Error t value Pr(>|t|)   
(Intercept) 16.0455 0.7355 21.816 < 2e-16 \*\*\*  
prek1 -0.3735 0.9601 -0.389 0.69732   
wksesq52 2.2931 0.9570 2.396 0.01663 \*   
wksesq53 2.9300 0.9127 3.210 0.00134 \*\*   
wksesq54 4.6310 0.9439 4.906 9.87e-07 \*\*\*  
wksesq55 7.2990 1.0343 7.057 2.19e-12 \*\*\*  
prek1:wksesq52 1.0638 1.2118 0.878 0.38012   
prek1:wksesq53 2.1087 1.1544 1.827 0.06785 .   
prek1:wksesq54 1.6715 1.1684 1.431 0.15267   
prek1:wksesq55 2.7972 1.2340 2.267 0.02349 \*   
---  
Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
  
Residual standard error: 6.9 on 2567 degrees of freedom  
Multiple R-squared: 0.1621, Adjusted R-squared: 0.1592   
F-statistic: 55.19 on 9 and 2567 DF, p-value: < 2.2e-16

To facilitate interpretation of the ouput, you can refer to the plot below. Each regression coefficient in the output corresponds to a feature of this plot.

visreg::visreg(mod, xvar = "wksesq5", by = "prek",   
 partial = F, rug = F, overlay = T,   
 strip.names = T, xlab = "SES",   
 ylab = "Math Achievement in K")



In order to summarize the model as an ANOVA table, we can use the following code. Note that the ANOVA output tests the variance explained (i.e., R-squared) of the original variables, and does not include dummy variables.

anova(mod)

Analysis of Variance Table  
  
Response: c1rmscal  
 Df Sum Sq Mean Sq F value Pr(>F)   
prek 1 3434 3434.3 72.1437 <2e-16 \*\*\*  
wksesq5 4 19914 4978.4 104.5805 <2e-16 \*\*\*  
prek:wksesq5 4 299 74.7 1.5701 0.1795   
Residuals 2567 122198 47.6   
---  
Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

In an ANOVA context, the R-squared statistics are called eta-squared. They are reported below:

effectsize::eta\_squared(mod, partial = F)

# Effect Size for ANOVA (Type I)  
  
Parameter | Eta2 | 95% CI  
--------------------------------------  
prek | 0.02 | [0.01, 1.00]  
wksesq5 | 0.14 | [0.12, 1.00]  
prek:wksesq5 | 2.05e-03 | [0.00, 1.00]  
  
- One-sided CIs: upper bound fixed at [1.00].

# 6. Model building

Up until this point we have considered models with only a handful of predictors. This chapter addresses how to scale up to real-life applications in which we may have many predictors. [Section 6.1](#sec-hierarchical-models-6) discusses the main ideas, [Section 6.2](#sec-worked-example-6) goes through a worked example, and [Section 6.3](#sec-too-many-predictors-6) addresses some technical details. Compared to the previous chapters, there aren’t a lot of mathematical details in this chapter. The focus is on using the techniques of the previous chapters in more realistic research scenarios (i.e., with lots of predictors).

There are two basic approaches to model building with many predictors. A *simulataneous* model is a single regression model that includes all predictors of interest. When there are many predictors, this is sometimes referred to the “kitchen sink” approach – just put everything in the model (including the kitchen sink). Conceptually, this approach has the advantage that it mitigates omitted variable bias and therefore helps justify the causal interpretation of regression coefficients. However, in practice it can lead to computational problems, especially when highly correlated predictors are included in the model. We discuss this situation in [Section 6.3](#sec-too-many-predictors-6).

*Hierarchical* model building is an alternative approach in which predictors are added into a model sequentially. A hierarchical approach is most appropriate when subsets of predictors can be grouped together conceputally. To illustrate this idea, let’s considering the following hypothetical example in which we seek to explain Reading Achievement using predictors that can be grouped into student factors and curriculum factors:

* Student factors
  + age
  + prior achievement
  + parental education
  + SES
* Curriculum factors
  + in-classroom reading (hrs)
  + peer discussion (hrs)
  + reading materials (level)
  + homework (hrs)

The student factors and curriculum factors are called “blocks” of predictors. A block is just a conceptual grouping of one or more predictors. It is up to the researcher to decide what counts as a conceptually meaningful block of predictors. The overall idea is to use your knowledge of your research area to design a regression model that addresses your research questions. This takes some practice. We will see some more examples of blocks of predictors in [Section 6.1](#sec-hierarchical-models-6) and [Section 6.2](#sec-worked-example-6).

In hierarchical regression, blocks of predictors are added into the model one at a time. The rationale for adding the blocks of predictors sequentially is to isolate the proportion of variance in the outcome (i.e., R-squared) attributable to each block, while controlling for the other block(s). Using our example, we could proceed as follows:

* **Step 1**. First we run a regression model with just the student factors. These variables are known to predict Reading Achievement, but they aren’t the types of factors that can be intervened upon via schooling. These types of predictors are often called “control variables”. What counts as a control variable is a largely non-statistical consideration that depends on the research context.
  + After we run the regression model with this first block of predictors, we record the R-squared statistic. We don’t care very much about the value of the regression coefficients at this point, since we know we have left out the curriculum factors.
* **Step 2**: Next we run a regression model with student factors *and* the curriculum factors, so that our model now includes all of the predictors. By comparing the the R-squared of the model in this second step (curriculum and student factors) to the R-squared of the model in the first step (only student factors), we can determine how much additional variation in Reading Achievement was explained by the curriculum factors. If the additional variation explained by the curriculum factors is substantial, we can reason that intervening on the curriculum may be a good way to improve Reading Achievement.

It is important to note the following points about this example:

* The second step in the hierarchical model is the same the simultaneous model – they both include all of the predictors. This is true in general. The final step of a hierarchical model is always the same as the corresponding simultaneous model.
* So, what does the hierarchical model give us that the simultaneous model doesn’t? The change in R-squared attributable to each additional block of predictors, controlling for the previous block(s) of predictors. This “change-in-R-squared” statistic is the main focus of this chapter.
* It is important to distinguish model building from variable selection. Model building is about research-question-driven strategies for partitioning the variation explained by a large number of predictors. Variable selection is about “weeding out” predictors that are not useful. So, in model building, the final model should always include all of the potential predictors, but in variable selection, the final model may include only a subset of the potential predictors.
* Variable selection is an important topic that is the focus of a lot current research in machine learning. There was also a large number of “step-wise” regression procedures used for variable selection in last century, but they didn’t work (i.e., different procedures resulted in different models and there were no good criteria by which to choose amongst them). If you have a research scenario that requires variable selection, this problem is best dealt with through machine learning. If you are interested, check out this resource: <https://glmnet.stanford.edu> (and feel free to ask questions in class!).
* A note on terminology: In some research areas, “hierarchical” is used to refer to multilevel models (e.g., “hierarchical linear models” in sociology). That is not the usage here. It would make sense to refer to the types of models we are calling “hierarchical” as “step-wise”, but, as mentioned in the previous point, the term “step-wise” already refers to a large number flawed techniques for variable selection. So… here we are.

One last thing: The overall idea of this chapter is that you should build models that answer your research questions. For concreteness, this chapter discusses the main statistical techniques for model building in the context of hierarchical regression. But, you don’t need to use hierarchical regression – the same techniques can be used in other ways creative ways as well.

## 6.1 Hierarhical models

The key ideas of hierarchical regression are:

1. Partition the predictors into conceptual “blocks”. Each block contains at least one predictor, but they usually contain more than one (e.g., “control variables”, “home factors”). No predictor can be included in more than one block.
2. Build a series of regression models in which the blocks are entered into the model sequentially. For this to be *hierarchical* model building, it is required that each new model in the series includes the blocks from all previous models. So, with three blocks we would have:
3. Compare the R-squared value of each model to that of the previous model. This allows us to determine whether each new block “adds anything” to the model. To see how this works, denote the R-squared values of our three models as , respectively. Then we proceed as follows:
   * Compute to evaluate the contribution of Block 1.
   * Compute to evaluate contribution of Block 2, controlling for Block 1
   * Compute to evaluate contribution of Block 3, controlling for Block 1 & 2.

The symbol is pronounced “delta R-squared.” It always has two subscripts, which denote the two models being compared.

The main advantage of hierarchical modeling is that we partition the total variance explained by all of the predictors into parts contributed uniquely by each block. In particular, it follows from the definition of delta R-squared that:

This means that the variation explained by all of the predictors together () can be separated into the 3 distinct components that are isolated by using hierarchical approach.

Sometimes we let denote a model that contains no predictors (only the intercept). Then can be equilvanently written as . This makes the notation of the previous equation a bit more consistent:

The above equation shows us that hierarchical modeling building is an approach to partitioning the total variance explained by a set of predictors (here denoted ) into the variance uniquely attributable to different blocks of predictors (the s). While there are many other approaches to model building, this equation is what uniquely defines *hierarchical* model building.

### 6.1.1 Example of blocks

We have already seen some examples of blocks of predictors in the previous chapters. This section highlights two examples.

* Categorical predictors with categories are represented as blocks of dummy variables. The dummies are conceptually equivalent to a single predictor.
* Interactions are blocks of predictors. Often interactions are entered into a regression model on separate step, after the “main effects” have been entered. Below is an excerpt from a study that illustrates this approac. The study is about equitable access to gifted-and-talented (GT) programs in New York City. In order to apply for GT programs, students need to take a test in Kindergarten. The test is optional and not everyone takes it. The study is mainly interested in whether attending Pre-K increases the probability that a child will take the test, especially for students who have been traditionally underrepresented in GT programs. (cite: Lu & Wienberg (2016). Public Pre-K and Test Taking for the NYC Gifted-and-Talented Programs: Forging a Path to Equity. Educational Researcher, 45, 36-47):

|  |
| --- |
|  |

More generally, blocks are any subset of predictors about which you want to ask a research question. It can be just one variable, or it can be a collection of conceptually related variables.

**Please take a moment to write down one or more examples of a block of variables from your area of research, and I will invite you to share your examples in class.**

### 6.1.2 Nested models

Whatever your blocks are, hiearhical modeling requires they are used to make a sequence of *nested models*. The idea behind nested models is depicted in [Figure 6.1](#fig-nested1). In the figure, Model 2 contains total predictors, and Model 1 contains a subset of those predictors. This relationship is shown by letting the area representing Model 1 be contained by the area representing Model 2. We can think of this like Matryoshka nesting dolls (<https://en.wikipedia.org/wiki/Matryoshka_doll>). Model 1 is contained within Model 2, in the sense that all of the predictors in Model 1 are also in Model 2.

|  |
| --- |
| Figure 6.1: Nested Models |

A bit more technically: One regression model is nested within another if the former can be obtained by removing some predictors from the latter. Other than the difference in the predictors, it is assumed that the models equivalent (same outcome variable, sample samples, etc.).

The difference between nested and non-nested models is illustrated in [Figure 6.2](#fig-nested2). The partially overlapping ovals are intended to indicate that the models share some predictors, but each model also has some unique predictors.

|  |
| --- |
| Figure 6.2: Nested vs Non-Nested Models |

As noted in the figure, a hierarchical model is a series of nested regression models. You might be wondering why it is so important for the models to be nested. Part of this is conceptual – the hierarchical approach allows us to partition the total variance explained into parts attributable to different subsets (blocks) of predictors (see [Section 6.1](#sec-hierarchical-models-6)). Nesting is also required to compute statistical tests for the statistics (more on this in [Section 6.1.4](#sec-delta-rsquared-6)).

In short, we can statistically compare Model 1 and Model 2 in the left hand panel of [Figure 6.2](#fig-nested2), but not in the right hand panel. (Side note: In the right hand panel, we could compare Model 1 vs Model 3 and Model 2 vs Model 3, but this would not be a hierarchical model because the s wouldn’t add up to the of Model 3.)

### 6.1.3 Some practical advice

So far we have discussed hierarchical modeling conceptually, but have not really addressed how to use it in practice. The following advice is paraphrased from the late Jacob Cohen (cite: Cohen, Cohen, West, & Aitken (2003). Applied Multiple Regression Analysis. Mahwah, NJ: Lawrence Earlbaum. Section 5.7.3).

* The hierarchical approach provides an effective strategy of inference if variables are entered into the model according to their relevance to the study.
* The variables that are most central to the study should be entered first, those that are secondarily of interest should be entered second, and those that fall into the category, “I wonder if” or “just in case” should be entered last.
* This principle may be succinctly stated as “least is last.” That is, “when research factors can be ordered as to their centrality, those of least relevance are appraised last in the hierarchy, and their results taken as indicative rather than conclusive”

[Figure 6.3](#fig-nested3) maps Cohen’s strategy onto our diagram for nested models. One addendum to this approach is that “control variables” are usually entered first. These are variables that have been established as important in past research, but are not of central interest to the present study. The rationale for entering them first is that new studies should add something beyond what has been established by past research.

|  |
| --- |
| Figure 6.3: Cohen’s ‘Least is Last’ Principle |

### 6.1.4 Delta R-squared

We have already seen how to compute – by subtracting the R-squared value of a nested model from that of a nesting model. This is a very widely used way of comparing nested regression models.

When adding predictors into a model one at a time (i.e., blocks with only a single predictor), testing is equivalent to testing the b-weight of the added predictor. But when adding multiple predictors into a model, provides important information that is not captured by any of the statistics we have considered so far. It is technically called a multiple semi-partial correlation, but “delta R-squared” or “R-squared change” are widely used terminology.

Statistical inference for uses the F-test shown below. Let Model 1 be a regression model with predictors that is nested within Model 2, with predictors. Define and . To test the hypothesis versus we can use the test statistic

which has an F-distribution on and degrees of freedom, when the null hypothesis true. This test assumes that Model 2 satisfies the linear regression population model (see @sec-population-model-2) and that Model 1 is nested within Model 2.

## 6.2 A worked example

To illustrate hierarchical modeling, let’s use the ECLS data to address whether reading achievement at the beginning of Kindergarten (c1rrscal) is related to SES (wksesl), parental (mother’s and father’s) education (wkmomed and wkdaded, respectively), and attendance in center-based care before K (p1center; pre-K status, for short).

This example is intended to be a realistic illustration of how to use the techniques we have covered up to this point. The sections below can be thought of as excerpts from a research paper, and also shows how to report results in a regression table. But there are some caveats that should be mentioned before starting:

* We used some tricks from later chapters to deal with non-linearity, and
* keep in the mind the subset of ECLS data we are using is not a representative sample and the results should not be assumed to generalize to the general population.

### 6.2.1 Research questions

We will address the following research questions.

* RQ1: Does SES predict Reading Achievement upon entry to K?
* RQ2: Does Parental Education (mother’s and father’s) predict Reading Achievement, after controlling for SES?
* RQ3: After controlling for both SES and Parental Education, is Pre-K Status associated with better Reading Achievement in K?
* RQ4: Does the relationship of SES or Parental Education to Reading Achievement change as function of Pre-K Status – in more causal language: does Pre-K participation reduce pre-existing disparities in Reading Achievement upon entry to K?

### 6.2.2 Data analysis plan

The research questions were addressed via hierarchical regression modeling with the following blocks of predictors.

* Block 1: wksesl
* Block 2: wkmomed and wkdaded
* Block 3: plcetner
* Block 4: Interactions between p1center and the other predictors

To aid the interpretation of regression coefficients in the presence of interactions, all continuous variables (predictors and outcome) were transformed to z-scores. plcenter was coded as binary indicator for Pre-K status.

The blocks were entered into the model in the indicated order. The RQs were addressed by examining the R-square change () for each corresponding block. (In practice, we would also interpret the sign and direction of the significant regression coefficients in the final model, but we will leave that as an exercise.) A significance level of .05 was used for all tests reported in this study.

Preliminary examination of the model residuals indicated violation of the assumption of linearity. To address this, the outcome variable was log-transformed and a quadratic term was added for wksesl (wksesl\_sq). These procedures for checking and addressing non-linearity are discussed in the coming chapters.

### 6.2.3 Results

The F-tests of R-squared change for each block of predictors are reported in [Figure 6.4](#fig-table1). The results show that block 1 (linear and quadratic trends in SES) explained about 18% of the variance Reading Achievement. While blocks 2 (parental education) and 3 (Pre-K) contributed statistically significant increases in variance explained, the change in R-squared was not large. Block 4 containing the interactions with Pre-K was not statistically significant after controlling for the main effects.

load("ECLS2577.RData")  
attach(ecls)  
  
# Recode variables  
p1center <- 2 - p1center  
z\_c1rrscal <- scale(c1rrscal)  
z\_wksesl <- scale(wksesl)  
z\_wksesl\_sq <- z\_wksesl^2  
z\_wkmomed <- scale(wkmomed)  
z\_wkdaded <- scale(wkdaded)  
log\_c1rrscal <- log(z\_c1rrscal - min(z\_c1rrscal) + 1)  
  
# Run models  
# models  
mod1 <- lm(log\_c1rrscal ~ z\_wksesl + z\_wksesl\_sq)  
mod2 <- lm(log\_c1rrscal ~ z\_wksesl + z\_wksesl\_sq + z\_wkmomed + z\_wkdaded)  
mod3 <- lm(log\_c1rrscal ~ z\_wksesl + z\_wksesl\_sq + z\_wkmomed + z\_wkdaded + p1center)  
mod4 <- lm(log\_c1rrscal ~ (z\_wksesl + z\_wksesl\_sq + z\_wkmomed + z\_wkdaded)\*p1center)  
  
R2\_1 <- summary(mod1)$r.squared  
R2\_2 <- summary(mod2)$r.squared  
R2\_3 <- summary(mod3)$r.squared  
R2\_4 <- summary(mod4)$r.squared  
  
rsquared <- c(0, R2\_1, R2\_2, R2\_3, R2\_4)  
delta\_rsquared <- c(0, R2\_1, R2\_2 - R2\_1, R2\_3 - R2\_2, R2\_4 - R2\_3)  
Models <- paste0("Model ", 0:4)  
names(delta\_rsquared) <- Models  
  
# Tests of R-square change  
model\_results <- knitr::kable(cbind(Rsquared = rsquared,   
 Delta.Rsquared = delta\_rsquared,   
 anova(lm(log\_c1rrscal ~ 1), mod1, mod2, mod3, mod4)),  
 digits = 3)  
model\_results

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  | Rsquared | Delta.Rsquared | Res.Df | RSS | Df | Sum of Sq | F | Pr(>F) | | --- | --- | --- | --- | --- | --- | --- | --- | --- | | Model 0 | 0.000 | 0.000 | 2576 | 343.636 | NA | NA | NA | NA | | Model 1 | 0.176 | 0.176 | 2574 | 283.114 | 2 | 60.522 | 277.886 | 0.000 | | Model 2 | 0.179 | 0.003 | 2572 | 282.032 | 2 | 1.083 | 4.970 | 0.007 | | Model 3 | 0.184 | 0.005 | 2571 | 280.271 | 1 | 1.761 | 16.168 | 0.000 | | Model 4 | 0.187 | 0.002 | 2567 | 279.538 | 4 | 0.733 | 1.683 | 0.151 |   Figure 6.4: R-squared change and F-tests |

Although we didn’t have research hypotheses about the individual predictors, it is often useful to report a table such as the one below to summarize that kind of information. The table was produced using the stargazer package in R and is representative of the usual format for reporting regression output from hierarchical regression models. The values reported for each variable are the regression coefficients, with the standard errors in parenthesis. **Please take a moment to look over the table and write down any questions you have about its interpretation**

|  |
| --- |
|  |

### 6.2.4 Interpretation

The focus of this section is on the interpretation of the statistics and their significance. In practice, we would also interpret the sign and direction of the significant regression coefficients in the final model. As noted above, the outcome variable was log transformed, but we ignore that detail here. It would usually not be necessary to report results both in a table and inline, but the latter are shown for illustrative purposes (practice, practice, practice!).

* RQ1: SES had significant linear and quadratic relationships with Reading Achievement. Together, both predictors explained about 17.6% of the variance in Reading Achievement (.
* RQ2: After controlling for SES, Parental Education explained a small but statistically significant proportion of the variance in Reading Achievement (),
* RQ3: After controlling for SES and Parental Education, Pre-K status explained an additional .5% of the variation in reading achievement ().
* RQ4: There was mixed evidence about the interactions between Pre-K Status and the other predictors. The F-test of R-squared change indicated that the interaction terms did not explain a significant amount of variation in reading (). However, the interaction with paternal education was significant at the .05 level. After re-running Model 4 with only the paternal education interaction, it was found that this effect was not significant at the .05 level (; see output below). It may be concluded that there were no significant interactions with Pre-K status.

mod4A <- lm(log\_c1rrscal ~ z\_wksesl + z\_wksesl\_sq + z\_wkmomed + z\_wkdaded\*p1center)  
summary(mod4A)

Call:  
lm(formula = log\_c1rrscal ~ z\_wksesl + z\_wksesl\_sq + z\_wkmomed +   
 z\_wkdaded \* p1center)  
  
Residuals:  
 Min 1Q Median 3Q Max   
-0.88751 -0.23151 -0.00025 0.20858 1.34884   
  
Coefficients:  
 Estimate Std. Error t value Pr(>|t|)   
(Intercept) 0.795829 0.015160 52.495 < 2e-16 \*\*\*  
z\_wksesl 0.115158 0.016322 7.056 2.2e-12 \*\*\*  
z\_wksesl\_sq -0.019645 0.005306 -3.702 0.000218 \*\*\*  
z\_wkmomed 0.018424 0.011174 1.649 0.099327 .   
z\_wkdaded 0.042770 0.017016 2.513 0.012016 \*   
p1center 0.060274 0.016163 3.729 0.000196 \*\*\*  
z\_wkdaded:p1center -0.014023 0.016373 -0.856 0.391830   
---  
Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
  
Residual standard error: 0.3302 on 2570 degrees of freedom  
Multiple R-squared: 0.1846, Adjusted R-squared: 0.1827   
F-statistic: 96.99 on 6 and 2570 DF, p-value: < 2.2e-16

In summary, it was found that SES and Parental Education were significant predictors of children’s Reading Achievement upon entry to Kindergarten, indicating that pre-existing disparities in reading due to economic factors are apparent at even this very early age. Attendance in Pre-K explained an additional .5% of variance in Reading Achievement and was associated with increased Reading Achievement; however, there was no evidence that participation in Pre-K reduced pre-existing disparities in Reading Achievement associated with the economic factors (i.e., there were no interactions between Pre-K participation and the economic factors).

## 6.3 Too many predictors?

What is better: more predictors or fewer? While there is no definitive answer to this question, there are two points that we should keep in mind when adding predictors into a model:

* The ratio of sample size to number of predictors: .
* The correlation among the predictors (AKA multicollinearity).

In order to understand the effect of these factors in multiple regression, its helpful to refer to the formula for the standard error of a regression coefficient (from [Section 3.7.1](#sec-inference-for-coeffecients-3)), which is presented again below:

Recall that denotes standard deviations and that is the R-squared from the regression of the -th predictor on the remaining predictors.

### 6.3.1

We can see from the equation for that increasing the number of predictors leads the term to get smaller. Consequently, the ratio

will get larger (and hence the standard error will get larger) if the increase in the number of predictors is not compensated by an increase in In other words, predictors that do not add explanatory power to the model (i.e., do not increase R-squared) will lead to reduced precision for all of the regression coefficients.

So long as , the effect of the number of predictors is negligible. But when the number of predictors gets close to the sample size, the estimates of the regression coefficients can become very imprecise. Many seek “rules of thumb” for the ratio of observations to predictors. This is just a complicated way of asking how large the sample size needs to be, and the correct answer to that question is provided by power analysis ([Section 2.8](#sec-power-2)). That said, if -ish you should start probably worrying about the number of predictors in your model.

### 6.3.2 Multicollinearity

Regardless of sample size, when a predictor is highly correlated with the other predictors, its precision is negatively affected via the term . When the situation is called multicollinearity. Note that multicollinearity can occur even if none of the pairwise correlations among the predictors is large – it has to do with the relation between predictor and *all* of the other predictors, which is why it is called *multi*collinearity.

The term is called the *tolerance* and its reciprocal is called the variance inflation factor (VIF).

Note that appears in formula for the standard error of the regression coefficient.

Some examples of how to interpret the are provided below:

* If , the estimated regression coefficient is 5 times less precise than it would be without multicollinearity.
* If , the estimated regression coefficient is 10 times less precise than it would be without multicollinearity.
* If , it is a good idea to reconsider whether the predictor needs to be recoded or omitted from the model.

In general, it is a good practice to check VIF as part of model diagnostics, which is the topic of the next chapter. For our example, the VIF statistics for Model 3 and Model 4 are reported below.

For Model 3:

# install.packages("car")  
car::vif(mod3)

z\_wksesl z\_wksesl\_sq z\_wkmomed z\_wkdaded p1center   
 6.290370 1.206007 2.950358 3.377283 1.053738

For Model 4:

car::vif(mod4)

there are higher-order terms (interactions) in this model  
consider setting type = 'predictor'; see ?vif

z\_wksesl z\_wksesl\_sq z\_wkmomed   
 28.157965 6.447179 12.799652   
 z\_wkdaded p1center z\_wksesl:p1center   
 13.816244 1.812096 28.320202   
z\_wksesl\_sq:p1center z\_wkmomed:p1center z\_wkdaded:p1center   
 7.562264 12.516708 14.011576

We can see that including the interactions led to a high degree of multicollinearity for some of the regression coefficients in the model. Based on this information, as well as the non-significant for the interaction block, it would be advisable to focus our interpretation on Model 3 rather than Model 4.

## 6.4 Workbook

This section collects the questions asked in this chapter. There aren’t many, we will mainly be focussed on going through the worked example during class time.

[Section 6.1](#sec-hierarchical-models-6)

Please take a moment to write down one or more examples of a block of variables from your area of research, and I will invite you to share your examples in class.

[Section 6.2](#sec-worked-example-6)

* Look over the tables below and write down any questions you have about their interpretation or the write up provided in [Section 6.2](#sec-worked-example-6).

|  | Rsquared | Delta.Rsquared | Res.Df | RSS | Df | Sum of Sq | F | Pr(>F) |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Model 0 | 0.000 | 0.000 | 2576 | 343.636 | NA | NA | NA | NA |
| Model 1 | 0.176 | 0.176 | 2574 | 283.114 | 2 | 60.522 | 277.886 | 0.000 |
| Model 2 | 0.179 | 0.003 | 2572 | 282.032 | 2 | 1.083 | 4.970 | 0.007 |
| Model 3 | 0.184 | 0.005 | 2571 | 280.271 | 1 | 1.761 | 16.168 | 0.000 |
| Model 4 | 0.187 | 0.002 | 2567 | 279.538 | 4 | 0.733 | 1.683 | 0.151 |

|  |
| --- |
|  |

## 6.5 Exercises

The exercises reproduce the worked example in [Section 6.2](#sec-worked-example-6). We will go through this material in class together, so you don’t need to work on it before class (but you can if you want.)

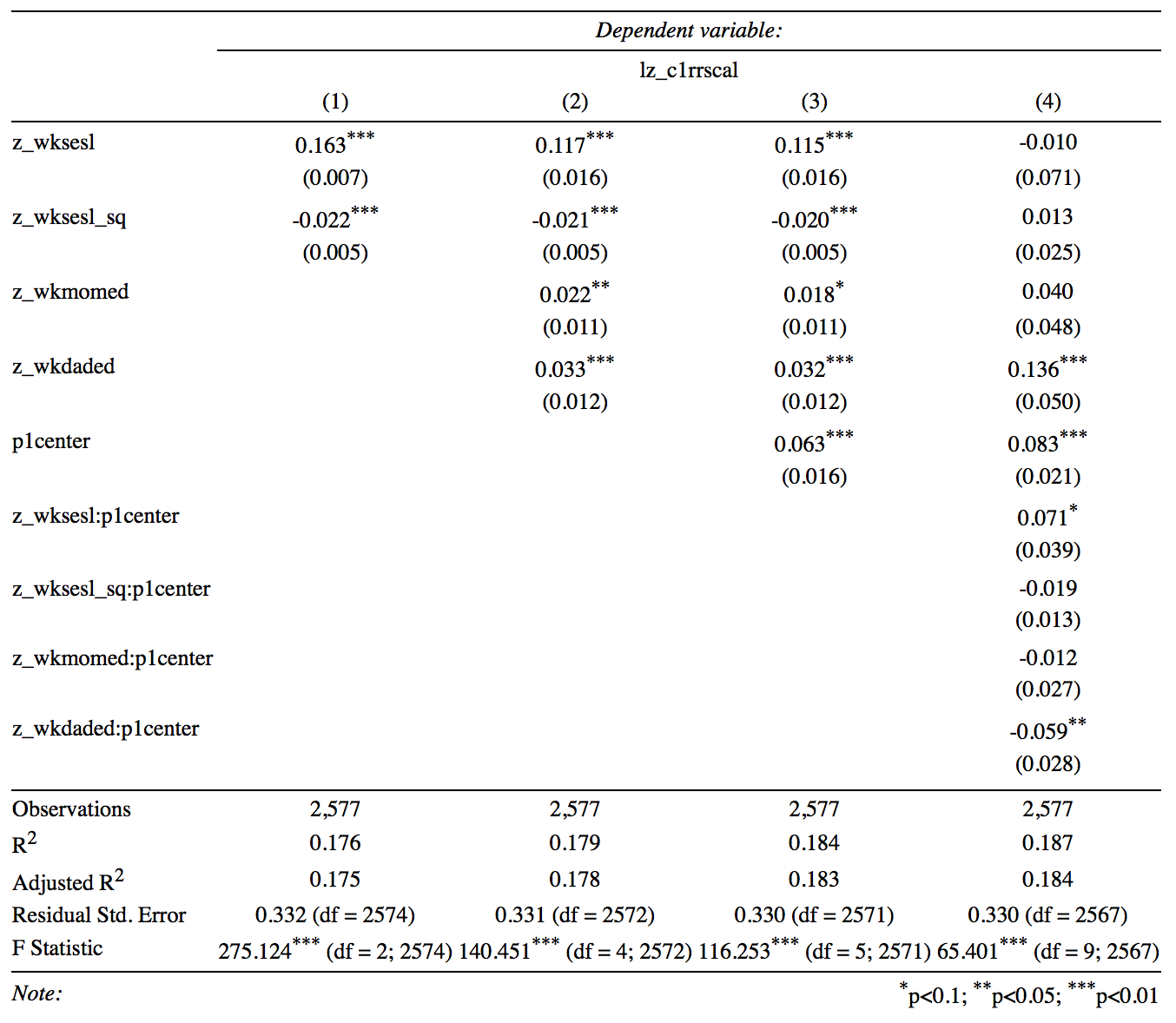
Before staring this section, you may find it useful to scroll to the top of the page, click on the “</> Code” menu, and select “Show All Code.”

First we set up the variables

# Clean up env. and load data  
#rm(list = ls())  
#load("ECLS2577.RData")  
#attach(ecls)  
  
# Recode variables  
p1center <- 2 - p1center  
z\_c1rrscal <- scale(c1rrscal)  
z\_wksesl <- scale(wksesl)  
z\_wksesl\_sq <- z\_wksesl^2  
z\_wkmomed <- scale(wkmomed)  
z\_wkdaded <- scale(wkdaded)  
log\_c1rrscal <- log(z\_c1rrscal - min(z\_c1rrscal) + 1)

Next, run the models.

# Run models  
mod1 <- lm(log\_c1rrscal ~ z\_wksesl + z\_wksesl\_sq)  
mod2 <- lm(log\_c1rrscal ~ z\_wksesl + z\_wksesl\_sq + z\_wkmomed + z\_wkdaded)  
mod3 <- lm(log\_c1rrscal ~ z\_wksesl + z\_wksesl\_sq + z\_wkmomed + z\_wkdaded + p1center)  
  
# Note the syntax here for interacting one variable with many  
mod4 <- lm(log\_c1rrscal ~ (z\_wksesl + z\_wksesl\_sq + z\_wkmomed + z\_wkdaded)\*p1center)  
  
# Regression table -- will discuss this in class  
# stargazer::stargazer(mod1, mod2, mod3, mod4, type = 'html', omit = "Constant")  
knitr::include\_graphics("files/images/stargazer.png")



The stargazer table doesn’t format nicely in markdown, it is attached as a screen shot instead. We will go through some different types of output with stargazer in class. You must install the stargazer package for the code to work.

The above table reports the for each model but does not provide the statistics or the corresponding F-tests. The statistics require some additional processing:

# Get R-squared   
R2\_1 <- summary(mod1)$r.squared  
R2\_2 <- summary(mod2)$r.squared  
R2\_3 <- summary(mod3)$r.squared  
R2\_4 <- summary(mod4)$r.squared  
rsquared <- c(0, R2\_1, R2\_2, R2\_3, R2\_4)  
  
# Compute R-squared change  
delta\_rsquared <- c(0, R2\_1, R2\_2 - R2\_1, R2\_3 - R2\_2, R2\_4 - R2\_3)  
  
# Set up for pretty printing  
results <- cbind(Rsquared = rsquared,   
 Delta.Rsquared = delta\_rsquared,   
 anova(lm(log\_c1rrscal ~ 1), mod1, mod2, mod3, mod4))  
   
knitr::kable(results, digits = 3)

| Rsquared | Delta.Rsquared | Res.Df | RSS | Df | Sum of Sq | F | Pr(>F) |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 0.000 | 0.000 | 2576 | 343.636 | NA | NA | NA | NA |
| 0.176 | 0.176 | 2574 | 283.114 | 2 | 60.522 | 277.886 | 0.000 |
| 0.179 | 0.003 | 2572 | 282.032 | 2 | 1.083 | 4.970 | 0.007 |
| 0.184 | 0.005 | 2571 | 280.271 | 1 | 1.761 | 16.168 | 0.000 |
| 0.187 | 0.002 | 2567 | 279.538 | 4 | 0.733 | 1.683 | 0.151 |

The variance inflation factors for Model 3 and Model 4 are obtained use the vif function of the car package (you must install the car package for this code to work).

# install.pacakges("car")  
car::vif(mod3)

z\_wksesl z\_wksesl\_sq z\_wkmomed z\_wkdaded p1center   
 6.290370 1.206007 2.950358 3.377283 1.053738

car::vif(mod4)

there are higher-order terms (interactions) in this model  
consider setting type = 'predictor'; see ?vif

z\_wksesl z\_wksesl\_sq z\_wkmomed   
 60.867975 12.682305 28.145377   
 z\_wkdaded p1center z\_wksesl:p1center   
 31.720745 1.812096 59.296155   
z\_wksesl\_sq:p1center z\_wkmomed:p1center z\_wkdaded:p1center   
 12.217023 28.440378 30.766199

# clean up  
rm(list = ls())  
detach(ecls)

# 7. Assumption checking

In [Section 2.5](#sec-population-model-2) we introduced the population model for linear regression. But how do we know whether this model applies to our data? That is the question we address in this chapter. In particular, we discuss data analyses that can be used to better understand whether the assumptions made when defining the population model for linear regression are consistent with our data. This is referred to *assumption checking*.

The three assumptions are (don’t worry, we review them again in the next section):

* Normality of residuals
* Homoskedasticity of residuals
* Linearity of the conditional mean (regression) function

Why do we care about these assumptions? Well, if all of the population assumptions of linear regression are not met, any of the following can happen:

* Regression coefficients and R-squared could be biased
* Standard errors could be too small (or too large)
* t- and F-tests could be too small (or too large)
* p-values could be too small (or too large)
* Confidence intervals could be too small (or too large)

Basically, all of the numbers we get in R’s summary(lm) output could be wrong.

The interpretation of assumptions in regression is a bit subtle. Assumptions are conditions we need to be true in order for the results of analysis to be valid. But, we don’t really care about the assumptions in their own right. For example, we don’t really care if the regression residuals are normally distributed – we just care if there are any violations of this assumption that might be affecting the results our analysis. If there is no evidence that an assumption is problematic, we proceed with the main analysis as intended. If there is evidence that an assumption is violated, we can modify the analysis as required. This chapter focuses on checking assumptions, and the following two chapters talk about strategies for dealing with assumption violations.

Assumption checking usually involves plotting the residuals from a regression model and trying to interpret what the plots tell us about the population model. The main focus of this chapter is to introduce you to these plots and how to interpret them. Sometimes assumption checking can feel a bit like reading tea leaves, because interpreting plots can be pretty subjective. Honing your interpretation will happen gradually with experience.

In this chapter we also address how to deal with heteroskedasticity in regression models. The short version is that heteroskedasticity does not affect the OLS estimates of the regression coefficients, it only affects their standard errors (and, consequently, the t-values, p-values, and confidence intervals). There are corrections to the standard errors that can be used to address heteroskedasticity. In particular, we will focus on one widely used procedure called heteroskedasticity-consistent (HC) standard errors. These “corrected” standard errors are often larger then than the “uncorrected” standard errors, which is why we prefer to use the latter when homoskedasticity is viable.

A related topic, *regression diagnostics*, is also introduced in this chapter. Diagnostics are procedures for detecting outliers. Unlike assumption checking, which focuses on the model *per se*, diagnostics focus on individual data points. We only cover the basics behind diagnostics in this chapter and the material is optional (i.e., it will not be assessed in this course, but you may find it useful for your research).

Regression diagnostics and outlier detection can be useful for identifying potentially problematic data points, but it is almost never the the case that data points should be omitted because they are outliers. Unless you can find something specifically wrong with a data point (e.g., a data entry error) you should not omit data. A better way to deal with outliers is by using statistical procedures that are specifically designed to deal with them, which is a field of study called *robust statistics*. Robust regression is an advanced topic that we won’t get to in this course, but check out this resource if you are interested and feel free to ask questions in class: <https://cran.r-project.org/web/views/Robust.html>.

## 7.1 Recap of population model

Let’s start with a recap of the population model for linear regression. This was introduced for simple linear regression in [Section 2.5](#sec-population-model-2). To restate the assumptions for multiple linear regression, we will use the vector notation

to represent the predictor variables. For our purposes, the vector is just a list of all of the predictors in a model. The outcome variable is denoted as , as usual.

1. Normality: The distribution of conditional on is normal for all values of .
2. Homoskedasticity: The conditional distributions have equal variances (also called homogeneity of variance, or just equal variances).
3. Linearity: The means of the conditional distributions are a linear function of .

These three assumptions can also be summarized in term of the regression residuals. Recall that residuals are computed as . If the three assumptions of linear regression hold, then the regression residuals should be normally distributed with mean zero and constant variance, for every value of the predictors:

[Figure 7.1](#fig-pop-model-e) presents the population model in terms of the residuals. This plot is similar [Figure 2.4](#fig-pop-model), but is modified for the multiple regression setting by using predicted values and residuals as the axes, rather than and . The plots we look at in the following sections are sample analogues to this population model – they plot residuals against predicted values.

|  |
| --- |
| Figure 7.1: The Multiple Regression Population Model. |

## 7.2 Linearity

This assumption is about whether the regression function is “really” a line or if it could be better represented as some other relationship. A classic example is shown below. We address this example, which is taken from “Anscombe’s quartet”, in more detail in [Section 7.6](#sec-diagnostics-7) (which is optional).

# Non-linearity in Anscombe's second example  
attach(anscombe)  
mod <- lm(y2 ~ x2)  
  
# Take a look at the raw data  
par(mfrow = c(1, 2))  
plot(x2, y2, col = "#4B9CD3", xlab = "X", ylab = "Y")  
abline(mod)  
  
# Compare to the residual vs fitted plot  
plot(mod, which = 1)  
detach(anscombe)

|  |
| --- |
| Figure 7.2: Anscombe’s second dataset |

The left hand panel of [Figure 7.2](#fig-linearity1) shows the scatter plot of the example data. It should hopefully be obvious that the relationship between and is not linear.

The right hand panel shows the residuals versus the predicted (“fitted”) values from the regression of on . It plots the residuals on the vertical axis and the fitted values () on the horizontal axis. This is the sample analogue of the population model in [Figure 7.1](#fig-pop-model-e).

It is important to note the following about residual vs fitted plot:

* The key idea is that deviations from the regression line in the left hand panel correspond to deviations from the horizontal line at Residuals = 0 in the right hand panel. Recall that the residuals should all be centered around this horizontal line if the population model is true (see [Figure 7.1](#fig-pop-model-e)). The non-linear trend is apparent in the in both panels, but in the residual vs fitted plot the nonlinearity is with reference to Residuals = 0.
* The red line in the right hand panel is a locally weighted smoothed (“lowess”) regression line – it follows whatever trend is in the residuals without assuming the trend is linear.
* The overall interpretation of the residual vs fitted plot is as follows:
  + If the red line is roughly horizontal at Residuals = 0, we conclude that the assumption of linearity is not problematic for the data.
  + If the red line deviates systematically from a horizontal line at Residuals = 0, this is evidence that the assumption of linearity is problematic.

In [Figure 7.2](#fig-linearity1), the assumption of linearity is clearly not met. In fact, this is so obvious that we could see it in the regular scatter plot! So, you might be asking, why do we need the residual vs fitted plot? Well, the regular scatter plot is only useful for diagnosing linearity with a single predictor, whereas the residual vs fitted plots works any number of predictors. So, in general, it is much easier to check the assumption using the residual versus fitted plot, even if the patterns are a bit harder to interpret.

[Figure 7.3](#fig-linearity2) illustrates the residual vs fitted plot using a model with 2 predictors. **Please write down whether you think the linearity assumption is problematic for the example below, and be sure to explain why with reference to the figure.** Keep in mind that interpreting plots takes a bit of practice and in general there is no “right” answer. Rather, what I am looking for is an explanation of *why* you think the assumption is problematic or not. Your explanation should refer to the interpretation of residual vs fitted plots, as outlined above.

load("ECLS250.RData")  
attach(ecls)  
  
# Run model for example  
mod2 <- lm(c1rmscal ~ ses\_orig + t1learn, data = ecls)  
  
# Plot resid vs fitted  
plot(mod2, which = 1)

|  |
| --- |
| Figure 7.3: An example from ECLS. |

Before moving on, let’s take a look at a few more examples. For each of the examples in [Figure 7.4](#fig-linearity3), **please write down whether you think the linearity assumption is problematic and explain why with reference to the plots.**. Hint: be careful not to over-interpret the lowess line in the tails of the plots, where only a few data points can have a big impact on the local trend. Focus your interpretation on the bulk of the data, and whether it shows a systemic trend away from a horizontal line at 0.

# Example: regression c1rmscal on ses\_orig and t1learn  
set.seed(101)  
par(mfrow = c(2, 2))  
for(i in 1:4) {  
 x <- rnorm(200)  
 e <- rnorm(200)  
 y <- x + e   
 plot(lm(y ~ x), which = 1)  
}

|  |
| --- |
| Figure 7.4: More examples. |

### 7.2.1 Summary

To check the assumption of linearity, we can use a residual vs predicted (fitted) plot.

* If the plot does not show a systematic trend other than a horizontal line at Residuals = 0, then there is no evidence against the assumption.
* If the residuals do show a trend away from Residuals = 0, then we should worry about the assumption.
* Don’t over interpret the tails of the lowess (red) lines in the R plots.
* If the assumption is violated: consider a non-linear transformations of the variable ([Chapter 8](#sec-chap-8)) or adding quadratic or other non-linear terms to the model ([Chapter 9](#sec-chap-9)).

## 7.3 Homoskedasticity

This assumption means that the variance of the residuals should not change as a function of the predicted values. Because we are again concerned with residuals and predicted values, we can re-use the same plot we used to check linearity. However, we are no longer interested in whether the lowess trend (red line) systematically deviates from zero – now we are interested in whether the range of the residuals (on the vertical axis) changes over the predicted values (on the horizontal axis).

[Figure 7.5](#fig-homo1) illustrates two data sets in which the assumption of linearity is met, but the right hand panel shows evidence of heteroskedasticity. This is apparent by observing the range of the residuals over values of .

# homoskedastic example  
set.seed(1)  
x <- sort(rnorm(250))  
e <- rnorm(250)  
y <- x + e  
mod3 <- lm(y~x)  
par(mfrow = c(1, 2))  
plot(mod3, which = 1)  
  
# Heteroskedastic example  
y2 <- y  
y2[x > 0] <- x[x > 0] + 3\* e[x > 0]  
y2[x < -1] <- x[x < -1] + .3\* e[x < -1]  
mod4 <- lm(y2~x)  
plot(mod4, which = 1)

|  |
| --- |
| Figure 7.5: Illustration of homo- and heteroskedasicity. |

To make it clearer what aspect of these plots is relevant for evaluating the assumption of homoskedasticity, the same figures are replicated below, but this time with blue lines represented my own “eye-balling” of the range of the residuals. In the left plot, the two lines are parallel, meaning the range is constant. In the right plot, the two lines form a cone, meaning the the range of the residuals increases for larger values of .

# homoskedastic example with ref lines  
par(mfrow = c(1, 2))  
plot(mod3, which = 1)  
segments(x0 = -1.5, y0 = 2, x1 = 1.5, y1 = 2, col = "#4B9CD3", lty = 2, lwd = 3)  
segments(x0 = -1.5, y0 = -2, x1 = 1.5, y1 = -2, col = "#4B9CD3", lty = 2, lwd = 3)  
  
# Heteroskedastic example  
plot(mod4, which = 1)  
segments(x0 = -1.5, y0 = 1, x1 = 1.5, y1 = 8, col = "#4B9CD3", lty = 2, lwd = 3)  
segments(x0 = -1.5, y0 = -1, x1 = 1.5, y1 = -8, col = "#4B9CD3", lty = 2, lwd = 3)  
  
# remove y2 from memory to avoid naming conflicts later on  
rm(y2)

|  |
| --- |
| Figure 7.6: Illustration of Homo- and Heteroskedasicity, with Reference Lines |

Note that I didn’t draw any lines in the tail ends of the plots – this is because there are fewer observations in the tails, so it is harder to make a judgment about the range of values. To avoid “reading the tea leaves” I focus on the values of for which there are sufficient observations to judge the range of the residuals.

To repeat, *the blue lines are just there for your reference*, to highlight the relevant information in the plot. You wouldn’t generally include these lines in the plot.

Let’s take another look at the plots in [Figure 7.4](#fig-linearity3). **Please write down whether you think the homoskedasticity assumption is problematic and explain why with reference to the plots**.

### 7.3.1 Dealing with Heteroskedasticity

Heteroskedasticity in linear regression, and corrections thereof, is a pretty big topic in the methodological literature (see (**cite-fox?**), section 12.2). In this section we are just going to discuss one widely used solution, and how to implement it in R.

As mentioned previously, heteroskedasticity affects the standard errors of the regression coefficients, and consequently their t-tests, p-values, and confidence intervals. In particular, the p-values for the regression coefficients will usually be too small if the data are heteroskedastic, but we mistakenly assume they are homoskedastic. Note that heteroskedasticity won’t affect the estimated values of the OLS regression coefficients (i.e., the ’s), and it also doesn’t affect R-squared or its F-test.

If our data exhibit heteroskedasticity, one solution is to use heteroskedasticity-consistent (HC) standard errors. HC standard errors are also sometimes called heteroskedasticity-robust, or just robust. The are also informally referred to as “sandwich” estimates – see (**cite-fox?**) section 12.2.3 for an explanation of this terminology.

Although there are many different version of HC standard errors, they are all equivalent with “large” samples. The simplest version is (see (**cite-fox?**), section 12.2.3)

In this equation, is the predicted value that results from regressing on the remaining predictors. The equation is not very intuitive to look at, but the general idea is that it can be derived without assuming homoskedasticity.

In terms of implementation, the procedure for using HC standard errors in R has three steps.

* First, we estimate the model as usual, (e.g., using the lm function)
* Second, we compute the HC standard errors (e.g., using the hccm function of the car package.)
* Third, we use the HC standard errors to compute the correct t-tests / confidence intervals (e.g., using the coeftest function of the lmtest package. )

You can find a more complete discussion of robust standard errors in R in the vignettes linked here: <http://jepusto.github.io/clubSandwich/>

The following output shows the results for the heteroskedastic (cone-shaped) example data in [Figure 7.6](#fig-homo2), using both the regular standard errors and HC standard errors. We can see that both sets of output are pretty similar: while the “Estimates” don’t change, the “Std. Errors” are a bit different in the two sets of output. In this case, the HC standard errors don’t affect conclusions about statistical significance, but in other cases they can lead to more dramatic differences in interpretation. **Please examine these two sets of output and write down any questions you have about their interpretation.**

* Regular OLS standard errors :

## Make sure the required packages are installed  
# install.packages("car")  
# install.packages("lmtest")  
  
# Regular SE:   
summary(mod4)

Call:  
lm(formula = y2 ~ x)  
  
Residuals:  
 Min 1Q Median 3Q Max   
-8.9985 -1.0402 -0.0504 0.9252 11.5002   
  
Coefficients:  
 Estimate Std. Error t value Pr(>|t|)   
(Intercept) -0.002712 0.154430 -0.018 0.986   
x 0.968044 0.160719 6.023 6.13e-09 \*\*\*  
---  
Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
  
Residual standard error: 2.441 on 248 degrees of freedom  
Multiple R-squared: 0.1276, Adjusted R-squared: 0.1241   
F-statistic: 36.28 on 1 and 248 DF, p-value: 6.135e-09

* HC standard errors:

# HC SE  
# Step 2. Use "hccm" to get the HC SEs for our model   
hcse <- car::hccm(mod4)  
  
# Step 3. Use "coeftest" to compute t-tests with the HC SEs  
lmtest::coeftest(mod4, hcse)

t test of coefficients:  
  
 Estimate Std. Error t value Pr(>|t|)   
(Intercept) -0.0027123 0.1520067 -0.0178 0.9858   
x 0.9680441 0.1908455 5.0724 7.701e-07 \*\*\*  
---  
Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

One final note: HC standard errors do not assume the data are homoskedastic. So, they can be used regardless of whether the homoskedasticity assumption is met or not. But, when the data *are* homoskedastic, the regular OLS standard errors are usually more precise (i.e., smaller). So, we generally don’t want to use HC standard errors unless there is evidence of heteroskedasticity in the data. This is why we do graphical checks first!

### 7.3.2 Summary

* The assumption of homoskedasticity (or homogeneity variance, or just equal variances) means that the variance of the residuals should not change as a function of the predicted values.
* Because we are again concerned with residuals and predicted values, we can re-use the same plot we used to check linearity. However, now we are interested in whether the range of the residuals (on the vertical axis) changes over the predicted values (on the horizontal axis).
* If we suspect that heteroskedasticity is a problem, we can adjust how the standard errors of the regression coefficients are computed. These adjusted standard errors are variously referred to as heteroskedasticity-consistent, heteroskedasticity-robust, robust, or “sandwich” estimates.
* To implement in HC standard errors in R, we can use a three step procedure:
* First, we estimate the model as usual, (e.g., using the lm function)
* Second, we compute the HC standard errors (e.g., using the hccm function of the car package.)
* Third, we use the HC standard errors to compute the correct t-tests / confidence intervals (e.g., using the coeftest function of the lmtest package. )

You can find a more complete discussion of robust standard errors in R in the vignettes linked here: <http://jepusto.github.io/clubSandwich/>

## 7.4 Normality

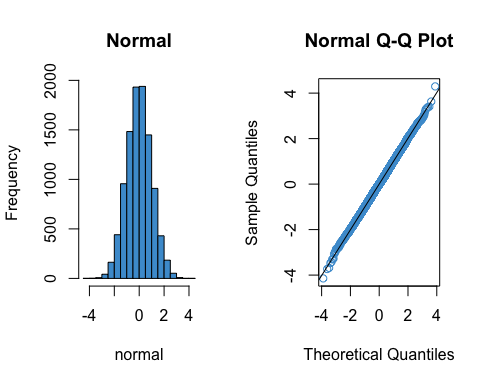
The last assumption we need to check is normality of the residuals. There are many ways to compare the empirical distribution of a variable (e.g., the residuals in a regression analysis) to a theoretical distribution (e.g., the normal). One general-purpose technique is a qq plot (short for quantile-quantile plot). A qq plot compares the quantiles (e.g., percentiles) of two different distributions.

For our assumption, we want to compare the quantiles of our standardized residuals to the quantiles of a standard normal distribution. Standardizing means the residuals should have variance equal to one, and, combined with the other population assumptions of linear regression, this implies that the residuals should have a standard normal distribution (see [Section 7.1](#sec-recap-7)).

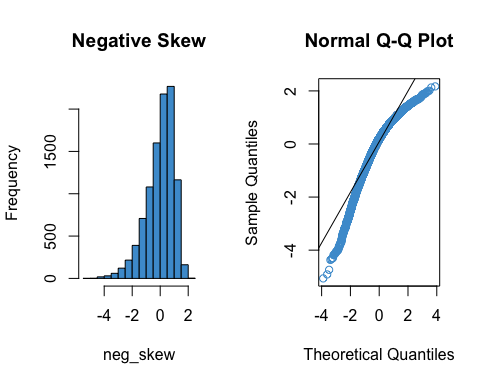
Since qq plots might not be something you have seen before, we’ll take a look at a few examples. Each figure below pairs a histogram and qq plot. In the qq plot, data points should fall on the diagonal line if the data are normally distributed. It should be emphasized that the line in the qq plot is *not a regression line!* It is just the diagonal line , and the bulk of the data points should fall on that line if the data were drawn from a normal distribution.

In the following examples, focus on how the pattern in the histogram shows up as deviations from the diagonal line in the qq plot. We will discuss the interpretation of these patterns together in class, but for now, **please write down any questions you have about the interpretation of the qq plots.**

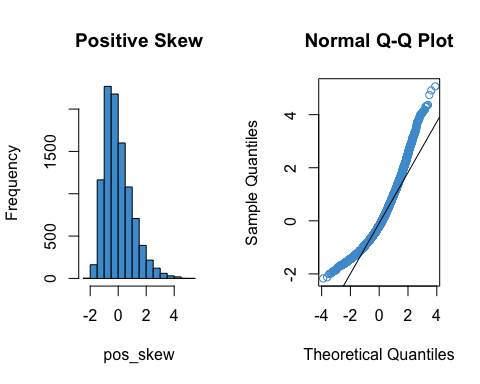
# Comparing histograms and q-q plots  
distributions <- read.csv("distributions.csv")  
names(distributions)[2] <- "normal"  
attach(distributions)  
dist\_names <- names(distributions)  
  
# Normal   
par(mfrow = c(1, 2))  
hist(normal, col = "#4B9CD3", main = "Normal")  
qqnorm(normal, col = "#4B9CD3")  
qqline(normal)



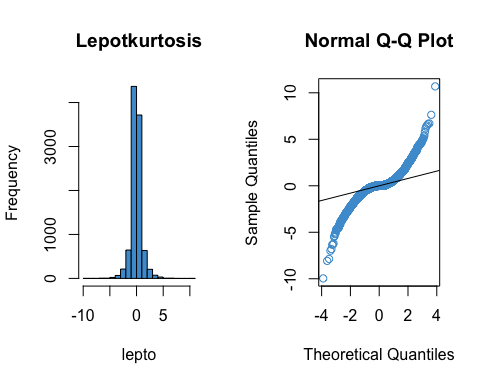
# Negative skew  
par(mfrow = c(1, 2))  
hist(neg\_skew, col = "#4B9CD3", main = "Negative Skew")  
qqnorm(neg\_skew, col = "#4B9CD3")  
qqline(neg\_skew)



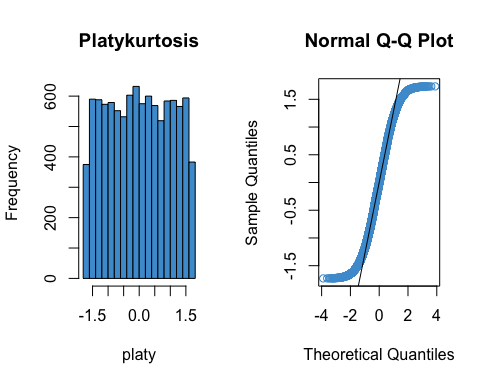
# Positive skew  
par(mfrow = c(1, 2))  
hist(pos\_skew, col = "#4B9CD3", main = "Positive Skew")  
qqnorm(pos\_skew, col = "#4B9CD3")  
qqline(pos\_skew)



# Leptokurtic  
par(mfrow = c(1, 2))  
hist(lepto, col = "#4B9CD3", main = "Lepotkurtosis")  
qqnorm(lepto, col = "#4B9CD3")  
qqline(lepto)



# Platykurtic  
par(mfrow = c(1, 2))  
hist(platy, col = "#4B9CD3", main = "Platykurtosis")  
qqnorm(platy, col = "#4B9CD3")  
qqline(platy)



detach(distributions)

Next, let’s consider a more realistic example using the default plotting from the lm function. **Please write down whether you think the normality assumption is problematic for the data in** [**Figure 7.7**](#fig-normality1)**, and be sure to explain why with reference to the plot.** Hint: if you think the data are non-normal, you should be able to interpret the pattern of deviations with reference to the examples given above (e.g. skew, kurtosis).

# Example: regression c1rmscal on ses\_orig and t1learn  
plot(mod2, which = 2)

|  |
| --- |
| Figure 7.7: An example from ECLS. |

Practice makes perfect, so lets work through a few more examples. In each of the examples in [Figure 7.8](#fig-normality2), **please write down whether you think the normality assumption is problematic and explain why with reference to the plots.**

# Example: regression c1rmscal on ses\_orig and t1learn  
set.seed(101)  
par(mfrow = c(2, 2))  
for(i in 1:4) {  
 x <- rnorm(200)  
 e <- rnorm(200)  
 y <- x + e   
 plot(lm(y ~ x), which = 2)  
}

|  |
| --- |
| Figure 7.8: More examples. |

### 7.4.1 Summary

To check the assumption of normality, we can use a qq plot of the standardized residuals against the standard normal distribution.

* If the points from the qq plot follows the line Y = X, then there is no evidence against the assumption.
* If the residuals do show a trend off of the diagonal line, then we should worry about the assumption.
* If the assumption is violated, the central limit theorem implies that significance tests used in OLS regression will be robust to violations of normality when sample sizes are large.
  + Practically this means we can often ignore mild violations of normality when (but other guidelines are used too).
  + For some specific types of violations, notably positive skew, it is also common to transform the Y variable (see [Chapter 9](#sec-chap-9)).
  + When we are worried that non-normality arises from a relatively small subset of the data that may be unduly influencing the results, this can be addressed through regression diagnostics (see [Section 7.6](#sec-diagnostics-7)) and robust statistics (which is an advanced topic).
* Oh, and one last thing: The line in a qq plot is *not a regression line!*

## 7.5 A worked example

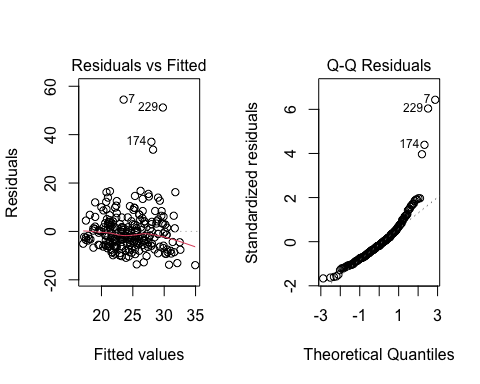
To illustrate the assumption checking procedures outlined above, let’s revisit the example from [Section 6.2](#sec-worked-example-6). In that example, it was noted that there was evidence that one (or more) of the assumptions were problematic. Let’s take a look at why this was the case.

For this example, we will use the ECLS data to regress reading achievement at the beginning of Kindergarten (c1rrscal) on SES (wksesl), parental (mother’s and father’s) education (wkmomed and wkdaded, respectively), and attendance in center-based care before K (p1center). This is model 3 from [Section 6.2](#sec-worked-example-6) (i.e., we don’t consider the model with the interactions).

The workflow for assumption checking requires first running the model and then producing the residual vs fitted plot and a qq plot of the residuals. If the plots look OK, we go ahead and interpret the model results. If the plots don’t look OK, we give up and wonder why we ever bothered with regression in the first place. Just kidding :) – the next two chapters of these notes address how to deal with violations of the linearity and normality assumptions, and we already know how to deal with heteroskedasticity from [Section 7.3.1](#sec-heteroskedasticity-7).

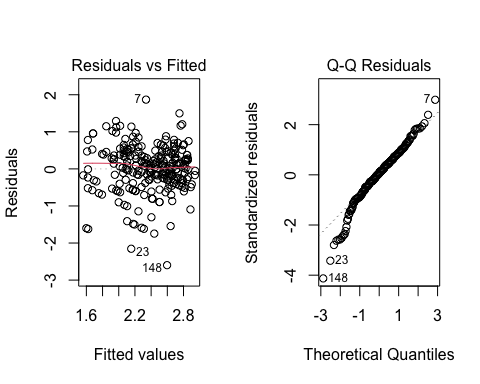
After running the model, we obtain the following two plots from the regression output. **For each of the three population assumptions of linear regression, please write down whether you think the assumption is problematic and explain why with reference to the plots.** Again, the purpose of this exercise is for you to think about how to interpret the plots with respect to the assumptions. I am looking for you to be explicit about how you reason from the plots to your conclusions. I am less interested in the conclusions per se, as this is something that requires practice to get right.

# Clean up and load data  
# rm(list = ls())  
# load("ECLS2577.RData")  
# attach(ecls)  
  
# Run model  
mod5 <- lm(c1rrscal ~ factor(p1center) + wksesl + wkmomed + wkdaded, data = ecls)  
  
# Check assumptions  
par(mfrow = c(1,2))  
plot(mod5, 1)  
plot(mod5, 2)



To foreshadow the next couple of chapters, here is what the plots looked like after dealing with positive skew of the residuals and the non-normality of the regression line:

# Run model  
log\_c1rrscal <- log(c1rrscal - min(c1rrscal) + 1)  
wksesl\_sq <- wksesl^2  
mod6 <- lm(log\_c1rrscal ~ factor(p1center) + wksesl + wksesl\_sq + wkmomed + wkdaded)  
  
# Check assumptions  
par(mfrow = c(1,2))  
plot(mod6, 1)  
plot(mod6, 2)



Neither assumption has been perfectly addressed, and, in particular, it looks like we may have over-corrected the positive skew and ended up with some negative skew. Also note that the data continue to exhibit heteroskedasticity, which is apparent from looking at the residual vs fitted plot in the range -1 to -3 of the residuals. Because heteroskedasticity is still an issue for these data, we can improve on the analysis reported in [Section 6.2](#sec-worked-example-6) by using HC standard errors rather than “regular” OLS standard errors. The difference between the two approaches is illustrated below. Remember, the estimates stay the same, but the SE’s change (and consequently the t-tests and p-values).

* Regular SE (same as Model 3 in [Section 6.2](#sec-worked-example-6)):

summary(mod6)

Call:  
lm(formula = log\_c1rrscal ~ factor(p1center) + wksesl + wksesl\_sq +   
 wkmomed + wkdaded)  
  
Residuals:  
 Min 1Q Median 3Q Max   
-2.59699 -0.29882 0.04381 0.37969 1.86787   
  
Coefficients:  
 Estimate Std. Error t value Pr(>|t|)   
(Intercept) -2.7941636 1.6307966 -1.713 0.08791 .   
factor(p1center)2 -0.1232061 0.1030926 -1.195 0.23321   
wksesl 0.1758849 0.0665763 2.642 0.00878 \*\*  
wksesl\_sq -0.0016162 0.0006514 -2.481 0.01378 \*   
wkmomed 0.0698645 0.0388073 1.800 0.07305 .   
wkdaded 0.0506601 0.0364953 1.388 0.16636   
---  
Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
  
Residual standard error: 0.6327 on 244 degrees of freedom  
Multiple R-squared: 0.2252, Adjusted R-squared: 0.2093   
F-statistic: 14.18 on 5 and 244 DF, p-value: 3.474e-12

* HC SE

lmtest::coeftest(mod6, car::hccm(mod6))

t test of coefficients:  
  
 Estimate Std. Error t value Pr(>|t|)   
(Intercept) -2.79416359 1.69074543 -1.6526 0.09969 .  
factor(p1center)2 -0.12320606 0.10855871 -1.1349 0.25752   
wksesl 0.17588489 0.06776405 2.5955 0.01002 \*  
wksesl\_sq -0.00161620 0.00065155 -2.4806 0.01379 \*  
wkmomed 0.06986450 0.03806165 1.8356 0.06764 .  
wkdaded 0.05066012 0.03077238 1.6463 0.10099   
---  
Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

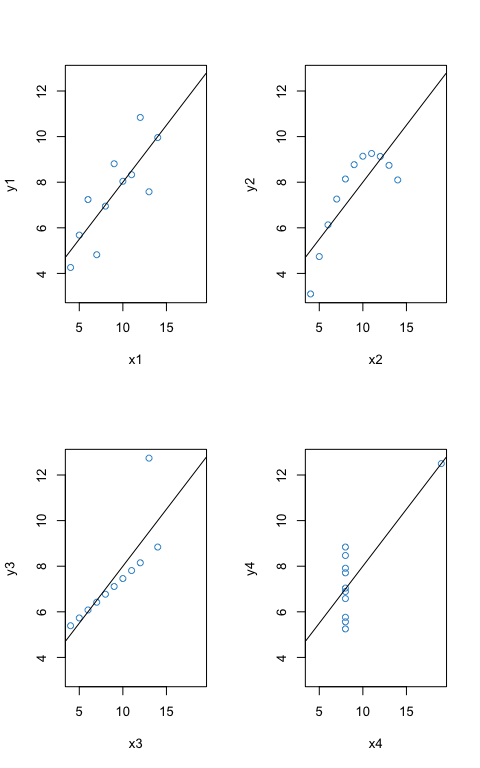
In this example, the HC SE’s didn’t lead to substantively different inferences, but we should report the results with HC SE’s based on the assumption checking.

## 7.6 Diagnostics\*

This section is optional and is currently under construction (i.e., many typos). It focuses on walking through the code so it is recommended to treat this like an Exercises section and scroll to the top of the page, click on the “</> Code” menu, then select “Show All Code.”

Like assumption checking, regression diagnostics also make extensive use of regression residuals, but this time the objective is to identify individual data points that are “outliers” with respect to the model. To work through the basic concepts of regression diagnostics, let’s again use the Anscombe’s quartet.

# Plotting Anscombe's quartet  
attach(anscombe)  
par(mfrow = c(2,2))  
ymax <- max(anscombe[,5:8])  
ymin <- min(anscombe[,5:8])  
xmax <- max(anscombe[,1:4])  
xmin <- min(anscombe[,1:4])  
  
  
plot(x1, y1, col = "#4B9CD3", xlim = c(xmin, xmax), ylim = c(ymin, ymax))  
abline(lm(y1 ~ x1))  
  
plot(x2, y2, col = "#4B9CD3", xlim = c(xmin, xmax), ylim = c(ymin, ymax))  
abline(lm(y2 ~ x2))  
  
plot(x3, y3, col = "#4B9CD3", xlim = c(xmin, xmax), ylim = c(ymin, ymax))  
abline(lm(y3 ~ x3))  
  
plot(x4, y4, col = "#4B9CD3", xlim = c(xmin, xmax), ylim = c(ymin, ymax))  
abline(lm(y4 ~ x4))



The interesting thing about these four examples is that they all have the same univariate and bivariate summary statistics – e.g., the same mean, variance, covariance, correlation, and regression coefficients. But first example is the only one that would be suitable for analysis by using these statistics. The second example shows a non-linear relationship, which we addressed in Section @ref(linearity-8). In this section we will focus on the last two examples, since they have clear outliers.

### 7.6.1 Leverage

Leverage describes how “unusual” a data point is on the X variable(s) – i.e., how far from the mean it is on each predictor. The function hatvalues computes the leverage for each data point. Let’s check out the leverage for the 3rd and 4th examples from Anscombe’s quartet.

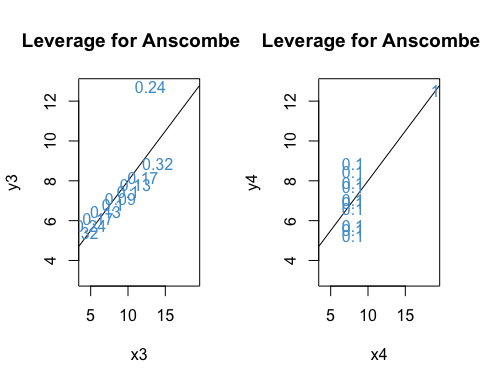
# leverage for Anscombe 3  
anscombe3 <- lm(y3 ~ x3)  
leverage3 <- hatvalues(anscombe3)  
  
# Take a look at the leverage values for each data point  
leverage3

1 2 3 4 5 6 7   
0.10000000 0.10000000 0.23636364 0.09090909 0.12727273 0.31818182 0.17272727   
 8 9 10 11   
0.31818182 0.17272727 0.12727273 0.23636364

# Show the leverage values in the scatter plot using the function "text"  
par(mfrow = c(1,2))  
  
plot(x3, y3, col = "white", xlim = c(xmin, xmax), ylim = c(ymin, ymax), main = "Leverage for Anscombe 3")  
abline(lm(y3 ~ x3))  
  
text(y3 ~ x3, labels = round(leverage3, 2), col = "#4B9CD3")  
  
# leverage for Anscombe 4  
anscombe4 <- lm(y4 ~ x4)  
leverage4 <- hatvalues(anscombe4)  
  
# Take a look at the leverage values for each data point  
leverage4

1 2 3 4 5 6 7 8 9 10 11   
0.1 0.1 0.1 0.1 0.1 0.1 0.1 1.0 0.1 0.1 0.1

# Show the leverage values in the scatter plot using the function "text"  
plot(x4, y4, col = "white", xlim = c(xmin, xmax), ylim = c(ymin, ymax), main = "Leverage for Anscombe 4")  
abline(lm(y4 ~ x4))  
  
text(y4 ~ x4, labels = round(leverage4, 2), col = "#4B9CD3")



Recall from the lesson that

* h should be smaller (closer to 0) for values closer to the mean of X
* The maximum value of h is 1

Based on the plots, we can see that the largest leverage is for the outlier in Anscombe 4.

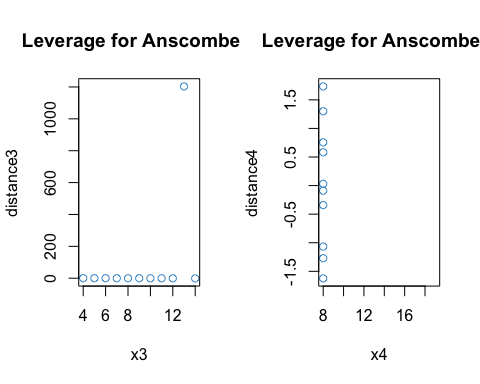
### 7.6.2 Distance (residuals)

Distance is about the size of the residuals. In order to judge the size of a residual, it helps to use the (externally) studentized residuals rather than the “raw” residuals. Because the studentized residual have a t-distribution on degrees of freedom, a rough ballpark for interpreting studentized residuals is that

* Values around +/- 2 are considered large.
* Values beyond +/- 3 are considered very large.

Let’s see what we have for our examples:

# Distance for Anscombe 3  
distance3 <- rstudent(anscombe3)  
  
par(mfrow = c(1,2))  
plot(x3, distance3, main = "Leverage for Anscombe 3", col = "#4B9CD3")  
  
# Distance for Anscombe 4  
distance4 <- rstudent(anscombe4)  
plot(x4, distance4, main = "Leverage for Anscombe 4", col = "#4B9CD3")



Clearly, the notion of distance is useful for describing what the problem is with Anscombe’s 3rd example. For the 4th example, the outlying data point is omitted because it has leverage of exactly 1, which means that the studentized residuals are undefined (divide by zero; R notes this in the console).

### 7.6.3 Influence

Influence describes how much the model results would change if a data point were omitted. Roughly, the conceptual relationships among influence, distance, and leverage are given by the following equation:

[ = ]

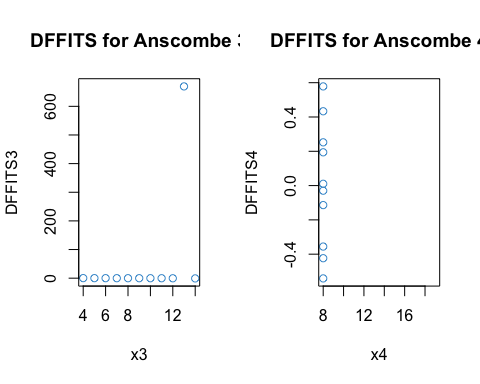
This equation tells us that, for a data point to have high influence, it must be a large distance from the regression line (have a large residual) *and* have high leverage (be far away from the mean on ).

There are a number of ways of computing influence. Like externally studentized residuals, they are all deletion statistics, or statistics computed using a “leave-one-out” approach.

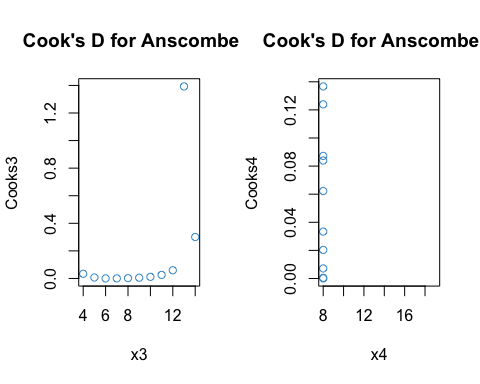
Influence statistics can also be classified into global versus local. Global approaches consider how a data point affects the predicted values. Local approaches consider how a data point affects the value of a specific regression coefficient.

Let’s start with DFFITS and Cook’s distance, two measures of global influence.

# DFFITS for Anscombe 3  
DFFITS3 <- dffits(anscombe3)  
  
par(mfrow = c(1,2))  
plot(x3, DFFITS3, main = "DFFITS for Anscombe 3", col = "#4B9CD3")  
  
# Distance for Anscombe 4  
DFFITS4 <- dffits(anscombe4)  
plot(x4, DFFITS4, main = "DFFITS for Anscombe 4", col = "#4B9CD3")



# Cook's distance for Anscombe 3  
Cooks3 <- cooks.distance(anscombe3)  
  
par(mfrow = c(1,2))  
plot(x3, Cooks3, main = "Cook's D for Anscombe 3", col = "#4B9CD3")  
  
# Distance for Anscombe 4  
Cooks4 <- cooks.distance(anscombe4)  
plot(x4, Cooks4, main = "Cook's D for Anscombe 4", col = "#4B9CD3")



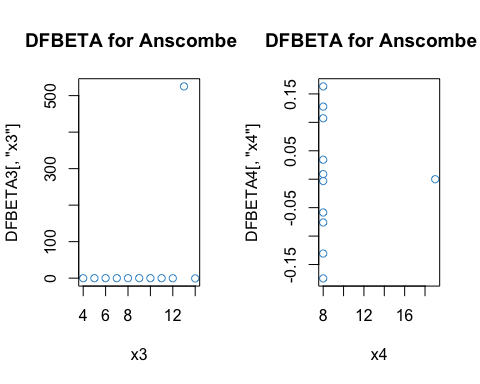
These statistics are similar but Cook’s distance is more interpetable. Values greater than 1 are considered indicative of high influence. We can see that the outlier in the 3rd example is highly influence. The outlier in the 4th example is “NA” because h = 1 hence there is a divide by zero problem.

For local measures of influence, the interpretation is roughly the same as global measures with simple regression (i.e., a single predictor). For multiple regression, local measures can provided additional insight to consider which regression coefficients are most influenced by an outlier.

# DFBETAs distance for Anscombe 3  
DFBETA3 <- dfbetas(anscombe3)  
  
#Take a look at the output: We get values for each coefficient, including the intercept  
DFBETA3

(Intercept) x3  
1 -4.625738e-03 -4.412673e-02  
2 -3.713338e-02 1.864368e-02  
3 -3.579096e+02 5.252677e+02  
4 -3.289981e-02 -1.737209e-18  
5 4.915510e-02 -1.172274e-01  
6 4.897424e-01 -6.674064e-01  
7 2.700082e-02 -2.088417e-02  
8 2.409027e-01 -2.089150e-01  
9 1.374342e-01 -2.313597e-01  
10 -1.970229e-02 1.342485e-02  
11 1.053656e-01 -8.740210e-02

# Plots for the regression coefficients  
par(mfrow = c(1,2))  
plot(x3, DFBETA3[,"x3"], main = "DFBETA for Anscombe 3", col = "#4B9CD3")  
  
# Distance for Anscombe 4  
DFBETA4 <- dfbetas(anscombe4)  
plot(x4, DFBETA4[,"x4"], main = "DFBETA for Anscombe 4", col = "#4B9CD3")



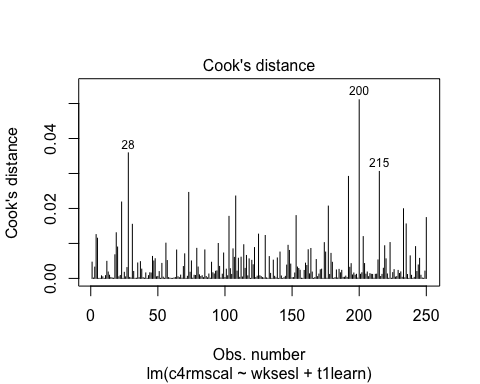
The output for DFBETA3 looks a lot like DFFITS3 (because we only have one predictor).

For DFBETA4, it is strange that our unusual data point (x4 = 18) was not identified as problematic – keep in mind that when this data point is removed, the variance of X is zero and so the regression coefficient for the leave-one-out model is not defined. Still, it is not clear why R reports the value as zero, rather than omitted.

### 7.6.4 A more realistic example

As a more realistic example, let’s consider Question 3 from Assignment 1 using the graphical output from lm. Note that the graphical output uses the internally studentized residuals, and refers to these as “standardized residuals”.

model <- lm(c4rmscal ~ wksesl + t1learn, data = ecls)  
  
# Influence via Cook's distance  
plot(model, which = 4)



We can see that, although R automatically labels the 3 data points with the highest values of Cook’s D, none of the data points are actually close to the cut off value of 1. In other words, none of the data points in this example have an undue influence on the results of the regression model.

## 7.7 Workbook

This section collects the questions asked in this chapter. The lesson for this chapter will focus on discussing these questions and then working on the exercises in [Section 7.8](#sec-exercises-7). The lesson will **not** be a lecture that reviews all of the material in the chapter! So, if you haven’t written down / thought about the answers to these questions before class, the lesson will not be very useful for you. Please engage with each question by writing down one or more answers, asking clarifying questions about related material, posing follow up questions, etc.

[Section 7.2](#sec-linearity-7)

* Please write down whether you think the linearity assumption is problematic for the example below, and be sure to explain why with reference to the figure. Keep in mind that interpreting plots takes a bit of practice and in general there is no “right” answer. Rather, what I am looking for is an explanation of *why* you think the assumption is problematic or not. Your explanation should refer to the interpretation of residual vs fitted plots, as outlined above.

plot(mod2, which = 1)

|  |
| --- |
| An example from ECLS. |

* For each of the four examples below, please write down whether you think the linearity assumption is problematic and explain why with reference to the plots. Hint: be careful not to over-interpret the lowess line in the tails of the plots, where only a few data points can have a big impact on the local trend. Focus your interpretation on the bulk of the data, and whether it shows a systemic trend away from the horizontal line at 0.

# Example: regression c1rmscal on ses\_orig and t1learn  
set.seed(101)  
par(mfrow = c(2, 2))  
for(i in 1:4) {  
 x <- rnorm(200)  
 e <- rnorm(200)  
 y <- x + e   
 plot(lm(y ~ x), which = 1)  
}

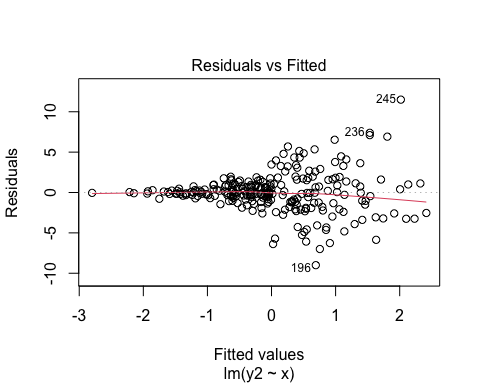
|  |
| --- |
| More examples. |

[Section 7.3](#sec-homoskedasticity-7)

* Let’s take another look at the four plots in the above figure. For each plot, please write down whether you think the homoskedasticity assumption is problematic and explain why with reference to the plot.
* The following output shows the results for the heteroskedastic (cone-shaped) example data, using both the regular standard error and HC standard errors. Please note the differences between these two sets of output and write down any questions you have about their interpretation.

Example:

plot(mod4, which = 1)



Regular SE:

## Make sure the required packages are installed  
# install.packages("car")  
# install.packages("lmtest")  
  
# Regular SE:   
summary(mod4)

Call:  
lm(formula = y2 ~ x)  
  
Residuals:  
 Min 1Q Median 3Q Max   
-8.9985 -1.0402 -0.0504 0.9252 11.5002   
  
Coefficients:  
 Estimate Std. Error t value Pr(>|t|)   
(Intercept) -0.002712 0.154430 -0.018 0.986   
x 0.968044 0.160719 6.023 6.13e-09 \*\*\*  
---  
Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
  
Residual standard error: 2.441 on 248 degrees of freedom  
Multiple R-squared: 0.1276, Adjusted R-squared: 0.1241   
F-statistic: 36.28 on 1 and 248 DF, p-value: 6.135e-09

HC SE:

# HC SE  
# Step 2. Use "hccm" to get the HC SEs for our piecewise model   
hcse <- car::hccm(mod4)  
  
# Step 3. Use "coeftest" to compute t-tests with the HC SEs  
lmtest::coeftest(mod4, hcse)

t test of coefficients:  
  
 Estimate Std. Error t value Pr(>|t|)   
(Intercept) -0.0027123 0.1520067 -0.0178 0.9858   
x 0.9680441 0.1908455 5.0724 7.701e-07 \*\*\*  
---  
Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

[Section 7.4](#sec-normality-7)

* Please write down whether you think the normality assumption is problematic for the data in the figure below, and be sure to explain why with reference to the plot. Hint: if you think the data are non-normal, you should be able to interpret the pattern of deviations (e.g. skew, kurtosis).

# Example: regression c1rmscal on ses\_orig and t1learn  
plot(mod2, which = 2)

|  |
| --- |
| An example from ECLS. |

* In each of the examples below, please write down whether you think the normality assumption is problematic and explain why with reference to the plots.

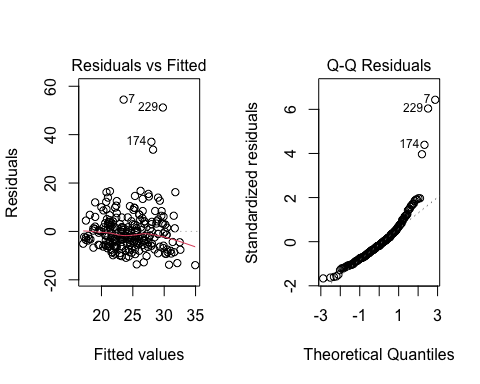
# Example: regression c1rmscal on ses\_orig and t1learn  
set.seed(101)  
par(mfrow = c(2, 2))  
for(i in 1:4) {  
 x <- rnorm(200)  
 e <- rnorm(200)  
 y <- x + e   
 plot(lm(y ~ x), which = 2)  
}

|  |
| --- |
| More examples. |

[Section 7.5](#sec-worked-example-7)

* For each of the three population assumptions of linear regression, please write down whether you think the assumption is problematic and explain why with reference to the plots. Again, the purpose of this exercise is for you to think about how to interpret the plots with respect to the assumptions. I am looking for you to be explicit about how you reason from the plots to your conclusions. I am less interested in the conclusions per se, as this is something that requires practice to get right.

# Clean up and load data  
# rm(list = ls())  
# load("ECLS2577.RData")  
# attach(ecls)  
  
# Run model  
mod5 <- lm(c1rrscal ~ factor(p1center) + wksesl + wkmomed + wkdaded, data = ecls)  
  
# Check assumptions  
par(mfrow = c(1,2))  
plot(mod5, 1)  
plot(mod5, 2)



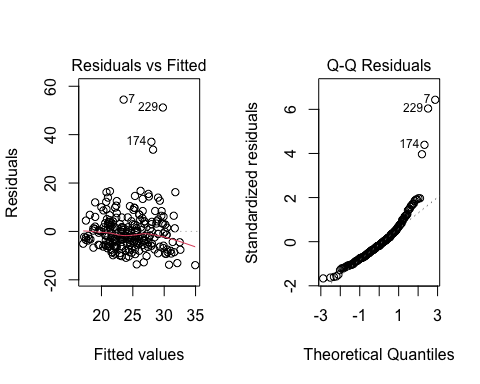
## 7.8 Exercises

These exercises collect all of the R input used in this chapter into a single step-by-step analysis. It explains how the R input works, and provides some additional exercises. We will go through this material in class together, so you don’t need to work on it before class (but you can if you want.)

Before staring this section, you may find it useful to scroll to the top of the page, click on the “</> Code” menu, and select “Show All Code.”

There isn’t much new in terms of R code in this chapter. Once we run a model with lm, we just call the plot function on the lm output to produce the graphics requires for assumption checking. This section shows these steps for the worked example in [Section 6.2](#sec-worked-example-6) and [Section 7.5](#sec-worked-example-7).

# Clearn up env and load data   
#rm(list = ls())  
#load("ECLS2577.RData")  
  
# Run model 3 from the example (all predictors, but no interactions)  
mod <- lm(c1rrscal ~ factor(p1center) + wksesl + wkmomed + wkdaded, data = ecls)  
  
# Puts both plots in one figure  
par(mfrow = c(1,2))   
  
# Check assumptions  
plot(mod, 1)  
plot(mod, 2)



The next bit of code shows how to adjust the statistical tests for heteroskedasticity using HC standard errors. In the next two chapters, we show how to address the linearity and normality assumption violations.

* Regular SE:

summary(mod)

Call:  
lm(formula = c1rrscal ~ factor(p1center) + wksesl + wkmomed +   
 wkdaded, data = ecls)  
  
Residuals:  
 Min 1Q Median 3Q Max   
-13.943 -5.080 -1.531 3.049 54.466   
  
Coefficients:  
 Estimate Std. Error t value Pr(>|t|)  
(Intercept) 7.0118 5.3078 1.321 0.188  
factor(p1center)2 -1.7608 1.3901 -1.267 0.206  
wksesl 0.2828 0.1724 1.640 0.102  
wkmomed 0.6685 0.5203 1.285 0.200  
wkdaded 0.2352 0.4935 0.477 0.634  
  
Residual standard error: 8.56 on 245 degrees of freedom  
Multiple R-squared: 0.1553, Adjusted R-squared: 0.1415   
F-statistic: 11.26 on 4 and 245 DF, p-value: 2.109e-08

* HC SE

## Make sure the required packages are installed  
# install.packages("car")  
# install.packages("lmtest")  
  
# Step 1: fit the model (see above)  
  
# Step 2: Use "hccm" to get the HC SEs   
hcse <- car::hccm(mod)  
  
# Step 3. Use "coeftest" to compute t-tests with the HC SEs  
lmtest::coeftest(mod, hcse)

t test of coefficients:  
  
 Estimate Std. Error t value Pr(>|t|)  
(Intercept) 7.01177 5.82938 1.2028 0.2302  
factor(p1center)2 -1.76083 1.11866 -1.5740 0.1168  
wksesl 0.28282 0.20506 1.3792 0.1691  
wkmomed 0.66853 0.66797 1.0008 0.3179  
wkdaded 0.23520 0.46189 0.5092 0.6111

As discussed above, the interpretation for both sets of output is essentially the same, but by using the HC SEs we can be sure that our inferences are not unduly affected by the assumption of homoskedasticity.

# 8. Log-linear regression

In the previous chapter we considered how to check the population assumptions of linear regression for a given data set. In this chapter we discuss one way to deal with assumption violations – by “transforming” the variable. In general, a transformation is just a function we apply to a variable to change it into another variable. We have already seen some examples of transformations: centering or deviation scores, standardizing or z-scores, and percentiles.

In this chapter we will apply the log transformation to the variable in a regression model, which results in the log-linear regression model. This transformation is widely used to address skew in the variable / residuals, and it also serves to illustrate the overall approach to transforming the variable in a regression model to address assumption violations.

There are two big themes to this chapter. First, whenever we apply a non-linear transform to the variable in a regression model, it always has three types of consequences. It affects:

1. The distribution of the variable / residuals.
2. The (non-)linearity of its relationship with predictor variables.
3. The interpretation of the regression coefficients.

We have to consider all three of these consequences when transforming the variable. In particular, we might find that addressing one problem (e.g., non-normality) can create other problems (e.g., nonlinearity). The moral is that addressing assumption violations by transforming the variable is not always as straightforward as we might hope.

The other big theme of this chapter is the overall strategy for dealing with assumption violations. In particular, when we are faced with data that are not compatible with the assumptions of regression modeling, we often proceed as follows:

1. Transform the data to make them compatible with linear regression (or some other model we know how to use).
2. Run the analysis on the transformed data.
3. “Reverse-transform” the results of the analysis so that they are interpretable in terms of the original variable(s).

We will see this “transformanalyzereverse- transform” strategy in this chapter, and again when we cover logistic regression in [Chapter 10](#sec-chap-10). Both of these approaches are exemplars of *generalized linear modeling*, which is an advanced regression topic that formalizes the intuitive idea behind this approach.

This chapter addresses transformations of the -variable only. Another approach for dealing with assumption violations in linear regression is to transform one or more of the variables instead of (or in addition to) the variable. These approaches are discussed in the next chapter.

## 8.1 Math review

Let’s start with a review the requisite math – logs and exponents. We are going to use these concepts later in the course as well, so it is worth sorting them out. If you haven’t seen logs or exponents in while, brace yourself. This fist section of the math review covers the basic interpretation of logs and their utility for reducing skewness. The following sections add some details.

In this math review, we use the lowercase symbols and for generic mathematical variables. These symbols do not correspond to the uppercase symbols and in regression notation. So, e.g., is not a predictor, its just a generic variable.

The function returns the power to which we need to raise 10 to get the value . It answers the question: .

Some examples:

* , because
* , because
* , because
* …

In [Figure 8.1](#fig-logx), the three values computed above are shown as vertical dashed lines. In general, the plots shows that for every order of magnitude that increases (e.g., from 10 to 100, or from 100 to 1000), its log only increases by one unit. Intuitively, this means that “big differences” on translate into “small differences” on .

|  |
| --- |
| Figure 8.1: Log base 10 |

This “compression” of large values down to a smaller scale turns out to be useful for dealing with positive skew, as illustrated in [Figure 8.2](#fig-logdata). This is a main reason that the log transform is widely used – to reduce positive skew.

|  |
| --- |
| Figure 8.2: Log transform and positive skew |

### 8.1.1 Natural logarithm

The symbol is read “log base 10”. In statistics, we usually use a different base – log base . Here is an irrational number called *Euler’s number*. We will see later on that using makes the interpretation of log-linear regression simpler. Note that in this context, is not the residual from a regression model. Whether denotes Euler’s number or regression residuals should be clear from context.

Regardless of which base we use, logs have the same overall compression effect, as shown in [Figure 8.3](#fig-loge). For example, and – both numbers are much smaller than , which means that both functions achieve the same overall result of making big values into smaller values. The same is true when applied to real data – in fact, the right hand panel in [Figure 8.2](#fig-logdata) was computed using .

|  |
| --- |
| Figure 8.3: Log base 10 vs log base e |

Log base , or ,is called the *natural logarithm* and often denoted “”. But can be hard to read so we will just stick with and omit the base symbol when we mean . This is consistent with what R’s log function does – its default base is .

### 8.1.2 log and exp

Recall that our overall modeling strategy in log-linear regression is “transformanalyzereverse- transform”. In mathematics, the reverse transform is called the *inverse* of the original function. The inverse of the logarithm is exponentiation. So, if we start by log-transforming our -variable, we are going to end up doing exponents later on.

To see how and work together, let’s start with the number , and take the log of that number:

This equation tells us that if we raise to the power of the result will be . We can think of as our original data and as our transformed value.

By definition of the logarithm, we can equivalently write as

This equation uses exponents (raise to a power) rather than logs, but it communicates the same information as the previous equation. In our second equation, we take the output of the log () and use it as input to the exponent. The exponent then returns the original value of .

The mathematical symbol is often used instead of to avoid having to write complicated expressions in the superscript. So we can more clearly write the above equation as

This is just a change of notation so we don’t have to typeset superscripts.

To summarize: the two equations presented below are equivalent by definition:

In the first one, we take the input value and transform it using the logarithm, which gives us the transformed value . In the second equation, we take the transformed value as input into the exponent, which gives us back the original value of .

In general, the relationship between the two functions is:

### 8.1.3 Pop quiz

I’ll ask some questions like the following to start class off. **Please write down your answers and be prepared to share them in class**. You check your answers by pasting the questions into or Google’s search bar.

* What is larger, or ?

### 8.1.4 Rules for log and exp\*

There are some rules for working with logs and exponents that we will use later on to derive some results about the interpretation of log-linear regression. You don’t need to know these but you might find them useful.

* Addition with logs:
* Multiplication with exponents:

## 8.2 The log-linear model

There are a various ways to apply log-transformations in regression analysis, which are outlined in the following table and discussed in more detail in (**cite-fox?**). The focus of this chapter is log-linear regression, but the overall approach to linear-log and log-log is the similar. A worked example is coming up in [Section 9.2](#sec-worked-example-9). (Note that the term “log-linear” is also the name of a family of models for contingency table data – that is not what we are talking about here.)

|  |
| --- |
| Figure 8.4: Logs in regression analysis |

Before getting into the worked example, recall that a main theme of this chapter is that transforming the variable in a regression has three interrelated consequences:

* Changing the distribution Y variable / residuals
* Changing the (non-)linearity of its relationship with other variables
* Changing interpretation of the regression coefficients

The next few sections address each of these points in more detail.

## 8.3 Distribution of

Below are some examples of the log transform applied to positively skewed data. We can see a range of results, from “wow, that definitely worked” in the first example to, “better, but still not great” in the second two examples. These are realistic reflections of how a log transform works to address skew.

|  |
| --- |
| Figure 8.5: Example 1: Log transforming ECLS math |

|  |
| --- |
| Figure 8.6: Example 2: Log transforming ECLS reading |

|  |
| --- |
| Figure 8.7: Example 3: Log transforming hourly wages |

### 8.3.1 Why ?

There are a few things to note about this approach to correcting positive skew. First, you might have nocited that we are computing instead of . This is because the log function has “weird” behavior for values of between 0 and 1. These values of were not shown in [Figure 8.1](#fig-logx) and [Figure 8.3](#fig-loge), but they are shown below in [Figure 8.8](#fig-log0).

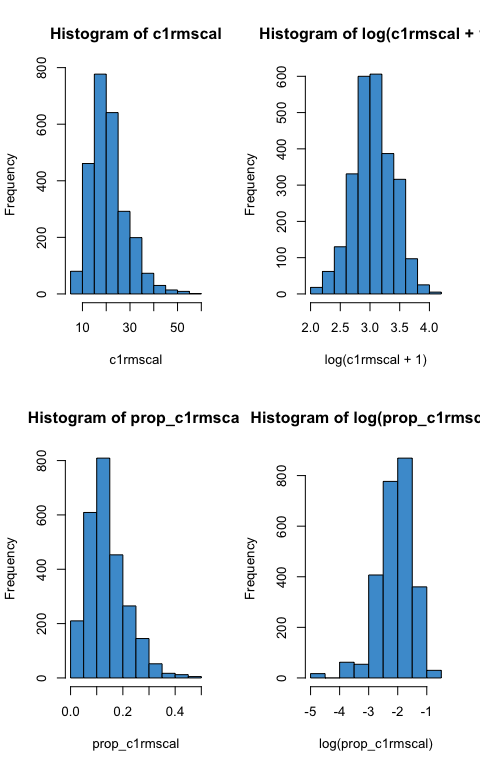
|  |
| --- |
| Figure 8.8: Logs for small values of X |

We can see that , which is true regardless of the base (i.e., for all choices of the ). But for values of , goes to negative infinity. This is because of how negative exponents are defined:

The upshot for log-linear regression is as follows: if your variable takes on values in the range , the log transform is going to change those values into large negative numbers. This can result in negative skew, rather fixing your positive skew.

**?@fig-skew4** provides an example. The top row shows is the example as [Figure 8.5](#fig-skew1) above, (math achievement in ECLS), and the bottom shows the same variable but transformed to proportion correct. In the left-hand panels we can see that the distribution of the proportion is qualitatively the same the as the distribution of the original variable. However, the log transform behaves differently for these two cases. Why? Because logs are “weird” on the range .

par(mfrow = c(2,2))  
prop\_c1rmscal <- (c1rmscal - min(c1rmscal))/100  
hist(c1rmscal, col = "#4b9cd3")  
hist(log(c1rmscal + 1), col = "#4b9cd3")  
hist(prop\_c1rmscal, col = "#4b9cd3")  
hist(log(prop\_c1rmscal), col = "#4b9cd3")

{#fig-skew4, fig-align=‘center’}

To avoid this situation we can add 1 to a variable before taking its log. Technically, there is no reason to do this if the variable does not take on values in the range 0 to 1, but I always apply the “add 1” rule just so I don’t have to worry about it.

### 8.3.2 What about negative values of ?

You may have noted that the “add 1” rule doesn’t cover cases where the variable can take on negative values. The problem here is that there is no way to turn a positive base into a negative number through exponentiation - i.e., is undefined whenever .

To address situations where you want to log transform a variable that can take on negative values, we can use the transformation

The following table shows how this transformation works. The first column shows the original data, the second column shows how subtracting the minimum makes all of the negative numbers positive, and the third column shows how adding 1 makes all of the numbers less than 1 equal to or greater than 1.

|  |
| --- |
| Dealing with negative values of x in log transforms |

The take home message: If we have negative values of , before we apply the log transform we need to:

* Step 1. “Add” the minimum value of to prevent undefined values of log
* Step 2. Add 1 to make sure no values are less than 1

**Can you think of a log transformation that would work to address data that were negatively skewed, rather than positively skewed? Hint: the trick is similar to the one in this section**

### 8.3.3 Skew versus censoring

It is important to distinguish between skew and floor (or ceiling) effects. Floor and ceiling effects are collectively known as “censoring”. While transforming you data can help with skew, it isn’t going to do anything about censoring, so it is important to be able to tell the difference.

Censoring can look like extreme skew, but is importantly different from skew because a large proportion of the observations have exactly the same value. An example is shown in [Figure 8.9](#fig-floor) below, which shows a floor effect in which a large proportion of the cases have a values of 0 (“are censored at zero”). Note that the log transformation does not do anything about the censored data. In general, if you transform a “heap” of values, then the same heap shows up in the transformed data.

To deal with censored data, we need to consider alternatives to linear regression (e.g., tobit regression). We won’t cover regression for censored data in this course. For our purposes, the moral is: don’t mistake censoring for skew, because they don’t have the same solution.

|  |
| --- |
| Figure 8.9: A floor effect is not the same as positive skew |

### 8.3.4 Final comments on dealing with skew

Recall that the assumptions of linear regression are not about the variable itself, but about the residuals ([Section 7.1](#sec-recap-7)). So, a skewed outcome variable is not necessarily a problem. In particular, it is entirely possible that is positively skewed but the residuals are not, especially if one of the variables has a similar distribution to . So, in practice, you should not overly worry about non-normality of – its the residuals of your regression model that matter.

Second, even if the residuals are non-normal, this still doesn’t matter very much with very large samples. The central limit theorem assures us that, as the sample size gets larger, the sampling distribution of the regression coefficients converges to a normal distribution, regardless of the distribution of the residuals.

The moral of this section is that, if you are considering log-transforming an outcome variable due to skew, you should keep in mind:

1. The problem may not really need to be fixed:
   * Check the residuals!
   * If you have sufficient sample size, mild to moderate violations of normality are not really an issue.
2. The log-transformation has other important implications for you model, which we discuss next.

## 8.4 Relationship with

In addition to affecting normality,(non-linear) transformations of the variable necessarily affect the linearity of regression. In particular, if is linearly related to , then cannot also be linearly related to .

Stated more formally, if

then

In practice, this means that addressing positive skew using a log transform can sometimes create more problems than it solves. [Figure 8.10](#fig-nonlinear1) illustrates this issue. In the top row, we see the residual versus fitted plot and qqplot of residuals for the regression of Math Achievement (c1rmscal) on SES (wksesl). The linear regression resulted in a clear violation of normality (positive skew). In the bottom row, we see same output after log transforming Math Achievement – it addressed the issue with normality, but also made the nonlinearity more apparent.

|  |
| --- |
| Figure 8.10: Nonlinear vs non-normality when log transforming Y |

The take home message is that when we transform (or ) we necessarily affect the relationship between the variables, as well as their individual distributions. It is often the case that when we try to “fix” the normality of the variable, we can inadvertently make its relationship to more non-linear.

As we discuss in the next chapter, we can address non-linearity of regression by transforming the variable(s). But we should be sure that the problem we are trying to solve (normality) isn’t worse than the the problem we end up with (non-linearity).

## 8.5 Interpretation

The final point about the log-linear model is how it affects the interpretation of the model parameters. For example, in linear regression we know that a one-unit increase in leads to a -unit increase in . But how do we interpret a -unit increase in ?

In general, one of the main shortcomings of “transforming your data” is that it can lead to uninformative or unclear interpretations of model parameters. If the results don’t make sense, its not much of a consolation to know that the model assumptions looked OK!

However, it turns out that there is very nice interpretation of the log-linear model. It is because of this interpretation that we often use the log transform to address skew in , rather than some other transformation.

There are two ways to interpret the regression coefficients in the log-linear model. First, we consider the easy but approximate way (you’re welcome!). Second, we consider the more complicated but accurate way, which involves reverse transforming the regression coefficients using exponentiation (see [Section 8.1](#sec-math-review-8)).

### 8.5.1 Approximate interpretation

In a log-linear model, the regression slope can be interpreted as the *approximate* proportionate change in , in the original units. So, in this equation,

we can interpret as the proportionate change in associated with a one unit increase in . This should seem like surprising: is not the outcome variable – is. But, we can still interpret the model with respect to !

Below are some numerical examples illustrating how to interpret the regression slope in [Equation 8.1](#eq-logy):

* if , then is expected to increase by 25% when increases by 1 unit.
* if , then is expected to be the same when increases by 1 unit (i.e., no relationship)
* if , then is expected to decrease by 45% when increases by 1 unit.

It is important to note that this interpretation is approximate and it only applies when is small (e.g., ). We will see how the approximation works and how to deal with other values of when we get to the exact interpretation in [Section 8.5.3](#sec-exact-8).

Before moving on, let’s get clear on what “proportionate change” means. Consider two values of a variable. We let denote the original value and denote a new value. The “change” is the difference . Treating the change as a proportion of the original value gives us the proportionate change:

Multiplying the proportionate change by 100 gives us the percentage change, which is usually easier to talk about in a sentence (the numerical examples listed above used percentage change). If we want to talk about the change without specifying the units (e.g, proportion or percentage), we say “relative change.”

Note that relative change is not the same as relative magnitude. is the magnitude of relative to . As shown in the above equation for proportionate change, we need to subtract 1 from relative magnitude to get relative change.

For example, say I loan dollars to a friend and they pay me back dollars. The amount my friend paid me back is 1.5 times, or 150%, of what I initially lent them:

This is the relative *magnitude* of the two amounts.

Alternatively, we could say that I have *gained* 50%, or 1/2, over the initial amount:

This is relative *change*.

As an example of relative loss rather than gain, assume and . Then is 40% of (relative magnitude) and the relative change (decrease) from 100 to 40 is 60%.

Here are some more examples. **Please write down your answers and be prepared to share them in class.**

1. How many times larger is than ?
2. What is the relative increase from to ?
3. What is the percent increase from to ?
4. What is the percent decrease from to ?
5. What is the percent change from to ?

The answers are hidden below and can be viewed by pressing the “code” button. But please keep in mind that you won’t learn anything by viewing the answers before trying the questions.

# 1. 6 is 3 times larger than 2, or 300% percent larger than 2 (6/2 = 3)  
# 2. 6 is a 200% increase from 2 (6/2 - 1 = 2)  
# 3. 600 is a 200% increase from 200   
# 4. 200 is a 66% decrease from 600  
# 5. 1 is a zero percent increase from 1 (but 100% as large as 1!)

### 8.5.2 A hypothetical example

This section provides a hypothetical example comparing the interpretation of linear regression and log-linear regression. We will work though the example in class, so **please write down your answers to the questions in this section and be prepared to share your responses**.

To get a better idea of how the interpretation of log-linear regression differs from the interpretation of linear regression, let’s consider the following variables:

* is wages (dollars per week)
* is hours worked per week

A linear relationship between and could be written

Let’s further assume that . With this information, you should be able to work out answers to the following two questions pretty easily:

1. For someone who is expected to earn $100/week, what would be their predicted earnings if they worked one hour more per week?
2. What about for someone who was expected to earn $1000/week?

Now let’s consider a log-linear model:

This time we will assume the regression coefficient is . Using this log-linear model, please answer the following questions:

1. For someone who is expected to earn $100/week, what would be their predicted earnings if they worked one hour more per week? To answer this question, use the interpretation of in terms of proportionate change, as outlined in the previous section.
2. What about for someone who was expected to earn $1000/week?

The answers are hidden below. You can use the “code” button to see them, but please try to answer on your own first.

# 1. 120 dollars per week  
# 2. 1020 dollars per week  
# 3. 120 dollars per week  
# 4. 1200 dollars per week

This example shows that, while the linear model represents change in fixed units (e.g., dollars), the log-linear model represents change that is relative to an initial value of . This is particularly useful in applications involving monetary outcomes, where it has been argued that people’s “subjective evaluation” of money is more like the latter than the former.

The contrast between “fixed units” and “subject value” interpretions can be illustrated as follows.

* At $100/week, $20 for an additional hour is equivalent to a 20% increase in wages for only a 2.5% (1/40) increase in hours worked.
* At $1000/week, $20 is a 2% increase in wages, which is less than the 2.5% increase in hours worked.

Thus, the same fixed amount, $20, doesn’t have the same “subject value” for someone making $100/week as compared to someone making $1000/week. On the other hand, a 20% increase for both people might be considered as having equal subjective value. On this logic, the person making $1000/ week would need a $200 increase in order experience the same subject value as the $20 increase provides to the person making $100/ week.

Some real life examples of this kind of interpretation are linked below.

* Sliding scales for parking, car pool / day care late pick-up, bail and fines, <https://www.theatlantic.com/business/archive/2015/03/finland-home-of-the-103000-speeding-ticket/387484/>
* Proportional tuition based on income and number of dependents. <https://ronclarkacademy.com/tuition-and-financial-aid>

### 8.5.3 Exact interpretation

If we want an exact interpretation of the regression coefficients in a log-linear model, we need to do some “follow-up math” with exponents. The main result is about the relative magnitude of corresponding to a one-unit change in

where is the predicted value of for a given value of . This is result derived in [Section 8.5.5](#sec-derivation-8) below (optional).

Based on [Equation 8.2](#eq-expb), we can interpret the exponentiated regression coefficient in a log-linear model in terms of relative magnitude. In practice, people often prefer to use relative change instead. From [Equation 8.2](#eq-expb) and the definition of proportionate change in [Section 8.5.1](#sec-approximate-8), it follows that the proportionate change in for a one-unit increase in is *exactly* equal to .

In the hypothetical example above (wages and hours worked), the exact proportion change in the log-linear model is

or about 22%.

### 8.5.4 Summary

While there are many transformations that can be used to address skew, the log-linear model has the advantage that it is still interpretable in terms of the original variable. In fact, in many applications, notably those in which is monetary, the interpretation is arguably better than for linear regression!

There are two ways to interpret the regression coefficients in the log-linear model.

* The approximate interpretation: a one-unit change in is expected to yield a change in . This interpretation only applies when is small (e.g., less than .5 in absolute value).
* The exact interpretation: a one-unit change in is expected to yield a change in , and this applies for any value of .

### 8.5.5 Derivation of [Equation 8.2](#eq-expb) \*

Start with the log-linear model, exponentiate both sides, and then use the multiplication rule for exponents:

Then do the same thing for in place of :

Finally, take the ratio of to :

This gives [Equation 8.2](#eq-expb).

The reason that the approximate interpretation in [Section 8.5.1](#sec-approximate-8) works is because

for . This is illustrated in the graph below, where the blue line is and black line is . We can see that the two lines are pretty close over the range (-.5, .5).

|  |
| --- |
| Figure 8.11: exp(b) - 1 (blue) versus b (black) |

## 8.6 Worked example

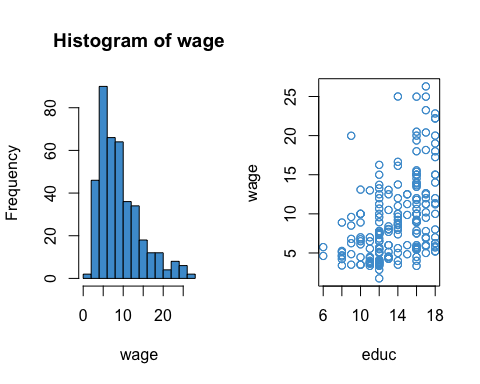
This section works through log-linear regression using a new example, the Wages.Rdata data (source: Weinberg & Abramowitz (2017) Statistics using Stata: An integrative approach. Cambridge University Press). The dataset was collected in 1985 from respondents and contains the following variables.

# Load the data and take a look  
load("Wages.RData")  
knitr::kable(head(wages))

| educ | south | sex | exper | wage | occup | marr | ed |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 12 | 0 | 0 | 17 | 7.50 | 6 | 1 | 2 |
| 13 | 0 | 0 | 9 | 13.07 | 6 | 0 | 3 |
| 10 | 1 | 0 | 27 | 4.45 | 6 | 0 | 1 |
| 9 | 1 | 0 | 30 | 6.25 | 6 | 0 | 1 |
| 9 | 1 | 0 | 29 | 19.98 | 6 | 1 | 1 |
| 12 | 0 | 0 | 37 | 7.30 | 6 | 1 | 2 |

We will focus on regressing hourly wage (wage) on years of education (educ). The distribution of the outcome variable and its relationship with the predictor are depicted below.

attach(wages)  
par(mfrow = c(1,2))  
hist(wage, col = "#4B9CD3")  
plot(educ, wage, col = "#4B9CD3")

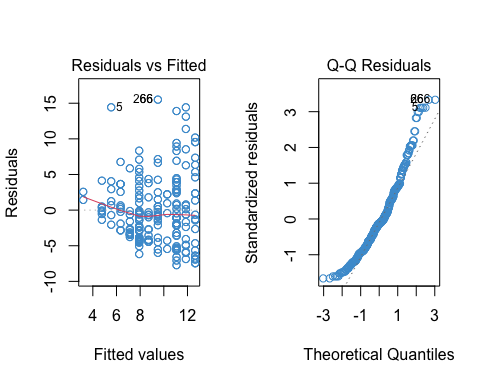


This example is a good candidate for log-linear regression because

* wages is positively skewed
* Its relationship with educ appears somewhat exponential
* The log-transform provides a suitable interpretation when the outcome is monetary

Moreover, a quick a look at the linear regression model (below) shows that *all* of the model assumptions are violated. So, time to try something other than regular regression!

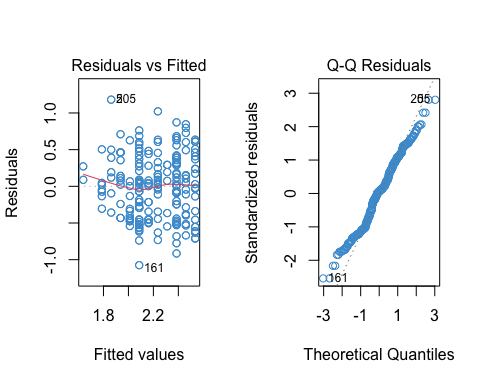
mod1 <- lm(wage ~ educ)  
  
# Check out model fit  
par(mfrow = c(1,2))  
plot(mod1, 1, col = "#4B9CD3")  
plot(mod1, 2, col = "#4B9CD3")



### 8.6.1 Log-linear regression of Wages on Education

To run a log-linear model, we can transform the -variable (wages) using the log function, and then run the model as usual. The diagnostic plots are shown below.

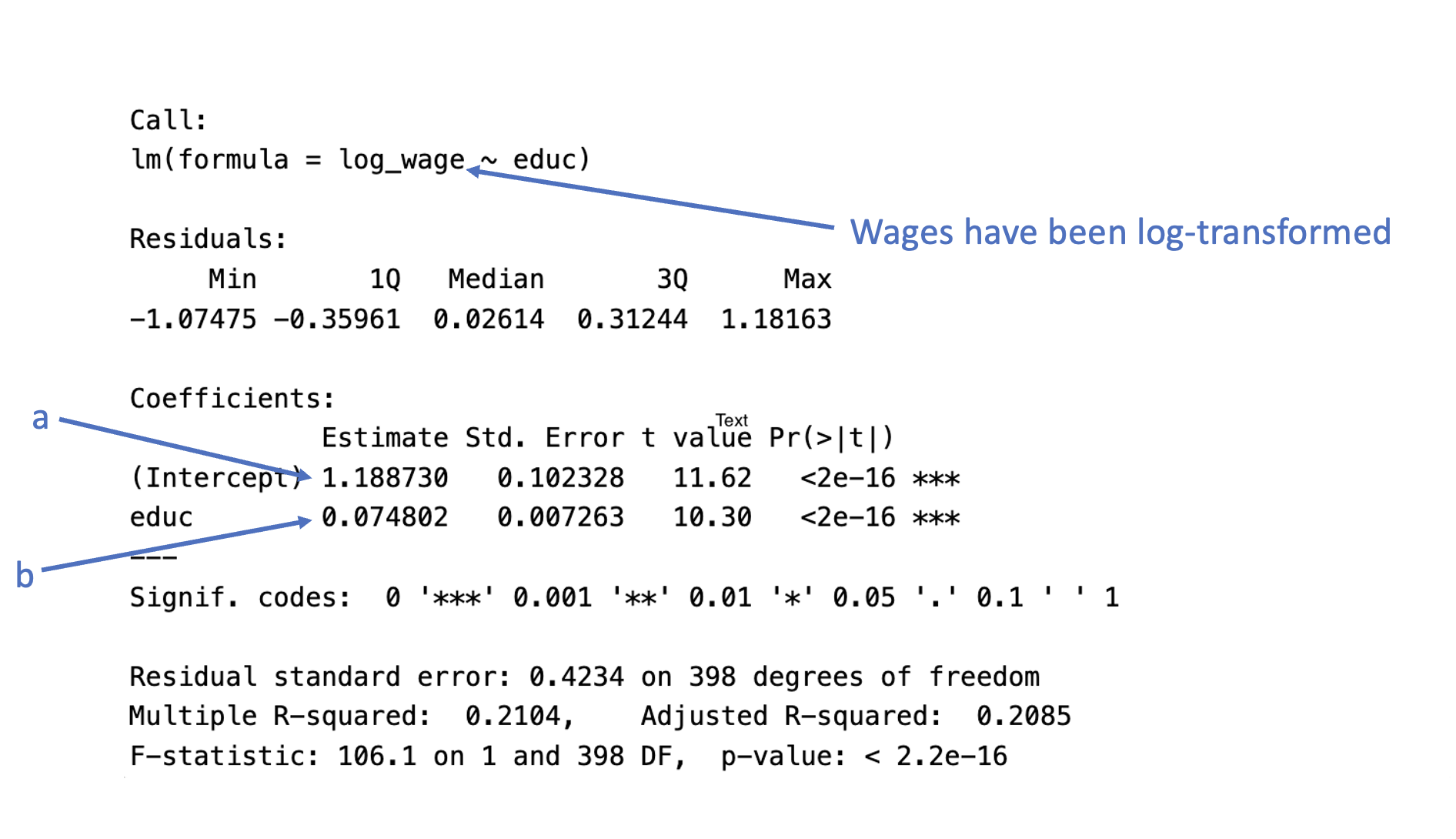
# Create log transform of wage  
log\_wage <- log(wage + 1)  
  
# Regress it on educ  
mod2 <- lm(log\_wage ~ educ)  
  
# Check out model fit  
par(mfrow = c(1,2))  
plot(mod2, 1, col = "#4B9CD3")  
plot(mod2, 2, col = "#4B9CD3")



We can see that the log transformation helped with positive skew, although didn’t do much for the negative tail. It also reduced the apparent heteroskedasticity. However, the relationship still appears non-linear, perhaps more so than in the original linear model. We will address non-linearity in the following chapter, and for now just focus on interpreting the log-linear model in this example.

The model output is shown below.

#summary(mod2)  
knitr::include\_graphics("files/images/log\_linear\_output.png", dpi = 125)



Plugging-in the values from the output using the equation for log-linear regression, we have

**Please take a moment to write down both the approximate and exact interpretation of the regression coefficient for educ. Hint: you need to do some follow-up math with exponents to get the exact interpretation.**

## 8.7 Workbook

This section collects the questions asked in this chapter. The lessons for this chapter will focus on discussing these questions and then working on the exercises in [Section 8.8](#sec-exercises-8). The lesson will **not** be a lecture that reviews all of the material in the chapter! So, if you haven’t written down / thought about the answers to these questions before class, the lesson will not be very useful for you. Please engage with each question by writing down one or more answers, asking clarifying questions about related material, posing follow up questions, etc.

[Section 8.1](#sec-math-review-8)

I’ll ask some questions like the following to start class off. You check your answers by pasting the questions into or Google’s search bar.

* What is larger, or ?

[Section 8.5](#sec-interpretation-8)

* Please write down your answers to the questions below and be prepared to share in clas: s

1. How many times larger is than ?
2. What is the relative increase from to ?
3. What is the percent increase from to ?
4. What is the percent decrease from to ?
5. What is the percent change from to ?

* Please write down your answers to the four questions about the hypothetical example in and be prepared to share your responses in class. \*

[Section 8.6](#sec-worked-example-8)

* Please take a moment to write down your interpretation of the regression coefficient for educ in the summary output below.

attach(wages)  
log\_wage <- log(wage + 1)  
  
# Regress it on educ  
mod2 <- lm(log\_wage ~ educ)  
summary(mod2)

Call:  
lm(formula = log\_wage ~ educ)  
  
Residuals:  
 Min 1Q Median 3Q Max   
-1.07475 -0.35961 0.02614 0.31244 1.18163   
  
Coefficients:  
 Estimate Std. Error t value Pr(>|t|)   
(Intercept) 1.188730 0.102328 11.62 <2e-16 \*\*\*  
educ 0.074802 0.007263 10.30 <2e-16 \*\*\*  
---  
Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
  
Residual standard error: 0.4234 on 398 degrees of freedom  
Multiple R-squared: 0.2104, Adjusted R-squared: 0.2085   
F-statistic: 106.1 on 1 and 398 DF, p-value: < 2.2e-16

detach(wages)

## 8.8 Exercises

There isn’t much new in terms of R code in this chapter. The code for the worked example is presented below, but please refer to [Section 8.6](#sec-worked-example-8) for the interpretation.

We will go through this material in class together, so you don’t need to work on it before class (but you can if you want.)

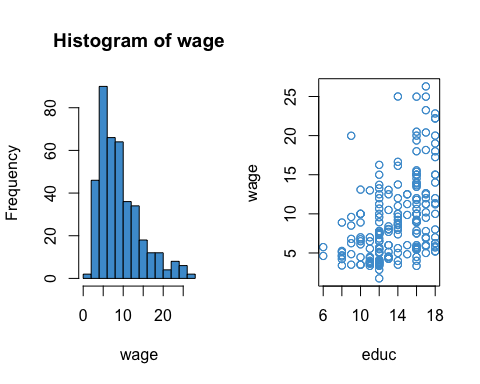
Before staring this section, you may find it useful to scroll to the top of the page, click on the “</> Code” menu, and select “Show All Code.”

# Load the data and take a look  
load("Wages.RData")  
knitr::kable(head(wages))

| educ | south | sex | exper | wage | occup | marr | ed |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 12 | 0 | 0 | 17 | 7.50 | 6 | 1 | 2 |
| 13 | 0 | 0 | 9 | 13.07 | 6 | 0 | 3 |
| 10 | 1 | 0 | 27 | 4.45 | 6 | 0 | 1 |
| 9 | 1 | 0 | 30 | 6.25 | 6 | 0 | 1 |
| 9 | 1 | 0 | 29 | 19.98 | 6 | 1 | 1 |
| 12 | 0 | 0 | 37 | 7.30 | 6 | 1 | 2 |

We will focus on regressing hourly wage (wage) on years of education (educ). The distribution of the outcome variable and its relationship with the predictor are depicted below.

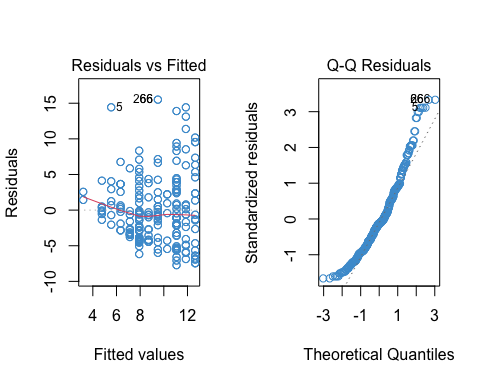
attach(wages)  
par(mfrow = c(1,2))  
hist(wage, col = "#4B9CD3")  
plot(educ, wage, col = "#4B9CD3")



### 8.8.1 Linear regression of Wages on Education

Working with linear model, we can see that *all* of the assumptions of the model are violated – we are going to need to try another approach!

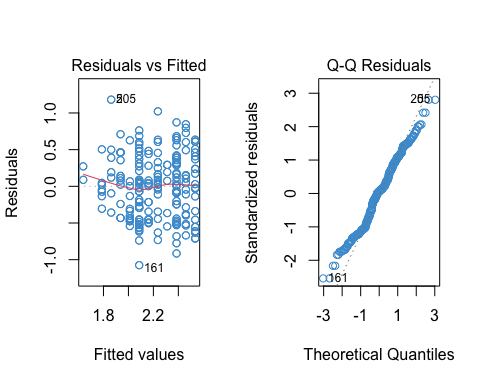
mod1 <- lm(wage ~ educ)  
  
# Check out model fit  
par(mfrow = c(1,2))  
plot(mod1, 1, col = "#4B9CD3")  
plot(mod1, 2, col = "#4B9CD3")



### 8.8.2 Log-linear regression of Wages on Education

To run a log-linear model, we can transform the Y-variable using the log function, and then run the model as usual.

# Create log transform of wage  
log\_wage <- log(wage + 1)  
  
# Regress it on educ  
mod2 <- lm(log\_wage ~ educ)  
  
# Check out model fit  
par(mfrow = c(1,2))  
plot(mod2, 1, col = "#4B9CD3")  
plot(mod2, 2, col = "#4B9CD3")



The log transformation helped with positive skew, although didn’t do much for the negative tail. It reduced the apparent heteroskedasticity. However, the relationship still appears non-linear, perhaps more so than in the original linear model. We will address non-linearity in the following chapter, and for now just focus on interpreting the log-linear model in this example.

The usual summary output provides the regression slope, which can be interpreted using the “approximate” interpretation of relative change:

summary(mod2)

Call:  
lm(formula = log\_wage ~ educ)  
  
Residuals:  
 Min 1Q Median 3Q Max   
-1.07475 -0.35961 0.02614 0.31244 1.18163   
  
Coefficients:  
 Estimate Std. Error t value Pr(>|t|)   
(Intercept) 1.188730 0.102328 11.62 <2e-16 \*\*\*  
educ 0.074802 0.007263 10.30 <2e-16 \*\*\*  
---  
Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
  
Residual standard error: 0.4234 on 398 degrees of freedom  
Multiple R-squared: 0.2104, Adjusted R-squared: 0.2085   
F-statistic: 106.1 on 1 and 398 DF, p-value: < 2.2e-16

Some follow-up math to get the exact interpretation of the regression slope in terms of relative change. First :

# Get the regression slope from the output   
conf.b <- coef(mod2)[2]  
  
# Exp and subtract one to get exact relative change  
exp(conf.b) - 1

educ   
0.07767034

Let’s also compute a confidence interval for :

# Compute a confidence interval for exact relative change  
exp(confint(mod2)[2, ]) - 1

2.5 % 97.5 %   
0.06239300 0.09316737

### 8.8.3 Write up

Approximate:

* Using a log-linear regression model, it was found that each additional year of education led to a predicted increase in wages of 7.4% ().

Exact:

* Using a log-linear regression model, it was found that each additional year of education led to a predicted increase in wages of 7.8% ().

# 9. Polynomial regression, etc

This chapter will cover two widely-used techniques for addressing violations of the assumption of linearity:

* Polynomial regression, which means raising -variables to a power (e.g., and ), and
* Piecewise or segmented regression, which involves using different regression lines over different ranges of a predictor. These techniques involve transforming the -variable(s), which can be done in addition to (or instead of) transforming the variable (see [Chapter 8](#sec-chap-8)).

Both polynomial and piecewise regression are useful in practice and lead to advanced topics like splines and semi-parametric regression. They also turn out to be special cases of interactions, so we have already covered a lot of the technical details in [Chapter 5](#sec-chap-5) – phew!

## 9.1 Polynomial regression

Polynomial regression means that we regress on a polynomial function of :

Your first thought might be, “doesn’t this contradict the assumption that regression is linear?” The answer here is a bit subtle.

As with regular linear regression, the polynomial model is linear in the coefficients – we don’t raise the regression coefficients to a power (e.g., ), or multiply coefficients together (e.g, ). This is the technical sense in which polynomial regression is still just linear regression, despite its name.

Polynomial regression does use nonlinear functions of the predictor(s), but the model is agnostic to what you do with your data. The situation here is a lot like when we worked with interactions in [Chapter 5](#sec-chap-5). In order to model interactions, we computed the product of two predictors and entered the product into the model as a third predictor. Well, is the product of a predictor with itself, so, in this sense, the quadratic term in a polynomial regression is just a special case of an interaction between two variables.

Although we did not cover interactions among more than two variables in this course, they are computed in the same way – e.g., a “three-way” interaction is just the product of 3 predictors. Similarly, is just the three-fold product of a variable with itself.

While polynomial regression is formally similar to interactions, it is used for a different purpose. Interactions address how the relationship between two variables changes as a function of a third. Their inclusion in a model is usually motivated by a specific research question that is formulated before doing the data analysis (see [Chapter 6](#sec-chap-6)).

By contrast, polynomial regression is used to address a non-linear relationship between and , and is usually motivated by a preliminary examination of data that indicates the presence of such a relationship (e.g., a scatter plot of versus ; a residual versus fitted plot). While it is possible to formulate research questions about polynomial terms in a regression model, this is not necessarily or even usually the case when polynomial regression is used – often its just used to address violations of the linearity assumption.

### 9.1.1 Recap of polynomials

In general, a polynomial of degree (i.e., highest power of ) produces a curve that can have up to bends (minima and maxima). Some examples are illustrated in Figure @ref(fig:poly) below.

* The (orange) linear function of is a polynomial of degree 1 and has zero bends.
* The (green) quadratic function of is a polynomial of degree 2 and has 1 bend (a minimum at ; this is also called a parabola).
* etc.

|  |
| --- |
| Examples of Polynomials |

As we can see, this is a very flexible approach to capturing non-linear relationships between two variables. In fact, it can be too flexible! This is the topic of the next section.

### 9.1.2 “Over-fitting” the data

[Figure 9.1](#fig-overfit) shows three different regression models fitted to the same (simulated) bivariate data.

* In the left panel, a standard linear regression model is used, and we can see that the model does not capture the nonlinear (quadratic) trend in the data.
* The middle panel uses a quadratic model (i.e., includes as a predictor, as well as ), and fits the data quite well.
* The right panel uses a 16-degree polynomial to fit the data. We can see that is has a higher R-squared than the quadratic model. But there is also something fishy about this model, don’t you agree?

|  |
| --- |
| Figure 9.1: Polynomial Regression Examples |

To help compare these three models, let’s simulate a second sample from the same population. In the plots below, the regression lines from [Figure 9.1](#fig-overfit) were added to the plots from a second sample. Note that the regression parameters were not re-estimated using the second data set. The model parameters from the first data set were used to produce the regression lines for the second data set, so the regression lines are the same as in [Figure 9.1](#fig-overfit).

|  |
| --- |
| Figure 9.2: Polynomial Regression Examples (With New Data) |

This procedure, which is called out-of-sample-prediction or cross-validation, is one widely used method for comparing the quality of predictions from different models. The R-squared values of the different models in the second sample are provided to help summarize the quality of predictions.

We will talk more about this example in class. **Before moving on, please take a moment to write down your intuitions about what is going in** [**Figure 9.1**](#fig-overfit) **and** [**Figure 9.2**](#fig-overfit2)**. In particular, you might consider whether the model in the right-hand panel is better than the one in the middle panel. I will ask you to share your thoughts in class.**

### 9.1.3 Interpreting the model

As mentioned, polynomial terms are often added into a model as a way to address nonlinearity. When this is the case, the polynomial terms themselves are not necessarily of substantive interest – they can be added just to “patch up” the model after assumption checking.

We saw an example of this in [Section 6.2](#sec-worked-example-6). In that example, linear and quadratic terms for SES were entered into the model in the first block. The R-squared was interpreted for the entire block, but the interpretation of the regression coefficient for the quadratic term was not addressed. This is a pretty common way of using polynomial regression – the polynomial terms are included so that the model assumptions (linearity) are met, but they are not necessarily interpreted beyond this.

However, we *can* interpret the regression slopes on the polynomial terms if we want to. This section addresses the interpretation of quadratic polynomials, which have the equation:

A similar approach applies to models with higher-order terms as well (cubics, etc.).

Let’s start with a classic example of a quadratic relationship: the Yerkes-Dodson law relating physiological arousal (“stress”) to task performance, which is represented in [Figure 9.3](#fig-YD). One way to interpret the law is in terms of the overall shape of the relationship. As stress goes up, so does performance – but only up to a point, after which more stress leads to a deterioration in performance.

|  |
| --- |
| Figure 9.3: Yerkes-Dodson Law (Source: Wikipedia) |

The overall shape of the trend depends on only the *sign* of the regression slope on the quadratic term:

* A U-shaped curve corresponds to a *positive* regression coefficient on (think of a parabola from high school math)
* An inverted-U-shaped curve corresponds to a *negative* regression coefficient on

Beyond the overall shape of the relationship, we might also want to know what level of stress corresponds to the “optimal” level of performance – i.e., where the maximum of the curve is. This exemplifies a more complicated interpretation of a quadratic relationship, and it requires some calculus (see [Section 9.1.4](#sec-deriviation-9), which is optional). The value of that corresponds to the maximum (or minumum) of the quadratic curve is

**Based on this discussion, please answer the following questions bout the Yerkes-Dodson Law, using the following model parameters. Make sure to explain how you know the answer!**

* **What is the overall shape of the relationship: U or inverted-U?**
* **What is the value of stress at which predicted performance reaches a maximum?**
* **Bonus: what is the maximum value of predicted performance?**

### 9.1.4 Analysing polynomials\*

This sections shows how we get [Equation 9.1](#eq-xmax) and some related details. Recall from intro calculus that the extrema (i.e., minima and maxima) of a function occur when the derivative of the function is equal to zero. The derivative of the quadratic regression equation is

Setting the derivative to zero

and solving for

gives the value of at which the reaches its minimum (or maximum) value. Let’s call this value of . Plugging into the original equation tells us the minimum (or maximum) of .

We can use the second derivative rule to determine whether is a minimum or maximum of .

If this value is positive (i.e., if ), then the second derivative rule tells that is minimum, hence the curve is “U-shaped”. If the value is negative (i.e., if ) then its a maximum, and hence the curve is “inverted-U-shaped”.

### 9.1.5 Model building vs curve fitting

Up to this point, we have discussed the use and interpretation of polynomials. In this section we consider how to build polynomial regression models in practice.

A typical model-building process for polynomial regression might proceed as follows.

1. Enter just the linear terms into the model and examine a residual versus fitted plot.
2. If there is evidence of non-linearity, look at the scatter plots between the outcome variable and each individual predictor to make a guess about which predictor(s) may be causing the non-linearity.
3. Add a quadratic term for a predictor of interest and examine whether there is a statistically significant increase in R-squared (see [Section 6.1.4](#sec-delta-rsquared-6)). If there is, you have found a source of non-linearity! If not, the quadratic term is not explaining variance in the outcome variable, so you can remove it from the model.
4. Keep adding polynomial terms (quadratic terms for other predictors; higher-order terms for the same predictor) one at a time until the model assumptions looks reasonable. This might take a bit of trial and error.

This overall approach is illustrated in the next section. However, there are a couple of important points to mention first.

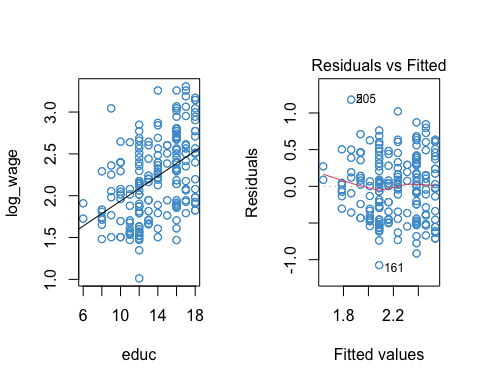
* Making good use of polynomial regression requires walking a fine line between curve-fitting and theory-based modeling (see [Figure 9.1](#fig-overfit)). Sometimes, adding polynomial terms can provide an elegant and intuitive interpretation of the relationship between two variables. But, if you find yourself adding more than a couple of polynomial terms into a model and still have unresolved issues with nonlinearity, it is probably best to consider another approach (such as piecewise regression, coming up in [Section 9.3](#sec-piecewise-9))
* Just like with interactions, higher-order polynomial terms are often highly correlated with lower-order terms (e.g., if takes on strictly positive values, and will be highly correlated). Recall that if two predictors are highly correlated, this can affect their regression coefficients ([Section 3.4](#sec-ols-3)) as well as their standard errors ([Section 6.3](#sec-too-many-predictors-6)). In the context of polynomial regression, there are a couple of things that can be done about this.
  + Interpret values rather than the individual regression coefficients and their -values. This is the easiest thing to do, conceptually.
  + Use “orthogonal polynomials”, which are designed to ensure the different polynomial terms for the same predictor are uncorrelated (orthogonal just means uncorrelated). The result of this approach is that numerical values of the regression coefficients are not directly interpretable beyond their sign, but the t-tests of the regression coefficients can be interpreted as testing the for each term in the polynomial. This is conceptually more complicated than first option, but leads to the same overall conclusions.

Both approaches are illustrated in the next section.

## 9.2 Worked Example

In [Section 8.6](#sec-worked-example-8) we saw that applying a log-transform to the Wages.Rdata example addressed non-normality of the residuals but did not do much to address nonlinearity. The summary output and diagnostic plots for the log-linear regression of wages on education are presented again below. **We will go through this example in class together, so please note any questions you have about the procedures or their interpretation and be prepared to ask them in class.**

# Load the data and take a look  
load("Wages.RData")  
attach(wages)  
  
# Create log transform of wage  
log\_wage <- log(wage + 1)  
  
# Regress log\_wages on educ  
mod1 <- lm(log\_wage ~ educ)  
  
# Check out model fit  
par(mfrow = c(1,2))  
plot(educ, log\_wage, col = "#4B9CD3")  
abline(mod1)  
plot(mod1, 1, col = "#4B9CD3")



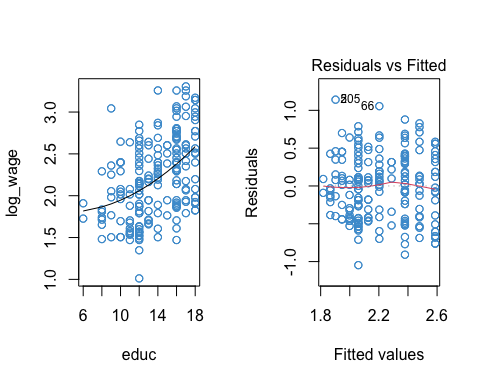
summary(mod1)

Call:  
lm(formula = log\_wage ~ educ)  
  
Residuals:  
 Min 1Q Median 3Q Max   
-1.07475 -0.35961 0.02614 0.31244 1.18163   
  
Coefficients:  
 Estimate Std. Error t value Pr(>|t|)   
(Intercept) 1.188730 0.102328 11.62 <2e-16 \*\*\*  
educ 0.074802 0.007263 10.30 <2e-16 \*\*\*  
---  
Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
  
Residual standard error: 0.4234 on 398 degrees of freedom  
Multiple R-squared: 0.2104, Adjusted R-squared: 0.2085   
F-statistic: 106.1 on 1 and 398 DF, p-value: < 2.2e-16

Because there is one prominent bend in our residual vs fitted plot (at ), let’s see if adding a quadratic term to the model can improve the model fit.

The poly function in R makes it easy to do polynomial regression, without having to hard-code new variables like educ^2 into our dataset. In the summary output below, poly(...)n denote’s the -th term in the polynomial. The diagnostic plots for the log-linear model with a quadratic term are also shown below.

mod2 <- lm(log\_wage ~ poly(educ, 2, raw = T))  
par(mfrow = c(1,2))  
plot(educ, log\_wage, col = "#4B9CD3")  
  
# To plot the trend we need to we first need to order the data and the predicted values ...   
sort\_educ <- educ[order(educ)]  
sort\_fitted<- fitted(mod2)[order(educ)]  
points(sort\_educ, sort\_fitted, type = "l")  
plot(mod2, 1, col = "#4B9CD3")



summary(mod2)

Call:  
lm(formula = log\_wage ~ poly(educ, 2, raw = T))  
  
Residuals:  
 Min 1Q Median 3Q Max   
-1.04723 -0.38939 0.01877 0.31820 1.14129   
  
Coefficients:  
 Estimate Std. Error t value Pr(>|t|)   
(Intercept) 1.862958 0.406587 4.582 6.18e-06 \*\*\*  
poly(educ, 2, raw = T)1 -0.031492 0.062468 -0.504 0.6144   
poly(educ, 2, raw = T)2 0.003985 0.002326 1.713 0.0875 .   
---  
Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
  
Residual standard error: 0.4224 on 397 degrees of freedom  
Multiple R-squared: 0.2162, Adjusted R-squared: 0.2123   
F-statistic: 54.77 on 2 and 397 DF, p-value: < 2.2e-16

Writing the R output in terms of the regression model, we have:

Let’s start by interpreting the plots. Based on the left-hand panel, it looks like a quadratic relationship provides a reasonable representation of the data. Based on the right-hand panel, I would conclude that the apparent non-linearity in the residual vs fitted plot has been sufficiently reduced. There is still a blip at , but there are 5 data points there so I am not to worried about it.

Turning to the summary output, there are three main take-aways:

1. As discussed in the previous section, the sign of the quadratic term tells us something about the overall shape of the relationship (do you remember what that is?). However, interpreting the numerical values of the regression coefficients in a polynomial regression is not always useful. For example, using the approach to interpreting quadratic regression from [Section 9.1.3](#sec-interpreting-the-model-9), it turns out that the minimum predicted wages occur for someone with 3.95 years of education. The lowest level of education in the sample is 6 years, so this interpretation isn’t super relevant for our example. Consequently, rather than focusing on the interpretation of the regression coefficients, it is often sufficient to focus on whether the two predictors (i.e., and ) together explained a significant proportion of variation in the outcome variable. This information is provided by the R-squared statistic and the F-test of R-squared in the summary output above.
2. There are two ways to test whether the addition of the quadratic term (poly(educ, 2)2 = 0.003985) improves the model. First, we can examine its test of significance. This test tells us that, controlling for the linear relationship between log-wages and education, the quadratic term is statistically significant at the .1 level (it is not statistically significant at the .05 level). Recall from [Chapter 7](#sec-chap-7) that this same information could be obtained by setting up a heirarhical model (Block 1 = linear term; Block 2 = quadratic term) and testing the change in R-squared. For the example, the F-test of R-squared change is

anova(mod1, mod2)

Analysis of Variance Table  
  
Model 1: log\_wage ~ educ  
Model 2: log\_wage ~ poly(educ, 2, raw = T)  
 Res.Df RSS Df Sum of Sq F Pr(>F)   
1 398 71.363   
2 397 70.840 1 0.52368 2.9348 0.08747 .  
---  
Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

Note that the p-value is exactly the same as the t-test of regression coefficient reported above. In both cases, it seems that we don’t really need the quadratic term, based on the .05 level of significance (more on this [Section 9.3](#sec-piecewise-9))

1. Finally, compared to the model without the quadratic term, we can see the linear term (poly(educ, 2)1 = - 0.031492 ) is now longer statistically significant. This can happen when we add higher order terms into a model. In this example, the linear and quadratic terms are highly correlated (the correlation is over .99 in the example!). Due to this correlation, the linear terms is not statistically significant, and the quadratic term is only “marginally” significant, even though the F-test of R-squared in the summary output is fatalistically significant with . In the next section, we will see how to avoid this issue of having highly correlated predictors in polynomial regression.

\*\* If you have any questions about the interpretation of the model results discussed in this section, please list them now and I will be happy to address the in class.\*\*

### 9.2.1 Orthogonal polynomails\*

Before moving on, a quick (and optional) note on orthogonal vs. raw polynomials. Orthogonal polynomials are the default approach in R, and they make life easier, so they are worth knowing about.

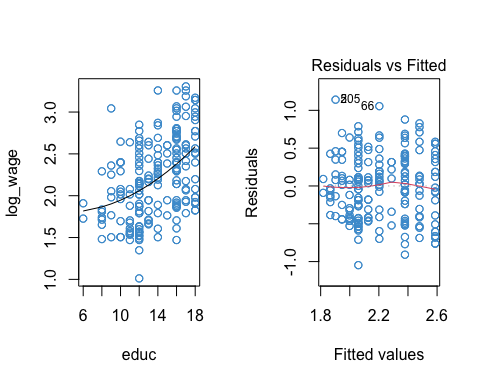
When using orthogonal polynomials, the different polynomial terms (e.g., ) are transformed so that they are uncorrelated. This means that the -test of each regression coefficient can be interpreted as testing the proportion of variance associated uniquely with that term of the polynomial. Basically, using orthogonal polynomials means that we don’t need to do the model building stuff (e.g., sequential blocks, adding in each term one at a time)– it’s already built into the coefficients.

The downside of orthogonal polynomials is that, beyond their sign, the regression coefficients are complicated to interpret. But, these coefficients aren’t easy to interpret anyway, and we often don’t care much about their exact values. If you are in a situation where you don’t really care about the interpretation of the coefficients beyond the overall shape of the relationship, the orthogonal polynomials are definitely a good choice!

The use of orthogonal polynomials is illustrated below. You’ll see that most of the output is the same as in the previous section, except the numerical value and associated tests of the regression coefficients in the summary table. In particular, the linear trend is statistically significant in the output below, because it is no longer correlated with the quadratic trend. In fact the t-test and p-value are exactly the same as the first model we fit to our example data above (the model with just the linear trend, and no quadratic trend).

In summary, orthogonal polynomials provide a shortcut to hierarchical model buiding with polynomials. By transforming the data so that the different terms of the polynomial are uncorrelated, we get the similar information from a single model using orthogonal polynomials as we would if we fitted a series of hierarchical models, adding each additional term into the model one at a time.

mod2a <- lm(log\_wage ~ poly(educ, 2, raw = F))  
par(mfrow = c(1,2))  
plot(educ, log\_wage, col = "#4B9CD3")  
  
# To plot the trend we need to we first need to order the data and the predicted values ...   
sort\_educ <- educ[order(educ)]  
sort\_fitted <- fitted(mod2a)[order(educ)]  
points(sort\_educ, sort\_fitted, type = "l")  
plot(mod2a, 1, col = "#4B9CD3")



summary(mod2a)

Call:  
lm(formula = log\_wage ~ poly(educ, 2, raw = F))  
  
Residuals:  
 Min 1Q Median 3Q Max   
-1.04723 -0.38939 0.01877 0.31820 1.14129   
  
Coefficients:  
 Estimate Std. Error t value Pr(>|t|)   
(Intercept) 2.21987 0.02112 105.103 <2e-16 \*\*\*  
poly(educ, 2, raw = F)1 4.36133 0.42242 10.325 <2e-16 \*\*\*  
poly(educ, 2, raw = F)2 0.72366 0.42242 1.713 0.0875 .   
---  
Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
  
Residual standard error: 0.4224 on 397 degrees of freedom  
Multiple R-squared: 0.2162, Adjusted R-squared: 0.2123   
F-statistic: 54.77 on 2 and 397 DF, p-value: < 2.2e-16

To find out more, use help(poly). A good discussion of this point is also available on StatExchange: <https://stats.stackexchange.com/questions/258307/raw-or-orthogonal-polynomial-regression?r=SearchResults&s=2%7C87.5473>

### 9.2.2 Summary

This section has addressed how to use, interpret, and implement polynomial regression. Some key points:

* We don’t want to overfit the data by adding too many higher-order terms. If a quadratic or cubic polynomial doesn’t sort out any issues with linearity (as diagnosed by the residual vs fitted plots), then you probably want to try something else (see next section).
* Often the overall shape of a polynomial regression is of interest. This is communicated by the sign of the regression slopes on the higher-order terms. However we aren’t interested in a more specific interpretation of the regression slopes – it can be done (#sec-interpreting-the-model-9), but it is not very common.
* Instead, we usually approach polynomial regression from the perspective of hierarchical model building – if the higher order terms lead to a significant increase in the variance explained (i.e., R-squared change), we keep them in the model.
* Orthogonal polynomials provide a shortcut to doing heirarhical model building with polynomials. The make our life easier, but they aren’t doing anything different than the hierarchical.
* In the worked example, it turned out that despite the apparent issue with linearity in the original model, and despite the apparently better fit of the quadratic model in terms of linearity, the statistical tests actually suggested we don’t need the quadratic term (using ). The next example provides another perspective on this modelling issue.

## 9.3 Piecewise regression

Piecewise or segmented regression is another approach to dealing with nonlinearity. Like polynomial regression, it is mathematically similar to interaction. Also like polynomial regression, it has a special interpretation and application that make it practically distinct from interaction.

In the simplest case, piecewise regression involves interacting a predictor variable with a binary re-coding of itself. To illustrate how the approach works, let’s again consider our wages and education example. The scatter plot of log-wages versus education is presented again below for reference.

|  |
| --- |
| Figure 9.4: The Wages Example |

Consider the following reasoning about the example:

* For people with 12 or less years of education (i.e., who did not obtain post-secondary education) the apparent relationship with wage is quite weak. This seems plausible, because if a job doesn’t require a college degree, education probably isn’t a big factor in determining wages.
* For people with more than 12 years of education, the relationship with wage seems to be stronger. This also seems plausible: for jobs that require post secondary education, more education is usually associated with higher wages.
* To restate this as an interaction: the relationship between wage and education appears different for people who have a post-secondary education versus those who do not.

To represent this reasoning visually we can modify [Figure 9.4](#fig-piecewise1) as shown in [Figure 9.5](#fig-piecewise2). This captures the basic idea behind piecewise regression – we have different regression lines over different ranges of the predictor, and the overall regression is piecewise or segmented. The next section shows how to build this model.

|  |
| --- |
| Figure 9.5: The wages example |

### 9.3.1 The piecewise model

We have reasoned that the relationship between wages and education might depend on whether people have post-secondary education. We also noted that this sounds a lot like an interaction (because it is!), which is the basic approach we can use to create piecewise models.

In order to run our piecewise regression, first we need to create a dummy-coded version of education that indicates whether a person had more than 12 years education:

Then, we enter the original variable, the dummy-coded indicator, and their interaction into the model:

As we can see, the resulting model is a special case of an interaction between a continuous predictor () and binary predictor ().

While the above model conveys the overall idea of piecewise regression, there are also more complex approaches that will search for breakpoints, smoothly connect the lines at the breakpoints, use nonlinear functions (e.g., polynomials) for the segments, etc. We won’t cover these more complex approaches here, but check out the following resource if you are interested and feel free to ask questions in class: <https://rpubs.com/MarkusLoew/12164>

### 9.3.2 Back to the example

The output for the example is provided below. Following the output, some questions are posed about the interpretation of the model.

* Diagnostic plots for the piecewise model:

|  |
| --- |
| Figure 9.6: The Wages Example |

* Summary output and estimated model (the model doesn’t fit nicely on one line!):

summary(mod4)

Call:  
lm(formula = log\_wage ~ educ \* educ12)  
  
Residuals:  
 Min 1Q Median 3Q Max   
-0.98101 -0.36398 0.02055 0.30687 1.10305   
  
Coefficients:  
 Estimate Std. Error t value Pr(>|t|)   
(Intercept) 1.78426 0.23531 7.583 2.43e-13 \*\*\*  
educ 0.01736 0.02159 0.804 0.4219   
educ12 -0.64255 0.35573 -1.806 0.0716 .   
educ:educ12 0.06144 0.02737 2.245 0.0253 \*   
---  
Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
  
Residual standard error: 0.4203 on 396 degrees of freedom  
Multiple R-squared: 0.2261, Adjusted R-squared: 0.2202   
F-statistic: 38.56 on 3 and 396 DF, p-value: < 2.2e-16

* Simple trends using of the emtrends function (see [Section 5.4](#sec-inference-for-interactions-5)).

emmeans::emtrends(mod4, specs = "educ12", var = "educ")

educ12 educ.trend SE df lower.CL upper.CL  
 0 0.0174 0.0216 396 -0.0251 0.0598  
 1 0.0788 0.0168 396 0.0457 0.1119  
  
Confidence level used: 0.95

We will discuss this model together in class. It should feel a lot like deja vu from [Chapter 5](#sec-chap-5), but even more complicated due to the interpretation of and the fact that the outcome is log-transformed. Fun!!! I’ll ask questions below in class:

Note that the intercept and main effect of the binary variable educ12 are not of much interest in this application.

* **Using the 2-step approach from** [**Section 5.3**](#sec-binary-continuous-interaction-5)**, please take a moment to work out the interpretation of main effect on and the interaction in the model above. It might help to draw a plot like** [**Figure 9.5**](#fig-piecewise2) **and label it accordingly. (The interpretation of and is not very interesting but you can work them out too if you like.)**
* **Please take a moment to write down your interpretation of the results of the piecewise regression, addressing the diagnostic plots, the parameter estimates, and the simple slopes.**

### 9.3.3 Summary

Piecewise regression is another approach to dealing with nonlinearity. It can be especially powerful when we can conceptualize the nonlinearity in as an interaction between a variable and categorical encoding of itself (e.g., the relationship between years of education and wages depends on whether you went to college.) The overall interpretation and implementation of the model is also based on the material we already covered in [Section 5.3](#sec-binary-continuous-interaction-5), so take a look at the summary of that section for additional pointers.

## 9.4 Workbook

This section collects the questions asked in this chapter. The lessons for this chapter will focus on discussing these questions and then working on the exercises in [Section 9.5](#sec-exercises-9). The lesson will **not** be a lecture that reviews all of the material in the chapter! So, if you haven’t written down / thought about the answers to these questions before class, the lesson will not be very useful for you. Please engage with each question by writing down one or more answers, asking clarifying questions about related material, posing follow up questions, etc.

[Section 9.1.2](#sec-overfitting-9)

* Please take a moment to write down your intuitions about what is going in [Figure 9.1](#fig-overfit) and [Figure 9.2](#fig-overfit2). In particular, you might consider whether the model in the right-hand panel is better than the one in the middle panel. I will ask you to share your thoughts in class.

[Section 9.1.3](#sec-interpreting-the-model-9)

* Please answer the following questions bout the Yerkes-Dodson Law, using the following model parameters. Make sure to explain how you know the answer.
* What is the overall shape of the relationship: U or inverted-U?
* What is the value of stress at which predicted performance reaches a maximum?
* Bonus: what is the maximum value of predicted performance?

[Section 9.2](#sec-worked-example-9)

* We will go through the example in class together, so please note any questions you have about the procedures or their interpretation and be prepared to ask them in class.

[Section 9.3](#sec-piecewise-9)

* The output for the example is provided below. Following the output, some questions are posed about the interpretation of the model,
* Diagnostic plots for the piecewise model:

|  |
| --- |
| The Wages Example |

* Summary output and estimated model:

summary(mod4)

Call:  
lm(formula = log\_wage ~ educ \* educ12)  
  
Residuals:  
 Min 1Q Median 3Q Max   
-0.98101 -0.36398 0.02055 0.30687 1.10305   
  
Coefficients:  
 Estimate Std. Error t value Pr(>|t|)   
(Intercept) 1.78426 0.23531 7.583 2.43e-13 \*\*\*  
educ 0.01736 0.02159 0.804 0.4219   
educ12 -0.64255 0.35573 -1.806 0.0716 .   
educ:educ12 0.06144 0.02737 2.245 0.0253 \*   
---  
Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
  
Residual standard error: 0.4203 on 396 degrees of freedom  
Multiple R-squared: 0.2261, Adjusted R-squared: 0.2202   
F-statistic: 38.56 on 3 and 396 DF, p-value: < 2.2e-16

* Simple trends using of the emtrends function (see [Section 5.4](#sec-inference-for-interactions-5)).

emmeans::emtrends(mod4, specs = "educ12", var = "educ")

educ12 educ.trend SE df lower.CL upper.CL  
 0 0.0174 0.0216 396 -0.0251 0.0598  
 1 0.0788 0.0168 396 0.0457 0.1119  
  
Confidence level used: 0.95

* Using the 2-step approach from [Section 5.3](#sec-binary-continuous-interaction-5), please take a moment to work out the interpretation of main effect on and the interaction in the model above. It might help to draw a plot like [Figure 9.5](#fig-piecewise2) and label it accordingly. (The interpretation of and is not very interesting but you can work them out too if you like.)
* Please take a moment to write down your interpretation of the results of the piecewise regression, addressing the diagnostic plots, the parameter estimates, and the simple slopes.

# clean up!  
detach(wages)

## 9.5 Exercises

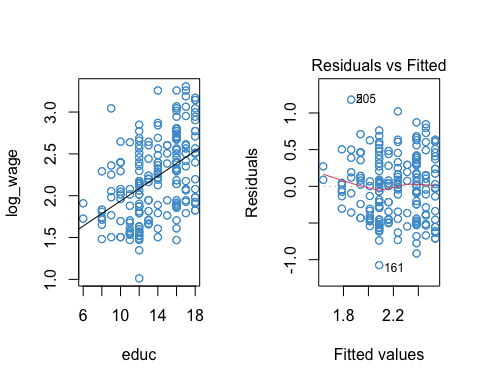
There isn’t much new in terms of R code in this chapter, but the workflows for the two types of model are pretty complicated so we review them here. You’ll see that some of the plots require a lot of fiddling about, especially for the piecewise regression model. We will cover some tricks and shortcuts for producing these types plots during the open lab sessions for Assignment 4. So don’t worry too much about the complicated-looking coded for the plots at this point!

We will go through this material in class together, so you don’t need to work on it before class (but you can if you want.) Before staring this section, you may find it useful to scroll to the top of the page, click on the “</> Code” menu, and select “Show All Code.”

### 9.5.1 Polynomial regression

Let’s start with the “Vanella” log-linear model for the wages examples

# Load the data and take a look  
load("Wages.RData")  
attach(wages)  
  
# Create log transform of wage  
log\_wage <- log(wage + 1)  
  
# Regress it on educ  
mod1 <- lm(log\_wage ~ educ)  
  
# Check out model fit  
par(mfrow = c(1,2))  
plot(educ, log\_wage, col = "#4B9CD3")  
abline(mod1)  
plot(mod1, 1, col = "#4B9CD3")



Because there is one prominent bend in our residual vs fitted plot of the log-linear model (at ), let’s see if adding a quadratic term to the model can improve the model fit.

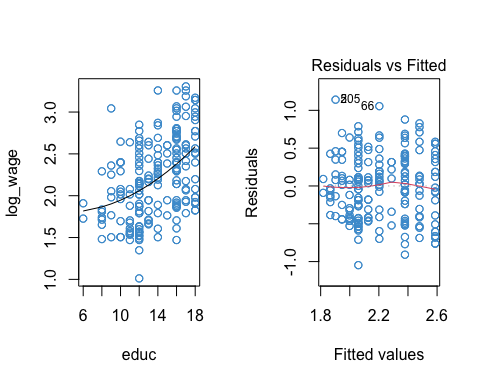
The poly function in R makes it easy to do polynomial regression, without having to hard-code new variables like EDUC^2 into our dataset. This function automatically uses orthogonal (uncorrelated) polynomials, so we don’t have to worry about centering, either. The basic interpretation of the model coefficients in an orthogonal polynomial regression is the same as discussed in #sec-polynomial-9, but the “more complicated” interpretation of the model parameters is not straightforward. To find out more, use help(poly).

The diagnostic plots for the log-linear model with a quadratic term included for education is shown below, along with the model summary. In the output, poly(educ, 2)n is the -th degree term in the polynomial. Orthogonal polynomials were used.

# Regress log\_wage on a quadratic function of eduction   
mod2 <- lm(log\_wage ~ poly(educ, 2))  
  
# Model output  
summary(mod2)

Call:  
lm(formula = log\_wage ~ poly(educ, 2))  
  
Residuals:  
 Min 1Q Median 3Q Max   
-1.04723 -0.38939 0.01877 0.31820 1.14129   
  
Coefficients:  
 Estimate Std. Error t value Pr(>|t|)   
(Intercept) 2.21987 0.02112 105.103 <2e-16 \*\*\*  
poly(educ, 2)1 4.36133 0.42242 10.325 <2e-16 \*\*\*  
poly(educ, 2)2 0.72366 0.42242 1.713 0.0875 .   
---  
Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
  
Residual standard error: 0.4224 on 397 degrees of freedom  
Multiple R-squared: 0.2162, Adjusted R-squared: 0.2123   
F-statistic: 54.77 on 2 and 397 DF, p-value: < 2.2e-16

# Diagnostic plots  
par(mfrow = c(1,2))  
  
# scatter plot with trend  
plot(educ, log\_wage, col = "#4B9CD3")  
# order the data and the predicted values ...   
sort\_educ <- educ[order(educ)]  
sort\_fitted <- fitted(mod2)[order(educ)]  
points(sort\_educ, sort\_fitted, type = "l")  
  
# residual versus fitted  
plot(mod2, 1, col = "#4B9CD3")



The F-test of R-squared change between the first and second models (not really required since we used orhtogonal polynomials, but for illustrative purposes):

anova(mod1, mod2)

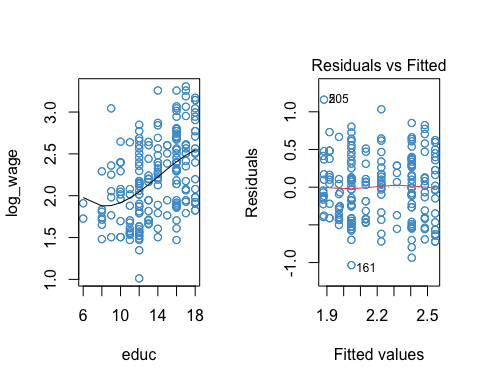
Analysis of Variance Table  
  
Model 1: log\_wage ~ educ  
Model 2: log\_wage ~ poly(educ, 2)  
 Res.Df RSS Df Sum of Sq F Pr(>F)   
1 398 71.363   
2 397 70.840 1 0.52368 2.9348 0.08747 .  
---  
Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

Illustrating the same overall approach for the cubic model:

mod3 <- lm(log\_wage ~ poly(educ, 3))  
summary(mod3)

Call:  
lm(formula = log\_wage ~ poly(educ, 3))  
  
Residuals:  
 Min 1Q Median 3Q Max   
-1.0344 -0.3734 0.0203 0.3107 1.1612   
  
Coefficients:  
 Estimate Std. Error t value Pr(>|t|)   
(Intercept) 2.2199 0.0211 105.213 <2e-16 \*\*\*  
poly(educ, 3)1 4.3613 0.4220 10.336 <2e-16 \*\*\*  
poly(educ, 3)2 0.7237 0.4220 1.715 0.0871 .   
poly(educ, 3)3 -0.5717 0.4220 -1.355 0.1763   
---  
Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
  
Residual standard error: 0.422 on 396 degrees of freedom  
Multiple R-squared: 0.2199, Adjusted R-squared: 0.2139   
F-statistic: 37.2 on 3 and 396 DF, p-value: < 2.2e-16

# Same plots as above, reusing variable names here  
par(mfrow = c(1,2))  
plot(educ, log\_wage, col = "#4B9CD3")  
sort\_fitted <- fitted(mod3)[order(educ)]  
points(sort\_educ, sort\_fitted, type = "l")  
plot(mod3, 1, col = "#4B9CD3")



anova(mod1, mod2, mod3)

Analysis of Variance Table  
  
Model 1: log\_wage ~ educ  
Model 2: log\_wage ~ poly(educ, 2)  
Model 3: log\_wage ~ poly(educ, 3)  
 Res.Df RSS Df Sum of Sq F Pr(>F)   
1 398 71.363   
2 397 70.840 1 0.52368 2.9410 0.08714 .  
3 396 70.513 1 0.32682 1.8354 0.17626   
---  
Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

### 9.5.2 Write up

This section illustrates the write-up for the quadratic regression based on heirarhical modeling and based on orthogonal polynomials. In practice, just pick one of these.

* *Based on heirachical modeling.* Starting with a model containing only the linear term, years of education explained about 21.6% of the variance in log-wages (). Adding a quadratic term to address potential nonlinearity, it was found that only an additional .5% of variance was explained ().
* *Based on orthogonal polynomials* Regressing log wages on a second-degree orthogonal polynomial function of wages, it was found that the linear term was statistically significant at the .05 level (), but the quadratic term was not ().

### 9.5.3 Piecewise regression

Moving on, let’s consider the piecewise model from [Section 9.3](#sec-piecewise-9)

# Create a dummy variable indicating if education is at least 12 years or more  
educ12 <- (educ > 12)\*1  
  
# Interact the dummy with educ  
mod4 <- lm(log(wage + 1) ~ educ\*educ12)   
  
# The model output  
summary(mod4)

Call:  
lm(formula = log(wage + 1) ~ educ \* educ12)  
  
Residuals:  
 Min 1Q Median 3Q Max   
-0.98101 -0.36398 0.02055 0.30687 1.10305   
  
Coefficients:  
 Estimate Std. Error t value Pr(>|t|)   
(Intercept) 1.78426 0.23531 7.583 2.43e-13 \*\*\*  
educ 0.01736 0.02159 0.804 0.4219   
educ12 -0.64255 0.35573 -1.806 0.0716 .   
educ:educ12 0.06144 0.02737 2.245 0.0253 \*   
---  
Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
  
Residual standard error: 0.4203 on 396 degrees of freedom  
Multiple R-squared: 0.2261, Adjusted R-squared: 0.2202   
F-statistic: 38.56 on 3 and 396 DF, p-value: < 2.2e-16

# The simple trends  
trends <- emmeans::emtrends(mod4, specs = "educ12", var = "educ")  
emmeans::test(trends)

educ12 educ.trend SE df t.ratio p.value  
 0 0.0174 0.0216 396 0.804 0.4219  
 1 0.0788 0.0168 396 4.686 <.0001

# The diagnostic plots  
par(mfrow = c(1, 2))  
plot(mod4, 1, col = "#4B9CD3")  
plot(mod4, 2, col = "#4B9CD3")

|  |
| --- |
| The wages example |

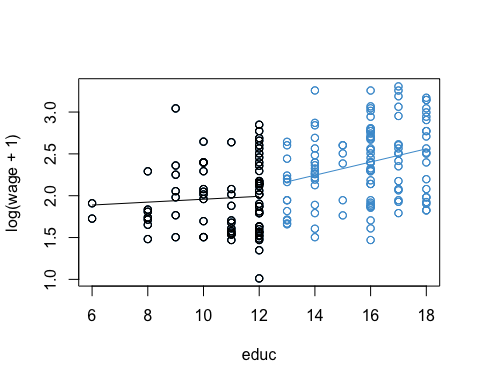
We can still see some evidence of heteroskedasticity in the residual versus fitted plot, so the last step is to use heteroskedasticity-corrected standard errors to ensure we are making the right conclusions about statistical significance

## Make sure the required pacakges are installed  
# install.packages("car")  
# install.packages("lmtest")  
  
# Step 1. Use "hccm" to get the HC SEs for our piecewise model   
hcse <- car::hccm(mod4)  
  
# Step 2. Use "coeftest" to compute t-tests with the HC SEs  
lmtest::coeftest(mod4, hcse)

t test of coefficients:  
  
 Estimate Std. Error t value Pr(>|t|)   
(Intercept) 1.784262 0.188861 9.4475 < 2e-16 \*\*\*  
educ 0.017362 0.017859 0.9722 0.33156   
educ12 -0.642555 0.323851 -1.9841 0.04793 \*   
educ:educ12 0.061436 0.024565 2.5010 0.01279 \*   
---  
Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

The next bit is optional. It shows how to produce the piecewise regression plot, which takes quite a bit of messing about with R…Let me know if you find an easier way to do this (in base R).

# Building the piecewise regression plot -- yeeesh  
  
# Add fitted values to dataset  
wages$fitted <- fitted(mod4)  
  
# Sort data on educ  
wages <- wages[order(educ), ]  
  
# Plot  
par(mfrow = c(1, 1))  
plot(educ, log(wage + 1), col = "#4B9CD3")  
  
# Change color for the points with educ ≤ 12  
with(subset(wages, educ <= 12), points(educ, log(wage + 1)))  
  
# Plot the predicted values for educ > 12  
with(subset(wages, educ > 12), lines(educ, fitted, col = "#4B9CD3"))  
  
# Plot the predicted values for educ ≤ 12  
with(subset(wages, educ <= 12), lines(educ, fitted))



### 9.5.4 Write up

Using a piecewise regression model with heteroskedasticity consistent standard errors, it was found that the relationship between log-wages and years of education depended on whether a person attended post-secondary education (). Using simple trends, it was found for people with 12 or less years of education, the relationship between log-wages and years of education was not significant (), whereas the relationship was significant for people with at least some post-secondary education (). The results indicate that, simply for those with some post-secondary education, each additional year of education was associated with a 8.1% increase in hourly wages (). Due to a software limitation, the simple trends did not utlize heteroskedastic consistent standard errors.

# 10. Logistic regression

The topic we address in this chapter is logistic regression. Like the previous two chapters, logistic regression is an extension of multiple linear regression to situations in which the “standard” model does not directly apply. This time, the extension is to binary *outcome* variables.

There are many situations in which the outcome of interest can be thought of as a binary or “yes / no” or “true / false” outcome:

* Death (the original logistic outcome; [Berkson, 1944](https://doi.org/10.2307%2F2280041))
* Onset of disease or condition
* Employment status
* College enrollment
* Passing a course or completing a credential
* Provide a correct response to a test question
* …

Just like with linear regression, we often we want to relate these types of binary variables to predictors, such as medical history, family background, personal characteristics, etc. That is what logistic regression does.

The main theme of this chapter is also an extension of the previous two chapters. We have seen a general strategy for how to deal with data that do not “fit” the assumptions of linear regression:

1. Transform one or more variables so that the data *do* fit the assumptions linear regression.
2. Run the analysis as usual.
3. Then work out the interpretation of results in terms of the original variable(s).

We will see this basic approach again in this chapter. In fact, its kind of a general-purpose hack for quantitative research – when you are faced with a problem you don’t know how to deal with, turn it into something you do know how to deal with. Certainly this is not the most creative approach, but it has the advantage of letting us “port over” many of the tools we have developed in one context (regression with a continuous outcome) into a new context (regression with a binary outcome).

In terms of statistical modeling, the move to binary outcome variables is a pretty big deal. Everything we have done up until now has focused on OLS regression, using the same basic principles we discussed in Chapters [Chapter 2](#sec-chap-2) and [Chapter 3](#sec-chap-3). However, logistic regression takes us into the wider framework of *generalized linear models* (GLMs), which are estimated using maximum likelihood (ML) rather than OLS. Thus we will need to start at the “ground floor” to build up our knowledge of logistic regression, which then provides a stepping stone to GLMs, which can additionally handle other types of outcome variables (e.g., count data, ordered categorical data).

Since we are starting with a new modeling approach, let’s kick things off with a new example.

## 10.1 The CHD example

As a working example, we will use data contained in the file CHD.RData to explore the relationship between age in years (“age”) and evidence (absence or presence) of coronary heart disease (“chd”). The data set contains 100 cases. Respondents’ ages range from 20 to 69, while clinical evidence of CHD is coded 0 when it is absent and 1 when it is present. A sample of 20 cases is shown below. (Source: Applied Logistic Regression by David W. Hosmer and Stanley Lemeshow, 1989, John Wiley and Sons.)

For the sake of brevity, I will say that a person either “has CHD” or not. This is less accurate (but much simpler) than saying that the variable denotes clinical evidence about the presence or absence of CHD.

load("CHD.RData")  
attach(chd.data)  
knitr::kable(list(chd.data[sample(1:100, 10),2:3],   
 chd.data[sample(1:100, 10),2:3]),   
 row.names = F,   
 caption = "The CHD example")

The CHD example

| chd | age |
| --- | --- |
| 0 | 49 |
| 1 | 39 |
| 1 | 34 |
| 0 | 57 |
| 1 | 53 |
| 0 | 30 |
| 1 | 55 |
| 1 | 57 |
| 0 | 41 |
| 0 | 45 |

| chd | age |
| --- | --- |
| 1 | 59 |
| 0 | 33 |
| 1 | 56 |
| 0 | 46 |
| 1 | 57 |
| 0 | 47 |
| 0 | 23 |
| 0 | 26 |
| 0 | 33 |
| 1 | 42 |

If we regress CHD on age using linear regression, this is referred to as the “linear probability model.” The diagnostic plots and summary output are below:

Call:  
lm(formula = chd ~ age, data = chd.data)  
  
Residuals:  
 Min 1Q Median 3Q Max   
-0.85793 -0.33992 -0.07274 0.31656 0.99269   
  
Coefficients:  
 Estimate Std. Error t value Pr(>|t|)   
(Intercept) -0.537960 0.168809 -3.187 0.00193 \*\*   
age 0.021811 0.003679 5.929 4.57e-08 \*\*\*  
---  
Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
  
Residual standard error: 0.429 on 98 degrees of freedom  
Multiple R-squared: 0.264, Adjusted R-squared: 0.2565   
F-statistic: 35.15 on 1 and 98 DF, p-value: 4.575e-08

|  |
| --- |
| Figure 10.1: Linear probability model with CHD example |

**Before moving on, please take a moment to write down you conclusions (and rationale) about whether the assumptions of linear regression are met for these data.**

I’ll note that researchers who have a strong preference for OLS methods (AKA economists) often approach binary outcomes using the linear probability model. As we can see, this approach violates all of the assumptions of linear regression, can lead to predicted probabilities outside of the range [0,1], produces incorrect standard errors for model parameters (need to use HC standard errors), and is, in a word, wrong. Yet, despite all this, it works pretty well in some situations and has the benefit of being easier to interpret than logistic regression. We will consider the situations in which the linear probability model is “close enough” at the end of the next section.

## 10.2 Logit & logistic functions

The general game plan for dealing with a binary outcome is to transform it into a different variable that is easier to work with, run the analysis, and then “reverse-transform” the model coefficients so that they are interpretable in terms of the original binary variable. This strategy should sound familiar from [Chapter 8](#sec-chap-8) – it’s the same overall approach we used for log-linear regression. Also in common with [Chapter 8](#sec-chap-8), we are going to use logs and exponents as the main workhorse for this approach (that is where the “log” in logistic comes from).

However, the overall strategy for transforming the variable in logistic regression is a bit more complicated than the log-linear model. So, it is helpful to start wit an overall “roadmap”.

* Step 1 (from binary to probability). First, we are going to work with probabilities rather than the original binary variable. In terms of our example, we are going to shift focus from whether or not a person has CHD to the *probability* of a person having CHD.
* Step 2 (from probability to logistic). The logistic function is widely-used model for probabilities. In terms of our example, we are going to use the logistic function to relate the probability of a person having CHD to their age.
* Step 3 (from logistic to logit). The logistic function has a nice interpretation, but it is not a linear function of age. So, we are going to transform it into something that is linear in age, which will let us “port over” a lot of what we have learned about linear models. Actually, the reason we choose the logistic function as a model of probability is because this transform is relatively straightforward and can be “undone” afterwards when interpreting the model coefficients, just like with log-linear regression. The transformation two steps:
  + Step3A (probability to odds). First we transform the probability of having CHD into the *odds* of having CHD. If denotes probability then odds are just . We will spend a while talking about how to interpret odds.
  + Step 3B (odds to logit). Then we take the log of the odds, which is called the *logit*. The logit turns out to be a linear function of age, so we can model the relationship between age and the logit of CHD in a way that is very similar to regular linear regression.

So, that’s the overall approach to dealing with a binary variable in logistic regression. Clear as mud, right? Don’t worry, we will walk through each step in the following subsections. If you find yourself getting lost in the details, it can be helpful to refer back to this overall strategy. In short, the overall game plan is:

Once we have all these concepts in play, we can start doing logistic regression.

### 10.2.1 From binary to probability

The following table presents the example data in terms of the proportion of cases with CHD, broken down by age groups. The first column shows the age groups, the second shows the number of cases without CHD, the third shows the number of cases with CHD, and the last column shows the proportion of cases with CHD.

knitr::include\_graphics("files/images/props.png")

|  |
| --- |
| Figure 10.2: From a binary variable to proportions |

Recall that a proportion is computed as the number of cases of interest over the total number of cases. In terms of the table above:

were denotes the number of cases with CDH, and is the number of cases without.

The number can be interpreted in many ways, which leads to a lot of terminology here.

* The **proportion** of cases in our sample with CHD.
* If we multiply by 100, it is the **percentage** of cases with CHD (i.e., cases per 100) in our sample.
* If we multiply by a number other than 100 (say 1000), it is the **rate** of CHD (e.g., cases per 1000) in our sample.
* Since a proportion is just the mean of binary variable, it is the **mean** or expected value of CHD in our sample.
* And finally, since proportions are one interpretation of probability, it is the **probability** of CHD in our sample.

You might hear all of these terms (i.e., proportion, percentage, rate, mean, probability) used in connection with logistic regression. But, they are all just different ways of interpreting the rightmost column of [Figure 10.2](#fig-props). I will try to make a point of using all of these terms so you get used to interpreting them in this context :)

Another concept that will be useful for interpreting our data is *odds*. Odds are closely related to, but not the same as, probability. The figure below adds the odds of having CHD to Figure [Figure 10.2](#fig-props).

knitr::include\_graphics("files/images/odds.png")

|  |
| --- |
| Figure 10.3: Proportions and odds |

As shown in the table, the odds are also a function of the two sample sizes, and :

Let’s take a moment to compare the interpretation of probability versus odds.

* The first row of the table tells us that the probability of having CHD in your 20’s is “1 in 10”. Loosely, this means that for every 10 people in their 20s, one of them will have CHD.
* By contrast, the odds of having CHD in your twenties is “1 to 9”. Roughly, this means that for every person in their twenties with CHD, there are nine without CHD.

Clearly, probabilities and odds are just two different ways of packaging the same information. The following equations shows the relation between odds and probability (these are derived from Equations [Equation 10.1](#eq-prob) and [Equation 10.2](#eq-odds) using algebra)

We will see these relations again shortly. But, before moving on, let’s get some more practice interpreting odds and probabilities using the data in [Figure 10.3](#fig-odds). **Please write down your answers to the following questions and be prepared to share them in class. For each question provide a verbal interpretation of the numerical answer (e.g, odds of 2 to 1 means that for every two people with a trait, there is one without.)**

1. What is the probability of a person in their 40s having CHD?
2. What are the odds of a person in their 40s having CHD?
3. What is the probability of someone in their 50s **not** having CHD?
4. What are the odds of someone in their 50s **not** having CHD?
5. What is probability of having CHD in your 40s, compared to your 30s? (e.g., is 3 times higher? 4 times higher?)
6. What are the odds of having CHD in your 40s, compared to your 30s?

The answers hidden below (use the Code button to reveal), but you won’t learn anything if you don’t try the question yourself first!

# 1. .39, so about 40% of people  
# 2. 11/17, so for 11 people with CHD there are 17 without  
# 3. 1 - .72 = .28, so about 28% of people   
# 4. (18/7)^-1 = 7/18, so 7 out ever 18 people  
# 5. .39 / .19 ~= 2, so the probability of having CHD in your 40s is about 2 times higher than the probability of having CHD in your 30s. This is called a relative risk, or a risk ratio.   
#6. (11/17)/(5/22) ~= 2.8, so the odds of having CHD in your 40s is about 2.8 times higher than the odds of having CHD in your 30s. This is called an odds ratio.

### 10.2.2 From probability to logistic

On thing you may have noted about the CHD data is that the proportion of cases with CHD increases with age. This relationship is shown visually in [Figure 10.4](#fig-chd2).

# Sample proportions  
prop <- tapply(chd, catage, mean)  
  
# Age categories  
years <- unique(catage)\*10  
  
# Plot  
plot(years,   
 prop,   
 type = "l",   
 lwd = 2,   
 col = "#4B9CD3",   
 ylab = "p(CHD =1)",   
 xlab = "Age categories")

|  |
| --- |
| Figure 10.4: Proportion of cases with CHD as a function of age |

Looking at the plot, we might suspect that the relationship between the probability of CHD and age is non-linear. In particular, we know that probabilities cannot take on values outside of the range , so the relationship is going to have to “flatten out” in the tails. For example, even if you are a baby, your probability of having CHD cannot be less than 0. And, even if you are centenarian, the probability can’t be great than 1.

Based on this reasoning, we know that the relationship between age and the rate of CHD should take on a sort of “S-shaped” curve or “sigmoid”. This S-shape is hinted at in [Figure 10.4](#fig-chd2) but is not very clear. Some clearer examples are shown in [Figure 10.5](#fig-sigmoids).

# Logistic function  
logistic <- function(x, a, b){exp(a\*x + b) / (1 + exp(a\*x + b))}  
  
# Generate data  
x <- seq(-5, 5, by = .1)  
  
# Plots  
plot(x, logistic(x, 1, 0),   
 type = "l",   
 lwd = 2,   
 col = 2,   
 ylab = "logistic")  
  
points(x, logistic(x, .75, -1.5),   
 type = "l",   
 lwd = 2,   
 col = 3,   
 ylab = "logistic")  
  
points(x, logistic(x, 1.5,- 1),   
 type = "l",   
 lwd = 2,  
 col = 4,   
 ylab = "logistic")  
  
points(x, logistic(x, 3, 2),   
 type = "l",   
 lwd = 2,   
 col = 5,   
 ylab = "logistic")

|  |
| --- |
| Figure 10.5: Examples of sigmoids |

The mathematical equation used to create these S-shaped curves is called the logistic function, the namesake of logistic regression. All you need to take-away from [Figure 10.5](#fig-sigmoids) is that there is mathematical function that produces the kind of relations we are expecting between age (continuous) the the probability of having CHD (bounded to the interval ).

Returning to our example, we can see in [Figure 10.6](#fig-chd3) that the logistic function provides a reasonable approximation for the relationship between the rate of CHD and age.

par(mfrow = c(1, 2))  
plot(years, prop, type = "l", lwd = 2, col = "#4B9CD3", ylab = "p(CHD =1)", xlab = "Age categories")  
plot(20:60, logistic(20:60, .12, -5.2), col = "#4B9CD3", ylab = "logistic", xlab = "Age in years")

|  |
| --- |
| Figure 10.6: Proportion of cases with CHD, data versus logistic |

One important thing to notice about [Figure 10.6](#fig-chd3) is that the plot on the left required re-coding age into a categorical variable and computing the proportion of cases with CHD in each age category (see [Figure 10.2](#fig-props)). However, the logistic plot on the right did not require categorizing age. So, one advantage of using the logistic function is that we can model the probability of CHD as a function of age “directly”, without having to categorize our predictor variables.

The take home message of this section is that the logistic function is a nice way to model how a proportion depends on a continuous variable like age. Next, we’ll talk about the math of the logistic function in a bit more detail.

### 10.2.3 Logistic to log odds (logit)

The formula for the logistic function (i.e., the function that produced the curves in [Figure 10.5](#fig-sigmoids) is

This function maps the variable onto the interval . In [Figure 10.6](#fig-chd3) we saw that the logistic function can provide a nice model for probabilities. We also saw that the logistic function is non-linear function of (i.e., it is sigmoidal or S-shaped).

However, a nice thing about the logistic function is that we can transform it into a linear function of . Since we already know how to deal with linear functions (that is what this whole course has been about!), transforming the logistic into a linear function of will let us port over a lot of what we know about linear regression to situations in which the outcome variable is binary. (In fact, the real motivation for choosing the logistic function in the first place, rather than some other S-shaped curve.)

So, let’s see how to get from our S-shaped logistic function of to a linear function of . Algebra with [Equation 10.4](#eq-logistic) shows that we can re-express the logistic function in terms of the odds:

Note that Equations [Equation 10.4](#eq-logistic) and [Equation 10.5](#eq-exp-odds) directly parallel the two expressions in Equation [Equation 10.3](#eq-p2o). The only difference is that, in the logistic model, the odds are represented as an exponential function of the variable , which is what Equation [Equation 10.5](#eq-exp-odds) is telling us.

In order to turn [Equation 10.5](#eq-exp-odds) into a linear function of , all we need to do is get rid of the exponent. Do you remember how?? That’s right, just take the log (see [Section 8.1](#sec-math-review-8)):

This equation is telling us that the log of the odds is linear in . The log-odds is also called the *logit*, which is short for “logistic unit.”

The relationship among the logistic, odds, and logit are summarized in [Figure 10.7](#fig-logit).

knitr::include\_graphics("files/images/logit.png")

|  |
| --- |
| Figure 10.7: Logistic, odds, and logit |

* The left-hand panel shows the logistic function. This is our “intuitive-but-nonlinear” model for probabilities. In terms of our example, this panel is saying that the probability of having CHD is a logistic or S-shaped function of age.
* The middle panel shows that the odds are an exponential function of . In terms of our example, this means that the odds of having CHD are an exponential function of age. This is the main assumption of the logistic model, and we will revisit this assumption again when we get to [Section 10.7](#sec-assumption-checking-10).
* Finally, the right-hand panel shows the “not-really-intuitive-but-definitely-linear” model for the logit. In terms of our example, the logit of having CHD is a linear function of age.

The next section discusses how to interpret the logit by reverse-transforming it back to the odds and probabilities. The situation is a lot like log-linear regression ([Chapter 9](#sec-chap-9)).

### 10.2.4 Pop quiz

Before moving, lets nail down the relation between probability, odds, and logits. [Figure 10.8](#fig-logit-table)) presents the relationship in tabular form.

|  |
| --- |
| Figure 10.8: Logistic, odds, and logit |

**I will asks some questions along the following lines in class.**

* If probability of an event is equal to .1, what are the odds the event? What is the logit of the event?
* If odds of an event are 4 to 1, what is the probability of the event? What is the logit of the event?
* If logit < 0 then probability < ? and odds < ?
* What is more likely: a event with probability of .9 or an event with odds of .9?
* If a probability of corresponds to a of , what is the logit corresponding to ? (Hint, try some numerical examples from the table).

### 10.2.5 Next steps

The logit is our workhorse for logistic regression. In [Section 10.3](#sec-simple-10), we will replace the variable with a simple regression model to get simple logistic regression. In **?@sec-multiple-10** we will extend simple logistic regression to multiple logistic regression, just like we did for multiple linear regression.

Although the logit is the workhorse, we generally don’t want to work with the logit when it comes time to interpret the results. The situation here is a lot like log-linear regression ([Chapter 9](#sec-chap-9)). In log-linear regression, we treated as a linear function of our predictor variable(s). However, we didn’t want to interpret the model in terms of , because, well, who thinks in log units? Instead we wanted an interpretation in terms of the original outcome, .

The same situation applies here. You may have already noted that the relationship between the logit (i.e., ) and in logistic regression is the same as the relationship between and in log-linear regression. The parallel between the two model is as follows:

* In the log-linear model we interpreted a unit increase in in terms of an change in (see Section (**ref-interpretation-8?**))).
* In logistic regression we will interpret a unit increase in in terms of an times change in .

So, while we use the logit function for modeling, we often use the odds for interpretation. One subtle difference to be aware of is that, in logistic regression, we usually report results in terms of relative magnitude (called the odds ratio) rather than relative change, although relative change is often used for verbal reporting. We will see examples in the next section.

Some authors have argued that people don’t really know how to interpret odds properly. These authors suggest that we interpret the logistic model in terms of probabilities, rather than odds. We will discuss how to do this as well.

### 10.2.6 Summary

At this point we have covered the overall logic of how we can model a binary outcome variable like CHD in terms of the logistic function. The overall situation is very similar to, but a bit more complicated than, log-linear regression. The main take aways are

* We use the logit (log-odds) for statistical analysis, because it results in a linear function, and we already know how to deal with linear functions.
* We use the odds for interpretation, because the logistic model leads to proportional change in the odds, in the same way that the log-linear model leads to proportional change in .
* We can also use probabilities for interpretation, but, since the logistic model implies that probabilities are non-linear (sigmoidal), things can get a bit complicated with this approach.

## 10.3 Simple logistic regression

In this section we move onto logistic regression proper. For the CHD example, the model we are interested in is

We are going to skip a few steps and go right into the interpretation of the R output. Once we know how to interpret the output, we will loop back to discuss details of estimation and inference in the following sections.

The summary R output for the example is below. The focus for now is just the interpretation of the values under the “Estimate” heading.

mod2 <- glm(chd ~ age, family = binomial, data = chd.data)  
summary(mod2)

Call:  
glm(formula = chd ~ age, family = binomial, data = chd.data)  
  
Coefficients:  
 Estimate Std. Error z value Pr(>|z|)   
(Intercept) -5.30945 1.13365 -4.683 2.82e-06 \*\*\*  
age 0.11092 0.02406 4.610 4.02e-06 \*\*\*  
---  
Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
  
(Dispersion parameter for binomial family taken to be 1)  
  
 Null deviance: 136.66 on 99 degrees of freedom  
Residual deviance: 107.35 on 98 degrees of freedom  
AIC: 111.35  
  
Number of Fisher Scoring iterations: 4

Plugging the estimates into our logit model, we have the following equation

The “literal” interpretation of this equation is:

* When , .
* Each unit of increase in age (i.e., each additional year) is associated with a .11 unit increase in .

While this interpretation is perfectly correct, most applied audiences are not going to know how to interpret . So, instead, we often work with the odds and probabilities, as outlined in the next few sections.

### 10.3.1 Odds ratio

The logistic regression model implies

where are the odds of the outcome associated with a given value of the predictor . Equation [Equation 10.7](#eq-OR) is called the *odds ratio* (abbreviated OR) associated with a one-unit increase in .

If you refer back to section [Section 8.5.5](#sec-derivation-8), you can see we are using the exact same approach from log-linear regression, but in [Equation 10.7](#eq-OR) we interpret the regression coefficient in term of the odds that , rather than the variable itself.

For the CHD example, the OR is:

This means that each additional year of age is associated with an OR of 1.11. For example, the odds for someone aged 21 having CHD is 1.11 time larger (relative magnitude) that someone aged 20. The really useful thing about the OR is that it is constant over values of the predictor. So, regardless of whether we are comparing a 21-year-old to a 20-year-old, or 41-year-old to a 40-year-old, the OR is the same.

Just like the log-linear model, we can also report the results of our analysis in terms of relative change rather than relative magnitude. In particular, the percent increase in the odds of CHD associated with each additional year of age is:

This means that the predicted odds of CHD increase 11.63% for each additional year of age.

Whether you use relative magnitude (i.e., the odds ratio) or relative change (i.e., percent change in odds) to report the results of logistic regression is up to you. In many fields, it is conventional to reports the odds ratios in tables, but to use percent change when writing about results in a sentence.

Before moving, **please practice your interpretation of the OR in simple logistic regression using the following examples**

* If what is the OR equal to? What is the percent change in the odds for a one unit increase in ?
* If what is the OR equal to? What is the percent change in the odds for a one unit increase in ?
* If what is the OR equal to? What is the percent change in the odds for a 10 unit increase in ?
* If the odds increase 100% for a one unit increase in , what equal to?

Answers hidden below (use Code button), but please try out the questions yourself first!

# 1. OR = exp(0) = 1 and percent change equals (exp(0) - 1) X 100 = 0%. So, "no relationship" means OR = 1.   
# 2. OR = exp(.25) = 1.2840 and percent change equals (exp(.25) - 1) X 100 = 28.40% increase  
# 3. OR = exp(-.025) = 0.9753 and percent change for one unit equals (exp(-.025) - 1) X 100 = (-.02469 X 100 = -2.469%. For 10 units of change, multiply by 10, which gives 24.69% decrease (negative sign is decrease).   
# 4. (exp(b) - 1) X 100 = 100 --> exp(b) = 2 --> b = log(2) = .6931

### 10.3.2 Other interpretations: Predicted probabilities

Another way to interpret the logistic model is in terms of the predicted probabilities, which are plotted below for the example data.

visreg::visreg(mod2, xvar = "age", scale = "response")

|  |
| --- |
| Figure 10.9: Predicted probabilities |

Using this plot, we can read off the probability of CHD for any given age. We might also want to report the probability of CHD for two or more chosen ages, which is an example of the MERV approach to marginal effects (see [Section 5.4](#sec-inference-for-interactions-5)):

library(emmeans)  
emmeans(mod2, specs = "age", at = list(age = c(20, 40)), type = "response")

age prob SE df asymp.LCL asymp.UCL  
 20 0.0435 0.0279 Inf 0.0121 0.145  
 40 0.2947 0.0578 Inf 0.1951 0.419  
  
Confidence level used: 0.95   
Intervals are back-transformed from the logit scale

The ratio of two probabilities is often called the *risk ratio* or the *relative risk*. So, we could also say that the risk ratio of CHD for someone who in their 40s as compared to someone who is 20 is .2947 / .0435 = 6.77. Otherwise stated, the risk of having CHD in you are 40s is almost 7 times higher than in your 20s (relative magnitude).

### 10.3.3 Other interpretations: Equal odds

Another interpretation of logistic regression is to report the value of at which the of the outcome are equal to 1 (equivalently, the probability of the outcome is equal to .5). This idea is illustrated in [Figure 10.10](#fig-equal-odds).

x <- data.frame(age = 20:70)  
prob <- predict(mod2, newdata = x, type = "response")  
plot(x$age, prob, xlab = "age", ylab = "p(CHD)", type = "l", col = "#4B9CD3")  
segments(x0 = 15, y0 = .5, x1 = 48, y1 = .5, lty = 2, lwd = 2)  
segments(x0 = 48, y0 = 0, x1 = 48, y1 = .5, lty = 3, lwd = 2)

|  |
| --- |
| Figure 10.10: The Equal Odds Interpretation |

First we find the probability of .5 on the axis and then follow the horizontal dashed line to the logistic curve. Then we follow the vertical dashed line down to the value of . This gives use the age at which the probability of CHD is “50-50”. Based on the plot we can say that, after your 48th birthday, your chances of having CHD are above 50%.

The math behind this interpretation is below. Since

we can solve

to find the age at which someone has equal odds of CHD, leading to

For the example data

which confirms the conclusion we made looking at the plot.

In summary, another way of interpreting regression coefficients in simple logistic regression is to compute , which gives the value of at which .

### 10.3.4 Other interpretations: Rate of change

Yet another interpretation is in terms of the slope of the straight line (tangent) through the point . The slope of this line describes the rate of change in the probability of CHD for people who are “close to” the age of equal odds (48 years in our example). The tangent line for our example is shown in [Figure 10.11](#fig-tangent).

x <- data.frame(age = 20:70)  
prob <- predict(mod2, newdata = x, type = "response")  
plot(x$age, prob, xlab = "age", ylab = "p(CHD)", type = "l", col = "#4B9CD3")  
abline(a = -.815, b = .11/4, lty = 2, lwd = 2)

|  |
| --- |
| Figure 10.11: The Rate of Change Interpretation |

It turns out that the slope of the tangent line is equal to exactly . The derivation requires calculus and is omitted (ask in class if you are interested!). For the example data . So, for every additional year, the predicted probability of CHD increases by .0275. Keep in mind, this interpretation only applies to people who “around” the age of equal odds (48 years old in this example). Looking at the plot, we can see that this approximation is pretty good for people between the ages of 40 and 60.

### 10.3.5 Relation to linear probability model

Notice that in our example, the logistic function is roughly linear for probabilities in the range . As mentioned in the introduction of this chapter, this is the situation in which using linear regression with a binary outcome (i.e., the linear probability model) works “well enough”. Also note that the regression coefficient from the linear probability model in [Section 10.1](#sec-chd-example-10) () is in the ballpark of the coefficient computed above (). These numbers are both describing how the probability of CHD is related to a person’s age.

The logistic model also provides us with a way of “diagnosing” whether the linear probability model is a good approximation. As noted, the logistic function is roughly linear for probabilities in the range . If we ran a linear regression on CHD and all of the fitted / predicted values were within the range , we would be in the situation where we might prefer to use linear regression (despite it being technically wrong). Referring back to residual vs. plotted in [Section 10.1](#sec-chd-example-10), we can see that the predicted values were outside of this range, so the logistic model is the better way to approach for this example.

### 10.3.6 Summary

The simple logistic regression model

has the following interpretations.

* In terms of the logit:
  + The intercept () is the predicted value of the log-odds of CHD when age = 0.
  + The slope () is how much the predicted log-odds of CHD changes for a one unit (year) increase in age.
* In terms of the odds:
  + The exponent of the regression parameter, , is the odds ratio associated with a one unit increase in age (relative magnitude).
  + is the percentage change in the odds of CHD for a one unit increase in age (relative change).
  + For the intercept, is the predicted odds of having CHD when . This isn’t a very useful number in our example, but it can be useful when the predictor(s) are centered.
* In terms of predicted probabilities:
  + The logistic plot provides a visual summary of how the probability of CHD changes as a function of age – if you want to report in terms of probabilities, this plot is usually a good choice.
  + Predicted probabilities can be reported for specific values of age (MERVs), and risk ratios / relative risk can be computed to compare the predicted probabilities at specific ages. This is the same approach we used for following-up interactions in [Section 5.4](#sec-inference-for-interactions-5) (i.e., you can use emmeans in R).
  + corresponds the age at which the probability of having CHD is “50-50”.
  + is the approximate rate of increase (slope) of the probability of having CHD for people close to the “50-50” point.

**Please write down the numerical values of each of the above summaries for the CHD example (except the predicted probability plot). You can select any values of age to report the predicted probabilities and risk ratios, and you can “eye ball” the probabilities using** [**Figure 10.9**](#fig-pred-prob)**.**

Answers are hidden below (use the Code button), but please try them yourself first.

# Logit:   
# predicted logit when age = 0: a = -5.31;  
# expected increase in logit when age increases by one year: b = .11  
  
# Odds:  
# OR = exp(b) = exp(.11) = 1.1163  
# % change in Odds = (exp(b) - 1) X 100 = 11.63%  
  
# Probability  
# In the example, used risk for Age = 40 relative to Age = 20: .2947/.0435 = 6.77  
# "50-50" age: -a/b = 5.31 / .11 = 48.27   
# Rate of change at "50-50" age: b/4 = .11/4 = .0275

## 10.4 Estimation

The previous section discussed the interpretation of simple logistic regression. In this section we discuss how to get the numbers in the “Estimate” column of the R summary() table, and in the next section we discuss how to compute statistical tests and confidence intervals.

The material in these next two sections involves a lot of mathematical and statistical concepts, although I try to keep the formulas to a minimum. Because the material is so technical, there will be a mini-lecture on these sections during class. It may also take some extra time to read through these next couple of sections, and you may want to write down any additional questions you have about the material so that we can address them in class together. Basically, the next two sections are pretty tough, so hang in there :)

The short version:

* In logistic regression, we use maximum likelihood (ML) rather than ordinary least squares (OLS) to estimate the model parameters.
* The resulting estimates are called the maximum likelihood estimates (MLEs).
* Unlike OLS estimates in linear regression, MLEs in logistic regression cannot be written down in “closed form.” This means that we need to define MLEs algorithmically, rather algebraically – more on this below.
* Inference for the MLEs is based on *asymptotics*. In statistics, “asymptotic” means “as the sample size goes to infinity…and beyond!” The implication is that using ML also changes how we compute standard errors, statistical tests, confidence intervals, etc.
* Our overall goal is to know enough about this technical stuff to be able to interpret and recognize the limitations of logistic regression.

### 10.4.1 ML vs OLS

Let’s start by contrasting ML with OLS. Recall that in linear regression, OLS leads to “closed form” expressions for the regression coefficients ( [Section 2.3](#sec-ols-2)). This means we can write down an explicit algebraic equation for the regression coefficients, e.g.,

As we discussed in [Section 10.1](#sec-chd-example-10), these estimates often aren’t very good when the outcome is binary. So, instead we use ML.

Unlike OLS with linear regression, ML with logistic regression does not lead to closed-form expressions for the regression coefficients. This means we can’t write down the other side of the equation using an algebraic expression like the one above. Instead, we must define the regression parameters in terms of a computational procedure (i.e., an algorithm) that produces the intended result. That computational procedure is maximization of the likelihood function.

We discussed the difference between algebraic and algorithmic definitions when we compared the mean and median last semester, and we can address it again in class if you have further questions. In the present context, the take home message is:

* In OLS, we didn’t really need to know much about least squares *per se*, because we could just work with the formulas for the regression coefficients.
* In ML, we don’t get any formulas for the coefficients, so were are going to have to dig into what the likelihood is and how to use it.

### 10.4.2 The likelihood

The likelihood is a function that tells us the probability of a sample being drawn from a population. For our CHD example data, we have cases in which there are 43 people with CHD and 57 people without CHD. The likelihood tells us the probability of this sample being drawn from the population. In this section, we build up a mathematical expression for the likelihood function. Note that when the outcome variable in a regression is binary, this is more specifically called the *binomial* likelihood.

The building block of the binomial likelihood is the probability of a person having CHD, which we denote as

Keep in mind that we are defining the probability of sample being drawn from a population. So, we are imaging that a person is sampled from a population, and some proportion of that population has CHD. That proportion is what we are representing as the probability .

For a single person (i.e., a sample of ), the likelihood describes the probability that the person either does or does not have CHD. The likelihood is given by the following formula:

This looks complicated, but its just a way of writing the population probability of having CHD, or not having CHD, in a single equation. All it says is:

* The probability of sampling someone with CHD is .
  + i.e., plug in into [Equation 10.8](#eq-Li) and the result is .
* The probability of sampling someone without CHD is .
  + i.e., plug in into [Equation 10.8](#eq-Li) and the answer is .

You should do the plugging-in now to see how this works.

For a simple random sample of people, the likelihood of observing the ful sample is just the product of the likelihoods for each person in the sample. To write the likelihood for person , we will use the notation , and to write the likelihood for the entire sample, we will keep using without a subscript:

(The symbol is shorthand for multiplication, just like is used for addition. )

If we plug-in to [Equation 10.9](#eq-L) using the values of in our sample, this gives us the likelihood of observing the sample. For example, the values of CHD for the first four cases in the data set (1, 1, 0, 1). So, the first four terms in the likelihood are:

**Please take a moment to verify this result for yourself using** [**Equation 10.8**](#eq-Li) **and** [**Equation 10.9**](#eq-L)**, and we will walk through the derivation together in class.**

### 10.4.3 Maximizing the likelihood

So far we have addressed the general form of the likelihood for a binary variable. But we haven’t talked about how to get the probabilities . This is where the logistic regression comes in. As discussed in [Section 10.2](#sec-logit-10), the purpose of logistic regression is to model the probability of having CHD. Thus, in order to turn [Equation 10.9](#eq-L) into the likelihood function for logistic regression, we just plug-in for using [Equation 10.4](#eq-logistic).

As you might expect, this equation turns out to be pretty complicated to write down, and we aren’t going to do that here. The point is that we *can* write down the likelihood for logistic regression, and once we have it written down we can use standard procedures from calculus to find its maximum (e.g., Newton’s method).

This is about as much math as we need to state the overall idea behind maximum likelihood estimation: We select the regression coefficients of the logistic model so that the likelihood in [Equation 10.9](#eq-L) is maximized.

You might be asking, why is maximizing the likelihood a good way to get regression coefficients for logistic regression? There are lots of ways to answer this question, but I think the best answer is that the resulting estimates have good properties – they are unbiased and precise, at least with large samples. We can discuss the rationale behind maximum likelihood more in class, so if you are interested please just let me know.

Doing the computations that maximize the likelihood is quite complicated, and the details are beyond the scope of this course (for more details, check out Chapter 15 of (**cite:fox?**)). However, there is one issue that we should be aware of: the likelihood can cause so-called “underflow” issues in computing. This is because we are multiplying together lots of numbers less than 1, which can lead to very, very small values – too small for a computer represent (i.e., underflow). To address this issue, we usually work with the log-likelihood, rather than the likelihood itself:

The last equality uses the “addition with logs” rule from [Section 8.1](#sec-math-review-8). For our purposes here, we just need to know that taking the log of the likelihood changes a very small number (the likelihood) into a large negative number (the log-likelihood), which prevents our computer from breaking when doing the computations.

The log-likelihood, , also shows up a lot in what follows. Its main uses are summarized below

* is called the *deviance* of the model. I am not sure how it got this name, but you will see it a lot in the R output.
* We can use the deviance to compute R-squared-like statistics that describe the overall quality of the predictions made using a logistic regression model (see [Section 10.6](#sec-pseudo-rsquared-10))
* Two logistic regression models can be compared using their deviances. This leads to the logistic regression equivalent of the F-test of R-squared change ([Chapter 6](#sec-chap-6)), which is called the likelihood ratio test (see [Section 10.5](#sec-inference-10)).

### 10.4.4 Summary

Lets take a look at our summary() output for the CHD data again:

summary(mod2)

Call:  
glm(formula = chd ~ age, family = binomial, data = chd.data)  
  
Coefficients:  
 Estimate Std. Error z value Pr(>|z|)   
(Intercept) -5.30945 1.13365 -4.683 2.82e-06 \*\*\*  
age 0.11092 0.02406 4.610 4.02e-06 \*\*\*  
---  
Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
  
(Dispersion parameter for binomial family taken to be 1)  
  
 Null deviance: 136.66 on 99 degrees of freedom  
Residual deviance: 107.35 on 98 degrees of freedom  
AIC: 111.35  
  
Number of Fisher Scoring iterations: 4

This section has addressed the question: how do we get the numbers in the “Estimate” column of the output shown above? The answer is maximum likelihood (ML). In short:

* The likelihood function tells us the probability of randomly sampling people from a population.
  + When the outcome variable is binary, it is called the binomial likelihood.
* ML is based on the idea that the values of the model parameters that make the data most likely are the “best”
  + The resulting values are called the maximum likelihood estimates (MLEs).
* In logistic regression, ML is accomplished using a computer algorithm – we can’t write down algebraic expressions for the MLEs, we only have an algorithmic definition, which is to maximize the likelihood
  + FYI: The algorithm R uses for logistic regression is a version of Newton-Raphson called Fisher Scoring.

## 10.5 Inference

In this section we talk about statistical tests and confidence intervals for logistic regression. There are three main types of inference for logistic regression:

1. *Wald test of regression coefficients*. The Wald test is a z-test that can be used with large samples (). This is what is reported in R’s summary() output for logistic regression.
   * Although the Wald test is reported by default in R, you should be careful when interpreting it. The Wald test is computed just like a regular t-test. However, rather than using a t-distribution, the Wald test is based on the normal distribution. This approach only applies in large samples (technically, when ). In smaller samples (), the standard errors can be too large, especially if the coefficient is large in value. This means that the Wald test can be under-powered, leading to too many Type I Errors in small samples. So, when you have fewer than 100 cases, it is good practice to use one of the other two procedures discussed below.
2. *Confidence intervals for odds ratios*. Odds ratios can be tested using confidence intervals. If the confidence interval includes the value 1, then the odds ratio is not statistically significant at the chosen level of alpha / confidence. Confidence intervals can be computed in two ways for logistic regression, either using the Wald test (just like a usual confidence based on a t-test), or using a procedure called *profile likelihood*. The latter is preferred because it is more accurate, and it is the default in R. Ask in class if you want to know more :)
3. *Likelihood ratio test of nested models*. The likelihood ratio test plays the same role as the F-test of in OLS regression. It allows you to compare nested models. We will talk about the analogue statistic for in [Section 10.6](#sec-pseudo-rsquared-10).

You might be wondering how the likelihood-based procedures (2 and 3) avoid the issues with the Wald test. Technically, the likelihood-based procedures also assume large samples sizes (), but they do not require computing the standard errors of the regression coefficients, which is the problem with the Wald test. So, although both procedures have the same assumptions, the likelihood-based procedures perform better than the Wald test in practice, especially with smaller samples.

The rest this section briefly illustrates each of these three methods using our example data.

### 10.5.1 Wald test

The MLEs are asymptotically normally (based on the central limit theorem), so we can divide the MLEs by their standard errors to get an asymptotic z-test. This test is called a Wald test, named after the person (Abraham Wald) who first invented it. The Wald test is what R reports in the summary() table below.

summary(mod2)

Call:  
glm(formula = chd ~ age, family = binomial, data = chd.data)  
  
Coefficients:  
 Estimate Std. Error z value Pr(>|z|)   
(Intercept) -5.30945 1.13365 -4.683 2.82e-06 \*\*\*  
age 0.11092 0.02406 4.610 4.02e-06 \*\*\*  
---  
Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
  
(Dispersion parameter for binomial family taken to be 1)  
  
 Null deviance: 136.66 on 99 degrees of freedom  
Residual deviance: 107.35 on 98 degrees of freedom  
AIC: 111.35  
  
Number of Fisher Scoring iterations: 4

At this point in the semester, you should have no problem interpreting whether these tests are statistically significant or not. The main wrinkle that comes up in logistic regression is that we are working with an (asymptotic) z-test rather than usual t-test from OLS regression. As noted, this z-test is also called Wald test.

In summary:

* The null hypotheses can be tested against the alternative using the following z-test, which has a standard normal distribution when the null hypothesis is true:
* This tests assumes:
  + is the MLE (this requires that model is correct – more on this in [Section 10.7](#sec-assumption-checking-10))
  + The sample size is “approximately” infinite ( is a practical lower limit).

The Wald test also applies to the intercept.

### 10.5.2 Odds ratio (OR)

In [Section 10.3](#sec-simple-10), the OR was introduced as a more intuitive way of interpreting logistic regression. But, how do we make inferences about this more interpretable quantity? The short answer: use confidence intervals.

The reason we use confidence intervals instead of tests of significance is because the null hypothesis corresponds to

So, if the regression coefficient is zero (i.e., no relationship), this means the OR is one. In terms of percent change, means 0% percent change:

To obtain confidence intervals on the OR, we can use the following two steps:

1. Compute the confidence intervals for the regression coefficient :
2. Then exponentiate the confidence intervals to get a confidence interval for the OR.

The output below shows the result of this procedure for the CHD data.

ci.b <- confint(mod2)  
ci.table <- cbind(coefs = coef(mod2), ci.b)  
exp(ci.table)

coefs 2.5 % 97.5 %  
(Intercept) 0.004944629 0.0004412621 0.0389236  
age 1.117306795 1.0692223156 1.1758681

In APA notation, we could write the results for age as . **Please write down your interpretation of the OR for age and its confidence interval. In your interpretation, please mention the relative change in the odds that is associated with each additional year of age and whether or not this is statistically different from zero.**

Note that the confidence intervals produced by use the likelihood rather than the Wald test. So, the problems we discussed with the Wald test don’t apply to these confidence intervals. (You can get Wald confidence intervals form the confint function, but this is not the default.)

### 10.5.3 Likelihood ratio test

The likelihood ratio (LR) test is analogous to the F-test of R-squared change in OLS regression. Its a general workhorse for model building and can be used in other creative ways. For example, the LR test can be used to replace the Wald test in simple logistic regression. To do this, we compare a model with only the intercept to a model with one predictor. And as noted above, the LR test avoids the problems associated with the Wald test, so it is usually preferred when the sample size is smaller ().

The LR tests works as follows:

* Assume that Model 1 with predictors is nested within Model 2 with predictors (i.e., the predictors in Model 1 are a subset of the predictors in Model 2; see [Chapter 6](#sec-chap-6) for a refresher on nested models).
* If it is true that the additional predictors in Model 2 all have regression coefficients equal to zero in the population, then it is also true that the models have equal likelihoods. This leads to the null hypothesis that the ratio of the likelihoods is equal to 1 (this is why its called a likelihood ratio test):
* When is true, the following test statistic has a chi-square distribution with degrees of freedom equal to .

In this equation, the symbol is the Greek letter for lower-case “x” and is pronounced “chi”. The chi-square test is computed as negative 2 times the log of the likelihood ratio, which ends up being the difference of the deviances of the two models (remember, the deviance is just a weird word for -2 times the log likelihood). Also note that the “hats” denote the likelihoods / log-likelihoods evaluated at the MLEs.

We have not covered the chi-square distribution previously, but it is a lot like the F-distribution. In fact, the F-static is a ratio of two chi-square statistics. Some examples of the chi-square distribution are shown below ( denotes the degrees of freedom).

|  |
| --- |
| Chi-square distributions (Source: https://en.wikipedia.org/wiki/Chi-squared\_distribution) |

To sum up:

* The LR test is the analogue of the F-test of R-squared change from linear regression (see [Chapter 6](#sec-chap-6)). It is used to compare two nested models.
* If LR test is significant, we conclude that model with more predictors does a better job or explaining our data (in the sense of R-squared). In practical terms, this means that we reject the models with fewer predictors in favor of the model with more predictors.
* If the LR test is not significant, we conclude that the model with more predictors does not add anything beyond the model with fewer predictors. In practical terms, this means that we reject the models with more predictors and stick with the model with fewer predictors.
* In the case where the two models differ by only one predictor, the LR test is conceptually equivalent to testing whether the regression coefficient of that predictor is equal to zero, and hence the LR test can replace the Wald test reported in R’s summary() output. This is again analogous to testing R-squared change in linear regression (see [Chapter 6](#sec-chap-6)).

Using our example, let’s conduct an LR test of the model with age as a predictor against a model without this predictor (i.e., the model with only the intercept). To do this, we can use the lrtest function of the lmtest package which produces the following output:

# The model with no predictors  
mod0 <- glm(chd ~ 1, family = "binomial", data = chd.data)  
  
# The LR test (enter the smaller model first)  
lmtest::lrtest(mod0, mod2)

Likelihood ratio test  
  
Model 1: chd ~ 1  
Model 2: chd ~ age  
 #Df LogLik Df Chisq Pr(>Chisq)   
1 1 -68.331   
2 2 -53.677 1 29.31 6.168e-08 \*\*\*  
---  
Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

In APA format, we could write this output as . **Please write down your interpretation of the LR test for the CHD example. Your interpretation should mention which two models were compared, whether the LR test was significant or not, and your conclusion about which model provided a better explanation of the data.**

## 10.6 (Pseudo) R-squared

As we have just discussed, the likelihood ratio test in logistic regression is analogous to the F-test of R-squared change in linear regression. But we have not yet discussed how to compute R-squared for logistic regression. That is what this section is about.

There are actually a number of statistics, called *pseudo R-squareds*, that have been developed to play the role of R-squared in logistic regression. The pseudo R-squareds are not computed as proportions of variance, which is why the are called *pseudo*. But they each try to describe the “preditive utility” of the logistic regression model, and each has their own shortcomings and limitations.

There is no clear “victor” in the battle of pseudo R-squareds, so we will just present one widely used exemplar that is straight forward to compute and interpret. You can learn about some other options here: <https://stats.oarc.ucla.edu/other/mult-pkg/faq/general/faq-what-are-pseudo-r-squareds/>

### 10.6.1 McFadden’s Pseudo R-squared

McFadden’s Pseudo R-squared focuses on how much the likelihood improves when comparing two nested models. This is close to, but not quite the same as, the interpretation of in linear regression.

McFadden’s Pseudo R-squared takes on values between 0 and 1 and its interpretation can be explained by considering these two extreme values.

For Model 1 nested within Model 2, McFadden’s R-squared is defined as:

Note that if Model 2 does not “add anything” to Model 1, then the two models have the same likelihood (see the discussion of the LR test in [Section 10.5](#sec-inference-10)). In this situation,

which implies

On the other hand, if Model 2 provides perfect predictions for all cases, (i.e., for each ), then

This implies

In summary, McFadden’s Rsquared

* is equal to zero if Model 2 does not add anything to Model 1,
* is equal to 1 if Model 2 provides perfect prediction of each data point (but model 1 does not).

Because McFadden’s R-squared is not a proportion of variance, we can’t interpret it the same way as OLS R-squared. Instead, it is usually interpreted in terms of model *fit*. Here “fit” means how much better a nesting model’s predictions are compared to a nested model, with zero meaning “not at all” and one meaning “as good as it gets”.

The following output shows McFadden’s R-squared for the example data. **Please write down an interpretation of the R-square value and we will discuss your interpretations in class.**

McFadden’s R-squared:

# Using the deviance of the models  
1 - mod2$deviance/mod0$deviance

[1] 0.2144684

You can use the “show code” button to see an example interpretation below (but try it by yourself first!)

# Interpretation: Relative a model with no predictors,   
# the inclusion of age improved model fit by about 21.4%   
# as much as a model with perfect predictions.   
  
# More elliptically: Using McFadden's R-squared the inclusion   
# of age improved model fit by about 21.4\%

## 10.7 Assumption checking

The final topic we need to cover is assumption checking. As we saw when deriving the binomial likelihood in [Section 10.4](#sec-estimation-10), there are two main assumptions in play when estimating logistic regression using maximum likelihood.

1. The different respondents (person, units) are independent (i.e., a simple random sample).
2. We have the right model for each individual’s probability of CHD, , which in this case is the logistic model.

Assumption 1 is ensured by random sampling. If your data wasn’t collected using random sampling, you’ll need to use a different modeling approach (e.g, multilevel modeling, see EDUC 935).

Assumption 2 can be checked with the Hosmer-Lemeshow (HL) test. The HL test is a chi-square test that compares the empirical probabilities of the outcome variable to the probabilities implied by the logistic regression model. In other words, we want to compare the two curves depicted below:

|  |
| --- |
| The idea behind the HL test |

The null hypothesis of the HL test is that these two curves are equivalent representations of the data. So, if we reject the null hypothesis, we reject claim that the model fits the data.

The HL test for the example data is reported below. The R code is shown by default to provide details about how the test is computed.

The logitgof function uses 3 arguments:

* obs is the binary outcome variable (chd)
* exp is the fitted values (predicted probabilities) from the logistic regression.
* g is the number of groups / bins to use. For this example, 5 is a good number.

## Note: Install package "generalhoslem" if you haven't do so already  
# install.packages("generalhoslem")  
  
hl\_test <- generalhoslem::logitgof(obs = chd, exp = fitted(mod2), g = 5)  
hl\_test

Hosmer and Lemeshow test (binary model)  
  
data: chd, fitted(mod2)  
X-squared = 0.046893, df = 3, p-value = 0.9973

Using APA notation, we could write the results of the test as . **Please write down your interpretation of the HL test and be prepared to share you answer in class. Your interpretation should mention whether the assumption that the logistic model fits the data is problematic or not.**

### 10.7.1 More details on the HL test\*

The HL statistic is computed using the observed and expected (model-implied) counts in each of the groups. Computational details are given below. The number of groups must be chosen by the researcher and should be based on considerations about sample size (e.g., the number of observation per group should be at least 20 to estimate a proportion).

# Observed and expected counts  
cbind(hl\_test$observed, hl\_test$expected)

y0 y1 yhat0 yhat1  
[0.0435,0.174] 18 2 17.870104 2.129896  
(0.174,0.318] 16 5 16.131098 4.868902  
(0.318,0.504] 12 9 12.268366 8.731634  
(0.504,0.734] 8 15 7.935627 15.064373  
(0.734,0.912] 3 12 2.794804 12.205196

The test statistic is the built up using the differences between the observed and expected values:

Hosmer and Lemeshow showed that this statistic has a chi-square distribution on when the null hypothesis is true (i.e., when the model fits the data).

## 10.8 Workbook

This section collects the questions asked in this chapter. The lessons for this chapter will focus on discussing these questions and then working on the exercises in **?@sec-exercises-10**. If you haven’t written down / thought about the answers to these questions before class, the lesson will not be very useful for you. Please engage with each question by writing down one or more answers, asking clarifying questions about related material, posing follow up questions, etc.

[Section 10.1](#sec-chd-example-10)

* Please take a moment to write down you conclusions (and rationale) about whether the assumptions of linear regression are met for the CHD data.

|  |
| --- |
| Linear probability model with CHD example |

Call:  
lm(formula = chd ~ age, data = chd.data)  
  
Residuals:  
 Min 1Q Median 3Q Max   
-0.85793 -0.33992 -0.07274 0.31656 0.99269   
  
Coefficients:  
 Estimate Std. Error t value Pr(>|t|)   
(Intercept) -0.537960 0.168809 -3.187 0.00193 \*\*   
age 0.021811 0.003679 5.929 4.57e-08 \*\*\*  
---  
Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
  
Residual standard error: 0.429 on 98 degrees of freedom  
Multiple R-squared: 0.264, Adjusted R-squared: 0.2565   
F-statistic: 35.15 on 1 and 98 DF, p-value: 4.575e-08

[Section 10.2](#sec-logit-10)

* Using the Table below, please write down your answers to the following questions and be prepared to share them in class. For each question provide a verbal interpretation of the numerical answer (e.g, odds of 2 to 1 means that for every two people with a trait, there is one without).
  1. What is the probability of a person in their 40s having CHD?
  2. What are the odds of a person in their 40s having CHD?
  3. What is the probability of someone in their 50s **not** having CHD?
  4. What are the odds of someone in their 50s **not** having CHD?
  5. What is probability of having CHD in your 40s, compared to your 30s? (e.g., is 3 times higher? 4 times higher?)
  6. What are the odds of having CHD in your 40s, compared to your 30s?

knitr::include\_graphics("files/images/odds.png")

|  |
| --- |
| Proportions and odds |

* Using the Table below, please answer the following questions.
  + If probability of an event is equal to .1, what are the odds the event? What is the logit of the event?
  + If odds of an event are 4 to 1, what is the probability of the event? What is the logit of the event?
  + If logit < 0 then probability < ? and odds < ?
  + What is more likely: a event with probability of .9 or an event with odds of .9?
  + If a probability of corresponds to a of , what is the logit corresponding to ? (Hint, try some numerical examples from the table).

|  |
| --- |
| Logistic, odds, and logit |

[Section 10.3](#sec-simple-10)

* Please practice your interpretation of the OR in simple logistic regression using the following examples:
  + If what is the OR equal to? What is the percent change in the odds for a one unit increase in ?
  + If what is the OR equal to? What is the percent change in the odds for a one unit increase in ?
  + If what is the OR equal to? What is the percent change in the odds for a 10 unit increase in ?
  + If the odds increase 100% for a one unit increase in , what equal to?
* For the CHD example (model given below), please write down the numerical values of each of the summaries listed below. You can select any values of age to report the predicted probabilities and risk ratios, and you can “eye ball” the probabilities using [Figure 10.9](#fig-pred-prob).
  + In terms of the logit.
  + In terms of the odds.
  + In terms of predicted probabilities.
    - Risk ratios / relative risk for some chosen ages.
    - The age at which the probability of having CHD is “50-50”.
    - The approximate rate of increase (slope) of the probability of having CHD for people close to the “50-50” point.

[Section 10.4](#sec-estimation-10)

To write the likelihood for person , we use the notation

and to write the likelihood for the full sample, we use use without a subscript:

If we plug into the last equation using the values of in our sample, this gives us the likelihood of observing the sample. For example, the values of CHD for the first four cases in the example are (1, 1, 0, 1). So, the first four terms in the likelihood are:

Please take a moment to verify this result for yourself and we will walk through the derivation together in class.

[Section 10.5](#sec-inference-10)

* The output below shows the odds ratios and confidence intervals for the example data. In APA notation we could write the results for age as . Please write down your interpretation of the OR for age and its confidence interval. In your interpretation, please mention the relative change in the odds that is associated with each additional year of age and whether or not this is statistically different from zero.

ci.b <- confint(mod2)  
ci.table <- cbind(coefs = coef(mod2), ci.b)  
exp(ci.table)

coefs 2.5 % 97.5 %  
(Intercept) 0.004944629 0.0004412621 0.0389236  
age 1.117306795 1.0692223156 1.1758681

* To conduct an LR test of the model with age as a predictor against a model without this predictor (i.e., the model with only the intercept), we can use the lrtest function of the lmtest package to obtain the following output. In APA format, we could write this output as Please write down your interpretation of the LR test for the example. Your interpretation should mention which two models were compared, whether the LR test was significant or not, and your conclusion about which model provided a better explanation of the data.

# The model with no predictors  
mod0 <- glm(chd ~ 1, family = "binomial", data = chd.data)  
  
# The LR test (enter the smaller model first)  
lmtest::lrtest(mod0, mod2)

Likelihood ratio test  
  
Model 1: chd ~ 1  
Model 2: chd ~ age  
 #Df LogLik Df Chisq Pr(>Chisq)   
1 1 -68.331   
2 2 -53.677 1 29.31 6.168e-08 \*\*\*  
---  
Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

[Section 10.6](#sec-pseudo-rsquared-10)

* The following output shows McFadden’s R-squared for the example data (i.e., comparing a model with age as a predictor to a model with just the intercept). Please write down an interpretation of the R-square value and we will discuss you interpretations in class.

# Using the deviance of the models  
1 - mod2$deviance/mod0$deviance

[1] 0.2144684

[Section 10.7](#sec-assumption-checking-10)

The HL test for the example data is reported below. Using APA notation, we could write the results of the test as . Please write down your interpretation of the HL test and be prepared to share you answer in class. Your interpretation should mention whether the assumption that the logistic model fits the data is problematic or not.

## Note: Install package "generalhoslem" if you haven't do so already  
# install.packages("generalhoslem")  
  
hl\_test <- generalhoslem::logitgof(obs = chd, exp = fitted(mod2), g = 5)  
hl\_test

Hosmer and Lemeshow test (binary model)  
  
data: chd, fitted(mod2)  
X-squared = 0.046893, df = 3, p-value = 0.9973

## 10.9 Exercises: Multiple logistic regression

These exercises are a bit different than usual. Rather than repeating the code that produced the previously discussed example, we extend the example to illustrate multiple logistic regression. The general idea is that, everything you can do in OLS linear regression you can also do in logistic regression, but some of the details are different.

We will go through this material in class together, so you don’t need to work on it before class (but you can if you want.) Before staring this section, you may find it useful to scroll to the top of the page, click on the “</> Code” menu, and select “Show All Code.”

### 10.9.1 CHD and smoking

This section works through multiple logistic regression with interactions using the CHD data again. Coronary Heart Disease (chd; 1 = yes, 0 = no) is still the outcome, and we predict it using a person’s age in years (age) as well as an indicator for whether a person has ever (or currently) smokes (smokes: 1 = yes, 0 = no).

# Load the data, take a look  
#load("CHD.RData")  
#attach(chd.data)  
  
summary(cbind(chd, age, smokes))

chd age smokes   
 Min. :0.00 Min. :20.00 Min. :0.00   
 1st Qu.:0.00 1st Qu.:34.75 1st Qu.:0.00   
 Median :0.00 Median :44.00 Median :0.00   
 Mean :0.43 Mean :44.38 Mean :0.42   
 3rd Qu.:1.00 3rd Qu.:55.00 3rd Qu.:1.00   
 Max. :1.00 Max. :69.00 Max. :1.00

Let’s start by regressing CHD on smoking and age, and refreshing our interpretations of the parameter estimates and tests. The regression coefficients are interpreted analogously to multiple regression (e.g., as holding constant or controlling for the other predictors).

# Regress CHD on smokes and age  
mod <- glm(chd ~ age + smokes, family = binomial)  
summary(mod)

Call:  
glm(formula = chd ~ age + smokes, family = binomial)  
  
Coefficients:  
 Estimate Std. Error z value Pr(>|z|)   
(Intercept) -6.72152 1.33396 -5.039 4.68e-07 \*\*\*  
age 0.11849 0.02639 4.489 7.16e-06 \*\*\*  
smokes 2.54953 0.59260 4.302 1.69e-05 \*\*\*  
---  
Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
  
(Dispersion parameter for binomial family taken to be 1)  
  
 Null deviance: 136.663 on 99 degrees of freedom  
Residual deviance: 84.004 on 97 degrees of freedom  
AIC: 90.004  
  
Number of Fisher Scoring iterations: 5

For illustrative purposes, let’s double check that smoking predicts CHD over and above age using the LR test. Recall that this test is analogous to the F-test of R-squared change in OLS regression

# LR test coefficient on age, and age + smokes   
mod1 <- glm(chd ~ age, family = binomial)  
mod2 <- glm(chd ~ age + smokes, family = binomial)  
lmtest::lrtest(mod1, mod2)

Likelihood ratio test  
  
Model 1: chd ~ age  
Model 2: chd ~ age + smokes  
 #Df LogLik Df Chisq Pr(>Chisq)   
1 2 -53.677   
2 3 -42.002 1 23.349 1.351e-06 \*\*\*  
---  
Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

Both the Wald test and LR test lead to the same overall interpretation for smoking.

How should we compute R-squared for the two-predictor model? Two different approaches are shown below. Both of these are defensible, but note the difference in interpretation as we change the “baseline model”.

# Compare against model 1  
(R\_21 <- 1 - mod2$deviance/mod1$deviance)

[1] 0.2174937

# Compare against model with only the intercept  
mod0 <- glm(chd ~ 1, family = binomial)  
(R\_20 <- 1 - mod2$deviance/mod0$deviance)

[1] 0.3853166

Recall that it is common to report the output using odds ratios rather than logits. Make sure to check your interpretation of the ORs.

exp(cbind(OR = coef(mod2), confint(mod2)))

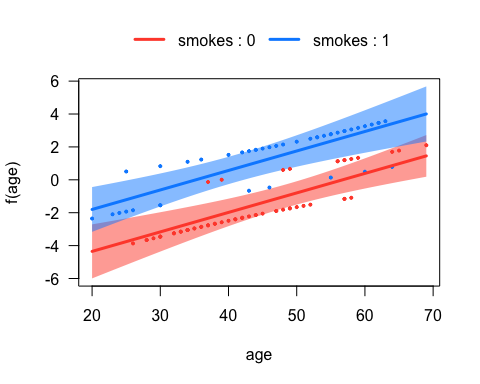
Waiting for profiling to be done...

OR 2.5 % 97.5 %  
(Intercept) 0.001204708 6.635861e-05 0.01314643  
age 1.125791093 1.073105e+00 1.19145228  
smokes 12.801117760 4.292057e+00 45.11382912

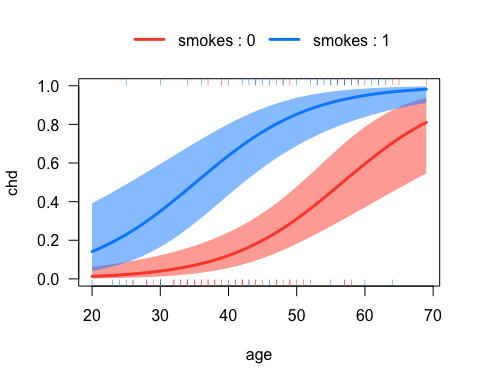
### 10.9.2 Predicted probabilities

We can plot the regression lines for smokers and non-smokers using the same techniques as for linear regression.

# Plots of the regression lines in logit scale  
visreg::visreg(mod2,   
 xvar = "age",   
 by = "smokes",   
 scale = "linear",   
 overlay = T)



# Plots of the regression lines in probability scale  
visreg::visreg(mod2,   
 xvar = "age",   
 by = "smokes",   
 scale = "response",   
 overlay = T)



We can also test for differences between smokers and non-smokers at specific ages, just like we did for the OLS regression. Below is an example of using emmeans to compute the risk ratio associated with smoking at +/- 1SD on age.

# Step 1. Make a function to compute +/- SD on age  
sd\_function <- function(x) { c(mean(x) - sd(x), mean(x) + sd(x)) }  
  
# Step 2. Compute and test the values on the odds scale  
library(emmeans)  
gap\_odds <- emmeans(mod2,   
 specs = "smokes",   
 by = "age",   
 cov.reduce = sd\_function,   
 type = "response")  
  
# Convert odds to probabilities  
gap\_prob <- regrid(gap\_odds, "log")  
  
# Compute risk ratios for smoking, at different ages.  
contrast(gap\_prob, method = "revpairwise")

age = 32.6586734987249:  
 contrast ratio SE df null z.ratio p.value  
 smokes1 / smokes0 7.79 3.898 Inf 1 4.100 <.0001  
  
age = 56.1013265012751:  
 contrast ratio SE df null z.ratio p.value  
 smokes1 / smokes0 1.92 0.388 Inf 1 3.214 0.0013  
  
Tests are performed on the log scale

### 10.9.3 Interactions

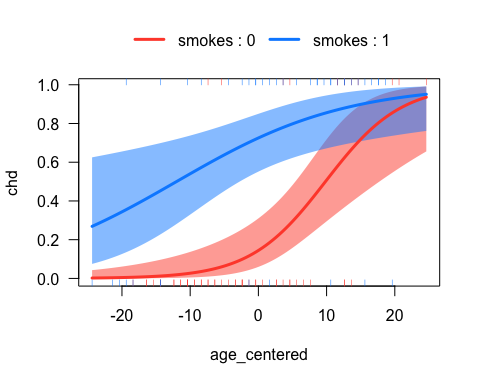
Next, let’s consider the interaction between age (centered) and smoking. This interaction addresses whether the relationship between age and CHD changes as a function of smoking status.

# Interacting smoking with age (centered)  
age\_centered <- age - mean(age)  
  
mod4 <- glm(chd ~ age\_centered\*smokes, family = binomial)  
exp(cbind(OR = coef(mod4), confint(mod4)))

Waiting for profiling to be done...

OR 2.5 % 97.5 %  
(Intercept) 0.1686540 0.05274623 0.3977169  
age\_centered 1.1984690 1.10011853 1.3473795  
smokes 15.6504935 4.96230836 62.2339766  
age\_centered:smokes 0.9046719 0.79508393 1.0087164

visreg::visreg(mod4,   
 xvar = "age\_centered",   
 by = "smokes",   
 scale = "response",   
 overlay = T)



detach(chd.data)

We will talk about the interpretation of this model in class.