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 3 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 3 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 just learn about the results as they become relevant and skip the math, 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 R-squared 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:

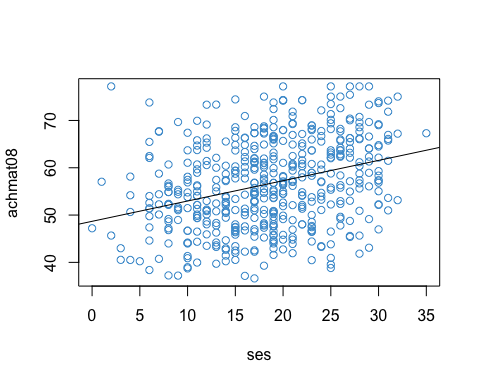
Y ~ X

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

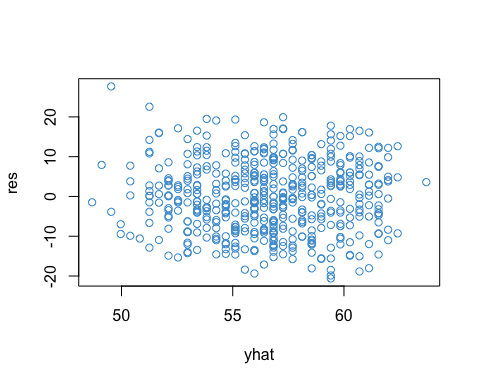
# Also check that R-squared is really equal to the square of the PPMC  
cor(achmat08, ses)^2

[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 interpretation of the regression coefficients correct. 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:

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.

Also note that we can see in the equations for 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 coefficients 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 **?@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 **?@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.

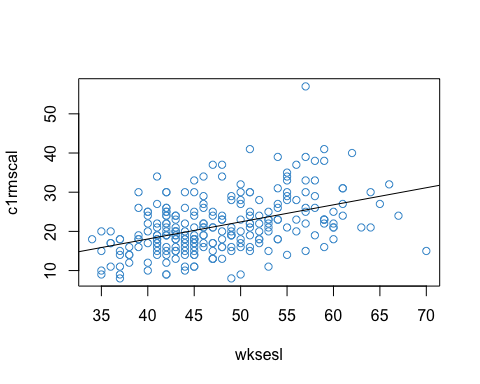
* Variable names begin with c, p, or t depending on whether the respondent was the child, parent, or teacher. Variables that start with wk were created by the ECLS using other data sources available in during the kindergarten year of the study.
* The time points (1-4 denoting fall and spring of K and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

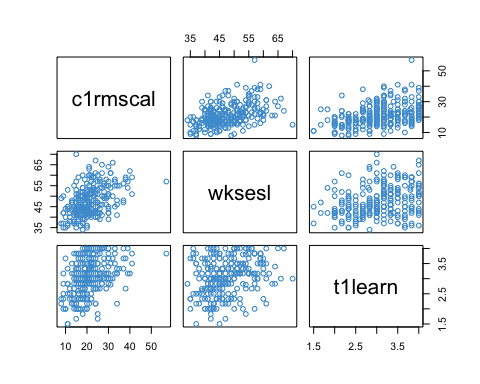
### 3.9.2 Multiple regression with lm

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
* …

This chapter will focus on the topic of “contrast coding” (also called “effect coding” or “dummy coding”). In this context, the term “contrast” means that we want to compare two (or more) groups of respondents in terms of their average levels on the outcome variable. For example, do urban students and rural students differ in their average levels of academic achievement?

In particular, we will address:

* 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 applied to 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 challenging to understand because, depending on the contrast coding used, the model results can appear quite different.

For example, the two models below uses 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.

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

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 the 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 so-called 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>.

In general, 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 [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. |

In this situation, the simple regression equation from Section @ref(regression-line-2) still holds

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 interpretations of the regression coefficients that apply in this situation. In this section we are interested in the this “more specific” interpretation of regression cofficients when is dichotomous.

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. This details, as we as some other interesting results on categorical variables, are shown in [Section 4.7](#sec-categorical-results-4)

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 the 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 the next chapter.

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 groups.

## 4.4 Reference-group coding

Now that we know how to do regression with a binary predictor works, let’s consider how to extend this approach to categorical predictors with categories. There are many ways to do this, and the general topic is variously called “contrast coding”, “effect coding”, or “dummy coding”.

The basic idea is to represent the categories of a 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.

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 coefficients 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 [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 greater than 2 groups.

### 4.4.1 A hypothetical example

Figure [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 (never married)
* 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 model parameters in the following regression equation:**
* **What happens if we include 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 dummy left out of the dummies used in 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 (e.g., in experimental conditions, comparisons are made 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, 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 an (arbitrarily chosen) reference-group, but instead represents the mean of the predicted values. When the groups have equal sample size, the deviation-coded intercept is also equal to overall mean on .

To summarize 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 indicated 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 the Section [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 grou 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 [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 the overall mean of the variable. This is not always a problem. For example, in the IDELA example, it is reasonable that each country should receive equal weight, even if the size of the samples in each country differ.

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. So, 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 all groups included\*

Another issue with deviation coding is that it requires leaving one group out of the model. This is a shortcoming of the approach. 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.

* Step A: Standardize the variable to have .
* Step B: Compute binary dummy variables (reference-group coding) for all groups,
* Step C: Regress on the 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 A), the group means can be interpreted as deviations from the overall mean on .

Note that to omit the intercept in R, you can use the formula syntax

Y ~ -1 + X1 + ...

in the lm function (the -1 removes the intercept from the model).

Again, keep in mind that 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 paragraph explains why, but it requires some material we won’t discuss until **?@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. In the present context, the focal models fit the data the same, regardless of whether we use the intercept and predictors or omit the intercept and use predictors. However, the reference model used by lm depends on whether the intercept is included in the model. In the model with the intercept, the reference model also has an intercept. In the model that omits the intercept, the reference also omits the intercept. Consequently, the F-statistic for the test of R-squared (as reported by summary(lm)) is different for the two approaches. You can get the “correct” R-squared from R, 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 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 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 the [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 use contrast coding and just report the F-test along with the test of the individual effects, without any discussion of familywise error rate control. Which approach is correct? Depends who you ask….

## 4.7 Some details 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 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 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 model parameters in the following regression equation:
* What happens if we include 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.

### 4.9.1 A single binary predictor

Regression with a single binary predictor is equivalent to an independent groups t-test of a difference between means. Let’s illustrate this using reading achievement in grade 8 (achrdg08) and gender in the NELS dataset. Note that, although the construct gender need not be conceptualized as dichotomous or even categorical, the variable gender reported in NELS data is dichotomous, with values “Female” and “Male”.

R treats variables like gender as “factors”. Factors have special properties that we are going to work with later on, but for now let’s recode gender to binary\_gender by setting Female = 0 and Male = 1

### 4.9.2 Recoding a factor to numeric

# 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

It is often hassle to get from factor to numeric or vice versa. Above we used the trick of first creating from the factor a logical vector (gender == "Male") and then coercing the logical vector to binary by multiplying it by 1. Other strategies can be used. See help(factor) for more information.

### 4.9.3 Binary predictors and independent samples t-tests

Now let’s get back to regressing reading achievement (achrdg08) on binary\_gender, and comparing this to an independent groups 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 mean difference (except for the sign):

In summary, a t-test of a b-weight of a binary predictor is equal equivalent to a t-test of the mean difference.

### 4.9.4 Reference-group coding

As discussed in Section @ref(reference-group-coding-5), 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 factor.

### 4.9.5 More about factors

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 "constrasts"  
contrasts(urban)

Rural Suburban  
Rural 1 0  
Suburban 0 1  
Urban -1 -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 back 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. levels should be assigned a text vector with length equal to the number of levels of the variable. The entries of the assigned vector will be the new level names of the factor.

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

[1] "Female" "Male"

### 4.9.6 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) 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

In the output, we see that two regression coefficients are reported, one for Suburban and one for Urban. As discussed in Section @ref(reference-group-coding-5), 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.

### 4.9.7 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)

Rural Suburban  
Rural 1 0  
Suburban 0 1  
Urban -1 -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 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.

### 4.9.8 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.

### 4.9.9 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) 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

If we compare the output from aov to the F-test reported by lm we see that the F-stat, 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 omega-squared and the ANOVA omnibus F-test is the same as the F-test of R-squared. The main difference is that lm focuses on contrasts for analyzing and interpreting the group differences, whereas ANOVA focuses on the F-test of the omnibus hypothesis and the procedures for analyzing group difference are conducted as a follow-up step.

### 4.9.10 Extra: Group-mean coding

As a step towards addresses the issues with deviation coding, let’s consider another coding procedure. If we omit the intercept term, all of the reference-group coded dummies can be included 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 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. When reporting R-squared and its F-test, we should use a model with the intercept.

By itself, group mean coding is not very interesting uninteresting – we don’t usually want to test whether the group means are different from zero. However, we will see in the next section that it can be used to provide an alternative to deviation coding.

### 4.9.11 Extra: Weighted versus unweighted deviation coding

This section presents a “hack” for addressing the two main issues with deviation coding noted in Section @ref(deviation-coding-5). This is a hack in the sense that we are working around R’s usual procedures rather than replacing them with a completely new procedure. The result of this hack is to provide deviation coding in which comparisons are made to grand mean on the outcome for all categories, not just categories. As a side effect, the F-test for the R-squared statistic is no longer correct, so you should not use this approach to report R-squared.

First, note that the group sample sizes are not equal for ubran

table(urban)

urban  
 Rural Suburban Urban   
 162 215 123

Because of this, the unweighted mean of the group means (i.e., the intercept in mod4 above) is not equal to the overall mean on achrdg08 – we can see that the overall mean and the unweighted group means are slightly different for these data:

group\_means <- tapply(achrdg08, INDEX = urban, FUN = mean)  
group\_means

Rural Suburban Urban   
54.99265 55.66019 58.12016

# Compare the overall mean and the unweighted mean of the group means  
# Center the outcome variable  
ybar <- mean(achrdg08)  
ybar

[1] 56.04906

mean(group\_means)

[1] 56.25767

Note that the intercept in mod4 is not equal to the overall mean, ybar, but is instead equal to the unweighted average of the group means, mean(group\_means). the difference isnt very big in this example, but it can be quite drastic with highly unequal sample sizes.

If you would like to compare the groups to the overall mean when the sample sizes are unequal, the simplest way to do this in R is by first centering the Y variable and then using group mean coding.

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.

# Change the contrasts back to reference group  
contrasts(urban) <- contr.treatment(n = 3, base = 1)  
colnames(contrasts(urban)) <- c("Suburban", "Urban")  
  
# Center the outcome variable  
dev\_achrdg08 <- achrdg08 - ybar  
  
# 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 Y variable is centered, but the F test **does** change from mod2, because there are now 3 variables included in the model. In general, when the intercept is omitted, R-square and its F-test do not have the usual interpretation. So, when reporting R-squared and F-test, we should use a model with the intercept, rather than the approach outlined here.