Bayesian multilevel models for repeated-measures data: A conceptual and practical introduction in R

Santiago Bareda and Noah Silbert

2022-01-19

# Preface

–

## 0.1 How to use this book

–

## 0.2 What this book is and what this book isn’t

–

### 0.2.1 It takes a village of books

–

## 0.3 What you need to know to use this book

–

## 0.4 What you need installed to ue this book

–

devtools::install\_github("santiagobarreda/bmmb")

## 0.5 Why go Bayesian?

–

## 0.6 Statistics as Procedural knowledge

–

# 1 Introduction: Experiments and Variables

Each chapter of this book will involve the analysis or discussion of data resulting from two perceptual experiment carried out by a group of fifteen listeners. The first 8 chapters will focus on an experiment investigating the perception of height from speech, and a second experiment investigating vowel perception will be introduced in Chapter 9. As noted in the preface, a basic working knowledge of R is assumed and a familiarity with basic statistics is helpful, though not strictly necessary. The preface also provides suggested readings for those wanting to do some background reading on R or statistical inference. In this chapter we will discuss the data for the first experiment, introduce some basic concepts related to variables and probabilities, and provide a very basic introduction to R along the way. For information about the software you need installed to follow along with the examples in the chapter, please see section X.

## 1.1 Experiments and effects

An experiment is a procedure or process than can help answer some research question. Obviously, when defined so broadly almost anything can be an experiment. In fact, when a child touches a hot stove to see what it feels like, they are conducting an experiment which provides essential information about their world. In an academic context, experiments are expected to be scientific. However, there is no definition of scientific that transcends space and time. What is considered ‘scientific’ is determined by what scientists in a specific time and place consider to be scientific, and this can change, and has changed, substantially over time. At the moment, in most contexts, a research project is ‘scientific’ when it generally conforms the scientific method. Of course, just as with science and scientific, there is no single scientific-method, no single true definition that can be referred to. Instead, the scientific method consists generally of a process in which researchers: 1) Ask questions based on gaps in their knowledge about the world, 2) Collect data using procedures developed to avoid certain pitfalls and maximize the chance that the collected data can answer their questions, 3) Evaluate their questions in light of their data, and 4) Reach conclusions where possible, synthesize their conclusions with their previous knowledge about the world.

Modern ‘scientific’ work usually involves the collection of empirical measurements, the quantification of patterns in these measurements, and the qualitative description of the quantitative patterns in the measurements. As a result, much modern scientific work yields large quantities of numeric values, observed under different conditions, which the researcher must then (statistically) analyze in order to understand. For example, imagine an experiment about whether caffeine makes people talk faster. Subjects are asked to drink either a cup of regular or decaffeinated coffee. After a 30-minute wait they are asked to read a passage aloud and the duration of the reading is measured. Basically we are measuring two different values, “the amount of time it takes people to read a passage of text after drinking decaf coffee”, and “the amount of time it takes people to read this passage of text after drinking regular coffee”. Our experiment allows us to ask: is “the amount of time it takes people to read this passage of text after drinking decaf” the same thing as “the amount of time it takes people to read this passage of text after drinking regular coffee”? Another way to look at this is that we are interested in the effect of caffeine on reading times. By ‘effect’ we mean the degree to which caffeine changes the characteristics of our observation (reading times) in some way. For example, if the average reading times were the same in both groups we would conclude that “caffeine has no effect on reading times”. In contrast, if reading times were 800 milliseconds seconds shorter in the caffeine group, we might conclude “caffeine has the effect of reducing reading times by 800 milliseconds”.

Our experiment on reading times is specifically constructed to investigate the effect of caffeine on readings times. If the speakers in our experiment were randomly assigned to conditions, there is no particular reason to expect that their reading times would be different in the absence of the caffeine. So, if we find that people read faster in the caffeine group, we may infer that it is the caffeine that has had the effect of causing the increase in speaking rate. This same logic applies in situations where we do not randomly assign subjects to groups, as long as we are careful in creating equivalent groups. Consider the same experiment about speaking rate carried out with groups based on speaker gender rather than drinking coffee. In this case the question would be “is the amount of time it takes men to read this passage of text the same amount of time that it takes women to read this passage of text”. If the speakers are generally similar in important characteristics (e.g., dialect, age, cultural background) apart from gender, then any group differences may be attributable to the effect of gender on speaking rate.

What we are describing in the above paragraph are controlled experiments, experiments where the researcher takes an active role in ensuring the ‘fairness’ of the experiment. The notions of control and fairness and somewhat hazy, and are perhaps more gradient than discrete (i.e. ‘controlled’ vs. ‘uncontrolled’). However, some situations clearly do not lead to ‘fair’ outcomes. For example, what if the caffeine group of readers were all first language English speakers, and the decaf group had substantial number of second language speakers. The caffeine group may very well read faster simply because they are more polished readers, independent of any effects of caffeine. Whenever possible, researchers avoid situations like this by exerting control over their experiments, both in the structure of their experiments and in the recruiting and assignment of their participants to experimental conditions.

All of the experiments discussed to this point would result in one (or more) reading time per subject per group. Due to random between-speaker variation (among other things), there is no chance whatsoever that the average readings times across both groups will be exactly identical, even if caffeine has no effect on reading times at all. Actually, if you re-ran the experiment, there is basically no chance that each group’s mean reading time would match each group’s mean exactly in the replication. And yet, there is the possibility that caffeinated reading times are systematically different - i.e., different in a way that the random variation of groups across replications is not. So, how can we ever establish that our measures are actually different and don’t just appear to be different because of randomness? It is precisely this problem that has motivated scientists to use statistical analyses to help answer their research questions.

## 1.2 Experiments and inference

This book is about statistical inference. We will talk about the ‘statistics’ part in more detail in the next chapter, but we can talk about the ‘inference’ part now. Inference is a form of reasoning that allows you to go from a limited number of observations to a general conclusion. For example, you may arrive at a newly discovered island and see white cats wandering around. If you are there for a while and continue to observe only white cats, you may conclude “all the cats on this island are white”. If you do this you have made what is called an inductive inference: You have gone from a set of observations (the cats you saw) to a general conclusion about all the cats on the island. Often, experiments are not just about observing and measuring certain effects, but also about drawing inferences regarding those effects. For example, in the reading time experiment described above the researchers are not specifically interested in the reading times of the people in the experiments (i.e., the cats they saw) but rather about the reading times of people more generally (i.e. all the cats on the island).

Since inductive inference seeks to go from limited observation to general rules or principles, it has a central weakness. For example, your inference that only white cats exist on the island is on solid ground until you see a cat that is not white. Can you be sure this won’t happen? You can’t, because fundamentally you don’t know what you don’t know and you can’t be sure that what hasn’t happened yet will never happen. This is called the problem of induction and it is a fundamental weakness of inductive reasoning.

It’s useful to be aware of the fundamental limitations of trying to understand general patterns given limited sets of observations, and to be generally skeptical. It is also useful to think about how we can reason in a way that might minimize the odds of inferential mistakes, especially by including our general knowledge of the world (and the specific topic) in our reasoning. For example, rather than observing white cats and leaving it at that we can ask: Why are the cats white? Do evolutionary pressures cause them to be white? How do their genetics ensure that all members of the species will be white? Is there any chance non-white cats could enter into the population? Considering the answers to questions like this, in combination with our observations, can make inferences like “all cats on this island are white” more reliable.

For example, the examples above involved the effect of caffeine on reading times. We are interested in generalizing to the human population based on what is a tiny sample of humans (relatively speaking). If we make the claim “caffeine speeds up reading times”, are we extending that to all humans, or at least to all English speakers? Past, present and future? That is a bold claim based on a small number of data points, or it would be in the total absence of any world knowledge and prior expectations. Of course, we know that caffeine is a stimulant and seems reasonably likely to make people read faster. As a result, the finding fits within our larger world view and, as a result, we may accept as likely to be ‘true’. In contrast, suppose that the two groups had instead drank plain water, one ‘regular’ and one dyed with blue food coloring. In this situation we may be skeptical of any finding for an effect for the food coloring. This is because there is no reason to suppose that there is an effect. Since this finding does not conform to any prior knowledge about the world, it is the sort of inference that may turn out to be less reliable, in the long run.

## 1.3 Our experiment

As noted above, each chapter in this book will feature the analysis of data from a perceptual experiment. In this section we provide information about the experiment in general, the design of the experiment, the general research questions this experiment can address, and an overview of the data resulting from our experiment.

### 1.3.1 Our experiment: Introduction

Any group of speakers will ‘sound’ different from each other even when they are all saying the ‘same’ word. These between-speaker differences can, in some cases, be systematically associated with speaker characteristics such as age, height, and gender. So, tall speakers may tend to sound one way, while shorter speakers may tend to sound different in particular ways. As a result, although it may sound odd to talk about how tall someone sounds, listeners are able to use the information in a speaker’s voice to guess information like this about the speaker. We call this information the speaker’s indexical characteristics, social and physical information regarding the speaker that is understood from the way someone speaks. We can ask two different questions with respect to assessments of indexical characteristics from speech: 1) Are they accurate, and 2) How do listeners arrive at their guesses? Generally speaking, listeners are often not very accurate in their judgments of indexical characteristics, however, they are very consistent in the errors that they tend to make. For example, if one voice is incorrectly assumed to belong to a particular sort of speaker, it will often be the case that this mistake is a regular occurrence.

Generally, the ‘guessing’ of speaker characteristics is dominated by two acoustic cues: Voice pitch and voice resonance. Voice pitch can be thought of as the ‘note’ someone produces with their speech. When you sing you produce different notes by producing different pitches. The pitch of a sound is related to the vibration rate of the thing that produced the sound, because repetitive vibration produces a repetitive sound wave that humans perceive as tonality. Human voice pitch is regulated by changing the vibration rate of the vocal folds in your larynx. You can feel this vibration if you hum a song and press your fingers against the middle of the front of your neck. Pitch is an auditory sensation, a feeling you have in relation to a periodic acoustic event, a sound. When you hear two sounds, you can order them based on which sounds lower/higher than the other. That’s pitch. Since this quality cannot be directly measured, scientists measure the fundamental frequency (f0) of the sound to quantify its pitch. The f0 of a sound is measured in Hertz (Hz), which measures how many times a sound repeats itself in a given second.

Generally speaking, smaller things tend vibrate at higher rates than larger things. This holds for vocal folds as well; shorter vocal folds tend to vibrate at higher rates than do longer vocal folds, thereby producing speech with a higher pitch. As a result, generally speaking, larger speakers tend to produce speech with a lower pitch. Since the vocal folds generally grow as one ages into adulthood, voice pitch may be an indicator of age between young childhood and adulthood. What we mean is that pitch may be able to help you distinguish a 5 year old from an 18 year old but maybe not an 18 year old from a 30 year old. In addition to general age-related changes, the vocal folds tend to increase in size quite a bit during male puberty so that post-pubescent males tend to produce speech with a lower pitch than the rest of the human population. As a result of these relations, a voice with a lower voice pitch is more likely to be produced by someone who is older, taller, and more male than a voice with a higher pitch. The relationships between age, height, gender and f0 are presented in Figure 1.1

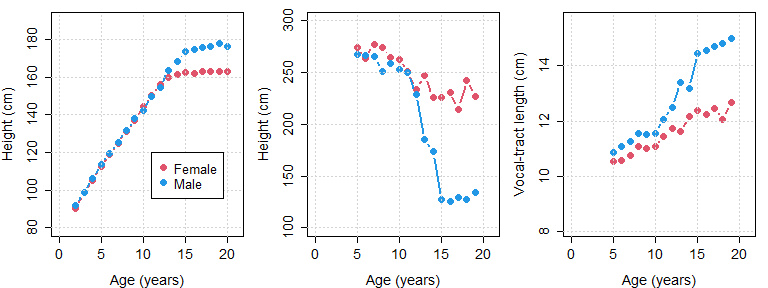


Figure 1.1: (left) Average height of males and females in the United states of America, organized by age (cite). (middle) Estimated vocal-tract length for male and female speakers between from 5 years of age until adulthood, based on the acoustic data provided in Lee at al. (?). (right) Average f0 produced by male and female speakers between from 5 years of age until adulthood.

Resonance can be thought of as the ‘size’ of a sound. For example, a violin and a cello can be playing the same note (with the same pitch), but a cello ‘sounds’ bigger. This is because lower frequencies resonate in its larger structure . In the same way, speakers with longer vocal tracts (the space from the vocal folds to the lips) tend to ‘sound’ bigger by producing speech with lower frequencies overall. We don’t really have good words to describe what resonance ‘sounds’ like, but a small resonance (short vocal tract) sounds ‘heliumy’. When a person breathes helium and speaks, their speech does not go up, but their resonance frequencies increase (for more information on this, see the Appendix). Long vocal tracts sound more like slow motion speech (think of someone saying “noooooooooooooooooo….” when something bad is happening in slow motion in a movie), and this is because slowing down the playback of a recording simulates a lowering of resonance frequencies in speech. In fact, size simulation by resonance manipulation is how the recordings for ‘Alvin and the chipmunks’ were originally created. A low-pitched male singer was recorded singing abnormally slow, and the recording was sped up in order to simulate a speech with a very high resonance (and an associated very short vocal tract).

There are many ways to measure the resonance of a voice. In our data we will use speech acoustics to directly estimate the length of the vocal tract that produced it, in centimeters (in the manner described in the Appendix). So, our measure of voice resonance will not be acoustic at all but will instead measure the physical correlate of the vocal tract expected to have produced the speech sound. In general, lower frequencies overall suggests a lower voice resonance which in turn suggests a longer vocal tract length in centimeters. There is a strong positive relationship between vocal-tract length and body length (i.e. height) across the entire human population. This means that the taller a person is, their vocal-tract is expected to be longer and their voice resonance is expected to be lower. Since height increases from birth into adulthood, this means that voice resonance can be used to predict both height and age. In addition, adult males tend to be somewhat taller than adult females in most populations, with an average difference of about 15 cm in the United States. As a result, voice resonance can be used to infer the gender of adult speakers, and possibly that of children as well. These relationships are shown in Figure 1.1.

So, voice pitch and voice size are independent ways that someone can acoustically ‘sound’ bigger/smaller, older/younger, and male/female. The experiment to be described below involves a perceptual experiment involving behavioral measures. This means that in this experiment human listeners listened to auditory stimuli (words) and were asked to listen to them and answer questions regarding what they heard. The experiment was designed to investigate the way that speech acoustics are used by listeners to determine the age, gender, and height of speakers, and the way that these decisions affect each other.

### 1.3.2 Our experimental methods

Our listeners were 15 native speakers of American English. Listeners were presented with the word “heed” produced by 139 different speakers of Michigan English. These speech samples were recorded by Hillenbrand et al (1995) and are available on the GitHub page associated with this book. So, this experiment featured 139 unique stimulus sounds that the listeners in the experiment were asked to respond to. The stimuli used were productions by 48 adult females, 45 adult males, 19 girls (10-12 years of age), and 27 boys (10-12 years of age). These speakers showed substantial variation in their voice pitch and resonance as measured by their f0 and estimated vocal-tract length (as will be discussed in section 1.5). In addition to the natural acoustic variation that exists between speakers, voice resonance was also manipulated experimentally. All stimuli were manipulated by shifting the spectral envelope down by 10%, simulating an increase in speaker size of approximately 10%. This acoustic manipulation is similar to the one carried out to make voices such as those of ‘Alvin and the Chipmunks’ sound small, but in reverse. (and pitch was not affected) By manipulating the spectral envelope of each word, we created two versions, the original and a manipulated version intended to ‘sound bigger’.

Each listener responded to all 278 stimuli (139 speakers x 2 resonance levels), for a total of 4170 observations across all listeners (15 listeners x 278 stimuli). Stimuli were presented one at a time, randomized along all stimulus dimensions. This means that tokens were thrown in one big pile and selected at random in a way that a stimulus was never predictable based on the previous one. For each trial, listeners were presented with a single word at random and were asked to:

* Indicate whether they thought the speaker was a “boy 10-12 years old”, a “girl 10-12 years old”, a “man 18+ years old”, or a “woman 18+ years old”. This is the **apparent speaker category**.
* Estimate the height of the speaker in feet and inches (converted to centimeters for this the discussion in this book). This is the **apparent speaker height**.

Our intention is to analyze the apparent height judgments provided by listeners in order to better understand them. To do this we will use acoustic descriptions of the different speakers’ voices, focusing on the fundamental frequency of their speech, and the vocal-tract length implied by their speech (estimated using the method described in the appendix). In addition, we will use the judgments made by listeners regarding the age and gender of the listener to better understand their use of acoustic characteristics when estimating speaker height.

### 1.3.3 Our research questions

This experiment is meant to investigate how listeners use speech acoustics to estimate the height of unknown talkers. Also, the results will let us investigate the possible relationship between the perception of talker size and the perception of talker category. Specific research questions will be discussed in each chapter, however, a general overview will be provided here. The expectations to be outlined below are based on the empirical relationships between these measurements and characteristics outlined above, and shown in Figure 1.1. The assumption is that listeners are familiar with the relationships between height and speech acoustics, and ‘somehow’ use the information in speech to guess the height of speakers. So, for example, if we know that a speaker with an f0 of 100 Hz is usually an adult male and is usually about 176 cm tall, we expect listeners will identify speech stimuli with an f0 near 100 Hz as produced adult male speakers who are about 176 cm tell.

Listeners were asked to provide two responses, speaker height and speaker group. The four speaker groups can be split according to two characteristics: The age of the group and the gender of the group (boy = male child, girl = female child, man = male adult, woman = female adult). So, we can consider that listeners reported the height, the age and the gender of the speaker, for each sound they listened to. In general, we expect that the perception of maleness will be associated with the perception of taller speakers, in particular for older speakers. The perception of adultness should be associated with taller speakers for either gender.

In terms of the acoustic variation in speaker voices, lower frequencies, whether f0 or resonances, are expected to be associated with taller and older speakers. For postpubescent speakers, low frequencies, particularly in f0, can also be an indicator of maleness. It is possible that the acoustic information in voices might be used differently based on the apparent category of the speaker. For example, maybe listeners used f0 one way when they think the speaker was an adult and another way when they think the speaker was a child. In addition, it is possible that different listeners useacoustic information in idiosyncratic ways that are systematic within-listener, but which differ arbitrarily from each other between listener.

### 1.3.4 Our experimental data

The data associated with this experiment is available in the bmmb package (discussed in section X), and can be accessed using the code below:

library ("bmmb")  
data (height\_exp)

The code above loads our data and places it into our workspace in an object called height\_exp. Below we use the head function see the first six lines of the data for the experiment. Our data is in long format so each row is a different individual observation and each column is a different piece of information regarding that observation. Each individual trial (a single row) represents an individual listener’s response to a single stimulus word played to them. So, we know that this data frame has 4170 rows to represent the 4170 observations in our data.

# see first 6 rows  
head (height\_exp)  
## L C height R S C\_v vtl f0 dur G A  
## 279 01 b 121.9 a 1 b 12.2 277 237 m c  
## 280 01 b 132.1 b 1 b 12.2 277 237 m c  
## 281 01 g 129.3 a 2 b 12.4 287 317 f c  
## 282 01 w 156.2 b 2 b 12.4 287 317 f a  
## 283 01 b 141.0 a 3 b 11.6 219 277 m c  
## 284 01 b 130.6 b 3 b 11.6 219 277 m c

If this were data that you collected and wanted to analyze, you would likely have it somewhere on your hard drive in a csv file, or some equivalent data file. If you were to open this data in Excel (or a similar software) you would see your data arranged in rows and columns. Below we write our data out as a csv file so that we can have a look at it outside of R.

write.csv (height\_exp, "height\_exp.csv", row.names = FALSE)

We can get more information about our data using the str function, which tells us that our data is stored in a data frame. A **data frame** is a collection of vectors that can be of different types, but which must be of the same length. A **vector** is a collection of elements of the same kind. Below, we see that the str function tells us about the vectors comprised by our data frame.

str (height\_exp)  
## 'data.frame': 4170 obs. of 11 variables:  
## $ L : chr "01" "01" "01" "01" ...  
## $ C : chr "b" "b" "g" "w" ...  
## $ height: num 122 132 129 156 141 ...  
## $ R : chr "a" "b" "a" "b" ...  
## $ S : num 1 1 2 2 3 3 4 4 5 5 ...  
## $ C\_v : chr "b" "b" "b" "b" ...  
## $ vtl : num 12.2 12.2 12.4 12.4 11.6 11.6 11.9 11.9 12.1 12.1 ...  
## $ f0 : int 277 277 287 287 219 219 260 260 244 244 ...  
## $ dur : int 237 237 317 317 277 277 318 318 242 242 ...  
## $ G : chr "m" "m" "f" "f" ...  
## $ A : chr "c" "c" "c" "a" ...

We see three kinds of vectors in our data: int indicating that the vector contains integers, num indicating that the vector contains floating point numbers (i.e. numbers that can have decimal points) numbers, and chr indicating that the vector contains elements made up of characters (i.e. letters or words), or numbers being treated as if they were letters (i.e. as symbols with no numeric value). For example our data contains a column called height that contains the numeric values 122, 132, 129, 156, 141, and so on. The information represented in each column is:

1. L: A number from 1-15 indicating which *listener* responded to the trial, being treated as a character.
2. C: A letter representing the speaker *category* (b=boy, g=girl, m=man, w=woman) reported by the listener for each trial.
3. height: A number representing the *height* (in centimeters) reported for the speaker on each trial.
4. R: A letter representing the *resonance* scaling for the stimulus on each trial. The coding is a (actual) for the unmodified resonance and b (big) for the modified resonance (intended to sound bigger).
5. S: A number from 1-139 indicating which *speaker* produced the trial stimulus.
6. C\_v: A letter representing the *veridical* (actual) speaker category (‘b’=boy, ‘g’=girl, ‘m’=man, ‘w’=woman) for each speaker for each trial.
7. vtl: An estimate of the speaker’s *vocal-tract length* in centimeters.
8. f0: The speaker’s average *fundamental frequency* (f0) measured in Hertz.
9. dur: The *duration* of the vowel sound, in milliseconds.
10. G: The *apparent gender* of the speaker indicated by the listener, f (female) or m (male).
11. A: The *apparent age* of the speaker indicated by the listener, a (adult) or c (child).

We can access the individual vectors that make up our data frame in many ways. One way is to add a $ after the name of our data frame, and then write the name of the vector after. This is shown below for our vector of heights.

height\_exp$height

Calling the command above will write out the entire vector to your screen, all 2780 observations of height responses that make up our data. Using the head function will show you the first six elements of an object, and you can get specific elements of the vector using brackets as shown below.

# show the first six  
head (height\_exp$height)  
## [1] 121.9 132.1 129.3 156.2 141.0 130.6  
  
# show the first element  
height\_exp$height[1]  
## [1] 121.9  
  
## show elements 2 to 6  
height\_exp$height[2:6]  
## [1] 132.1 129.3 156.2 141.0 130.6

Below, we use two sets of brackets to retrieve the height vector using its position in the data frame (first example), or its name (second example).

head( height\_exp[[3]] )  
## [1] 121.9 132.1 129.3 156.2 141.0 130.6  
  
head( height\_exp[["height"]] )  
## [1] 121.9 132.1 129.3 156.2 141.0 130.6

We can also retrieve the height vector by using a single set of parentheses as shown below. This method relies on treating the data frame as a matrix whose elements are arranged on a grid. Each element of the grid can then be accessed by providing x and y grid coordinates in single brackets as in [x,y]. Below we retrieve the entire third column by specifying a column number (or name) but leaving the row number unspecified.

head( height\_exp[,3] )  
## [1] 121.9 132.1 129.3 156.2 141.0 130.6  
  
head( height\_exp[,"height"] )  
## [1] 121.9 132.1 129.3 156.2 141.0 130.6

Below we use the same method to recover the entire first row of the data frame, and then the second element of the first row (or, from another perspective, the first element of the second column).

height\_exp[1,]  
## L C height R S C\_v vtl f0 dur G A  
## 279 01 b 121.9 a 1 b 12.2 277 237 m c  
height\_exp[1,2]  
## [1] "b"

## 1.4 Variables

Each of the columns in the height\_exp data frame can be thought of as a different variable. **Variables** are placeholders for some value, whether we know it or not. For example I can say “my weight is pounds”, or “this data represents a response provided by experimental subject ”. In our data, our variables take on different values from trial to trial, and the values of these variables tell us about the different outcomes and conditions associated with the trial. In this section we are going to discuss different aspects of variables, especially as they pertain to the analysis of experimental data.

### 1.4.1 Populations and samples

Anything that varies from observation to observation in an unpredictable manner can be modeled as a random variable. For example, your exact weight varies from day to day around your ‘average’ weight. In principle, you could probably explain exactly why your weight varies from day if you were so inclined. However, in practice you are probably not exactly sure why your weight is a bit higher one day and a bit lower the next. So, your weight is a random variable not necessarily because it is impossible to know why it varies, but simply because you don’t currently have the means to predict its value on any given observation.

In order to answer questions about reasonable values for variables of interest, scientists often collect measurements of that variable. These measurements can help us understand the most probable values of this variable, and the expected range of the variable, even if its value for any given observation is unpredictable. For example, although you may not know your exact weight on any given day, if you weigh yourself with some regularity you may have enough observations to have a pretty good idea of what your weight might be tomorrow. In addition, your expectation may be so strong that a large deviation from it would be more likely to result in your buying a new scale than believing the measurement.

A sample is a finite set of observations – measurements of a variable - that you actually have. A population is a (hypothetical) larger group of all possible observations that you are actually interested in. The population is the entire set of possible values of the random variable. For example, the population of “f0 produced by adult women in the United States” contains all possible values of f0 produced by the entire set of women from the United States. Our sample is the specific set of observations we have from our set of speakers.

Usually, a scientist will collect a sample to make inferences about the population. In other words, we are interested in the general behavior of the variable itself, not just of the small number of instances that we observed. For example, Hillenbrand et al. collected their data to make inferences about speakers of American English in general, and not because they were particularly interested in the specific speakers in their sample. Hillenbrand et al. collected speech samples from a relatively small sample of speakers to make inferences about the whole population of speakers in the United States. Similarly, we are not specifically interested in the opinions of the 15 listeners in our data, but about what their behavior might tell us about the population of human listeners in general.

### 1.4.2 Dependent and Independent Variables

We can make a very basic distinction between variables that we want to explain or understand, and variables that we *use* to explain and understand. The variables we want to explain are our **dependent variables**, they are usually the variables we measure or observe in an experiment. The variables that we use to explain and understand our measurements are our **independent variables** (sometimes called explanatory variables).

Dependent variables can often be **random**, which means their values are not knowable **a priori** (before observation). For example, you may have some expectation about what your weight might be before you get on a scale, but in general you can’t know exactly what it will say with certainty before collecting the observation. Although the exact values of our dependent variables can vary somewhat unpredictably from trial to trial, in the context of an experiment there is the general expectation that these values will *depend* in some way on the other variables in the experiment. For example, in this experiment we modified the stimuli so that some are expected to ‘sound’ bigger than others. As a result, the reported height we expect for any given trial *depends* on the value of the R (Resonance) variable in our data, among other things.

Variables that help predict the response (dependent) variables and are sometimes referred to as independent variables because their values are not considered to depend on those of the other predictors. More specifically, we can say the values of our independent variables are not assumed to depend on the values of the other variables in our experiment within the context of our experiment, or in a manner that directly relates to the relevant research questions.

Our experiment has two response variables: the apparent height (height) reported for each trial, and the apparent speaker category (C) reported for each trial. Our experiment also involves several variables that could be used to understand our responses (i.e. every other variable in the data). Whether a variable is dependent or independent depends on the research question and on the structure of the model more than on some inherent property of variables and data. For example, the data in height\_exp could be used to understand variation in voice pitch (f0) across speakers groups. In this case f0 would be the dependent variable and the veridical speaker category (C\_v) would be the independent variable. Another researcher may chose to model how perceived height varies as a function of f0 and speaker group. In this case height would be the dependent variable and f0 and C\_v would be the independent variables.

### 1.4.3 Categorical variables and ‘factors’

Categorical variables, also sometimes called nominal variables, are variables that take on some set of non-numeric, usually character values. Often, categorical variables are the labels that we apply to objects or groups of objects. For example, gender is a nominal variables with possible values of ‘male’ and ‘female’ among others. In our experiment data, C, S, L, R, and C\_v are nominal variables. Categorical predictors are often called factors. Factors can take on a limited number of values, called levels. For example if your factor is “word category” you factor levels may be “verb” and “noun” (among others). If your factor is “first language” your levels may be “Mandarin” and “Hindi”.

A factor is actually a data type in R. It’s very similar to a vector of words but it has some additional properties that are useful. For example, consider our C\_v predictor, which tells us which category each speaker falls into. Initially it is a character vector. We see that the first few tokens are produced by boys (b), and that there is no numerical value associated with these letter labels. The unique function returns all unique labels in the vector, in the order that they appear in the vector.

# see the first 6 observations  
head (height\_exp$C\_v)   
## [1] "b" "b" "b" "b" "b" "b"  
  
# class starts as a character vector  
class (height\_exp$C\_v)   
## [1] "character"  
  
# no numerical values, you will see NAs  
head (as.numeric (height\_exp$C\_v))   
## Warning in head(as.numeric(height\_exp$C\_v)): NAs introduced by coercion  
## [1] NA NA NA NA NA NA  
  
# we can see the number of unique groups  
unique (height\_exp$C\_v)   
## [1] "b" "g" "m" "w"

We can turn the character vector C\_v into a factor vector C\_v\_f. The benefit of this is that these nominal labels now have associated numerical values. Many R functions turn your nominal (non-numeric) predictors into factors, and doing this yourself gives you control over how this will be handled.

# we can turn it into a factor in R  
height\_exp$C\_v\_f = factor(height\_exp$C\_v)   
  
# now it has official levels  
levels(height\_exp$C\_v\_f)   
## [1] "b" "g" "m" "w"  
  
# now each level has numerical values  
table (height\_exp$C\_v\_f, as.numeric (height\_exp$C\_v\_f))   
##   
## 1 2 3 4  
## b 810 0 0 0  
## g 0 570 0 0  
## m 0 0 1350 0  
## w 0 0 0 1440

By default, factor levels are ordered alphabetically. You can control this behavior by re-ordering the factor levels as below:

height\_exp$C\_v\_f = factor (height\_exp$C\_v, levels = c('w','m','g','b'))  
  
levels (height\_exp$height\_exp)  
## NULL  
  
# note that 'm' is now the second category  
table (height\_exp$C\_v\_f, as.numeric (height\_exp$C\_v\_f))   
##   
## 1 2 3 4  
## w 1440 0 0 0  
## m 0 1350 0 0  
## g 0 0 570 0  
## b 0 0 0 810

Although our factors seem to have an ‘order’ this is only because items can only be discussed and presented one at a time, and so there must be some order in our nominal variables at some level of organization. For example, when presenting effects and plotting figures, you literally do have to decide to show one effect first and another second. However, the ordering of factors is **exchangeable** meaning it does not in any way affect our analysis. For example, the listeners and speakers in our experiment received unique numbers. However, listener 1 is not the listener who ‘most’ has the quality of listener, and speaker 8 is not twice the speaker that speaker 4 is. In other words, although we must commit to some order in our factors in order to organize our data, this ordering is arbitrary and not meaningful.

There is a special kind of categorical variable called an ordinal variable where the ordering of the categories is meaningful. These variables are halfway between numbers and labels: They faithfully represent the order (rank) of categories but not the magnitude of the difference between values. For example, consider the first, second, and third place runners in a race. These are ordinal values. You know who finished before/after who, but don’t know anything about how much of a difference there was between the runners. As a result, these variables seem to have some of the properties of numbers, while not being totally like ‘real’ numbers. We will discuss the prediction of ordinal dependent variables in more detail in Chapter X.

### 1.4.4 Quantitative variables

Unlike nominal variables, quantitative variables let us represent the relative ordering of different observations *and* the relative differences between different observations. Some examples of quantitative variables are time, frequency, and weight. In our experiment data, height is a quantitative dependent variable, and f0, vtl, and duration are quantitative independent variables.

A distinction is made between **continuous** and **discrete** quantitative variables. Continuous variables have infinitely small spaces between adjacent elements (like the real numbers), at least in principle. On the other hand discrete variables have gaps between the possible values of the variable, like the integers. For example, things like time are naturally continuous while things like counts are naturally discrete.

When we are using a quantitative variable as our dependent variable, there is usually the expectation that is is continuous rather than discrete. In practice all measures stored on computers are discrete and many continuous values (e.g. reaction times) can be measured with a maximal precision, resulting in discrete values. For example, a chronometer that measures reaction times to the millisecond contains only 1000 possible values between zero and one second. Similarly, human height is difficult to measure to much less than a centimeter of precision, making height measurements effectively discrete. Below are some more questions that will help you decide if you should treat a variable as quantitative, even if it ‘discrete’:

1. Is the variable on a ratio or interval scale? This is a prerequisite for a quantitative value to be used as a dependent variable. An interval scale means that distances are meaningful, and a ratio scale means that 0 is meaningful.
2. Is the underlying value continuous? Many variables are discrete in practice due to limitations in measurement. However, if the underlying value is continuous (e.g., height, time) then this can motivate treating the measurement as a quantitative dependent variable since fractional values ‘make sense’. For example, even if you measure time only down to the nearest millisecond, a value of 0.5 milliseconds is possible and interpretable. In contrast, a value of 0.5 people is not.
3. Are there a large number (>50) of possible values the measured variable can take? For example a die can only take on 6 quantitative values, which is not enough.
4. Are most/all of the observed values far from their bounds? Human height does not really get much smaller than about 50 cm and longer than about 220 cm, so it is technically bounded. However, in most cases our observations are expected to not be clustered at the boundaries.

If you answered yes to all or most of these questions, it is probably ok to treat a quantitative variable as if it were continuous, though this determination really needs to be made on a case by case basis.

### 1.4.5 Logical variables

Before finishing with variables, we need to talk about one type that does not appear in our data, but that will come up often. These are referred to as Boolean variables in many other situations, however, they are referred to as logical variables in R. Logical variables in R can only take one of two values: TRUE and FALSE. Below we use two equal signs to test for the equality of two values, and != to check for an inequality. Notice that we can check for the equality of numbers or characters.

2 == 1  
## [1] FALSE  
"hello" == "hello"  
## [1] TRUE  
"hello" != "hello"  
## [1] FALSE

We can also check for inequalities between numbers:

2 > 1  
## [1] TRUE  
2 >= 1  
## [1] TRUE  
2 < 1  
## [1] FALSE  
2 >= 1  
## [1] TRUE

One useful fact is that the logical values of TRUE and FALSE have numeric values of 1 and 0, as seen below. In each case, TRUE is equal to 1 so the expression evaluates to 2.

TRUE + 1  
## [1] 2  
(2 == 2) + 1  
## [1] 2

When logical operators are applied to vectors, the operation is evaluated for each element of the vector, as below, and a vector of logical values is returned. When combined with the numeric values of logical variables, this means that we can easily calculate the number of times a certain condition was met in the vector.

# is the values less than or equal to 3?  
c(1,2,3,4,5,6,7,8,9,10) <= 3  
## [1] TRUE TRUE TRUE FALSE FALSE FALSE FALSE FALSE FALSE FALSE

Below, we find whether each element of the vector is or is not greater then or equal to three. This results in a vector of logical values equivalent to a vector of ones and zeros. When we find the sum of the vector of logical values, we find the number of times in which the condition was met. Below, we see that three of the elements in this vector satisfy our condition.

logical\_vector = c(1,2,3,4,5,6,7,8,9,10) <= 3  
  
as.numeric (logical\_vector)  
## [1] 1 1 1 0 0 0 0 0 0 0  
  
sum (logical\_vector)  
## [1] 3  
sum (c(1,2,3,4,5,6,7,8,9,10) <= 3)  
## [1] 3

There is one other very important use for vectors of logical values, and this is to extract subsets of your data that meet certain conditions. Below we create a vector of logical values that indicate whether the f0 for a trial is below 175 Hz or not. We can see that this vector has 4170 elements, one for every row in our data, and that 1290 trials satisfied our condition. This is nothing more than a bigger version of the same process we just carried out above with our logical\_vector.

f0\_idx = height\_exp$f0 < 175  
str (f0\_idx)  
## logi [1:4170] FALSE FALSE FALSE FALSE FALSE FALSE ...  
sum (f0\_idx)  
## [1] 1290

Recall that we can access individual rows of our data frames by placing this information before a comma, inside brackets following the name of the data frame (as seen below). When we use a logical vector in this way, the effect is to include every row that equals TRUE and to omit every row that equals FALSE in the vector. Below we use our f0\_idx vector to create a new data frame called low\_f0 containing only productions with f0 below 175 Hz.

low\_f0 = height\_exp[f0\_idx,]  
str(low\_f0)  
## 'data.frame': 1290 obs. of 12 variables:  
## $ L : chr "01" "01" "01" "01" ...  
## $ C : chr "m" "m" "m" "b" ...  
## $ height: num 172 177 176 188 178 ...  
## $ R : chr "a" "b" "a" "b" ...  
## $ S : num 47 47 48 48 49 49 50 50 51 51 ...  
## $ C\_v : chr "m" "m" "m" "m" ...  
## $ vtl : num 14.8 14.8 15.6 15.6 15.5 15.5 14.5 14.5 15 15 ...  
## $ f0 : int 172 172 108 108 96 96 134 134 122 122 ...  
## $ dur : int 339 339 236 236 315 315 240 240 241 241 ...  
## $ G : chr "m" "m" "m" "m" ...  
## $ A : chr "a" "a" "a" "c" ...  
## $ C\_v\_f : Factor w/ 4 levels "w","m","g","b": 2 2 2 2 2 2 2 2 2 2 ...  
max(low\_f0$f0)  
## [1] 172

We can use the ! operator, which basically means ‘not’ to flip each TRUE to FALSE (and vice versa). When f0\_idx is flipped to select a subset of a data frame, the result is to select those rows where speaker f0 is *above* 175 Hz.

high\_f0 = height\_exp[!f0\_idx,]  
str(high\_f0)  
## 'data.frame': 2880 obs. of 12 variables:  
## $ L : chr "01" "01" "01" "01" ...  
## $ C : chr "b" "b" "g" "w" ...  
## $ height: num 122 132 129 156 141 ...  
## $ R : chr "a" "b" "a" "b" ...  
## $ S : num 1 1 2 2 3 3 4 4 5 5 ...  
## $ C\_v : chr "b" "b" "b" "b" ...  
## $ vtl : num 12.2 12.2 12.4 12.4 11.6 11.6 11.9 11.9 12.1 12.1 ...  
## $ f0 : int 277 277 287 287 219 219 260 260 244 244 ...  
## $ dur : int 237 237 317 317 277 277 318 318 242 242 ...  
## $ G : chr "m" "m" "f" "f" ...  
## $ A : chr "c" "c" "c" "a" ...  
## $ C\_v\_f : Factor w/ 4 levels "w","m","g","b": 4 4 4 4 4 4 4 4 4 4 ...  
min(high\_f0$f0)  
## [1] 175

## 1.5 Inspecting our data

After running an experiment but before running any kind of data analysis, you should inspect the patterns in your data. This gives you an opportunity to make sure the data has the characteristics you expect, and that there were no errors during the collection of your data or with the design of your experiment.

### 1.5.1 Inspecting categorical variables

One of the most useful functions for understanding the distribution of categorical variables is the table function. This function make a **cross tabulation** (or **contingency table**) of the variables passed to the function. If a single factor is passed, the function returns the number of times each level of the factor is found in the data. Since each of our listeners listened to 278 stimuli, we expect that each level of the factor L (representing listeners) will appear 278 times in our data, confirmed below.

We can use this approach to confirm basic expectations about our data, and to rule out problems with the design of the experiment. This is always a good idea since mistakes happen, and sometimes only get noticed when attempting to process the data. For example, if any of the levels below appeared more than or fewer than 278 times, we would have a problem.

table (height\_exp$L)  
##   
## 01 02 03 04 05 06 07 08 09 10 11 12 13 14 15   
## 278 278 278 278 278 278 278 278 278 278 278 278 278 278 278

We can also provide two (or more) factors at a time and the table function will return counts for every combination of factor levels. The table below reflects the fact that each listener heard 54 boys, 38 girls, 90 men, and 96 women, for a total of 276 total responses. When you provide multiple factors to table, it will vary the first factor along the rows of the table and the second factor along the columns of the table. If a third factor is provided, it makes a different table for factors one and two, for each level of factor three. More and more factors can be provided to the function, but these tables before harder and harder to work with.

table (height\_exp$C\_v, height\_exp$L)  
##   
## 01 02 03 04 05 06 07 08 09 10 11 12 13 14 15  
## b 54 54 54 54 54 54 54 54 54 54 54 54 54 54 54  
## g 38 38 38 38 38 38 38 38 38 38 38 38 38 38 38  
## m 90 90 90 90 90 90 90 90 90 90 90 90 90 90 90  
## w 96 96 96 96 96 96 96 96 96 96 96 96 96 96 96

Below we see that unlike our veridical categories, the distribution of *apparent* speaker categories varies across listeners. This is because the equal distribution of speakers for each listener is an aspect of the experimental design. However, how listeners interpreted each voice, whether they though it sounded like a boy or girl for example, may vary across individual listeners.

table (height\_exp$C, height\_exp$L)  
##   
## 01 02 03 04 05 06 07 08 09 10 11 12 13 14 15  
## b 59 89 59 32 51 12 95 98 63 40 68 58 59 90 38  
## g 24 45 44 67 44 60 23 16 43 34 24 29 34 50 61  
## m 90 89 99 88 91 95 88 88 89 95 86 90 92 84 97  
## w 105 55 76 91 92 111 72 76 83 109 100 101 93 54 82

We can visualize relationships between categorical variables using a mosaic plot. In figure 1.2 we mosaic plots representing the two tables shown immediately above. Mosaic plots use rectangles of different sizes to reflect the relative frequencies of different combinations of categorical variables. For example, in the left mosaic plot we see that the size of the rectangle for each category is identical across listeners. This tells us these variables do not affect each other: changing the listener does not affect the distribution of veridical speaker class in any way. In contrast, the distribution of apparent speaker class is affected by the listener and this is shown in the right plot where columns differ randomly from each other.

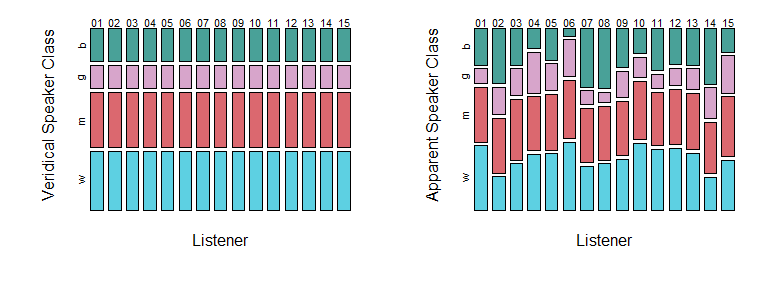


Figure 1.2: Comparisons of mosaic plots showing variables that do not (left), and do (right), affect each other.

Below we make a three-dimensional table, and inspect the table and each dimension. Notice that to index the table along the third dimension we need to add two commas inside the brackets.

tmp\_tab = table (height\_exp$C, height\_exp$L, height\_exp$R)  
tmp\_tab  
tmp\_tab[,,1]  
tmp\_tab[,,2]

When we plot the relationship between apparent speaker class, listener, and resonance, we see a three-way relationship between the variables. First, we see that the chances of observing different speaker categorizations depends on the listener. Second, we see that the chances of observing each category depends on resonance. And third, we see that the effect of resonance potentially affects each listener a somewhat different way. The first chapters of this book will focus on understanding patterns in continuous variables. However, we will discuss the prediction and modeling of categorical dependent variables beginning in Chapter X.

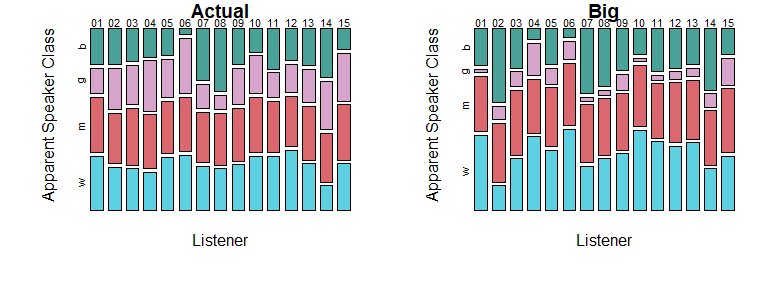


Figure 1.3: Mosaic plots highlighting a three-way relationship: The two-way relationship varies as a function of the third variable (indicated along the top of each plot).

### 1.5.2 Inspecting quantitative variables

Using R, we can easily find useful information about any quantitative variable. Below, we calculate the sample mean, the number of observations, the sample standard deviation, and some important quantiles for our speaker height judgments. The **quantiles** of a set of values such that a given percentage of observations fall above and below the value. Quantiles are found by ordering the observations and selecting the observation that is greater than of the sampled values and less than () of the sampled values. For example, the 50% quantile, also called the **median**, is the value such that 50% of the distribution is below it and 50% is above it, and the 25% quantile (the *first quartile*) is the value such that 25% of the distribution is below it and 75% is above it.

# calculate the mean  
mean (height\_exp$height)  
## [1] 162.7644  
  
# find the number of observations  
length (height\_exp$height)  
## [1] 4170  
  
# find quantiles  
quantile (height\_exp$height)  
## 0% 25% 50% 75% 100%   
## 106.7 154.9 164.8 173.5 198.1

We can use this information to make some basic, and potentially useful statements about our data. The mean and median are 162.8 and 164.8 cm respectively, and height values range from 106.7 to 198.1 cm. However, there are not many observations at the extremes, and 50% of values are between 154.9 and 173.5 cm. We know this because these are the values of the first and third **quartiles**, the 25% quantiles that divide our distribution into four equal parts. Since , we know that 50% of the distribution of observations must fall inside of these boundaries. We can look at the distribution of apparent height judgments in several ways, as seen in Figure 1.4. In the top row each point indicates an individual production. Points are jittered (randomly shifted) along the y axis to make them easier to distinguish so that dense and sparse locations can be compared. In the middle row we see a **box plot** of the same data. The edges of the box correspond to the 25% and 75% quantiles of the distribution, and the line in the middle of it corresponds to the median. So, the box spans the **interquartile range** of your observations and 50% of observations are contained in the box.The boxplot **whiskers** extend from the edge of the boxplots. By default, these extend out 1.5 times the interquartile range. These whiskers are simply intended to give you an estimate of the amount of ‘typical’ variation in your sample. Beyond the whiskers we see individual **outliers**, points considered to be substantially different from the rest of the sample. We can see that the boxplot does a good job of summarizing the information in the top plots, and provides information related to both average f0 values and to the expected variability in these values.

The bottom row presents what is knows as a histogram of the same data. The histogram divides the x axis into a set of discrete sections (‘bins’), and gives you the count (or frequency) of observations in each bin. Bins with lots of observations are relatively taller (more dense) than bins with fewer observations in them. As a result, histograms can be used to summarize where observations tend to be. For example, we can see that the bins under the interquartile range have the most observations, and that values further from the mean value become increasingly less frequent. In addition, histograms can provide us with information that boxplots can’t. For example, in the right column we see that our distribution of height judgments actually has two distinct peaks, with a little gap in the middle. This information does not really come across in the boxplot representation of the same data.

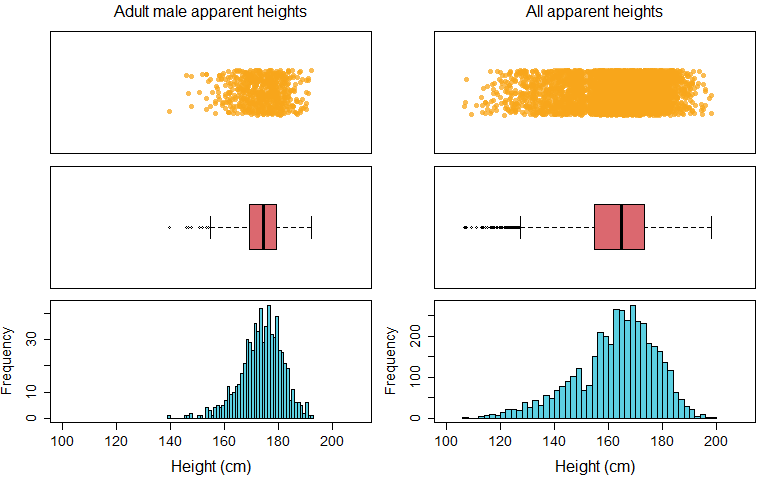


Figure 1.4: Each row presents data in a different way, with each column containing the same data across rows.

**Scatter plots** are plots that represent two variables at a time using a set of points on a coordinate space. Each point represents a single observation, the x-axis location represents the value of one variable, and the y-axis location represents the value of the other variable. Scatter plots are useful to understand relationships between continuous predictors. Below we consider the relationships between our quantitative predictors using a pairs plot (pairs). A pairs plot creates scatter plots for all pairs of quantitative variables provided, resulting in plots for variables. Each plot below contains a single point for each different stimulus used in this experiment (height values represent averages across all listeners).

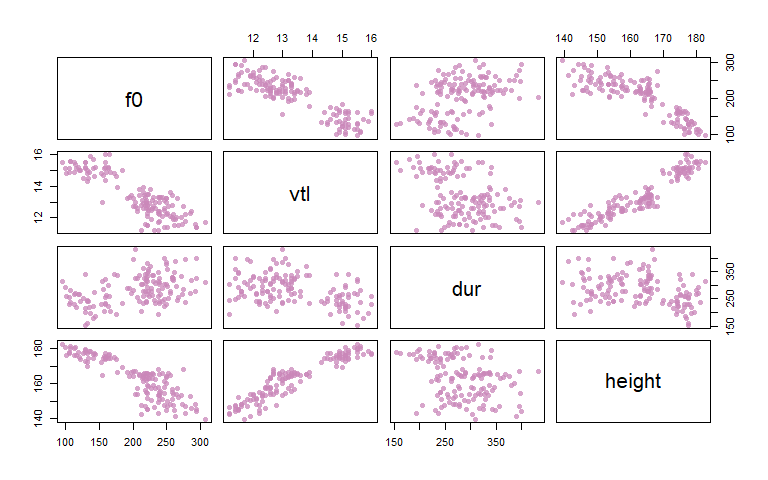


Figure 1.5: A pairs plot of the continuous variables in our data, showing different sorts of relationships between our variables.

In the plot above we can see several apparent relationships between our quantitative variables. For example, pitch (f0) and vocal-tract length (vtl) are *negatively* related. This means that as the value of f0 increases (left to right), the value of vtl decreases (top to bottom). In other words, if the f0-vtl relationship were a hill it would have a negative, decreasing, slope. In contrast we see that height and vtl enter into a positive relationship: As you increase vtl, height also increases. Finally, we see that duration (dur) and height do not seem to have much of a relationship. Unlike the other two scatterplots which looked a bit like ramps or lines, the scatter plot of dur and height resembles a Rorschach test inkblot. This suggests either that these two variables are not strongly related, or that the nature of the relationship is more complicated than what can be understood using these simple plots.

### 1.5.3 Exploring continuous and categorical variables together

We can also consider the relationships between our quantitative and categorical variables. We can use the boxplot function as below:

boxplot (y ~ factor)

To make a set of boxplots for the variable y. The function call above will create a plot with a separate box for each level of the factor in the function call. In figure 1.6, we see different quantitative variables organized according to veridical speaker category. For example, the left panel shows the distribution of observations of f0 for boys, girls, men, and women respectively. In this case the differences between the boxplots for each level of the factor tell us about the values of f0 usually observed for speakers in that category.

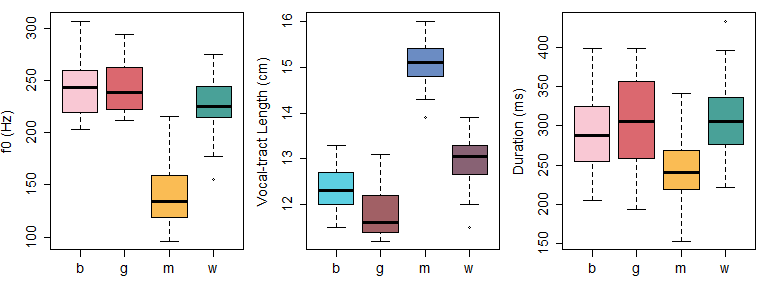


Figure 1.6: Boxplots showing the distribution of different quantitative variables in our data according to the veridical speaker categories of boy (b), girl (g), man (m), and woman (w).

Another way to think of the relationships between our categorical and quantitative variables is using the plot in figure 1.7. In the scatter plot below, each point indicates a single speaker from our experiment, and the position of each point is determined by the f0 and vocal-tract length of the speaker. However, rather then plot using symbols, each point is labeled using a letter which indicates the veridical category that the speaker falls into. Using a plot like the one below helps us understand the relationship between our important acoustic predictors and our speaker categories. For example, it is clear that adult males are fairly distinct acoustically compared to the other speaker categories. In addition, it seems that boys, girls, and women are easier to separate along the vocal-tract length dimension than the f0 dimension. What we mean by this is that it would be easier to draw horizontal lines separating the groups than vertical lines separating the groups.

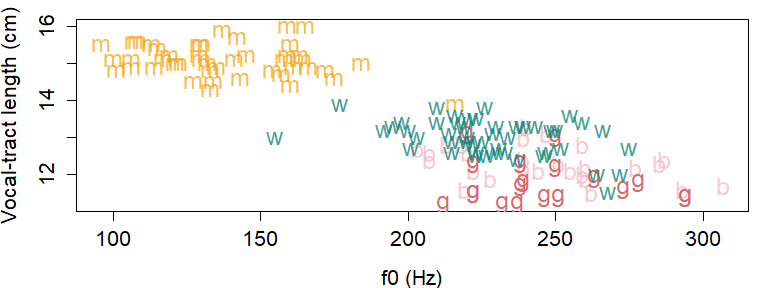


Figure 1.7: Speakers plotted according to their fundamental frequency (f0) and vocal-tract length. Letters indicate if speaker is a boy (b), girl (g), man (m), or woman (w).

# 2 Probabilities, likelihood, and inference

In the previous chapter we introduced some experimental data and talked about variables and experiments. In this chapter we’re going to talk about probabilities and explain how they can be used to make inferences about our data and research questions. Before beginning this chapter we should note that it’s normal if a some of the topics to be discussed don’t make sense the first time you read this chapter. It will make more sense once you start to actually build models and it becomes less hypothetical and more practical. In addition, as noted in the preface, it is a mistake to think that you can read a chapter in a statistics textbook once and move on having fully absorbed the content. If many of the topics in this chapter are new to you, you should probably: 1) Read this chapter, 2) wait a few days and read it again, and then 3) wait a few more days (or weeks) and read it again. It may also be useful to return to this chapter even once you are working through the following chapters. In our experience, you may find that you *see* information for the first time that was there all along, but that finally makes sense given your increased experience with the subject.

## 2.1 Data and research questions

We’re going to think about a hypothetical value “the average apparent height of adults males in the United States (US)”. To put it another way, we might wonder “how tall do men from the US ‘sound’?”. The apparent heights of men from the US cannot be known **a priori**. In other words, you don’t know how tall a random man from the US will ‘sound’ until you actually observe the judgment. For this reason, “the average apparent height of men from the US” is a random variable. We’re going to think about how we can use our experimental data to try to answer the following two questions:

Q1) How tall does the average man from the US sound?

Q2) Can we set limits on credible average apparent heights based on the data we collected?

These two questions can be though of as relating the the central location and the spread of the data, respectively. Answering the first question tells you what values your variable tends to take on, while answering the second question tells you how much variation you can expect around the most typical values. Scientific research is often focused on questions such as (1) regarding the central location, the average value, of some variable. For example someone might ask “how tall do adult male speakers from the US sound?” and you can say, for example, “I have some data that suggests 174 cm is a reasonable estimate”. However, reliable inference requires answering question (2) as well, and determining what range of values are plausible for a certain variable.

Think of the average (actual) height of the people in a large city. You can go out and sample 100 people and find the mean of your sample, arriving at a single estimate of the population mean. Now imagine that 50 people went out in the same city and each sampled 100 random people. There is no chance that every single of those 50 people would find identical means across all of their samples. Instead, there will be a distribution of sample means, in the same way there is a distribution of the original data used to calculate the means. Another way to look at this is that there is some degree of **uncertainty** involved when answering any research question.

As a result of this uncertainty, it can be difficult to rule out alternative possible answers to our research questions. For example, if 174 cm is a good estimate, what about 173.99 cm? What about 173 cm? 172 cm? Where do estimates of average height stop being ‘good’? Without being able to say what is *not* a good estimate, it is not quite as useful to be able to say what *is*. A related issue arises with respect to the interpretation of average values. Imagine that you read about a miracle diet that was guaranteed to make you lose one gram of weight a day. You know that is not very impressive. How? Because you understand that a difference of one gram is not large relative to the variation that exists in the weight of a human body on a daily basis. You could gain the weight back (and more) by drinking a teaspoon of water. Without knowing how much human weights tend to vary between and within people, it’s impossible to know whether a reduction of one gram constitutes a meaningful change in the mass of a human. In contrast, a diet that causes one gram of weight loss in the average hamster may actually be of interest to hamster owners, as this is a relatively large value relative to natural variation in hamster weight.

So, we see that imposing limits on credible ranges for our average values can be as important as finding the average values themselves. Further, in order to properly contextualize values and effects, we need to have some idea about the underlying variation in the measurements. Clearly, we need some principled way to ‘guess’ reasonable ranges based on our sample of observations, in addition to just talking about average values. In this chapter we will discuss how statistics provides us with a framework to answer both questions above using only our sample of values.

In order to discuss the apparent heights of adult males, we need to extract the subset of speakers judged to be adult males from our data. Below we load the book package (bmmb) and select the relevant rows from our experimental data. Recall that our experiment contained an acoustic manipulation such that speech resonances were changed to make speakers sound bigger (see section 1.3.2). For now, we’re going to focus only for the unmodified productions, the ‘natural’ speech produced by the men in our sample.

We can have a look at some of the quantiles (see section 1.4.4 to get an idea of what range of values this variable tends to have. We see that the minimum and maximum values are 139.7 and 192.3 cm, that 174.5 cm is the median, and that half of the observed height judgments for adult males fell between 169.2 and 179.1 cm.

quantile (mens\_height)

## 0% 25% 50% 75% 100%   
## 139.7 169.2 174.5 179.1 192.3

Obviously, an inspection of the distribution of our observed height judgments only gives us direct information about the judgments we *did* observe. To make inferences about the probable characteristics of the height judgments we did *not* observe, or to talk about height judgments for adult males from the US more generally, we rely on methods of statistical inference, as will be described below.

## 2.2 Empirical Probabilities

The sample space of a variable is the set of all possible outcomes/values that a variable can take. Classic examples are a coin flip, which can take on the values ‘heads’ or ‘tails’, or the roll of a die which can take on the values one through six. In other cases the sample space may have an infinite or practically infinite number of members. For example, since time is continuous there are an infinite number of durations an event may have, given adequate precision in measuring time. If we think of the human population, or the population of fish in the sea, these are theoretically finite but practically infinite. It would be extremely difficult to fully sample either of these populations, and impossible to do so before they changed substantially (i.e. before some members have died and others have been born).

The **probability** of an event/outcome is the number of times an outcome occurs, relative to every possible outcome that can occur (i.e. all of the outcomes in the sample space). By convention, the probability of each event is assigned a value between 0 and 1 and the total probability of all of the possible outcomes in the sample space is equal to one. As a result of this convention, you know that a probability of 0.5 means something is expected to occur half the time (i.e. on 50% of trials), and a probability of 0.25 indicates that something should happen 25% of the time, about one in every four trials. **Empirical probabilities** are the probabilities of different outcomes in a sample of data. For example, we can flip a coin 100 times and observe 65 heads. This means that the empirical probability of observing heads in our data is 0.65 (65% of trials).

Suppose we want to know the probability of observing an apparent height judgment greater than 180 cm in our sample. To calculate the empirical probability of this occurring in our data we need to find 1) The number of times it occurred, and 2) the total number of observations for the variable. We can do this easily using the logical operators and variables discussed in Chapter 1. Below, we find the total number of outcomes that satisfy our restrictions (being over 180 cm), and divide this by the total number of observations being considered.

# the evaluation in the parenthesis will return 1 if true, 0 if false  
# number of observations the fall below threshold  
sum (mens\_height > 180)   
## [1] 136  
  
# divided by total number of events  
sum (mens\_height > 180) / length (mens\_height)   
## [1] 0.2014815  
  
# a shortcut to calculate probability, mean = sum/length  
mean (mens\_height > 180)  
## [1] 0.2014815

The top value is the frequency of the occurrence. This is not so useful because this number can mean very different things given different sample sizes (e.g., 136/675, 136/675000). In contrast, the middle and bottom values have been divided by the total number of observations. As a result, these now represent an estimate of the probability of occurrence in a way that is independent of the total number of observations.

### 2.2.1 Conditional and marginal probabilities

Figure 2.1 presents boxplots of the overall distribution of height judgments in our data (left), and of the distribution of height responses provided by each listener individually. We can see that height judgments range from about 140 to 200 cm, with most responses falling between 170 and 180 cm. Notice that our overall boxplot does not give us any information about the ranges used by different listeners, nor even the fact that the data was contributed by different listeners. This overall distribution of height responses is the **marginal distribution** of height judgments. The marginal distribution of a variable is the overall distribution, *across* all values of all other variables. The marginal boxplot on the left compresses all of the listener-specific boxplots on the right into one single box. It’s common to denote marginal probabilities using notation like this , meaning we might refer to the marginal probabilities of height responses like this .

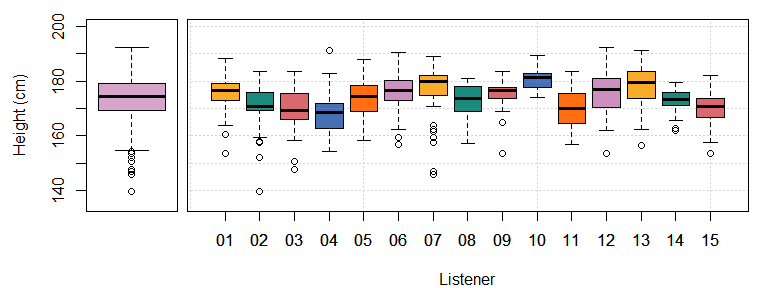


Figure 2.1: (left) Boxplot showing all height judgements for adult male speakers in our experiment. (right) Individual boxplots for each listener’s responses

A look at figure 2.1 reveals that the probability of observing a height response of over 180 cm can depend substantially on the listener that provided it (e.g. compare listener 10 vs listener 15). Recall that height is a quantitative variable and listener is a categorical predictor (a factor) with 15 levels, one for each listener (see section 1.4.3). We can talk about how height judgments vary across the levels of our listener factor (i.e. for different listeners) by considering the **conditional probability** of height given listener. A conditional probability is the probability of an outcome given that some other outcome has occurred. For example, rather than ask “what is the probability of observing an apparent height over 180 cm?”, we can ask “what is the probability of observing an apparent height over 180 cm *given* that we are observing data from listener 10?”. Conditional probabilities basically reduce the sample space by including only the subset of events that satisfy the given condition.

Conditional probabilities are often denoted like this , which in this case would look like . For example, the first box in the right panel of figure 2.1 is the distribution of , and the second box is the distribution of . Below we divide our vector of height judgments (mens\_height) into those contributed by listener 10 and listener 15. We then find the probability of observing a height judgment over 180 cm conditional on listener, and see that these can differ quite a bit from each other.

# create subsets based on listener  
L10 = mens\_height[men$L==10]  
L15 = mens\_height[men$L==15]  
  
# find the conditional probability of height>180 for each listener  
mean (L10 > 180)  
## [1] 0.6444444  
mean (L15 > 180)  
## [1] 0.02222222

In the boxplots in figure 2.1, we see that the distributions of heights vary substantially as a function of the value of L, our listener variable. Contrast this with the boxplots seen in figure 2.2, which shows the distribution of stimulus durations conditional on listener. Since every individual recording had a fixed duration and all listeners heard the same sounds, we know that the distribution of stimulus durations is identical across all levels of the listener variable. As a result, we can see that all of the conditional distributions of duration given listener look just like each other, and just like the marginal probability. This tells us that duration and listener are *statistically independent*. When two variables are statistically independent, the distribution of one variable is not affected by the values of the other. As a result, the conditional distribution of one variable given the other will be the same as its marginal distribution, as seen in figure 2.1. This can be stated for the general case as . In our example above, is *not* equal to , and so we conclude that the variables apparent height and listener are not independent. Instead, they are **statistically dependent**, meaning that these variables *do* affect each other in some way, and that knowing the value of one may tell you something about probable values of the other.

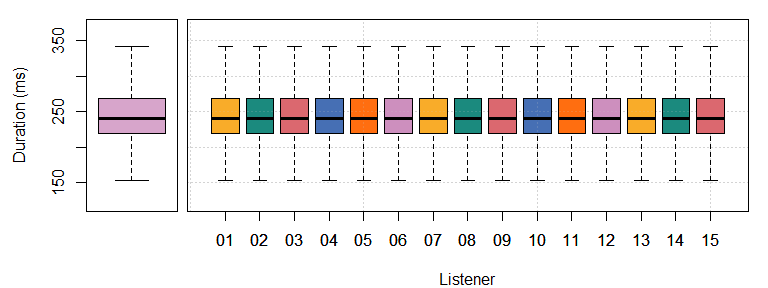


Figure 2.2: (left) Boxplot showing the duration of adult male speakers in our experiment. (right) Individual boxplots for the stimuli presented to each listener.

### 2.2.2 Joint probabilities

**Joint probabilities** reflect the probabilities of two or more things occurring together. We can refer to the joint probability of and using the notation or . Here are some important things to know about joint probabilities:

1. The formula for calculating the joint probability of two outcomes and is given by . In plain English this means that the probability of and is equal to the conditional probability of given , multiplied by the marginal probability of .
2. Recall from section @ref{c2-conditional} that when and are independent . In other words when and are independent the conditional probability of given equals the marginal (unconditional) probability of . As a result of this, when and are independent . Thus, when and are independent their joint probability can be found by simply multiplying their individual marginal probabilities.
3. The probability of and is equal to the probability of and . As a result of this , and as a result of that . This relation will become useful later.

We can demonstrate the above properties using the empirical probabilities in our data. Consider the joint probability of observing a response in our data that was contributed by listener ten (L=='10'), and also being longer than 250 ms in duration. We can find this by joining two logical variables using the & (and) symbol as shown below.

# TRUE is the listener is 01  
L10 = men$L=='10'  
# TRUE if the Resonance is 'a'ctual  
dur\_250 = men$dur > 250  
  
# The probability of A and B  
mean (L10 & dur\_250)  
## [1] 0.02814815  
mean (men$L=='10' & men$dur > 250)  
## [1] 0.02814815

So, we see that the probability of observing this event is 0.028, indicating that we expect this in about 3% of trials. Below, we see that this same joint probability can be calculated based on .

# Marginal probability of observing listener 10 (i.e. P(L=10))  
p\_L10 = mean (men$L=='10')  
  
# Subset containing only listener 10   
L10 = men[men$L == '10',]  
# Conditional probability of dur>250 given listener 10 (i.e., P(dur>250 | L=10))  
p\_dur\_250\_given\_L10 = mean (L10$dur > 250)  
  
# Joint probability = P(dur>250 | L=10) \* P(L=10)  
p\_dur\_250\_given\_L10 \* p\_L10  
## [1] 0.02814815

Because of the experimental design, we know that every listener heard the same vowel durations (see Section 1.3 for a review of the experimental design). This means that duration is independent of listener: Knowing who the listener is in no way changes the fact that every listener heard the same number of trials with durations longer than 250 ms. As a result of this independence, we could also have calculated the joint probability of the above events by simply multiplying their marginal probabilities, as seen below.

mean (p\_L10) \* mean (dur\_250)  
## [1] 0.02814815

This short cut will not work for variables that are not independent. For example, let’s consider the probability of observing a height judgments of greater than 180 cm provided by listener 10. Below we calculate this the *wrong* way by multiplying the marginal probabilities, substantially underestimating the probability of the event. The problem is that this method of calculation does not take into account that listener 10 was among the most likely listeners to report heights greater than 180 cm.

# TRUE is the listener is 10  
L10 = men$L=='10'  
# TRUE if the height is over 180  
under\_180 = men$height > 180  
  
# Empirical probability of the observation  
mean(men$L=='10' & men$height > 180)  
## [1] 0.04296296  
  
# Wrong: multiplying marginal probabilities  
mean(L10) \* mean(under\_180)  
## [1] 0.0134321

Below we calculate the joint probability correctly by using the conditional probability.

# Marginal probability of observing listener 01  
p\_L10 = mean (men$L=='10')  
  
# Subset containing only listener 01 (i.e., given listener 01)  
L10 = men[men$L == '10',]  
# Conditional probability of a height < 160 given listener 01  
p\_over\_180\_given\_L10 = mean (L10$height > 180)  
  
# Joint probability  
p\_over\_180\_given\_L10 \* p\_L10  
## [1] 0.04296296

This highlights a very important point: The calculation of joint probabilities is much (much) simpler when you can assume that your observations are independent. For example, consider the comparison of the joint probability of four events, one dependent and one independent:

This means that we not only need to calculate many conditional probabilities, but most of these feature the conditioning of one variables on several other variables (i.e., ). Although this may not matter when calculating the joint probability of a handful of observations, calculating the joint probability of hundreds or thousands of correlated variables can become difficult if not impossible in practice.

## 2.3 Probability distributions

A probability distribution is a function that assigns probabilities to all the different outcomes in your sample space. We will discuss exactly what this means with reference to histograms. In the left panel of figure 2.3 we see a histogram of counts. This histogram shows the number of times a range of values was observed along the y axis. The heights of the bars reflect the frequencies of different sort of observations and so can tell you about the values you expect, and don’t expect, for the variable. This sort of representation makes it difficult to compare distributions across samples of different sizes. For example, if the number of observations were tripled, so would the heights of all the bars in the histogram.

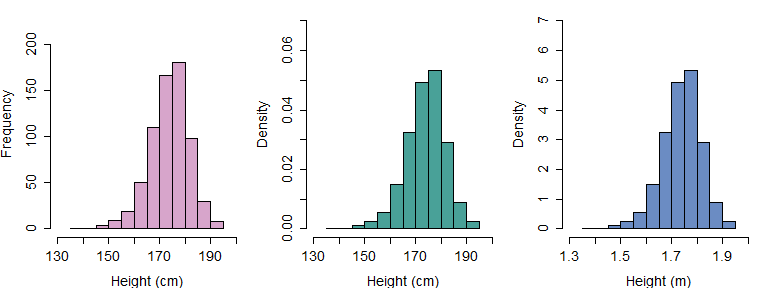


Figure 2.3: (left) A histogram of adult male height judgments showing counts in each bin. (middle) The same data from the left panel, this time showing the density of the distribution. (right) The same data from the middle plot, this time with heights expressed in meters.

Histograms can also be used to show the **probability densities** associated with different values. Imagine a circle, like a Venn diagram, that contains all possible values of your variable (i.e. the sample space). This circle has an area of one since it contains all possible values in the population. Imagine we took this circle and pressed it down against the number line, squishing it. We then spread out this circle along the x axis so that its shape reflected the relative frequencies of different values present in the population. For example, if some outcomes were five times more probable than others, the shape should be five times thicker there, and so on. If we managed to do this, the thickness (or **density**) of this shape would exactly correspond to the probability density of the variable.

In the middle in figure 2.3 we see a histogram that shows the *density* of the distribution of height judgments, i.e. the thickness of the distribution at different locations along the number line. The values of the density are constrained by the fact that the area under the curve must equal one, since these reflect probabilities. In the case of our histograms this means that the total area of all the bars (rectangles) is constrained to equal one. As a result of this, when variables have wide ranges densities tend to be very small (as in the middle panel in Figure 2.3).

In the right panel of figure 2.3 we present heights in meters rather than centimeters. This has the effect of substantially increasing the values of our density (one-hundred fold) but does not otherwise affect the shape of the curve. The relationship between variable ranges and density values means that densities can’t really be interpreted in an absolute sense. A density value of 0.06 means very different things in the histograms in Figure 2.3. Instead, density values need to be interpreted relative to the other values along the same curve. A higher density in a certain location tells us values in that vicinity are more probable than values in locations with lower densities, and differences in densities reflect differences in the relative probability of different values or outcomes.

Probability distributions sometimes have shapes that can be represented using mathematical functions. These functions have a limited number of **parameters** that determine their exact shape. Think of parameters like ways that things can be ‘set’ differently from each other. For example, a radio has three parameters: tuner frequency, band (AM/FM), and volume. A toaster may have only one, a single knob determining the degree of toasting required. The more parameters something has, the more complicated it is (an airplane has thousands). For example, consider the code below, which defines a slope and intercept and draws a line based on these parameters. You can change the values of the intercept and the slope and draw many kinds of lines. However, there no way to change the characteristics of a line other than by changing its slope and intercept: A line is entirely defined by its slope and intercept.

intercept = 3  
slope = 1  
x = seq (-10,10,.1)  
y = intercept + x\*slope  
  
plot (x,y, type='l',lwd=3,col=4)

Sometimes the same sorts of probability density shapes pop up over and over, and these shapes are often well-defined mathematically. Parametric probability distributions are those density shapes that can be understood in terms of curves that vary in terms of a limited number of parameters. Just like with lines, the characteristics of a parametric probability distribution are entirely defined by the values of its parameters.

## The normal distribution {#c2-normal}

The distribution of many random variables (at least approximately) follows what’s known as the **normal** or **Gaussian** distribution. This means that if you take a sample of a random variable and arrange observations into bins, they resulting histogram will tend to have the familiar, bell-shaped curve common to normally-distributed data (seen in the histograms in figure 2.3). The normal distribution has the following important characteristics:

1. The distribution is symmetrical - i.e., producing a higher or lower than average f0 is equally likely.
2. The probability of observing a given value decreases as you get further from the mean (i.e., *average*) value.
3. It’s easy to work with, very well understood, and naturally arises in many domains.

Normal distributions have two parameters, meaning they vary from each other in only two ways. The parameters are:

1. A mean parameter () which determines the location of the distribution along the x axis. When the mean changes, the whole shape of the distribution ‘slides’ along the number line. The mean is the 50% halfway point of the ‘mass’ of the distribution. If the distribution were an physical object, its mean would be its center of gravity and you would balance the distribution on your fingertip at this point.
2. A variance () or standard deviation () that determines its *spread* along the x axis. When the standard deviation changes the distribution is stretched wide or made very narrow, but stays in place. Since every distribution has an area under the curve equal to one (i.e., they all have the same ‘volume’), distributions with small variances must necessarily be very dense.

In principle, a given probability distribution can be thought of as having fixed parameters. For example, we might imagine that the average apparent height of adult male speakers from the US is exactly 175.91254… cm. However, in most situations we can’t be certain of what the ‘true’ parameters of a distribution are. Instead, we must be satisfied with estimating the values of parameters based on our samples. These are **statistics**, estimates of our population parameters based on our sample. **Statistical inference** consists of using statistics (based on our sample) to make inferences about the characteristics of the overall population (i.e. the ‘true’ parameters). In the case of the normal distribution we are interested in two statistics: The sample mean and the sample standard variance.

### 2.3.1 The sample mean

The sample mean is our ‘best’ guess for the population mean. We’ll be more specific about what this means later, but for now we can consider that if you don’t know the population parameter for a given distribution, the sample mean will provide you the best available estimate. The formula for the sample mean is given in equation (2.1). Initially, reading these mathematical formulas may seem daunting. However, learning to read these is just a skill that is developed with practice. In addition, you will begin to see the same ‘chunks’ or structures come up in formulas over and over, and reading these becomes much easier once you start to recognize the meaning of these repetitive structures intuitively.

Equation (2.1) says that the sample mean of () is equal to the sum of all of the elements of the vector , divided by where is equal to the length of the vector. We use the little hat symbol () to indicate that this is an estimate of the mean, and to distinguish it from the population mean which goes hatless (). The summation () symbol represents the repetitive adding of whatever is the the right of the symbol to some total. The summation begins at the number below the and performs one operation for every integer value of between the starting point and the end point (indicated above the ). The counter variable, in this case , is also often used to index values of a vector (or other structure) that is being summed (as seen below).

This behavior analogous to the behavior of a for loop in R. Below we define the variables N (equal to the length of the vector) and initialize a variable to receive the summation (mean\_height). The for loop then proceeds to increment a count variable (i) from one to N along the integers. For each iteration it adds the value of the vector, divided by N, to the summation variable mean\_height. As we can see below, this results in a value identical to that returned by the mean function in R.

# initialize values  
N = length (mens\_height)  
mean\_height = 0  
  
# summation equivalent to equation 2.1 above  
for (i in 1:N) mean\_height = mean\_height + mens\_height[i]/N  
  
# replicates values of the mean function  
mean (mens\_height)  
## [1] 173.7877  
mean\_height  
## [1] 173.7877

Here are some useful things to know about sample means, in no particular order:

1. The mean of a set of observations is affected by addition and multiplication. This means that adding to a set of observations increases its mean by , and multiplying observations by results in an change in the mean by the same factor.
2. The mean of the sum of two sets of variables (of the same length) an is equal to . In other words, the average of the sum is just the sum of the averages.
3. The sum of sample’s deviations from the sample mean equals zero (seen in (2.2)). This means that the sum of the distances between positive and negative differences from the sample mean exactly balance out. To some extent this makes sense since the mean is the ‘center of gravity’ of a distribution. It is worth noting that this does not apply to deviations form the *population* mean since this is not specifically fit to the characteristics of the sample.

### 2.3.2 The sample variance (or standard deviation)

The formula to calculate the sample variance is seen in (2.3). Note that it is quite similar to the structure of (2.1) and clearly involves the averaging of a value. In fact, if we were to replace with the two equations would be identical. The value being averaged () consists of finding a difference () followed by a squaring operation. So, we see that what’s being averaged is squared deviations from the sample mean. This is what the variance is: The expected value of squared deviations around the mean of the variable.

Below we see that we can use a for loop to recreate Equation (2.3). However, we don’t manage to exactly recreate the output of the var (variance) function included in R. The reason for this is that the formula above is our best guess for the *sample* variance, but not for the *population* variance. This is because Equation (2.3) uses the sample mean to calculate variation.

# initialize variable  
variance\_height = 0  
  
# equivalent to 2.3 above  
for (i in 1:N) variance\_height = variance\_height + (mens\_height[i]-mean\_height)^2/N  
  
# this time the values don't match  
var (mens\_height)  
## [1] 60.27316  
variance\_height  
## [1] 60.18386

Recall above that we said that the sample mean is our ‘best’ estimate of the population mean given a sample. A more formal way to state this is that the sample mean is the value which minimizes the sample variance. In other words, if we choose any value of to calculate the sample variance other than the sample mean, the variance will necessarily be larger. However, we know that our sample mean is just an estimate of the population mean and will never be exactly equal to it. As a result of this, the true variance *must* be greater than the sample variance when calculated using the sample mean. We can put it like this: , the sum of squares around the sample mean will always be less than or equal to the sum of squares around the population mean. For reasons that we won’t get into (but which aren’t too complicated), this expected difference may be offset by dividing the squared deviations by rather than as in (2.4).

We can update our R code to reflect this change, and see that this now matches the calculation of the variance carried out by R.

# initialize variable  
variance\_height = 0  
  
# equivalent to 2.4 above  
for (i in 1:N) variance\_height = variance\_height + (mens\_height[i]-mean\_height)^2/(N-1)  
  
# this time the values do match  
var (mens\_height)  
## [1] 60.27316  
variance\_height  
## [1] 60.27316

The sample standard deviation () is simply the square root of the sample variance, as in (2.5)

Here are some useful things to know about variances, in no particular order:

1. Variances are always positive, and can only be zero for variables that do not actually take on different values (i.e., constants).
2. The variance of a set of observations is not affected by addition. So, adding or subtracting some arbitrary value from a data set will not affect the variances in that data.
3. Multiplication *does* affect the variances of a set of values. Multiplying numbers by results in a change of the variances equal to . So, if we took our heights and multiplied them by 10 to express them in milliliters, we would expect the value of to increase by a value of . Since standard deviations are the square roots of variances, this implies that multiplying data by results in an increase in the standard variation of the data by a factor of .
4. The variance of the sum of variables depends on whether they are independent or not. When variables are independent, the sum of their variances is simply equal to for variables x and y. However, when variables are not independent, the variance can be much greater or smaller than this based on the nature of the relationship between the variables. As a result, just as with the calculation of joint probabilities, we need to take into account whether variables are independent or not when we considering the variance of the sum of variables.

### 2.3.3 The normal density

The parameters of a probability distribution are used to draw its shape, which can be used to make inferences about likely values. Think back to high school math and the function defining the shape of a parabola . This function draws a shape based on the settings of its parameters and . The parameter determines the width of the parabola (and whether it points up or down), while the vertex of the parabola will have x and y axis coordinates of and respectively. In the same way, the formula defining the density of the normal distribution draws a shape given the settings of its and parameters. The formula for the probability density function of the normal distribution is seen in (2.5). The function returns a density value for the probability distribution as a function of the value of , and the values of its parameters.

The equation in (2.5) features **exponentiation** (), that is raising the base (Euler’s number) to some power as in . In (2.5), the value being exponentiated is . The **logarithm** (log) is the inverse function to exponentiation, it basically *erases* or *undoes* exponentiation. We can apply a logarithmic transformation to both sides of (2.5), resulting in the **log density** seen in equation (2.7). Before explaining equation (2.7), we will discuss some basic properties of logarithms that are useful to understand probabilities and probability distributions, as these often involve exponentiation and logarithms. The first line in (2.6) shows the basic behavior of logarithms. The next four lines pertain to the values expected, or undefined, for values of . The next two lines highlight the fact that exponentiation of numbers is equivalent to the multiplication of their logarithms. The final two lines highlight the fact that multiplication of two numbers is equivalent to the addition of the logarithms of the numbers.

Armed with knowledge of the behavior of logarithms, we can see that compared to equation (2.5), we have removed the function around the rightmost term , and added the function around all terms that were *not* previously exponentiated. In addition, the multiplication of the two terms in the right hand side of the equation has turned into addition, or subtraction in this case because we are adding a negative term.

We can use the properties of logarithms to turn into and move this term to the rightmost end of the equation as in (2.8). At this point the function is that of a parabola in vertex form, , where , , and . So, we can see that the normal distribution is just an exponentiated parabola that is scaled by so that the area under the curve is equal to one. The parabola has its vertex at , and opens downwards since the term is negative. When a parabola is in vertex form, the relationship between parabola width and the value of is inverted, its width decreases as the value of increases. However, because , this means that in the case of normal distributions the parabola width increases as grows larger. As a result, a larger standard deviation leads to wider parabolas and wider probability densities.

Equation (2.8) shows how the and parameters work to make observations further from the means less probable. First, we know that negative logarithmic values will fall between 0 and 1, with more negative values being closer to zero (i.e. less probable). As observations () are further from the mean, the value of will be greater so that values further from the mean will be generally less probable. However, whether a deviation is big or small is relative, and so this distance is scaled with respect to the average expected squared deviation from the mean (i.e. the variance ). Variation of 1 cm in body length means different things for an earthworm as opposed to an anaconda. As a result, large values of can be offset by large values of when determining the probability of an outcome.

The nice thing about working in R is that you don’t need to take our word for any of this, you can see it for yourself. In fact, we strongly encourage you to ‘test’ the things we say or check to see if your intuitions about things using toy examples in R. You can use the code below to confirm that the ‘shape’ of the normal distribution is just an exponentiated parabola:

par (mfrow = c(1,2), mar = c(4,4,1,1))  
# draw an parabola  
curve (-x^2, xlim = c(-3,3), lwd=2)  
# same thing but exponentiated  
curve (exp (-x^2), xlim = c(-3,3), lwd=2)

The code below will generate densities for normal distributions based on the equations in the text above:

# you can change the ranges, mean and standard deviation (sigma) to any value  
x = seq (-4,4,.01)  
mu = 0  
sigma = 1  
eq25 = 1/(sigma\*sqrt(2\*pi)) \* exp (-(1/(2\*sigma^2))\*(x-mu)^2)  
eq27 = log(1/(sigma\*sqrt(2\*pi))) + -(1/(2\*sigma^2))\*(x-mu)^2  
eq28 = -(1/(2\*sigma^2))\*(x-mu)^2 - log((sigma\*sqrt(2\*pi)))

And these plots can be used to confirm the relations outlined above. For example, the output of equations (2.7)) and (2.8)) should equal, and this can be confirmed below.

par (mfrow = c(1,2), mar = c(4,4,1,1))  
plot (x, eq25, cex=1.2, col=skyblue,pch=1)  
lines (x, exp(eq27), col = coral, lwd = 2)  
plot (x, eq27, cex=1.2, col=skyblue)  
lines (x, eq28, col = coral, lwd = 2)

### 2.3.4 The standard normal distribution

The **standard normal distribution** is a normal distribution with a mean of zero and a standard deviation of one. Variables drawn from a standard normal distribution are often represented by the symbol (sometimes called a **z score**). Any normally distributed variable can be turned into a standard normal variable by an operation known as **standardization**, which consists of **centering** and then **scaling** the variable as in (2.9). To center a variable we subtract the mean from the value of each observation, making the new mean equal to zero. By dividing our observations by the standard deviation, we scale these values so that the new standard deviation is equal to one (since anything divided by itself is equal to one).

Equation (2.10) re-arranges the terms in (2.9) to isolate on the left-hand side. From (2.10) we can see that any normally-distributed variable can be thought of as a standard normal that has been multiplied by a standard deviation and then had a mean added to this product.

Normally-distributed data is often discussed in terms of ‘standard deviations from the mean’. This is because stating things in terms of standard deviations from the mean effectively standardizes a variable, making all variables seem standard normal. For example, if someone says “my test score was two standard deviations above the mean” what do you know about their score? You don’t know what the mean is, nor what the standard deviation is. All you know is that their score is two distance units (standard deviations) above the mean, so their test score can be thought of as a standard normal variable like . This is despite the obvious fact that the true average test score was not zero and the true average standard deviation was not one. This is an extremely useful property because it means that we can discuss the probability of any given event from any given distribution in consistent terms. For example, we can say that an observation four standard deviations from the mean is very unusual in *any* normal distribution. This means that if you have an observation equal to 140, the mean is 174, and the standard deviation is 8, you know that this observation is very improbable. That is because it is 4.25 (34/8) standard deviations from the mean of 174. Figure (2.4) presents our height data again, but this time compares the data to its centered and scaled versions.

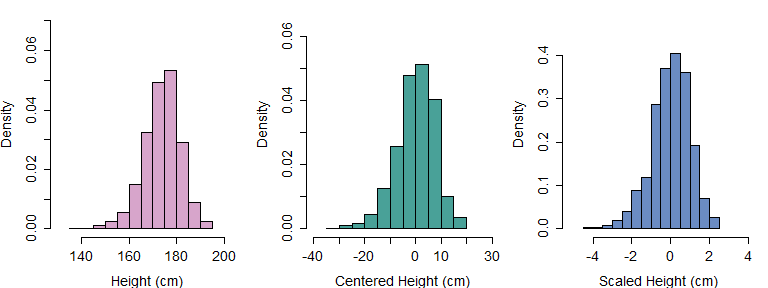


Figure 2.4: (left) A histogram of adult male height judgments in our data. (middle) The same data from the left panel, this time the data has been centered around the mean. (right) The same data from the middle plot, this time the data has been scaled according to the standard deviation.

## 2.4 Models and inference

Models are simplified representations that help us understand things. For example, we may want to understand the movement of balls on a billiards table, perhaps to create a video game about playing pool. To do this we may assume the balls are spherical and that their mass is evenly distributed among its volume. Neither of these things are exactly true but assuming this helps us keep our model simple and manageable. It also helps us build our model in terms of things, like regular spheres, whose properties and behaviors are well understood and are easy to work with. To build a real *exact* model we would need to include friction from the felt on the table, the effect of wind resistance, the gravitational effect of the moon, and a large number of other factors. As a result, a perfect or exact model is not really possible for most things (and maybe for anything). And yet, the simulation of realistic behavior of billiards balls is easy and can be done with great accuracy, suggesting that a simplified model can still be useful to understand the behavior of the more-complicated phenomenon it is meant to represent.

In general, it’s impossible to know what the ‘true’ data distribution is, so that *perfect* inference is not possible. As a result, scientists often use theoretical probability distributions to make inferences about real-life populations and observations. If our measurements more or less follow the ‘shape’ predicted by the theoretical normal distribution, we may be able to use the characteristics of an appropriate normal distribution to make inferences about our variables. Using a normal distribution to make inferences about your data is like using a mathematical model for spheres to understand the behavior of billiard balls. The billiards balls are *spherical enough* to allow us to make useful predictions based on the simplified model.

It is useful to keep in mind that reality will never exactly conform to our model. This can result in unpredictable errors in our conclusions. In general, the things you don’t know you don’t know are the things that will cause the most problems. If you had known that your model was wrong, you would have fixed it! Further, using models to make inferences about the general properties of data assumes that the things you have not seen are more or less just like those you have. Under those conditions then the conclusions you draw may be reliable. It is important to keep this limitation in mind, because you never know for sure that what you have not seen will conform to your model, and as a result the fit between a model and some set of observations can never definitively *prove* the truth of the relations encoded in the model (this is the problem of induction, discussed in section 1.2).

Parametric distributions can be used to establish **theoretical probabilities**, that is expectations about which events are and are not likely based on the general shape expected for the distribution. Basically, if we expect our distribution of values to have the shape of the normal distribution, we can use the shape of the normal distribution to make inferences about our distribution of values. When we used *empirical probabilities* above, our probabilities were estimated only with respect to the data we observed. However, when we refer to theoretical probability distributions we can also think about the behavior of values we did not observe, or the behavior of the distribution in general. In order to calculate theoretical probabilities you first needed to commit to a model of the data. You may be thinking, what model? It may seem too simple to be a model, but by assuming that our data can be understood as coming from a normal distribution with some given and , we have already created a simple model for our data. This is analogous to committing to a spherical shape for our model of our billiards balls: Saying that you expect your data to be normally distributed commits you to a certain distribution ‘shape’ and to more and less probable parameter values for your variable.

In figure 2.5 we compare the histogram of apparent height judgments to the density of a normal distribution with a mean equal to the sample mean () and a standard deviation equal to the sample standard deviation () of our mens\_height vector. The density was drawn using the dnorm function, which draws a curve representing the shape of a theoretical normal distribution with a given mean and standard deviation. Clearly, there is a good alignment between our random sample of real-world data and the theoretical normal density. This suggests that we could potentially use the *theoretical* shape of the normal distribution to talk about the characteristics of our observed random sample of data. Although the distribution of our sample is unlikely to be perfectly normal, it is *normal enough* to make the comparison worthwhile.

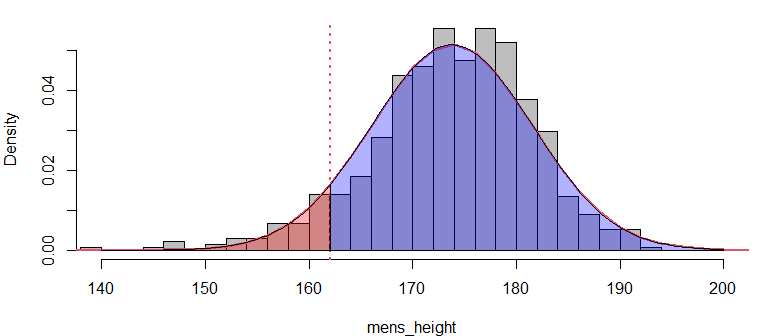


Figure 2.5: The histogram shows the empirical distribution of height judgments for adult male speakers. The shaded area shows the theoretical density of the equivalent normal distribution. The read area corresponds to the theoretical probability of observing a height under 162 cm.

The average female over 20 in the United states is 162.1 cm (CDC). A vertical line has been place at this value in figure 2.5. We might wonder, what is a probability of observing an height judgment for an adult male that is shorter than this average adult female height? Asking this question is equivalent to asking: What is the area under the curve of the density above, to the *left* of the vertical line? Since the *total* area of the density is always equal to one, the area of the red portion below corresponds to a percentage/probability of observing values less than 162.1 cm. One way to answer this question is to calculate the empirical probability of observing an apparent height less than 162.1 cm for male speakers in our data. Another way to do this is by calculating the *theoretical probability* by finding the proportion of values expected to be less 162.1 Hz in the normal distribution that has approximately the same ‘shape’ as our data distribution (i.e. the one seen in figure 2.5).

Below, we use the function pnorm to find the proportion of values that are expected to be less than 162.1 cm. This function takes in a value, a mean, and a standard deviation. It then tells you the proportion of the distribution that is to the *left* of (i.e. less than) a given value. Below, we use parameters estimated from our sample to run the pnorm function, as these are our best guesses of the population parameters. The output of this function is equal to the area of the red section in the density above. As we can see, the theoretical and empirical probabilities are very similar to one another. If we subtract this area from one, we get the area under the curve to the *right* of the vertical line, the blue section of the density above.

# empirical probability of height < 162.1  
mean (mens\_height<162.1)  
## [1] 0.07703704  
  
# Red area of distribution, x < 162.1  
pnorm (162.1, mean (mens\_height), sd(mens\_height))  
## [1] 0.06610382  
  
# Blue area of distribution, x > 162.1  
1 - pnorm (162.1, mean (mens\_height), sd(mens\_height))  
## [1] 0.9338962

Imagine you had 1 pound of clay and you were asked to make a shape *exactly* like the normal density above. This shape should be perfectly flat, i.e., it should have a constant depth (like a coin). If you had this shape made of clay used a knife to remove the part to the left of 162.1 cm (the red subsection) and weighed it, it should weigh 6.6% of a pound (0.066 pounds). The ‘area under the curve’ of this clay sculpture would just correspond to the amount of clay in a certain area, and in this case we know that only 6.6% of the clay should be in that section of the shape. So, the area under the curve, the probability, is just the amount of the *stuff* in the density that falls below/above a certain point, or between two points. The pnorm function allows you to slice and ‘weigh’ the sections of the distribution to tell you how much of it is in any given section.

What our theoretical probabilities tell us is this: *If* height judgments come from a normal distribution, *and* that distribution has a mean and standard deviation that is close to the sample estimates, then we expect (in the long run) that 6.6% of height judgments will be lower than 162.1 cm. What we are expressing here is effectively a conditional probability, we are saying *if* the parameters have certain values, *and* the probability distribution has a certain shape, then we expect certain height judgments to be more or less probable. Of course, if you change any part of that, either the values of the parameters or the probability distribution, then your estimated theoretical probabilities are likely to change. This is an important thing to keep in mind because it means that inference based on theoretical probabilities can change when our assumptions change, and many of these assumptions cannot be ‘proven’ to be true.

## 2.5 Probabilities of events and likelihoods of parameters

We’re going to change from talking about *probabilities* to talking about *likelihoods*. Probability is the odds of observing some data/event/outcome, given some parameter(s). A **likelihood** inverts this, and places odds on different *parameter* values given some observed data. For example, you could say “how probable is it that a random man from the US will be sound shorter than 162.1 cm in height?”. In contrast, you could ask “how likely is it that the average man from the US sounds 162.1 cm tall?”. The likelihood of a parameter represents the joint probability (density) of observing all the data you observed, given a certain parameter value. In other words, the likelihood relates to the probability that a specific probability distribution would produce your first observation, *and* your second observation, *and* your third observation, and so on, for all observations.

The **likelihood function** is a curve showing the relative likelihoods of different parameter values, given a fixed set of observations/data. The likelihood function tells you what parameter values are *credible* given your data. If a value is very unlikely, that means that it is not supported by your data. In other words, unlikely parameter estimates represent conclusions that your data is rejecting as not viable, and hence they are not **credible**. Every parameter for every probability distribution has a likelihood function, given some data. Here, we’re only going to discuss the likelihood of the normal mean parameter, , in detail.

The likelihood of the sample mean reflects the joint probability of observing all of your data, given different values of the mean. An example of how this is calculated is given in Figure 2.6. The left panel shows the likelihood function for based on a random sample of ten height judgments from our data (indicated by the blue points at the bottom of the plot). We can see that the most *likely* values of are centered on the bulk of the observations, and that values become less likely as we deviate from them. The vertical dotted lines indicate three possible mean values that will be highlighted in this discussion.

The likelihood of a parameter value (e.g., = 174 cm in the right panel of Figure 2.6) is equal to the product of the density of each observation in the sample, given the value of the parameter. This sounds like a mouthful but is actually deceptively simple. For example, to calculate the likelihood that , we:

1. Assume that the data is generated by a normal distribution with and equal to your sample standard deviation (7.8 cm).
2. Find the the height of the curve of the probability distribution (the density) over each point (indicated by the vertical lines in the right panel below). This reflects the relative probability of each observation given your parameter value.
3. The joint probability of all of the observations (the likelihood) is the product of all of these densities (heights). This assumes that all of your observations are statistically independent of each other (see section 2.2.2).

So, the value of the likelihood function in the left panel of Figure 2.6 at 174 Hz (rightmost vertical line) is equal to the product of the probabilities of the points in the right panel (i.e., the heights of the lines in the panel). Imagine we did this for a range of values along the axis, recording the likelihood values at each step. If we do this and then plot the product of the densities for each corresponding value the result would be a curve identical to that of the left panel in figure 2.6.

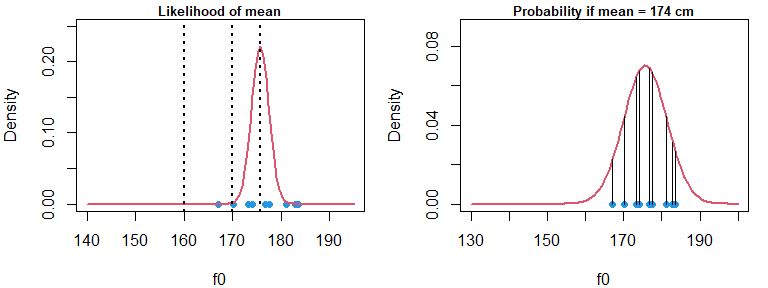


Figure 2.6: (left) The red curve indicates the likelihood of the population mean given the blue points in the figure. The vertical lines indicate three different parameters, or hypotheses, that will be considered. (right) The red curve indicates the probability of the points given an assumed mean of 174 cm. Vertical lines highlight the dentisty over each point, given the assumed mean

The right panel of figure 2.6 shows the probability of points assuming that the population mean is equal to the sample mean (for our tiny sample). We can see in the left panel of figure 2.6 that this is the most likely value for the mean parameter. When we said earlier that the sample mean is the ‘best’ estimate of the population mean, what we really meant was that the sample mean is the **maximum-likelihood** estimate of the population mean. This means that the sample mean provides an estimate of that maximizes the value of the likelihood function given the data. This is related to the fact that, as mentioned in section 2.3.1, the sample mean minimizes the variance of the sample. As a result, if you want to know which mean estimate is most likely given your data, you simply need to calculate the sample mean as in equation (2.1).

In the left panel of figure 2.7 we see that a normal distribution with a of 170 cm is a reasonable fit to the data. However, several observations are very improbable and this relative lack of fit is reflected by the low value of the likelihood function at 170 in Figure 2.6. In the right panel of Figure 2.7 we see that a normal distribution with a mean of 160 is very unlikely to generate this data: Many points are extremely improbable and have densities close to zero. Correspondingly, the value of the likelihood corresponding to in figure 2.6 reflects a very unlikely parameter value.

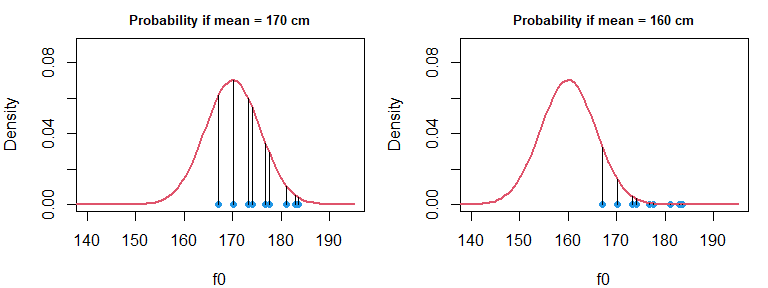


Figure 2.7: (left) The red curve indicates the probability of the points given an assumed mean of 170 cm. Vertical lines highlight the dentisty over each point, given the assumed mean. (right) The same information as in the left plot, but given a mean of 160 cm.

We need to talk about why it makes sense to multiply densities to calculate joint probabilities. Above, we stated that probabilities relate to areas under the curve. A problem we have is that the area under the curve of a single point is always zero. This is because a single point is so thin (it’s width is basically zero) that the area under the curve under the point will equal zero regardless of its density. Suppose that we said “ok let’s agree to use a fixed width, , around our point to calculate our area under the curve”. We can make so very tiny that is is almost as if we were calculating just the area under our density. We can use to represent the height of a density for a given value of . This means that we could approximate the value of the area under the curve at by treating it like a trapezoid and finding . If we wanted to calculate the joint probability of a number of observations, we could multiply a series of these areas, as seen in (2.11).

Since each term in (2.11) contains , this can be factored out as in (2.12).

At this point, we see that the joint probability of a set of values () can be thought of as the the product of the densities over those values () times some arbitrary constant . If we agree to use the same constant for all our calculations, we can ignore it and use the product of the densities to evaluate the relative probabilities of different combinations of observations.

### 2.5.1 Characteristics of likelihoods

We can think about the characteristics of the likelihood of to make some predictions about its expected shape given different numbers of observations and underlying data distributions. The likelihood is the joint probability of your data given different some parameter values. We can calculate the joint probability of two observations and by multiplying their individual densities as in equation (2.13), assuming these observations are independent.

Equation (2.13) defines the likelihood of given the data and and some . We will update the left-hand side to reflect this, replacing a term representing the joint probability of and () with a term representing the likelihood of the mean given the data, . Notice that nothing has changed except our perspective. In equation (2.13) we are treating the parameter as fixed and using it to calculate the joint probability of some data. In equation (2.14) we are treating the data as fixed and using it to calculate the likelihood of values of .

We can log-transform both sides of the equation to turn multiplication into addition, get rid of the exponentiation, and re-arrange the terms as in equation (2.15) (as explained for equation (2.8) above). Because they make working with likelihoods much simpler, statisticians often refer to the logarithms of likelihood functions, referred to as **log-likelihoods**, denoted using the symbol . When you see this, , just think “the logarithm of the likelihood of the mean given the data ”. In equation (2.15) we see that the log-likelihood of these two observations is the sum of two parabolas. Since the sum of parabolas will (almost always) also be a parabola, we know that the log-likelihood of the mean given these, and any number of other observations, will also be a parabola. As a result, we see that the (non-log) likelihood of the mean is an exponentiated parabola and has the same general shape as the normal distribution.

Equation (2.15) is specifically for two variables, however, Because each term is identical except for , the log-likelihood function can be greatly simplified as in equation (2.16). In (2.16) we can see that the log-likelihood of given observations of is equal to the sum of the individual squared deviations from the mean, divided by , with the value subtracted from it. So, we see that the log-likelihood is a parabola that has its vertex at . This means that this parabola has its maximum value when is equal to the sample mean. This again, reflects the fact that the sample mean is the maximum-likelihood estimate for the population mean.

There are two mechanisms by which the likelihood function may get narrower or wider. The first is the mechanism discussed in section 2.3.3 where a smaller value of results in a narrower parabola (and narrower likelihood), and it also applies here. However, there is another way that likelihood functions can become narrower, and that is by increasing the sample size. To see why this is the case, consider what happens to the relative value of our parameter () as the sample size grows. Imagine that we are calculating the sum of squared deviations about our sample mean. We know that the average squared variation we expect from our mean is equal to the variance. So, let’s replace in (2.16) with in (2.17). Now, instead of adding *N* squared deviations around the mean, , we are multiplying our expected squared deviation by *N*, .

We can also move the over from under the to make the following point simpler, as in (2.18).

Multiplying the numerator of a fraction is exactly equivalent to dividing the denominator of the fraction by the same amount, as in for example . Thus, we see that calculating the likelihood using data points is expected to have about the same effect on parabola width as dividing the data variance by , as shown in (2.19). As a result, increasing numbers of observations reduce the uncertainty in parameter estimates by making the likelihood narrower and narrower.

### 2.5.2 Making inferences using likelihoods

In Section 2.4 we discussed using the normal distribution to make inferences about the probable values of a random variable. When variables are normally distributed we can use the theoretical normal distribution and functions such as pnorm to answer questions about values we expect, and don’t expect, to see. In the same way, we can use likelihood functions to understand probable, and improbable, values of parameters given our data. For example, suppose that you measured the heights of 100 women in a small town (pop. 1500) and found the average height was 160 cm, with a standard deviation of 6 cm. You might accept that the *actual* population average is 161 cm, but may find it difficult to accept that it was actually 180 cm. This is because a true mean of 180 cm is *unlikely* given your observed data: The observations you have are *improbable* given a true mean of 180 cm. The logic is quite simple: a true mean of 180 cm is unlikely to ‘produce’ so many women around 160 cm. You *know* you observed the short women, therefore, you have no reason to believe that the true mean is 180 cm.

Below we calculate the likelihood of different mean parameters given our data. We do this by finding the log-density of each observation and then adding the points together. Log-densities are used because the likelihood is often a number so small that computers have a hard time representing them otherwise. For example, the highest point of the probability densities in figure 2.7 is about 0.07. Let’s pretend it’s 0.1 for the sake of simplicity. Recall that to find joint probabilities we need to multiply the densities above each of the points. This means that the probability of two observations at the mean is , and the probability of observing observations at the mean is equal to . Since we have 675 observations in our mens\_height vector, the probability of observing every one of those at the mean, the most probable outcome, would be equal to , or a decimal point followed by 674 zeros and then a one.

By relying on the logarithms of probabilities instead, we can accurately represent very small numbers more comfortably. This is because adding together the logarithms of two numbers is equivalent to multiplying those numbers, and multiplying a logarithm by another number is equivalent to raising it to that power (see (2.6)). For example, the number can also be expressed like , which equals -1554.245 (i.e. ). This is obviously a much easier number to deal with than one with 675 decimal places.

# make candidates for mean parameter  
mus = seq (172.5,175, .01)  
  
# easy way to make zero vector of same length as above  
log\_likelihood = mus\*0  
  
# add the log-density of all observations. Notice only the mean changes.  
for (i in 1:length(mus))   
 log\_likelihood[i] = sum (dnorm (mens\_height, mus[i], sd(mens\_height), log = TRUE))

In figure 2.8 we plot the log likelihood and the ‘scaled likelihood’. This is just the likelihood that has been scaled so that its peak is equal to one, and the peak of the log-likelihood equals zero. This allows us to actually plot and consider the likelihood function, though the values of the density no longer reflect the actual values of the likelihood function. However, the *relational* characteristics are maintained by this scaling. So, a scaled likelihood value of 0.2 is still five times less likely than a value of one.

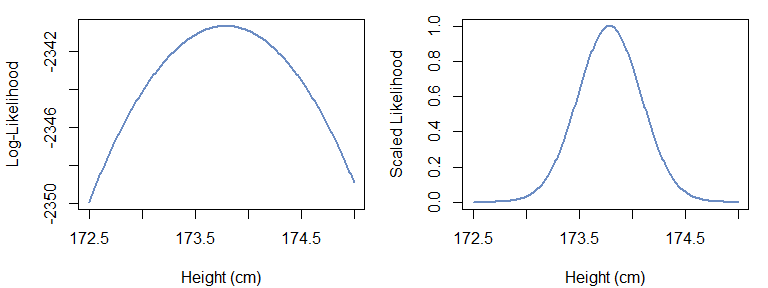


Figure 2.8: (left) Histogram of data. (right) Histogram of centered data, basically the error distribution.

At this point we can provide answers to the questions posed in section 2.1. The questions were:

Q1) How tall does the average man from the US sound?

Q2) Can we set limits on credible average apparent heights based on the data we collected?

Below, we see that the maximum likelihood estimate for the mean corresponds to the sample mean:

# find index number of highest values in log-likelihood  
maximum = which.max(scaled\_log\_likelihood)  
  
# print and compare to sample mean  
mus[maximum]

## [1] 173.79

mean (mens\_height)

## [1] 173.7877

So, we may conclude that the average male speaker is *most likely* to sound about 174 cm tall. We can also conclude informally based on Figure 2.8 that the most likely mean values fall between (approximately) 173 and 174.5 cm. This means that although the sample mean was 174 cm, it is reasonable that the true population mean might actually be 173.5 cm. This is because mean parameters in this range are also reasonably *likely* given our data. Basically, maybe our sample mean is wrong and arose by accident, and 173.5 cm is the true population . This outcome is compatible with our data. However, a value of 172.5 cm is very *unlikely* given our data. Since we think that 172.5 cm is not a plausible mean parameter given our sample, we can rule it out as a plausible value for the parameter of the underlying probability distribution. Using this approach, we can use the information in likelihood functions to rule out implausible values of based on the characteristics of our data.

# 3 Fitting Bayesian regression models with brms

In this chapter we’re going to start to answer basic research questions with Bayesian regression models using the brms package in R. The model we’ll use initially is not ‘correct’ for our data, but it is simple enough to work as an introduction to Bayesian regression models. In the next chapter we’ll use brms to build models that are closer to ‘correct’ given the structure of our data. Before using a Bayesian regression model to investigate our data, we will explain what we mean by *regression model* and what specifically makes the models in this book *Bayesian*. We’ll leave a discussion of the ‘multilevel’ aspect of our models for the following chapter.

## 3.1 What are regression models?

It’s difficult to offer a precise definition for **regression** because the term is so broad, but regression models can be thought of as models that help you understand systematic variation the mean parameter () of a normal distribution. Actually, you can model variation in other parameters (and use other probability distributions), but for now we will focus on models based on the normal distribution. Basically it goes like this:

1. You have a variable you are interested in, , which is is a vector containing observations. We can refer to any one of these observations like this for the observation. Although it’s not necessary, we’re going to put the index variables associated with trial number () in brackets like this . This is just to make it easier to identify, and to highlight the similarity to vectors (e.g., mens\_height[i]).
2. You assume that the random variation in your data is well described by a normal probability distribution. This is a mathematical function () that describes what is and is not probable based on two parameters. You also assume that the random variation in your data is independent. This means, for example, that you can’t really say why any two observations are above or below the mean.
3. The mean of this distribution is either fixed, or varies in a systematic manner. The standard deviation of the error distribution is usually fixed, but can vary (more on this in chapter X).
4. The variation in the mean of this distribution can be understood using some other variables, and regression tries to understand these relations.

We can write our model more formally as in Equation (3.1). This says that we expect that the tokens of the variable we are interested in are distributed according to () a normal distribution with parameters equal to and . Notice hat gets a subscript while and do not. That means that in this model those parameters are fixed for all observations, while the value of changes for each observation based on the subscript.

We’re going to learn to read and write formal descriptions of our models. The relationship between statistical concepts and the formal notation used to represent them is very similar to the ability to play music and musical notation. Someone who can play a song undoubtedly *knows* that song. However, in the absence of formal musical training that same person might not recognize sheet music representing the song. This person would also lack the vocabulary to discuss components of the song, and may find it difficult to learn to play new pieces. In the same way, most people have an excellent intuitive understanding of many statistical concepts (to be discussed in upcoming chapters) such as slopes, interactions, error, and so on. However, they lack the notational knowledge to understand these concepts when expressed in the formalisms often used to express these ideas. As a result, learning to read/write this notation will help you describe your models efficiently, and understand the models used by other people more effectively.

Equation (3.1) formalizes the fact that we think the *shape* of our data distribution will be like that of a normal distribution with a mean equal to and a standard deviation equal to . When you see this, , picture in your mind the shape of a normal distribution just like if you see this you may imagine a parabola. Really just represents that shape of the normal distribution, and the associated expectations about more and less probable outcomes. The above relationship can also be presented as in Equation (3.2).

Notice that we got rid of the symbol, moved out of the distribution function (), and that the mean of the distribution function is now 0. Under this interpretation, it is not the case that the entirety of the variable is random and unknowable. This representation breaks up our variable into two components:

1. A systematic component, the **predicted value** , that is expected for the variable .
2. A random component, the *error* , that causes unpredictable variation around .

Regression models separate variables into their **systematic** and **random** components. In this case, the systematic component is predictable for all observations in the data. The random component represents the *noise*, or *error* in our data, random variation around our predicted values that can’t be explained. This doesn’t mean that it’s inexplicable in general, it only means that we’ve structured our model in a way that doesn’t let us explain it. In other words, a model like this thinks all variations from the mean as noise because it is structured in such a way that treats all deviations as noise.

In regression models, we can try to understand variation in using predictor variables . These predictor variables co-vary (vary with) our variable, and we think help explain the variation in . For example, in (3.3) we’re saying we think is equal to some combination of the three predictor variables , , and . For example, we might expect that the apparent height of a speaker is affected by their f0 (), vocal-tract length (), and resonance (). So, when we combine these predictors we think we can come up with a pretty good estimate of how tall someone will sound.

The values of the predictor variables will vary from trial to trial, and are not fixed. In fact, often the whole point of running an experiment is to predict differences in observations based on differing predictor values. If we expect our predictors to vary from trial to trial, that means that the equation above should include subscripts indicating that the equation refers to the value of the predictors *for that trial* rather than overall. If we expect the predictors to change from trial to trial, naturally it is possible that may take on different values from trial to trial, and it therefore also needs an subscript. This update is reflected in equation (3.4).

The predicted value () for a given trial is very unlikely to be an equal combination of the predictors (as in equation (3.4)), so that a *weighting* of the predictors will be necessary. We can use the symbol for these weights as in equation (3.5). For example, maybe is twice as important as the other two predictors and so , while and . Actually, maybe one predictor has a *negative* effect so that . The ‘weights’ associated with each predictor are the **coefficients** (or parameters) of our model. Note that the weight terms () do not get an subscript. This is because they do not change from trial to trial. The *values* of the predictors change from trial to trial, but the way that these are combined does not, they are a stable property of the model.

We can insert equation (3.5) into equation (3.2) resulting in equation (3.6). At this point our model consists of an average value that has been broken up into three component parts, and the random component represented by normally-distributed noise.

Often, is used to represent the random component, the error term, as in equation (3.7). Notice that the error term *does* get an, subscript, as in . That is because the exact value of the error *does* change from trial to trial, even though the general characteristics of the error (i.e., ) do not. When expressed in this way this is now a *regression equation* or a *regression model*. **Fitting** a regression model basically consists of trying to find the ‘best’ values of its coefficients, , , and given our data and model structure.

Notice that according to equation (3.6), regression models do not require that our *data* be normally distributed, but only that the *random variation* in our data () be normally distributed. In (3.8), we see the sort of representation of our model that we will use in this book. This representation splits the systematic and random components of our regression model and clearly and succinctly informs us of the structure of out model. In plain English this is:

“We expect the average value of our variable to vary from trial to trial based on three predictors. The combination of these predictors is based on model specific coefficients that are static across trials. Our observations are expected to be randomly distributed around the mean value according to a normal distribution with a standard deviation equal to sigma”.

## 3.2 What’s ‘Bayesian’ about these models?

Traditional approaches to statistics (sometimes generally referred to as **frequentist**) estimate parameters by trying to find the most likely values for parameters. They do this by referring to likelihood functions based on theoretical probability distributions (i.e., maximum-likelihood estimation) as discussed in Chapter 2. We’re not going to discuss this approach to statistical inference in any detail as there are hundreds, if not thousands, of books available on the subject (for reading on the subject see section X). Rather than dwell on these ‘traditional’ approaches to statistical inference, we’re going to focus on what makes Bayesian inference *Bayesian*. Rather than rely on the likelihoods of parameter values, Bayesian models rely on the **posterior probabilities** of parameters. To explain what posterior probabilities are we need to talk about joint probabilities again. We’ll begin by stating something obvious: The probability of events and occurring is the same as the probability of and occurring.

Equation (3.9) can be reformulated as in (3.10) (see section 2.2.2).

Recall that and are simply placeholders for the probability of some hypothesis or assertion. We can replace these with and , which represent “the probability of observing your data” and “the probability of observing a certain parameter value” respectively. This is seen in equation (3.11), which states that “the probability of the parameter and the data is the same as the probability of the data and parameter”.

We can isolate on the left hand side by dividing both sides by , resulting in equation (3.12). When structured in this way, this is called **Bayes theorem**, and it underlies Bayesian statistical inference. Note however, that all we did was state a basic principle of probability theory (equation (3.9)) and then re-arrange some terms.

Each of the components in (3.12) have a name:

1. is the **prior probability** (of the parameter). This is the **a priori** probability of parameter values *before* the current experiment, and independently of the data . This a priori expectation can come from world knowledge, previous experiments, common sense, or some combination thereof. For example, before you measure the height of adults in San Francisco, you know the average is not 4 feet and it is not 7 feet.
2. is the likelihood. This is the probability that the data would be observed/generated for particular values of . The likelihood establishes the distribution of possible/credible parameter values given the *current* data and probability model. If we assume our data is normally distributed, can be thought of as the probability density of individual points given normal distributions as specified by .
3. is the **posterior probability**, the probability of parameter values *after* your current experiment. This is the **a posteriori** probability that the parameter is given your data . You get the posterior probability by combining the prior distribution and the likelihood, and in doing so combining your prior beliefs with your current observations.
4. is the **marginal probability**. This is necessary to scale the numerator so that the posterior density has a total area under the curve equal to one. However, you will note that the marginal probability does not vary as a function of . As a result, this does affect the posterior probability of different values of , and so you don’t really need to worry about the marginal probability very much.

As noted above, ‘traditional’ models focus primarily (or exclusively) on how likely different conclusions are given your data. In contrast, Bayesian models focus on the posterior probability of different parameter values, that is on the combination of the likelihood and prior probabilities of parameters.

#### 3.2.0.1 Prior probabilities

In a Bayesian model, every parameter whose value is being estimated needs a prior probability distribution to be specified. For example, imagine you are interested in estimating the mean of a set of values, . You decide to use a Bayesian model and decide that will have a normal prior with a mean and standard deviation equal to and . To estimate this model you would need to provide fixed values for and , for example we could use and . Note that the parameters of the prior do not get prior distributions themselves. This is because we are not estimating values of and , but only of .

The use of prior probabilities is sometimes said to make Bayesian models inherently ‘subjective’ but this concern is a bit overblown in most cases. First, in cases where you have plenty of data (as is often the case for experiments), prior probabilities often have little to no effect on outcomes. This is because when you have many observations, the posterior probability of parameters is dominated by the likelihood (as will be discussed in 3.2.0.2). Second, a researcher will always use ‘common sense’ (i.e. their prior expectations) to interpret their data. For example, if a listener reported that all adult males were 90 cm tall a researcher would have to wonder if this subject understands height in centimeters or if they were carrying out the experiment in good faith. So, even when they do not explicitly assign prior probabilities to parameter values, researchers still often use their expectations to ‘screen’ results in a manner broadly consistent with the use of prior probabilities in Bayesian reasoning. A Bayesian model simply requires that you build your expectations into your model. It *formalizes* it, makes it *definable* and *replicable*. Finally, every model involves arbitrary decisions which can substantially affect our results so that the design of a model can never said to be strictly ‘objective’. As a result,there is no particular reason to worry about the objectivity involved in establishing a prior in Bayesian modeling without also worrying about the objectivity involved in model building more generally.

#### 3.2.0.2 Posterior distributions

The calculation of posterior distributions involves the combination of the likelihood function with the prior probability distribution and the marginal probability. The marginal probability does not affect the ‘shape’ of the posterior distribution, and exists to scale the posterior so that the area under the curve is equal to one (to satisfy the requirements of basic probability theory). As a result, we will focus on the combination of the likelihood and prior probabilities.

The combination of the likelihood and prior probability distributions is straightforward conceptually: You multiply the values of the two densities at each x-axis location. This works because we are interested in the *joint* probability of the likelihood and the prior, and since these are independent of each other we know we can just multiply them. The resulting curve then represents the **joint density** of the two distributions. In Figure 3.1 several likelihoods and priors are combined, showing the effects of variations in these on posterior distributions. Each column of figure 3.1 differs in terms of the number of observations used to calculate the likelihood (n = 3, 10, 675), and rows differ in terms of the standard deviation of the prior probabilities of the mean with equal to 180 cm and = 100, 15, 1. In each case, the calculation of the likelihood assumes that the data has the same standard deviation as our apparent height data (7.8 cm). In each plot, all curves have been scaled to have the same height in the figure. This is only to make the figures visually interpretable but does not affect any of the points made below.

The different standard deviations used for the prior probabilities of encode different levels of prior belief regarding what heights of adult male speakers from the US. The top, middle, and bottom rows represent what can be referred to as **vague**, **weakly informative** and **informative** priors. The boundaries between these kinds of priors is somewhat fuzzy, however, they can be distinguished as follows. A vague prior provides almost no information about the expected range of plausible results, a weakly informative prior provides some information, and an informative prior provides a lot. You can think of these as three different approaches to using prior information.

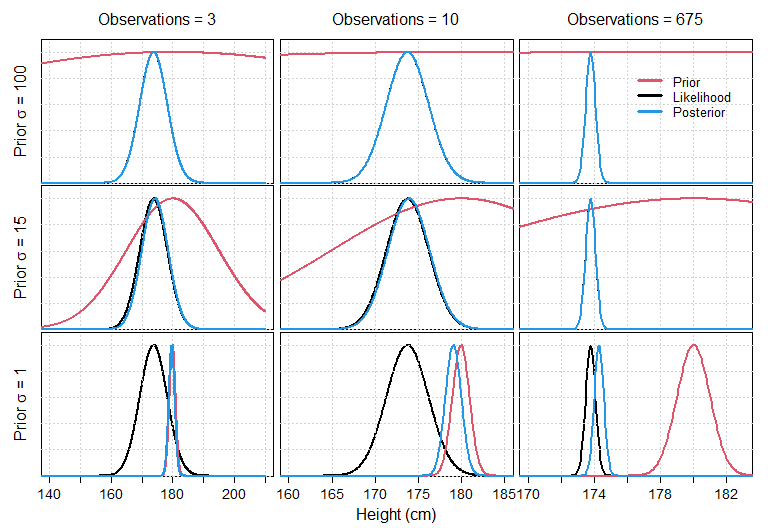


Figure 3.1: Demonstration of the effect of different prior probabilities and number of observations on likelihoods on posterior distributions.

In the top row of Figure 3.1 we see the influence of a *vague* or *diffuse* prior on inference. Since this distribution has a standard deviation of 100 and a mean of 180, this means that basically all heights from zero to 400 cm are plausible a priori. A prior probability this wide is only going to place very minimal constraints on the posterior probabilities. This is reflected in the top row of figure 3.1 where even in the case of only three observations the posterior probability almost exactly matches the likelihood. So, in the presence of very weak prior beliefs the most credible values for your parameters a posteriori (after experimentation) will be dominated by the most likely values. This makes sense.

In the middle row of Figure 3.1 we see the influence of a *weakly informative* prior. This distribution has a standard deviation of 15 and a mean of 180. The standard deviation was set to twice the value of the standard deviation of adult male heights in the United States (cite). As a result, it places reasonable constraints on our expectations but it will not overly influence results within that reasonable range. We see that in this case, the likelihood still dominates the posterior even when only five observations are available.

In the bottom row of Figure 3.1 we see the influence of a *informative* prior. This distribution has a standard deviation of 1 and a mean of 180. A prior distribution this narrow basically says that we are *sure* that the average height is around 180 cm. This is because the prior probability of any value outside of 178-182 cm is nearly zero, meaning that we do not believe values outside this bound before collecting our data. We can see that under these conditions, the prior distribution actually can have a strong effect on the posterior, especially in cases with few observations. However, with a large enough sample size the likelihood can still come to dominate even the narrowest prior distributions.

### 3.2.1 Characteristics of posterior distributions

If we focus on the general characteristics of posterior distributions with respect the the characteristics of priors and likelihoods, a pattern emerges. Specifically, the posterior is a mix of the prior and likelihood which takes their relative ‘variances’ into account. We know (see section 2.5.1) that the width of the likelihood function is dependent on the underlying error in the data and on the sample size: Error increases the width of the likelihood while more observations result in a narrower and narrower likelihood. So, we see that the characteristics of the posterior distribution will depend on: The ‘width’ of the prior distribution, the amount of noise in your dependent variable, and the number of samples involved in the calculation of your likelihood.

A consequence of the ‘merging’ of prior information with the likelihood is that posterior distributions can be **shrunk** towards the prior. Recall from chapter 2 that the maximum likelihood estimate of a parameter can be thought of as the ‘best’ parameter in that it is the parameter that makes your data as probable as it can be given the model structure. In figure 3.1 we see that in some cases the posterior distribution is not exactly like the likelihood and has been *pulled* towards the prior. This *pull* is referred to as **shrinkage**, because it tends to *shrink* the magnitude of effects by pulling them towards the mean (which is often zero). Broadly speaking, deviations from prior expectations are maintained when there is good enough evidence for them, and shrunk when there is not. What constitutes ‘enough’ evidence is based on the structure of the model and the nature of the data. As seen in figure 3.1 a wide enough prior will not meaningfully affect estimates even for extremely small sample sizes. You may wonder, is shrinkage a good thing? It turns out that shrinkage can help models arrive at more reliable parameter estimates by reducing weakly supported values that deviate substantially from prior expectations (cite). This will be discussed further in the following chapter.

## 3.3 Sampling from the posterior using STAN and brms

We want to understand the posterior distribution of parameters. How do we get this information? For very simple models posterior distributions can be understood **analytically**, that is by finding exact solutions to a series of equations. However, for more complicated models such as the ones we’ll discuss in this book, it can be difficult (if not impossible) to understand the characteristics of the posterior distributions of parameters in this way. As a result, these questions are answered **numerically** using software that randomly samples from posterior distributions. Given enough enough random samples from the posterior distribution, we can begin to draw the general expected shape of the distribution (just as with sampling from any other variable).

Many different software approaches to sampling from posterior distributions have been developed through the years including *winBUGS*, *JAGS*, and *STAN*. The software used in this book is **STAN**. We use this because it is (quick)relatively) fast, reliable, and widely adopted. However, the modeling and statistical principles explained in this book apply generally to all Bayesian models regardless of the software used to fit them, and the general concepts extend to all linear modeling more general.

One downside with working with STAN directly is that you need to write your own models. This is not too difficult, but it is not *easy* either, and it can be a bit time consuming especially for complicated models that we want to fit quickly. In this book we will rely on the brms package to use STAN. brms effectively simplifies the use of STAN by making the specification of highly-efficient models very simple, and providing us with a great deal of flexibility in doing so. It also includes many of helper functions that make working with Bayesian models very convenient. So, even though we will use brms for simplicity, everything we discuss in this book could be used to directly write your own models for STAN (or any other statistical software).

In order to sample from the posterior distribution using software like STAN, the user provide some data and a description of a model which specifies:

1. The relations between the variables in the data. For example, what is the dependent variable? What are the independent variables? How do these relate?
2. The nature of random variation in the model. To this point we have only discussed single sources of normally-distributed noise in our models.
3. Prior distributions for all estimated parameters.

Given this information, STAN takes random samples from the posterior distribution and returns them all, rather than looking for the single ‘best’ estimate. For each parameter, STAN does a more-complicated version of the following algorithm:

1. Pick a random value for the parameter (i.e., = 173 cm).
2. Calculate the posterior probability for the current estimate of .
3. If the posterior estimate meets some criteria (e.g., it is better than the last one, it is not too low, etc.), then the value of is recorded, and becomes . If not it is just discarded.
4. Go back to step 1.

The result of this process is a **chain** of estimates of , and any other parameter you are estimating in your model. This *chain* is a vector of sequential samples of the posterior distribution of , which together tell us about the characteristics of the parameter it represents. Incredibly, under a very reasonable set of conditions the process below will result in a distribution of that will converge on the posterior distribution of given your data and model structure (including prior probabilities).

## 3.4 Estimating a single mean with the brms package

### 3.4.1 Data and Research Questions

We’re going to use the same experimental data we looked at last chapter: The height judgments collected for the adult male speakers in our experiment. For more information on the experiment, see section 1.3.2. Below we load the book package and subset our experimental data to only include those rows involving adult male speakers. In addition, we will focus only on the natural productions (the actual resonance, men$R=='a'), excluding those trials involving the manipulated ‘big’ resonance level.

We’re going to revisit the research questions posed at the beginning of Chapter 2:

Q1) How tall does the average adult male from the US sound?

Q2) Can we set limits on credible average apparent heights based on the data we collected?

However, this time we are going to approach these questions using a Bayesian regression model using brms (and STAN).

### 3.4.2 Description of the model

We’re beginning with a model that treats all of our data as random deviations drawn from a single, undifferentiated normal distribution. Our model for a single group of normally distributed values can be thought of in several different ways. In (3.13) the value of your dependent variable for any given trial () is thought of as being a normally-distributed variable with a trial-specific mean of , and a fixed standard deviation .

We can also think of this model as in (3.14), which says that your dependent variable is the sum of some of some average *expected value* for that trial, () and some specific random error for that trial (). The random error is expected to be normally distributed with a mean of 0 and some unknown standard deviation (as in: ).

In general, we use regression models to understand orderly variation in from trial to trial by breaking it up into predictors () that are combined using based on weights as determined by the model coefficients (). However, in this case we expect the value of to actually be equal for all trials. When we are only trying to estimate a single average, we don’t have any predictors to explain variation in . In fact, our model structure suggests we expect no variation in from trial to trial. However, mathematically we can’t just say ‘we have no predictor’ since everything needs to be represented by a number. As a result, we use a single ‘predictor’ with a value of 1 so that our regression equation is as in (3.15)). Now, our model is trying to guess the value of a single coefficient (), and we expect this coefficient to be equal to since it is being multiplied by a ‘predictor’ with a constant value of 1.

This kind of model is called an **Intercept only** model. Regression models are really about representing *differences*, differences between groups and across conditions. When you are encoding differences, you need an overall reference point. For example, saying that something is ‘5 miles north’ is only interpretable given some reference point. The ‘reference point’ used by your model is called your ‘Intercept’, and it is the center of your model’s universe. At this point our model consists *only* of a single reference point, and the parameter reflects its value (as shown in Equation (3.15)). As a result, the coefficient is called the ‘Intercept’ in our model. When a coefficient is just being multiplied by a ‘fake’ predictor that just equals one, we can omit it from the regression model (but its still secretly there). So, our model investigating the apparent heights of adult males from Michigan can be formalized like this:

Put in plain English, each line in the model says the following:

1. We expect that apparent height for a given observation is normally distributed according to some trial-specific expected mean value and some unknown (but fixed) standard deviation.
2. The expected value for any given trial () is equal to the intercept of the model for all trials. This means its fixed and we have the same expected value for all tokens.

What the model also implicitly says that the error, the random variation around , is drawn from a normal distribution with a mean of 0 and a standard deviation of . This distribution represents all deviations in apparent height around the mean apparent height for the sample (). In other words, the error for this model is expected to look like:

We can rearrange the terms in (#eq:313b) to isolate the random term on the left side. When we do this, we see that **error** is what we call the difference between the value of an observation and the expected value for that observation.

In practice, you never know the true expected value, the *real* exact parameter for whatever distribution you are working with. Instead, you work with an estimate of the predicted value . As a consequence, you do not have access to the exact errors but instead to estimated errors . Estimated errors are called **residuals**.

As noted in section 3.1, regression models assume that the random error, the unpredictable deviations about the expected value for a trial, are independent and identically distributed. We can now be more specific and say that we expect that our model *residuals* to be independent of each other. This assumption is obviously violated for this data since we have multiple observations from each listener, each of which had their own tendencies (as discussed in section 2.2.1). For this reason, we can say that this model is ‘wrong’; it is built in such a way that we know it is not a good fit for our data. We will discuss this, and the problems it causes, in the following chapter.

### 3.4.3 The model formula

Model structures are expressed in R using a very specific syntax. Think of writing a model formula as a sub-language within R. Generally, model formulas in R have the form:

y ~ predictors

The variable we are interested in understanding () goes on the left hand side of the , and on our predictors go on the right hand side. Notice that the random term () is not included in the model formula. The formula above can be read as ‘y is distributed according to some predictor’, which really means “we think there is systematic variation in our y variable that can be understood by considering its joint variation with our predictor variable(s).”

For intercept only models, the number 1 is included in the model formula to indicate that a single constant value is being estimated (as in (3.15)). As a result, our model formula will have the form seen below. This model could be said out loud like “we are trying to estimate the mean height” or “we are predicting mean height given only an intercept”.

height ~ 1

### 3.4.4 Fitting the model: Calling the brm function

The brms package contains the brm function, which we will use to fit our models. The brm function takes a model specification, data, and some other information, and fits a model that estimates all the model parameters. Unless otherwise specified, brm assumes that the error component () of your model is normally distributed. The first argument in the function call is the model formula, and the second argument tells the function where to find the data (a dataframe called men). The other arguments tell the function to estimate a single set of samples (chains = 1) using a single processor on your CPU (cores = 1). These arguments will be discussed in more detail later.

# To ensure predictable results in examples, we will using the same random   
# seed throughout, and resetting it before model fitting.   
set.seed (1)  
model = brms::brm (height ~ 1, data = men, chains = 1, cores = 1)  
  
## Compiling Stan program...  
## Start sampling  
##   
## SAMPLING FOR MODEL '03859e54349182b6cd9cd51aa7ca25d3' NOW (CHAIN 1).  
## Chain 1:   
## Chain 1: Gradient evaluation took 0 seconds  
## Chain 1: 1000 transitions using 10 leapfrog steps per transition would take 0 seconds.  
## Chain 1: Adjust your expectations accordingly!  
## Chain 1:   
## Chain 1:   
## Chain 1: Iteration: 1 / 2000 [ 0%] (Warmup)  
## Chain 1: Iteration: 200 / 2000 [ 10%] (Warmup)  
## Chain 1: Iteration: 400 / 2000 [ 20%] (Warmup)  
## Chain 1: Iteration: 600 / 2000 [ 30%] (Warmup)  
## Chain 1: Iteration: 800 / 2000 [ 40%] (Warmup)  
## Chain 1: Iteration: 1000 / 2000 [ 50%] (Warmup)  
## Chain 1: Iteration: 1001 / 2000 [ 50%] (Sampling)  
## Chain 1: Iteration: 1200 / 2000 [ 60%] (Sampling)  
## Chain 1: Iteration: 1400 / 2000 [ 70%] (Sampling)  
## Chain 1: Iteration: 1600 / 2000 [ 80%] (Sampling)  
## Chain 1: Iteration: 1800 / 2000 [ 90%] (Sampling)  
## Chain 1: Iteration: 2000 / 2000 [100%] (Sampling)  
## Chain 1:   
## Chain 1: Elapsed Time: 0.103 seconds (Warm-up)  
## Chain 1: 0.057 seconds (Sampling)  
## Chain 1: 0.16 seconds (Total)

By default, brms takes 2000 samples, throwing out the first 1000 samples and returning the last 1000. The first 1000 samples are the **warmup**, the time the model uses to find appropriate parameter values for the model. The output above shows you that the sampler is working, and tells you about the progress as it works. This is a small amount of data and a simple model so it should be pretty fast. This is the only time we will be actually fitting a model in the code chunks. For all of the models discussed in this book, you can fit any of these models yourself locally using the code provided, or you can download them directly from the course GitHub using the code below.

# Download it from the GitHub page:  
model = bmmb::get\_model ('3\_model.RDS')

### 3.4.5 Interpreting the model: the print statement

Typing the model name into the console and hitting enter prints the default brms model print statement:

# inspect model  
model

The first part provides you with some basic information and part tells you some technical details that we don’t have to worry about for now (though some are obvious).

## Family: gaussian   
## Links: mu = identity; sigma = identity   
##Formula: height ~ 1   
## Data: men (Number of observations: 675)   
## Draws: 1 chains, each with iter = 2000; warmup = 1000; thin = 1;  
## total post-warmup draws = 1000

Next we see estimated effects for our predictors, in this case only an intercept. This is a **population-level effect** because is is shared by all observations in our sample, and not specific to any one observation.

## Population-Level Effects:   
## Estimate Est.Error l-95% CI u-95% CI Rhat Bulk\_ESS Tail\_ESS  
## Intercept 173.80 0.31 173.20 174.38 1.00 1038 598

The information above provides the mean (Estimate) and standard deviation (Est. Error) of the posterior distribution of (Intercept). The values of l-95% CI and u-95% CI represent the upper and lower 95% **credible interval** of the posterior distribution. An credible interval of a parameter is an interval such that the parameter has an chance ( probability) of falling inside the interval. brm calculates credible intervals using quantiles so that the l-95% CI and u-95% CI represent 2.5% and 97.5% quantiles of the posterior samples of a parameter. Based on its 95% credible interval, we see that there is a 95% probability that is between 173.2 and 174.4 cm given our data and model structure. Our model also provides us an estimate of the error standard deviation(), under ‘Family Specific Parameters: sigma’. This estimate closely matches our sample standard deviation estimate (sd(mens\_height)) of 7.76 cm.

In addition, we also get a 95% credible interval for this parameter (2.5% = 7.38, 97.5% = 8.21). Although our focus is often on estimation mean parameters, it is very imprtant to keep in mind that our model in (3.16) involves the estimations of *two* parameters, and .

## Family Specific Parameters:   
## Estimate Est.Error l-95% CI u-95% CI Rhat Bulk\_ESS Tail\_ESS  
## sigma 7.78 0.22 7.38 8.21 1.00 1060 736

This last section is just boilerplate and contains some basic reminders which will generally always look the same.

## Samples were drawn using sampling(NUTS). For each parameter, Bulk\_ESS  
## and Tail\_ESS are effective sample size measures, and Rhat is the potential  
## scale reduction factor on split chains (at convergence, Rhat = 1).

### 3.4.6 Seeing the samples

In section 3.3 we discussed that Bayesian modeling software (like STAN) takes *samples* of the posterior distributions of parameters given the data and model structure. It is helpful to see that our model is really just a series of **posterior samples**. Compact description of our models, such as the one in the print described above, are just summarizes of the information contained in the posterior samples. Below we get the posterior samples from the model we fit above, in the form of a matrix. As expected, we have 1000 samples of each parameter. The first column represents the model intercept (b\_Intercept), the middle column is the error (sigma). The third column (lp\_\_) is the **log posterior density**: the logarithm of the product of the prior and the likelihood, without dividing by the marginal probability (the importance of this value will be discussed in later chapters).

# get posterior samples from model  
samples = brms::as\_draws\_matrix (model)  
head (samples)  
## # A draws\_matrix: 6 iterations, 1 chains, and 3 variables  
## variable  
## draw b\_Intercept sigma lp\_\_  
## 1 174 7.7 -2346  
## 2 174 7.8 -2345  
## 3 174 7.6 -2345  
## 4 174 7.5 -2346  
## 5 174 8.1 -2346  
## 6 174 7.9 -2345

We can plot the individual samples for the mean parameter on the left in figure 3.2, and on the right we can see a histogram of the samples.

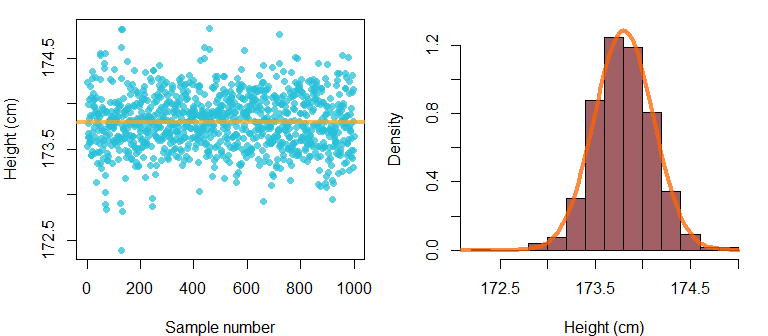


Figure 3.2: (left) Individual samples from posterior distribution of the model intercept paramter. (right) A histogram of the samples on the left. The curve shows a normal distribution with a mean of 173.8 and a standard deviation of 0.31.

Recall that our model output provides information about 95% credible intervals for the mean parameter:

## Population-Level Effects:   
## Estimate Est.Error l-95% CI u-95% CI Rhat Bulk\_ESS Tail\_ESS  
## Intercept 173.80 0.31 173.20 174.38 1.00 1038 598

We know that these simply correspond to the 2.5% and 97.5% quantiles of the posterior samples. We can confirm this by checking the quantiles on the vector containing our posterior samples and see that these exactly correspond to the values of Estimate, l-95% CI, and u-95% CI in the model print statement above.

quantile (samples[,"b\_Intercept"], c(.025, .975))  
## 2.5% 97.5%   
## 173.2024 174.3812

One of the great things about Bayesian models is that you can make your own summaries of the posterior samples, summarize them in several ways as required, and ask different questions easily. For example, there is no special status for the 2.5 and 97.5% quantiles, and we can easily check the values of other ones:

quantile (samples[,"b\_Intercept"], c(.25, .75))  
## 25% 75%   
## 173.5890 174.0052

We can also use the posterior distribution to find the probability that the mean parameter is over/under any arbitrary value:

mean (samples[,"b\_Intercept"] < 174)  
## [1] 0.746

Let’s take a second to think about why this works. Recall that the probability is the odds that something will occur, relative to all other outcomes. Our vector samples[,"b\_Intercept"] represents 1000 observations of a random variable, 1000 possible values of the average apparent height of adult males from the US. If we find the total number of these observations that were below 174 cm and then divide by the total number of observations (1000), we are calculating the probability of observing a mean estimate below 174 cm. As a result, the calculation above says that there is a 0.75 probability (a 75% chance) that the mean apparent height of adult male speakers in this population is under 174 cm, given our data and model structure. We come to this conclusion by finding that 75% of the posterior samples of the parameter of interest are below 174 cm.

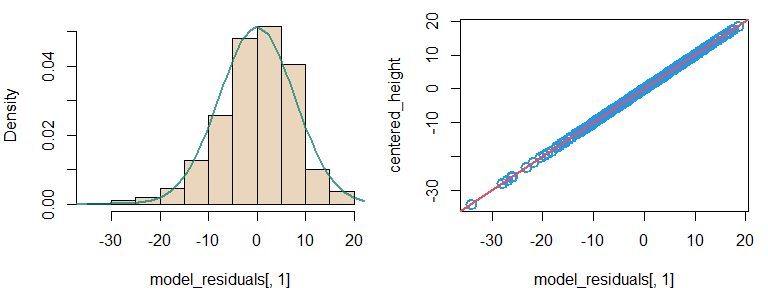
### 3.4.7 Getting the residuals

We can get the model residuals using the residuals function. By default it returns a data frame where one row corresponds to each observation in your data, and the different columns provide you information about the estimate. You will notice that you get credible intervals for each of the estimated residuals. This is because there is one prediction for each set of posterior samples. Since we have 1000 posterior samples that means we have 1000 slightly different predicted values for each observation and therefore 1000 slightly different estimated errors for each observation. By default the distribution of residuals is presented, but you can get the individual estimates themselves by calling residuals(model,summary=FALSE).

model\_residuals = residuals (model)  
head (model\_residuals)  
## Estimate Est.Error Q2.5 Q97.5  
## [1,] -2.0979616 0.3084632 -2.681229 -1.5023669  
## [2,] 2.5020384 0.3084632 1.918771 3.0976331  
## [3,] 4.0020384 0.3084632 3.418771 4.5976331  
## [4,] 5.0020384 0.3084632 4.418771 5.5976331  
## [5,] -1.5979616 0.3084632 -2.181229 -1.0023669  
## [6,] -0.7979616 0.3084632 -1.381229 -0.2023669

In the left panel of figure @ref(fig:F32.5) we show a histogram of our residuals and compare this to the standard deviation estimated for the error distribution in our model (sigma = 7.78). It is no surprise that these match because this is precisely what the sigma parameter in our model is an estimate of, the standard deviation of the residuals. Since our model consists of only an intercept, a single expected value () for all instances of the variable, all variation around the mean constitutes error. We can compare this by comparing our residual estimates to the centered data (right panel of figure @ref(fig:F32.5)) and seeing that these are basically the same.

par (mfrow = c(1,2), mar = c(4,4,1,1))  
hist(model\_residuals[,1],main="", col = bmmb::cols[13], freq=FALSE)  
curve (dnorm (x,0,7.8), xlim=c(-40,25),add=TRUE,lwd=2,col=deepgreen)  
  
centered\_height = mens\_height - mean(mens\_height)  
plot (model\_residuals[,1], centered\_height,lwd=2,col=4,cex=1.5)  
abline (0,1,col=2,lwd=2)



(#fig:F32.5)(left) Histogram of the residuals of our model. (right) A comparison of our residuals and centered height judgments shows that these are the same thing.

## 3.5 Checking model convergence

Our parameter estimates are based of a set of samples from the posterior distribution of a parameter. As with any other inference based on samples, our parameter estimates will be unreliable if we don’t have enough samples, or if our samples do not represent the population we are trying to understand. For this reason, it’s important to look at the **ESS** values (the *expected sample size*), and the **Rhat** values provided for brm model print statements. ESS tells you about how many *independent* samples you have taken from the posterior. Bulk ESS tells you how many samples the sampler took in the ‘thick’ part of the density, and Tail ESS reflects how much time the sampler spent in the ‘thin’ part, in the tails of the distribution. Rhat tells you about whether your ‘chains’ have converged (more on this later). As noted in the ‘boilerplate’ at the end of the brm model print statement, values of Rhat near 1 are good, and values higher than around 1.1 are a bad sign. Ideally we would like several hundred samples (at least) for mean estimates, and thousands to be confident in the 95% confidence intervals. If you fit a model and get a warning message like this:

## Warning messages:  
## 1: Bulk Effective Samples Size (ESS) is too low, indicating posterior means and   
## medians may be unreliable. Running the chains for more iterations may help. See:  
## http://mc-stan.org/misc/warnings.html#bulk-ess  
## 2: Tail Effective Samples Size (ESS) is too low, indicating posterior variances   
## and tail quantiles may be unreliable.  
## Running the chains for more iterations may help. See  
## http://mc-stan.org/misc/warnings.html#tail-ess

That is brms telling you that you need to collect more samples in order to be confident in your parameter estimates. To get more samples we can run the model longer, or we can use more **chains**. A chain is a separate set of samples for your parameter values. A model can be fit in parallel across several cores resulting in several independent chains. Since these chains are all supposed to be sampling from the same posterior distribution, their samples can be merged across chains after sampling. There is a fixed number of samples a single core of your computer can take in a fixed amount of time. When you do this across cores, you can get (approximately) times as many samples in the same amount of time. Since many computers these days have 4-8 (or more) cores, we can take advantage of parallel processing to fit models faster. Before fitting a model across multiple cores, you should confirm how many you have. You can use the following command (you may need to install the parallel package):

parallel::detectCores()

The example code throughout this book will use four cores to fit models. If you only have four total cores, change the models to use 2-3 chains and cores. One thing to keep in mind is that these models can be computationally intensive to fit. As the data sets become larger and the models become more complicated, more powerful computers are needed in order to fit a model in a reasonable amount of time. Below, we re-fit our initial model but run it on four chains, and on four cores at once.

# Fit the model yourself  
set.seed (1)  
model\_multicore =   
 brms::brm (height ~ 1, data = men, chains = 4, cores = 4)

# Or download it from the GitHub page:  
model\_multicore = bmmb::get\_model ('3\_model\_multicore.RDS')

We can print the model below, and can see that using four chains has substantially increased our ESS, without taking up much more computing time. Towards the top of our print statement we see that 4 chains have collected total post-warmup samples = 4000. This means our model has 4000 samples for every parameter in the model. However, for some parameter we have only about 3000 ‘effective samples’. This means some of our samples are basically dead weight, taking up space and slowing down future computations for no good reason. The discrepancy between the number of samples and the ‘effective’ number of samples is due to something called **autocorrelation**, the self-similarity of nearby observations in a series of observations. Sometimes consecutive samples can be too similar and so don’t given you that much *independent* information. When this happens you end up with less information about a parameter than you might think based on the number of samples you have. Think of it like measuring the temperature of a place to get an idea of its average annual weather. Measurements need to be well separated in order to be really independent. If you were to measure the temperature every 5 minutes these measurements would have a high autocorrelation, and would not give you a good impression of the range of temperatures that place tends to experience in a calendar year.

# inspect model  
model\_multicore  
## Family: gaussian   
## Links: mu = identity; sigma = identity   
## Formula: height ~ 1   
## Data: men (Number of observations: 675)   
## Draws: 4 chains, each with iter = 2000; warmup = 1000; thin = 1;  
## total post-warmup draws = 4000  
##   
## Population-Level Effects:   
## Estimate Est.Error l-95% CI u-95% CI Rhat Bulk\_ESS Tail\_ESS  
## Intercept 173.78 0.30 173.19 174.36 1.00 3920 2665  
##   
## Family Specific Parameters:   
## Estimate Est.Error l-95% CI u-95% CI Rhat Bulk\_ESS Tail\_ESS  
## sigma 7.77 0.21 7.38 8.20 1.00 2956 2804  
##   
## Draws were sampled using sampling(NUTS). For each parameter, Bulk\_ESS  
## and Tail\_ESS are effective sample size measures, and Rhat is the potential  
## scale reduction factor on split chains (at convergence, Rhat = 1).

One way to increase the ESS without increasing the total final sample size is to run longer chains and keep only every one. This strategy is called **thinning**. This lets your models be smaller while containing approximately the same information. To do this you have to change the iter, warmup and thin parameters when you fit your model. Default behavior is that the models you fit keep every sample after the warmup is done, up to the iter maximum. So if iter=3000 and warmup=1000 you will end up with 2000 samples. After this, you keep only one every thin samples. Basically, you will end up with samples per chain. If you are doing this across Ncores cores, then you will end up with samples in total. Below, we ask for 3000 samples per chain. Since the warmup is 1000 this means we will keep 2000 post warm-up. However, since thin=2, we will keep only 1000 of these. Finally, since we are fitting the model on four cores, we will end up with 5000 samples in total (i.e. ).

# Fit the model yourself  
set.seed (1)  
model\_thinned =   
 brms::brm (height ~ 1, data = men, chains = 4, cores = 4,  
 warmup = 1000, iter = 3000, thin = 2)

We inspect the model print statement and see that despite having the same number of samples as the model\_multicore, the ESS for this model is higher than for the previous model, in particular ofr the sigma parameter.

# inspect model  
model\_thinned  
## Family: gaussian   
## Links: mu = identity; sigma = identity   
## Formula: height ~ 1   
## Data: men (Number of observations: 675)   
## Draws: 4 chains, each with iter = 1500; warmup = 500; thin = 2;  
## total post-warmup draws = 2000  
##   
## Population-Level Effects:   
## Estimate Est.Error l-95% CI u-95% CI Rhat Bulk\_ESS Tail\_ESS  
## Intercept 173.78 0.29 173.22 174.35 1.00 3528 3547  
##   
## Family Specific Parameters:   
## Estimate Est.Error l-95% CI u-95% CI Rhat Bulk\_ESS Tail\_ESS  
## sigma 7.76 0.21 7.38 8.18 1.00 4063 3665  
##   
## Draws were sampled using sampling(NUTS). For each parameter, Bulk\_ESS  
## and Tail\_ESS are effective sample size measures, and Rhat is the potential  
## scale reduction factor on split chains (at convergence, Rhat = 1).

## 3.6 Specifying prior probabilities

In section 3.2 we mentioned that in Bayesian models all estimated parameters *must* have prior probability distributions specified for them. And yet, to this point we’ve been fitting models without explicitly specifying prior probability distributions for the parameters. It turns out that if you don’t specify prior probabilities for your parameters, brm will use its own default priors for parameters given the characteristics of your data. We can use the function get\_prior in brms to see what the default priors are for our model, and to see which parameters in our model require priors. Of course, we should know this based on the structure of our model but this method is useful to help verify our expectations.

Below we can see that our model requires priors for our two estimated parameters, the Intercept () and sigma () parameters, and that these have been given default values. The default values use a t distribution (student\_t()), which we will discuss in section X.

brms::get\_prior (height ~ 1, data = men)

## prior class coef group resp dpar nlpar bound source  
## student\_t(3, 174.5, 7.1) Intercept default  
## student\_t(3, 0, 7.1) sigma default

brms makes it easy to specify prior probabilities for specific parameters or whole ‘classes’ of parameters. Setting priors for entire classes of parameters is faster for you and makes the model run faster. Right now, our model only includes the following classes of parameters:

1. Intercept: this is a unique class, only for intercepts.
2. sigma: this is for the standard deviation of our error parameters. Our model only has one for now, sigma (), but it will have more later.

We’re going to set *weakly informative* prior probabilities for our parameters. To set these you have to use what you know about your variables and the world in general. Since we know that the average male over 20 in the US is 176 cm tall (cite@@), this seems like a reasonable prior expectation for how tall the adult males in our sample will sound. We also know that the standard deviation of adult male heights in the US is 7.5 cm, and will double this for our priors. This is to account for that fact that there may be more variation in how tall people ‘sound’ compared to how tall they *are*. The code to set the priors for our model looks like this:

prior = c(brms::set\_prior("normal(176, 15)", class = "Intercept"),  
 brms::set\_prior("normal(0, 15)", class = "sigma"))

The code above tells our model to use a normal distribution with a mean of 176 and a standard deviation of 15 (normal(176, 15)) for the prior distribution of the Intercept. Around 90-95% of the mass of normal distributions is within two standard deviations of the mean. This means that we are saying that we expect, a priori, that the intercept should be between around 146 (176-15x2) and 206 Hz (176+15x2). This is too broad, but at least places the supposed outcomes within reasonable human ranges. The random error, sigma, was given a prior with a normal distribution with a mean of 0 and a standard deviation of 15 (normal(0, 15)). This is likely an overestimation of the magnitude of the random error in this data. However, it is likely to be in the ballpark. Our prior specifies a normal distribution centered at 0 for the standard deviation. Since standard deviations, like variances, can only be positive the sampler (STAN) used by brm ignores the negative half and uses only the positive half of the prior distribution. This prior basically says that we expect the average variation around the mean to be less than 30 cm, which it is very likely to be.

The left panel in figure 3.3 compares the normal distribution we used (blue line) to a histogram of our height judgments. As we can see, the prior distribution we used for the intercept is much broader (more vague) than the data distribution so that it will have basically no effect on our results (but will help our model fit properly). The right panel compares the prior for the standard deviation parameters to the absolute value of the centered apparent heights. This presentation shows how far each observation is from the mean apparent height (at 174 cm), and again we see that most of these deviations are in the thicker part of the prior density. As a result, neither of these priors is going to have much of an effect on our parameter estimates given the size of our sample (see figure 3.1.

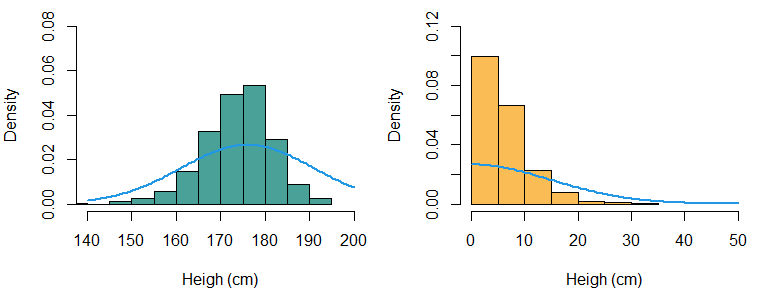


Figure 3.3: (left) A comparison of the densities of the prior for our intercept, compared to a histogram of our height judgments for male speakers. (right) The distribution of absolute deviations from the mean height judgment, compared to the prior distribution for the error parameter in our model.

We can update the description of our model to include the specification of prior distributions for each estimated parameter, as in (3.18). In the future, our model descriptions will always include these. Our model specification now makes it clear: We expect height judgments to be normally distributed, we expect the mean to always equal the intercept, and we have specific distributions in mind for all estimated model parameters (i.e., ).

We can fit this new model below using the new lines which specify the prior distributions for our parameters.

# Fit the model yourself, or  
set.seed (1)  
model\_priors =   
 brms::brm (height ~ 1, data = men, chains = 4, cores = 4,  
 warmup = 1000, iter = 3500, thin = 2,  
 prior = c(brms::set\_prior("normal(176, 15)", class = "Intercept"),  
 brms::set\_prior("normal(0, 15)", class = "sigma")))

We can use the short\_summary function in the bmmb package to get a ‘smaller’ version of the model print statements. These shorter versions are not a replacement for the complete statement as they omit important information about model fit. However, they do help us to compare models while minimizing redundant information in this text, thereby making a more efficient use of space on the page. If we compare the output of model\_thinned:

# inspect model  
bmmb::short\_summary (model\_thinned)  
## Formula: height ~ 1  
## Population-Level Effects:  
## Estimate Est.Error l-95% CI u-95% CI  
## Intercept 173.78 0.29 173.22 174.35  
##   
## Family Specific Parameters:  
## Estimate Est.Error l-95% CI u-95% CI  
## sigma 7.76 0.21 7.38 8.18

To that of the model where we specified wider priors (model\_priors), we see that there is no noticeable effect on our results. This is because the prior matters less and less when you have a lot of data, and because we have set wide priors that are appropriate (but vague) given our data. Although the priors may not matter much for models as simple as these, they can be very important when working with more complex data, and are a necessary component of Bayesian modeling.

bmmb::short\_summary (model\_priors)

## Formula: height ~ 1  
## Population-Level Effects:  
## Estimate Est.Error l-95% CI u-95% CI  
## Intercept 173.8 0.3 173.21 174.39  
##   
## Family Specific Parameters:  
## Estimate Est.Error l-95% CI u-95% CI  
## sigma 7.78 0.21 7.37 8.22

## 3.7 Answering our research questions

Finally, let’s return again to the research questions we posed initially in chapter 2, and again at the beginning of this chapter:

Q1) What is the average apparent height of the whole *population* likely to be?

Q2) Can we set bounds on likely mean f0 values based on the data we collected?

We can consider the answers to these questions provided by our final model, model\_priors. Usually, parameters should be reported with *at least* the mean/median and standard deviations of the posterior distribution, in addition to some useful credible interval (e.g. 50%, 95%) around that parameter. Based on the result of our final model, an answer to each question might be something like this:

A1) Based on our model the average apparent height for adult males is likely to be 174 cm. In a paper we might report this like: “The mean height is 174 cm (s.d. = 0.3, 95% CI = 173.2, 174.7)”.

A2) Yes we can. There is a 95% probability that the population mean is between 173.5 and 174.7 given our data and model structure. In other words, 95% of the posterior density is concentrated between the values of 173.2 and 174.4.

Notice that our answers correspond closely to what we concluded at the end of last chapter, that “the average male speaker is *most likely* to sound about 174 cm tall. We can also conclude informally based on Figure 2.8 that the most likely mean values fall between (approximately) 173 and 174.5 cm”. The reason for this correspondence is because we made our inferences at the end of chapter 2 using only the likelihood and, due to the shape of the prior and the number of observations in our data, the shape posterior distribution of our model is being dominated by the likelihood.

## 3.8 Frequentist corner

In frequentist corner, we’re going to compare the output of brms to some more ‘traditional’ approaches. We’re not going to talk about the traditional models in any detail, the focus of this section is simply to highlight the similarities between different approaches, and to point out where to find equivalent information in the different models. If you are already familiar with these approaches, these sections may be helpful. If not, feel free to skip these sections of the book, although they may still be helpful. If you want to know more about the statistical methods being discussed here, please see section X for a list of suggested background reading in statistics.

### 3.8.1 Intercept-only Bayesian models vs. the one-sample t-test

We can fit a one-sample t-test to our vector of apparent-height judgments.

t.test (mens\_height)

##   
## One Sample t-test  
##   
## data: mens\_height  
## t = 581.58, df = 674, p-value < 2.2e-16  
## alternative hypothesis: true mean is not equal to 0  
## 95 percent confidence interval:  
## 173.2010 174.3744  
## sample estimates:  
## mean of x   
## 173.7877

This usage is incorrect however, since this test requires that all the observations be independent from each other. Since we have multiple observations from each speaker and listener, our observations are not independent of each other. Notice however, that the interval provided around the mean (95 percent confidence interval: 173.2010 174.3744) corresponds very well to the 95% credible interval around the intercept provided in model\_priors (173.21, 174.39). The reason they align so well is because our Bayesian estimate is being dominated by the likelihood, and the more ‘traditional’ t-test *only* consideres the likelihood. In addition to this, both the t-test and our current Bayesian model (inappropriately) treat all our observations as independent when they are not. This issue will be discussed at length in the following chapter.

# 4 Inspecting a ‘single group’ of observations using a Bayesian multilevel model

In the last chapter we built a Bayesian regression model suitable for inspecting the average of a single group of observations. However, as we noted multiple times this model was ‘wrong’ for the structure of our data. The reason for this is that this model did not properly account for the *repeated measures* structure in our data. To properly handle repeated measures data, we need a *multilevel model*. In this chapter we will explain what we mean by ‘repeated measures’ and ‘multilevel’, in addition to fitting our first proper multilevel Bayesian model using brms.

## 4.1 Multilevel models and repeated measures data

Experiments often result in what is called **repeated measures data**, data where multiple observations can come from the same person/source. As a practical matter, setting up experiments is often time consuming such that it makes more sense to collect, for example, 100 observations from 50 different people, rather than 1 observation from 5000 people. In addition, collecting more than one measurement from each source can go a long way towards reducing uncertainty in a model. However, the statistical analysis of repeated measures data requires models that take the repeated nature of the measurements into account. Treating repeated measures data as if it were *not* repeated measures data can cause problems for the inferences we make using statistical models. This is because it can give us a warped perspective regarding how much variability there really is in a group of observations.

For example, if we have 50,000 samples of speech from adult males from Los Angeles you may be confident that we can reliably estimate the speech characteristics of males speakers from Los Angeles. But what if all these samples were from only three different speakers? It’s obvious that this makes data less reliable for making inferences about Los Angeles in general, although it may provide excellent information about the three people we measured. The reason repeated-measures data can cause problems is because measurements are not independent: Multiple measurements from the same person are probably going to be similar to each other. As a result, the repeated observation of a limited number of subjects may give you good information about the subjects, but may be less useful to make inferences about the population more generally.

We’re going to consider the same data we discussed in Chapter 2, the apparent height judgments made for the adult male speakers in our experiment in the actual, unmodified resonance condition (see section 1.3). Below we load and subset our full data to include only these trials:

library (bmmb)  
data (height\_exp)  
men = height\_exp[height\_exp$C\_v=='m',]  
men = men[men$R=='a',]  
  
# create vector of height judgments  
mens\_height = men$height

Figure 4.1 presents the height judgments collected for adult male speakers in our experiment, organized in three different ways. In the right panel we see the marginal distribution of the data with no differentiation made between listeners. This boxplot shows the individual observations of our vector around the overall mean with a standard distribution equal to . In the middle panel we see each listener’s height judgments in a different boxplot. Each listener has a different mean value which we will refer to as for listener , for any individual unspecified listener, and with an empty subscript when referring to all listener mean values. Each listener also has a distribution of height judgments around their average value. For the sake of simplicity, we will assume that the standard deviation of these distributions is equal to for all listeners. Finally, in the left panel we see the distribution of individual speaker averages (). This distribution is similar to considering the distribution of thick bars inside each individual boxplot (although these bars represent the medians and not the means). We can call the standard deviation of this distribution , since it represents between-listener variation in average height judgments.

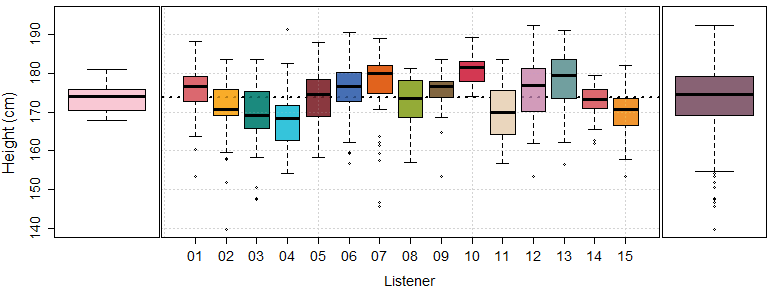


Figure 4.1: (left) Boxplot of distribution of average height judgments made by each listener in the experiment for adult male speakers. (middle) Individual boxplots of height judgments for adult male speakers, for each listener. (right) The marginal distribution of the height judgments shown in the middle plot.

A single normal distribution has a single fixed standard deviation. A visual inspection of the boxplots in figure 4.1 suggests that , , and may not all be equal. Conceptually, they are not the same thing: measures variation between random listeners *and* observations, measures variation conditional on a fixed listener, and measures variation between listener averages while ignoring random variation in observations within-listener. So, we potentially have three different standard deviations, and three different sources of variance, in our data based on how we conceptualize the variation. The final model we used to analyze our data in the last chapter looked like this:

This model above estimates ( in the model above) but not or . As a result, this model treats all deviations from the mean as random and therefore acts as if systematic between-speaker variation did not exist. However, it is clear that listeners show consistent tendencies in the heights they tend to report, so that systematic between-speaker variation () should not necessarily be treated as random, unpredictable error. For example, a height response of 181 cm may be 7 cm above the overall mean, but is perfectly average for listener 10. From the perspective of the model above, an observation of 181 cm is +7 above the mean regardless of who produced it. All variation is noise. However, if you know listener 10 provided a response then a response of 181 cm should not be surprising and would be perfectly ‘on target’ for this listener. So, an observation of 181 would be an error of 0 cm when provided by listener 10.

If we consider our data *conditional* on the listener who provided it, it is variation from that listener’s average () value that should be considered *noise*, and not all deviations from the mean (). So, from the perspective of a model that includes information about listeners the *real* random error is not but . This is because our model would ‘know’ why the boxplots in the middle panel of figure 4.1 have different means (because different listeners provided the judgments), but not why the boxplots have an internal distribution (i.e. why listeners provided a range of height judgments).

### 4.1.1 ‘Levels’ of variation

We can think of repeated measures data as having multiple ‘levels’ of variation, as shown in figure 4.2. For our experimental height data, our ‘levels’ are:

1. The ‘lower’ level: Is the distribution of the *data* itself, our individual observations and data points. We expect that each listener will produce judgments around their specific average (), with a standard deviation equal to for all speakers (referred to as above). Clearly, there may be many good reasons for listeners’ height judgments to vary from their average. However, our model cannot explain this and so this is treated as ‘error’ by our model.
2. The ‘upper’ level: The distribution of *parameters* between your sources of data. In this case this is the average height reported by a given listener . The speaker averages can be thought of as random variables since listeners are chosen randomly from a larger population of listeners and so any given speaker’s is unpredictable a priori. We can assume that this upper-level distribution is also normal with a mean of and a standard deviation equal to (referred to as above).

A **multilevel model** is a special kind of regression model that is able to simultaneously estimate random variation at multiple levels. A multilevel model fit to our height judgments would be able to estimate the values of for all listeners, the variation in listener means (), and the random within-listener variation (), all at the same time. The structure of such a multilevel model can be compared to the structure of the ‘unilevel’ model we fit to our height judgments in Chapter 3 (and seen in (4.1)). In our ‘unilevel model’ there is only one level and one distribution: The data distribution. In a multilevel model there are two levels: The distribution of *parameters* above, and the distributions of *data* below.

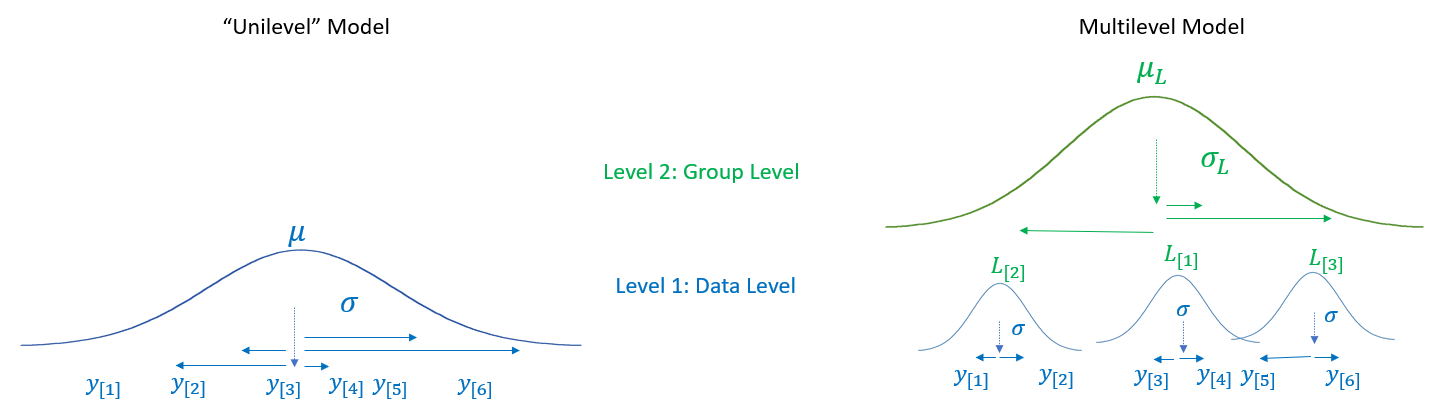


Figure 4.2: A comparison of a statistical model with only a single level of variation to the structure of a multilevel model.

Model parameters describe some tendency in the data. For now we are just talking about average values. So, our models can basically describes listeners in terms of their average response, which it encodes using parameters related to the speaker means. As a result, when we say “the distribution of parameters” to refer to the distribution of , we are really saying “the distribution of listener characteristics”. So, when we say “multilevel models model variation at the data level and the parameter level”, we are basically saying “multilevel models allow us to model variation of data within listeners, and of parameters/characteristics between listeners”.

## 4.2 Strategies for estimating factors with many levels

Our model so far has only included an intercept. However, it seems that our model should include coefficient that would allow us to model variation in apparent height judgments between different listeners. R treats nominal, categorical predictors as *factors* and assumes that each different label represents a different group. Each value of a factor is called a *level*. Although we can indicate listeners using numbers, we are not treating this predictor as numerical (quantitative). Instead, we will treat listener (L) as a factor, and the individual listeners in the experiments are its levels. As far as our models are concerned, participant/speaker/subject/listener has no special status as a predictor and it is just a factor with many levels.

In order to estimate a separate effect for each listener we need to add 15 parameters to our model, , one for each listener. Our data could then include a predictor for each listener coefficient that would equal one or zero based on whether the row was contributed by that listener or not. For example, in (4.2) we see a line consisting of a set of 15 predictors associated with the coefficients, all multiplied by zero save for the second one. The effect of this, seen in the second line in (4.2), is to only have the second listener coefficient contribute to the value of the predicted value for observation .

A careful consideration of (4.2) suggests that the coefficients can’t equal the average height reported by each listener. For example, if the overall mean (the intercept) is 170 cm and listener three reports heights of 175 cm on average, the equation above would predict a height response of 345 cm () for this listener. Clearly that is not how the model should be working. Recall that regression models encode *differences* rather than absolute values. Our model already represents the overall data average using the intercept parameter. Thus, the listener predictors only need to contain information about *differences* between the listener average and the intercept, i.e. the *effect* for that listener. So, if the overall average is 170 cm and listener three reports an average height of 175 cm, then and . For this reason, multilevel models usually set equal to zero and model listener (or group) specific variation around zero, rather than the absolute values of listener (or group) averages.

If every single factor level were to get an independent parameter represented in our regression equation, these would become very long and difficult to interpret. For example, we may want to include a factor with dozens or even hundreds of levels. Instead, we can treat the effects associated with each level of a factor as a vector. So, rather than use two separate predictors, and , to represent the first and second listener effects, we will use and , the first two elements of a vector of length 15. In order to know which listener contributed to each trial, we also need an index variable, a variable that equals a number from one to 15 that indicates which listener parameter to use on that trial. This is the L column in our height\_exp data. For example, below we convert our listener vector to a factor and check out the order of the levels.

L = as.factor (men$L)  
levels(L)

## [1] "01" "02" "03" "04" "05" "06" "07" "08" "09" "10" "11" "12" "13" "14" "15"

When you give brms a factor as a predictor, it uses the property of factors as numbers (discussed in section 1.@@) to index the predictor vector. So, when you see the order of the levels provided by R, that is the order of the coefficients in the model that represent the levels of that factor. When we then use this listener vector to estimate the average height reported by each listener, the ordering of the levels is maintained in the listener averages.

# find average reported height for each listener  
listener\_means = round(tapply (men$height, L, mean),1)  
listener\_means

## 01 02 03 04 05 06 07 08 09 10 11 12 13 14 15   
## 175.4 170.9 169.7 167.8 174.1 175.9 176.7 173.1 175.3 180.9 170.1 175.9 178.0 173.0 169.9

If we center the listener\_means vector above, this is basically equivalent to our vector of listener effects .

listener\_effects = round (listener\_means - mean (listener\_means),1)  
listener\_effects

## 01 02 03 04 05 06 07 08 09 10 11 12 13 14 15   
## 1.6 -2.9 -4.1 -6.0 0.3 2.1 2.9 -0.7 1.5 7.1 -3.7 2.1 4.2 -0.8 -3.9

Imagine that we knew some data was provided by the second, third and first listeners. We could represent these effects in the proper order as below:

listener\_effects[c(2,3,1)]

## 02 03 01   
## -2.9 -4.1 1.6

The L column in our data tells us which listener provided our data for each data point we have. So, if we index our effects vector with the L vector, we end up with a vector that shows you which listener effect applies for each observation in our data. Below, we see that the first six elements of L indicate listener 01, and as a result the first six elements of listener\_effects[L] would represent the coefficient for listener 01.

head (L)  
head (listener\_effects[L])

Unfortunately, R names the vector of coefficients in your model using the same name as your predictor. So, the vector of listener effects in your model won’t have a nice name like listener\_effects. Instead, we have a predictor called L which represents listeners, *and*, we have a set of coefficients called in our model that represent the effects for each of our listeners. As a result, our regression equations include terms that look like this, , which means: “Coefficient , and the level is the one indicated by the L predictor for trial ”. This term is now included in our prediction equation in (4.3). For example, in the code below we the equivalent notation in terms of our listener\_effects vector, where we arbitrarily set to show the listener affect associated with the 300th trial in our dta.

i = 300  
L[300]  
listener\_effects[L[i]]

When you see terms like these in your models, this is nothing more than a more compact way of representing the same model structure shown in (4.2). The regression equation in (4.2) says that the expected value for any given trial is equal to the intercept plus the listener effect specific to the listener that contributed that trial (as indiciated by the L vector).

We know that the 15 values of are related by being different listeners in our experiment. The way you tell your model that these coefficients are related is by treating them as levels of the same factor rather than as unrelated coefficients. One way to think of factors is that they represent ‘batches’ of coefficients in your model that are thematically or conceptually related. Once you encode the fact that there are ‘batches’ of parameters in your model, you have several options with respect to how these batches are estimated. Specifically with respect to our height data, we have several options regarding how we estimate the value of our listener parameters . Gelman and colleagues (CITE) discuss three general ways that parameters related to levels of a factor can be estimated: 1) *Complete pooling*, 2) *No pooling*, and 3) *Partial pooling*. The distinction between complete, no, and partial pooling is made for each factor in a model. So, one factor may be estimated using partial pooling while another may be fit using no pooling. This makes understanding the distinction between these approaches very important.

### 4.2.1 Complete pooling

Last chapter we carried out a **complete pooling** analysis of our data. That is, we threw everything into one big pile and analyzed it as if it came from one undifferentiated group (as in figure 4.2). There are several problems with this approach. First, the complete pooling approach does not allow us to make any statements about variation between listeners since it does not estimate values of . In fact, this model assumes that is equal to zero (or does not exist). If listeners do not differ from each other in any way, then will be equal to zero and the two models in figure 4.2 will converge on each other. However, if is not zero, then this approach will miss out on important information.

There is perhaps a more serious problem with the complete pooling approach. The complete pooling model assumes that all deviations from the mean are due to random error. Recall that regression models assume that all random variation around predicted values (i.e. the residuals) are independent. Since independence is assumed, likelihoods are calculated the ‘easy’ way by multiplying individual marginal probabilities (see sections 2.2.2 and 2.5.1. However, as we can see in figure 4.1, each listener had a slightly different average judged height. This means that the residuals associated with listener 14 are not independent, we actually expect almost all of them to be positive (i.e. larger than average) since nearly all of this listener’s responses were above the average. The violation of this assumptions means that if we are using a no-pooling approach for our repeated-measures data we are calculating likelihoods the ‘wrong’ way. Basically, our data has 675 rows but not 675 totally-independent observations, but our model doesn’t know this.

We’ve previously mentioned that our models are not exactly ‘right’ or ‘true’, so why does it matter all of the sudden that the complete pooling model is ‘wrong’? Recall that in section 2.5.1 we mentioned that the likelihood function gets narrower as a function of the sample size ? This applies only to *independent* observations, but not necessarily to *dependent* observations. As a result, when we treat dependent observations as independent our likelihood may end up being narrower than is warranted. This gives us a false sense of security about how precise our parameter estimates are. So, it matters that this is wrong because it can end up being very wrong, and it can easily be improved. Basically, spherical models of billiards balls are wrong but useful, while cubic models of the balls are wrong and much less useful.

### 4.2.2 No pooling

To account for systematic variation between listeners in the data we can include listener predictors () in our model, as in as in (4.4). When we estimate the levels of this factor with **no pooling**, we estimate every level of the factor totally independently. One way to model this mathematically is to think of the coefficients as coming from a uniform distribution that extends from negative to positive infinity. The uniform distribution is just a flat line, meaning any value in the interval is equally likely. So, a uniform distribution extending from positive to negative value places absolutely no restrictions on the possible values of a parameter and says absolutely any value is equally likely. This is equivalent to setting . As a result, we can see complete and no pooling as two extremes, in one case we assume there is no between-listener variation and in the other we assume it can potantially be infinitely large.

In (4.4) we see that in our no pooling model, the expected value () for each trial is the sum of our intercept and the listener predictor, and that each of the terms is drawn from the same uniform distribution.

There is one main weaknesses to this approach: Assuming that does not allow shrinkage or regularization to occur. We know that shrinkage occurs when an estimate is pulled towards its prior. Well, a perfectly uniform prior does not exert any influence on posterior distributions, so that these are completely reflective of the likelihood function. As a result, a factor estimated with no pooling is one in which shrinkage cannot occur. The absence of shrinkage is a *bad* thing, as this is considered to be a strength of Bayesian modeling. Shrinkage represents the pooling of information across the levels of your factor. For example, below we see some quantiles for the average apparent height reported for adult male speakers by the listeners in our experiment:

listener\_means = tapply (men$height, men$L, mean)  
quantile(listener\_means)

## 0% 25% 50% 75% 100%   
## 167.8356 170.5289 174.1000 175.9256 180.8911

The listeners in this experiment each carried it out independently and did not know each other. Despite this, we see below that half of listeners’ average height judgments are between 171 and 176 cm, and that all listener averages fell between 168 and 181 cm. The consistency of these judgments suggest a common behavior between listeners, and an underlying similarity in listener characteristics and parameters. Given this, it might be surprising if a new listener carried out the experiment and provided a height judgment of 130 cm for an adult male. In the absence of strong evidence to suggest that this is a good representation of their true opinion, perhaps we should be skeptical of this value (e.g. perhaps they clicked somewhere by mistake). Unfortunately, our no pooling model has no mechanism by which this can occur. In the absence of shrinkage the characteristis of each listener are estimated as if only a single listener had taken part in our experiment. But we don’t have one listener, we have 15, and they all performed the experiment in quite similar ways. Clearly, it may be useful if our model could consider what it knows about the values of in general when it estimates the value of any given in particular.

### 4.2.3 (Adaptive) Partial pooling

Complete pooling and no pooling offer extremes, we either assume is zero or we assume it is infinite. Partial pooling offers a sort of middle ground between complete pooling and no pooling, a potential ‘just right’ compromise solution. **Partial pooling** estimates the values of using non-zero values of , meaning that the parameters representing the levels of a factor are neither completely independent nor totally merged.

We know that our prior probability can help to pull outliers towards the prior (as discussed in section 3.2.0.2), resulting in *shrinkage*. In order for this to work effectively, the variance of the prior needs to be set at a reasonable value given the amount of variation in the data. Below, we see that the standard deviation of listener means is 3.6 cm. It would be nice to be able to use a value close to this, since we know this to be a good fit to our data. Unfortunately, if we did this it would not really be a *prior* probability anymore, since we would be analyzing the data, finding the most likely parameter values, and then using that to set what is supposed to be the *prior* probability.

sd(listener\_means)

## [1] 3.593794

Instead of doing this, a partial pooling model allows us to estimate using our data. This allows us to have a prior distribution for our parameters that is tailored to the data, without us having to actually specify it a priori. When standard deviation parameters like are estimated from the data rather than a priori, this is called **adaptive partial pooling**. However, by convention the ‘adaptive’ part is usually dropped when partial pooling is dicussed. So, when you read that a factor was estimated using ‘partial pooling’, most often that means that the standard deviation parameters of the model coefficients for that factor (e.g.  for ) are estimated from the data.

The (adaptive) partial pooling version of our model is presented in (4.5). There are two important differences with respect to the model in (4.4). First, the terms are drawn from a normal distribution with a mean of zero and a standard deviation equal to . The reason the mean of the distribution is equal to zero is that our listener coefficients only encode differences to intercept (effects), and not the actual listener average. So, a listener that is perfectly average will have a coefficient equal to 0. Second, since the term is estimated from the data a prior distribution for is specified.

This approach is called partial pooling because the estimate of each level of is potentially influenced by every other level of by virtue of being drawn from the same distribution . For example, we saw that half of listener mean height judgments were between 170 and 176 cm, and our model will now take that into account when estimating the individual levels of . So, unlike the complete pooling case we do not ignore variation between listeners, and unlike in the no pooling case we do not ignore the similarities between them.

#### 4.2.3.1 Hyperpriors

You may have noticed that in (4.5) we assign a prior distribution to in the model above. However, is a parameter in the prior distribution of our listener parameters . So actually, the prior for is the prior of a prior. We might call this a ‘grand prior’ (like grandparent) but the actual term for it is **hyperprior**. One way to think of it is that in a multilevel model each level effectively acts as the prior for the level below it. This leads to a nested or hierarchical model structure (sometimes multilevel models are called **hierarchical models** for this reason).

We’re going to update the equation defining our posterior distributions (originally presented in (3.12)) to take into account the structure of our partial pooling models. Our model will now reflect not only our prior beliefs but also the information encoded in our hyperpriors, and it will estimate the characteristics of from the data. We can begin with the posterior probability of observing a given value of for a single listener in isolation. The equation below says that the posterior probability of some speaker effect () given the data is equal to the product of the likelihood of given the data () and the prior of , divided by the marginal probability of the data ().

If we want to consider the posterior probability of given the data, we can add to every term that previously included in (4.6). Basically, we are saying *and* , instead of just . Equation (4.7) can be read aloud as “the posterior probability of observing a certain listener effect and a certain amount of between-listener variation given some data is equal to the likelihood of the data given the listener effect and the amount of variation between listeners, times the joint prior probability of the listener effect and the amount of variation between listeners, divided by the marginal probability of the data”.

We know that we can represent the joint probability of a listener effect and the listener standard deviation, , using normalized conditional probabilities as in . We can also remove the listener standard deviation from the likelihood of effect given the data () because we know that the probability of any given data point in a normal distribution depends only on the mean of the distribution and not the standard deviation of means *between* distributions. These two changes are reflected in (4.8). The first term in the numerator on the right hand side is the likelihood, this was in our original equation in (4.6). The second term is the prior distribution of listener effects. This also appeared in (4.6), however, this distribution now allows for variation in the value of . Finally, the third term represents the hyperprior, the prior distribution of one of the parameters of another prior distribution.

Our posterior estimates of may differ from our maximum likelihood estimates of the listener effects. This is because in addition to the likelihood, the posterior probability of in (4.8) takes into account the distribution of values of with respect to estimates of . When our posterior estimates of the listener (or any other) effects are more similar to the population average than the maximum-likelihood estimate, *reguralization* has occurred. Importantly, the amount of pooling/shrinkage that will occur will be determined by , which is estimated from the data itself.

## 4.3 Estimating a multilevel model with brms

We’re going to fit a model to the same data we investigated in Chapter 3, however, we’re going to use a multilevel model that reflects the repeated-measures nature of the data.

### 4.3.1 Data and Research questions

Below we load and subset our full data to only include those trials involving adult male speakers, and only when the actual resonance size was presented. This is the same data we considered in chapter 3.

data (bmmb::height\_exp)

## Warning in data(bmmb::height\_exp): data set 'bmmb::height\_exp' not found

men = height\_exp[height\_exp$C\_v=='m',]  
men = men[men$R=='a',]

Once again, we’re going to try to address the following basic research questions:

Q1) How tall does the average adult male from the US sound?

Q2) Can we set limits on credible average apparent heights based on the data we collected?

To analyze our data using brms in R we need it to be in a data frame with each row containing a different observation. We need one column containing our dependent variable (the variable we want to analyze) and one column that indicates the ‘source’ of the data. We can see that our data satisfies these conditions below, where the relevant columns (for this model) are L for listeners, and height for our dependent variable.

head (men)

## L C height R S C\_v vtl f0 dur G A C\_v\_f  
## 371 01 m 171.7 a 47 m 14.8 172 339 m a m  
## 373 01 m 176.3 a 48 m 15.6 108 236 m a m  
## 375 01 m 177.8 a 49 m 15.5 96 315 m a m  
## 377 01 m 178.8 a 50 m 14.5 134 240 m a m  
## 379 01 m 172.2 a 51 m 15.0 122 241 m a m  
## 381 01 m 173.0 a 52 m 14.8 101 259 m a m

Often, experiments (or data more generally) will feature a single source of information. However, in this experiment we have both speakers producing our data and listeners providing judgments about our data. This leads to non-independent observations based on both speaker and listener. For example, a speaker who sounded tall to one person may have sounded tall to other listeners, and listeners who identified all speakers as generally tall/short would have done so in a similar way across all speakers. We will begin by building a model that incorporates information about listeners before discussing a model that includes information regarding both speakers and listeners.

### 4.3.2 Description of the model

We’re going to keep working on developing an intuitive understanding for model notation, and using it to describe models. Just keep in mind that this is nothing more than an efficient way to denote the concepts being discussed in the text. To specify a multilevel model, you need to write a slightly more complicated model formula. This explanation assumes that you have a data frame where one column contains the variable you are interested in predicting (in this case height), and another column contains a vector containing unique labels for each speaker/listener/participant (in this case a unique listener label L). Before, the model formula looked like this:

height ~ 1

Which meant ‘predict height using only an overall intercept’. To indicate that your model contains an ‘upper’ level where you have repeated-measures clusters of data coming from different sources, you have to put another model formula *inside* your main model formula. The model formula corresponding to the model shown in (4.5) looks like this:

height ~ 1 + ( 1 | L)

Which means ‘predict height using only an overall intercept, but also estimate a different intercept for each level of the L (listener) factor’. When you place a predictor in parenthesis on the right-hand-side of a pipe |, like this ( | predictor ), you tell brm that you would like to estimate separate parameters for each level of the factor. Whatever you put in the left-hand-side of the parentheses ( in here | predictor ) specifies the parameters you wish to estimate for each level of the factor. In addition, any parameters specified on the left-hand side of the pipes in brackets (( in here | predictor )) will be estimated using (adaptive) partial pooling. By convention, effects estimated using partial pooling are modeled as coming from a normal distribution with a mean of zero, so that aspect of the model does not need to be directly specified.

So the model formula height ~ 1 + ( 1 | L) tells brm to build an intercept only model with a separate intercept for each listener, and that the listener intercept terms should be estimated with partial pooling. Effectively, this model formula specifies a subset of the information provided in our full model specification shown in (4.9) (originally (4.5))

In plain English, the model above says:

We expect height judgments to be normally distributed around the expected value for any given trial, , with some unknown standard deviation . The expected value for a trial is equal to a fixed overall average (Intercept) and some value associated with the individual listener who uttered the trial (). The listener coefficients () were modeled as coming from a normal distribution with a mean of zero and a standard deviation () that was estimated from the data.

There is a very important difference in how the ‘unilevel’ (complete pooling) model from Chapter 3, and this model partition the variance in apparent talker heights. The initial complete pooling model broke down the total variation in the data () like this:

In other words, all variation was error. We don’t know why values vary from the mean, so all variation is just random noise. Our multilevel model views the variation in our data like this:

The total variance is equal to the within listener variance () plus the between-listener variance (). From the perspective of this model, only the variation *within* a speaker’s individual boxplot is error. The differences from listener to listener represent random (but systematic) between-listener variation in apparent height judgments. To some extent, investigating the *effects* for the different listeners in our sample involves asking how much the listener averages tended to differ from the overall response. As noted earlier, if listeners do not differ from each other in any way, then is equal to zero and the two models in figure 4.2 will converge on each other. So, a crucial part of investigating the overall effect of different listeners on our results is understanding the magnitude of .

### 4.3.3 Fitting the model

We can use brms to fit a model that more closely resembles our model specification in (4.5). To do this we need to use the formula given above, and we also need to specify priors for four parameters: , , and . We can confirm our expectations using the get\_prior function as seen below.

brms::get\_prior (height ~ 1 + (1|L), data = men)

## prior class coef group resp dpar nlpar bound source  
## student\_t(3, 174.5, 7.1) Intercept default  
## student\_t(3, 0, 7.1) sd default  
## student\_t(3, 0, 7.1) sd L (vectorized)  
## student\_t(3, 0, 7.1) sd Intercept L (vectorized)  
## student\_t(3, 0, 7.1) sigma default

We see a new class of prior, sd. Our classes of priors are now:

1. Intercept: this is a unique class, only for intercepts.
2. sd: this is for our standard deviation parameters that relate to ‘batches’ of parameters. In our example this is sd(Intercept) for L ().
3. sigma: the error term.

The information presented by get\_prior does not just list the priors that must be specified, but also different ways that priors can be set, so there are some redundancies. This is not so necessary for our very simple models now but it will become very useful later so it is worth understanding now while it is still relatively simple. Top to bottom, here is the information presented in the output above. First, you can specify a prior for the Intercept, which is its own unique class. Second, you can specify a prior for *all* sd parameters in your model. Third, you can specify a prior for all the sd parameters, specifically for your L group. The fourth line indicates that you can also specify a prior *only* for the Intercept for your L group (i.e. ) and not to any other listener-related predictors in your model (there are none for now). The fifth line lets you specify the prior for the error in the model (i.e. ).

Below we fit a model and specify prior distributions for our intercept, for *all* variables in the sd class, and for the error term (basically the first, second, and fifth lines above).

# Fit the model yourself  
set.seed (1)  
multilevel\_model =   
 brms::brm (height ~ 1 + (1|L), data = men, chains = 4, cores = 4,  
 warmup = 1000, iter = 3500, thin = 2,  
 prior = c(brms::set\_prior("normal(176, 15)", class = "Intercept"),  
 brms::set\_prior("normal(0, 15)", class = "sd"),  
 brms::set\_prior("normal(0, 15)", class = "sigma")))

### 4.3.4 Interpreting the model

We can inspect the model print statement:

# inspect model  
bmmb::short\_summary (multilevel\_model)  
## Formula: height ~ 1 + (1 | L)  
##   
## Group-Level Effects:  
## ~L (Number of levels: 15)  
## Estimate Est.Error l-95% CI u-95% CI  
## sd(Intercept) 3.77 0.84 2.48 5.81  
##   
## Population-Level Effects:  
## Estimate Est.Error l-95% CI u-95% CI  
## Intercept 173.8 1.03 171.8 175.89  
##   
## Family Specific Parameters:  
## Estimate Est.Error l-95% CI u-95% CI  
## sigma 7.03 0.2 6.67 7.42

This model contains one new chunk its print statement, the Group-Level Effects, which tells us about the standard deviation for the Intercept (sd(Intercept)) for the ~L factor. The term sd(Intercept) in this section corresponds to , the standard deviation of listener effects () in the sample. Below, we calculate the listener mean heights values, the standard deviation of these, and the amount of within-speaker variation in apparent height judgments. We can see that these resemble the value of the analogous parameters in our model.

# find mean height for each speaker  
listener\_means = aggregate (height ~ L, data = men, FUN = mean)   
# find the within speaker standard deviation This is the within-talker 'error'.  
listener\_sigmas = aggregate (height ~ L, data = men, FUN = sd)   
  
# the mean of the speaker means corresponds to our Intercept  
mean (listener\_means$height)  
## [1] 173.7877  
  
# the standard deviation of the speaker means corresponds to 'sd(Intercept)',   
# the estimate of the standard deviation of speaker intercepts  
sd (listener\_means$height)  
## [1] 3.593794  
  
# the average within-speaker standard deviation corresponds to 'sigma',   
# the estimated error  
mean (listener\_sigmas$height)  
## [1] 6.822099

We’re going to discuss the calculations we made above in terms of what they represent in the boxplots below, our model coefficients in the print statement above, and in our model definition in (4.9). The overall mean height in our data (173.8) corresponds quite well to our model estimate of 173.8, seen in the horizontal line in the figure below. This is the intercept in our model. The standard deviation of the listener means is 3.6 cm. This is very similar to our model estimate of 3.8 cm. The standard deviation of the listener effects reflects the average distance from each speaker’s average to the overall average, and reflects in our model. Finally, the average of the within-speaker standard deviations in our data (6.8 cm) corresponds closely to our model’s error estimate (sigma = 7.0), represented by in our model. This reflects the average spread of each speaker’s data relative to their own mean, within their own little boxplot. We can see that the information provided by brms is quite similar to what we can estimate directly using our data. However, *brms* does this all for us, in addition to giving us information regarding credible bounds for different parameters.

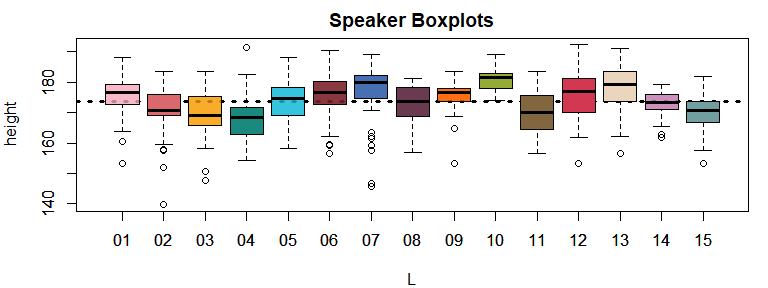


Figure 4.3: Listener-specific boxplots for apparent height judgments made for adult male speakers. The horizontal line indicates the model intercept.

## 4.4 ‘Random’ Effects

If you have some familiarity with statistical modeling or the analysis of experimental or repeated-measures data, you may have heard of **random-effects models** or **mixed-effects models**. People often distinguish the predictors in their models between those that are ‘fixed’ and those that are ‘random’. So, a researcher might describe their model as including ‘random effects’ for so and so and ‘fixed effects’ for some other predictor. In fact, a very common way of describing our model above would be that we used a ‘mixed-effect model with random intercepts by listener’. What makes a predictor fixed or random and what is the practical effect of this in our models? It turns out that although they are commonly used, the terms ‘fixed’ and ‘random’ effects do not have a single agreed-upon definition. Gelman (2005) highlights at least five different definitions of the distinction given in the literature:

1. Fixed effects are constant across individuals/groups, while random effects can vary across them.
2. Effects are fixed when you are interested in them specifically, but random if you are actually interested in the population.
3. “When a sample exhausts the population, the corresponding variable is fixed; when the sample is a small (i.e., negligible) part of the population the corresponding variable is random” Green and Tukey (1960).
4. “If an effect is assumed to be a realized value of a random variable, it is called a random effect” LaMotte (1983).
5. Fixed effects are estimated using maximum likelihood (no pooling) estimates least squares, while random effects are estimated using partial pooling (and shrinkage).

You will note that some of these definitions (2 and 4 in particular) would have an effect change from random to fixed based on the researcher’s conception of it. This is the **mind projection fallacy**, the mistaken assumption that the way we choose to represent the world in our model reflects the true nature of the thing itself (McIllwreath, p81). Rather than focus on our conception of effects as fixed or random in theory, we may ask: What is the difference between these sorts of effects in practice? What special treatment are we referring to when we say that a certain predictor in our model is a ‘random effect’?

The answer is that when researchers discuss the inclusion of **random effects** in their models, what they are usually (if not always) referring to is that the effects were estimated using (adaptive) partial-pooling. When researchers indicate that their effects were **fixed effects**, they are usually saying that they were fit with no pooling (or at least with weak priors). This applies to both Bayesian models, and to models fit using more ‘traditional’ approaches such as the popular lmer function in the lme4 package in R (cite). As a result, although researchers may define ‘random’ effects using any of the definitions shown above, in practice terms labelled ‘random’ are usually those estimated using adaptive partial pooling. In light of this, Gelman (cite) recommends avoiding the temrs ‘fixed’ and ‘random’, since these are vague and overly-broad. Although we agree with this position, the fact is that in many research areas it is still most common to refer to effects as being either fixed or random. For that reason we will follow this convention, although we will continuously highlight the relevant underlying distinction: That so-called ‘random’ effects are simply those that are estimated using (adaptive) partial pooling.

In general, when you have many levels of a factor, it may be a good idea to include these as ‘random’ effects, regardless of how ‘random’ they might actually be (cite). There is not much downside to it: You get more information from the model (e.g., information about ), there are several modeling advantages associated with estimating large batches of coefficients with partial pooling (cite), and you can always fit multiple models to see what, if any differences, result form the different approaches. Rather than considering the more ‘philosophical’ position outlined in 1-5 above, some useful things to consider when deciding between fixed and random effects are: Do you believe that treating a predictor as a ‘random’ effect offers a modeling advantage? Does it better reflect the process/relationships you are trying to understand? Does it provide you with information you find useful? Is it realistic to fit this kind of model given the amount of data you have, and the nature of the relationships in the data? Right now the last point is not something we have talked about very much, but it is something we will need to worry about more as our models become more complicated.

### 4.4.1 Inspecting the random effects

You may have noted that the estimates of our coefficients fit with partial pooling () do not appear in the model print statement. Instead, only the standard deviation of the effects is presented (, or sd(Intercept)). The reason for this is that there may be a large number of random effects coefficients, which would result in models that are difficult to read if they were all shown. A second reason for this is that researchers are often not directly interested in the individual values of their random effects, but rather they are usually focused on the overall effects common to all listeners such as the model intercept. For example, to this point we have been talking about “what is the average apparent height of male speakers” and never “what did listener 04 think the average height of men is?”. However, there are situations in which we may be interested in investigating the random effects, and the brms package has several functions to facilitate this process.

We will present two ways to consider the random listener effects using the ranef function, which gets information about your random effects from your brm model. In the first of these you set summary=FALSE in the call to ranef as shown below. This function returns a list of three-dimensional matrices, where each list element is named after the grouping factor it represents. Below, we collect the random effects and take the element named L, corresponding to a matrix representing our listener factor. The matrix representing each factor has three dimensions, with individual posterior samples varying across the first dimension (rows), factor levels varying along the second dimension (columns), and parameters varying across the third dimension. Effectively, listener\_effects is a set of two-dimensional matrices stuck together, one for each random parameter.

# get random effects, without summarizing  
random\_effects = brms::ranef (multilevel\_model, summary = FALSE)  
# get only the listener random effects  
listener\_effects = random\_effects$L

You can select the matrix corresponding to the intercept by putting the name of the parameter in the third dimension of the matrix, as in listener\_effects[,,"Intercept"],reducing our three-dimensional matrix to a two-dimensional one representing the random listener intercepts in our model.

# along the third (coefficient) dimension, take only the intercept dimension  
listener\_intercepts = listener\_effects[,,"Intercept"]

The above process results in a two-dimensional matrix representing the individual samples of our listener effects. We can see below that this matrix has 5000 rows, corresponding to the individual samples, and 15 columns, representing the individual listeners. We use the head function to see the first six samples of the listener effects.

# Our matrix of posterior samples showing 5000 samples for each of 15 parameters  
dim (listener\_intercepts)

## [1] 5000 15

# we can see the first six samples for each of the 15 listener parameters  
head ( round (listener\_intercepts, 2))

## 01 02 03 04 05 06 07 08 09 10 11 12 13 14 15  
## [1,] 1.09 -2.11 -2.04 -5.40 0.71 4.43 6.04 0.71 4.20 8.74 -2.65 3.31 4.54 2.08 -2.30  
## [2,] 0.69 -3.18 -6.51 -5.78 -1.06 1.95 1.19 -1.14 0.33 6.24 -4.13 2.60 4.15 -1.72 -5.55  
## [3,] 0.81 -4.54 -4.52 -7.54 -0.91 1.83 2.46 -0.60 -0.84 8.34 -4.69 0.61 1.75 -1.26 -5.00  
## [4,] 0.47 -3.45 -4.99 -6.46 -1.07 -0.14 2.47 1.08 -2.00 5.14 -3.19 -0.01 4.96 -1.00 -4.53  
## [5,] 1.50 -2.74 -4.88 -7.70 1.05 1.05 1.79 -1.44 1.30 6.99 -4.37 1.00 3.43 -2.48 -3.96  
## [6,] 1.95 -2.17 -6.06 -4.72 -2.36 0.63 4.13 -1.17 1.97 6.21 -3.12 0.99 2.55 0.63 -4.36

If we were to find the average of the samples within each column, the result would be the posterior mean estimate for each of our individual speaker effects.

colMeans (listener\_intercepts)

## 01 02 03 04 05 06 07 08 09 10   
## 1.5105944 -2.6620551 -3.8161356 -5.4853563 0.2766460 1.9620211 2.6837029 -0.6775710 1.3843760 6.5341588   
## 11 12 13 14 15   
## -3.3391433 1.9169167 3.8045962 -0.6911198 -3.5850982

We can repeat the process above but without specifying summary=FALSE like we did above. As a result, rather than the individual samples, you get summaries of the posterior distribution. We will still get a list of three-dimensional matrices. However, this time factor levels vary across the first dimension (rows), information about each factor level varies across columns, and parameters vary across the third dimension. We again take the list element corresponding to our list element L and take only the third dimension corresponding to our Intercept.

# get random effects, \*with\* summarizing (by default)  
random\_effects = brms::ranef (multilevel\_model)  
# get only the listener random effects  
listener\_effects = random\_effects$L  
# along the third (coefficient) dimension, take only the intercept dimension  
listener\_intercepts = listener\_effects[,,"Intercept"]

This time, the process results in a two-dimensional matrix representing a summary of the posterior, as seen below. The leftmost column of the output below represents the posterior mean effect estimated for each listener, the second column represents the standard deviation of posterior samples, and the third and fourth columns provide he 2.5% and 97.5% credible intervals for each parameter, respectively. We can see that the information in the first column below exactly matches the posterior means calculated for each listener effect above. This is because they are the same thing. In the un-summarized representation above we were seeing the individual samples varying across columns. In the summarized representation we are seeing summarizes of those samples, varying across rows.

listener\_intercepts

## Estimate Est.Error Q2.5 Q97.5  
## 01 1.5105944 1.366470 -1.18585973 4.20220273  
## 02 -2.6620551 1.400853 -5.54711749 -0.03228644  
## 03 -3.8161356 1.371796 -6.56687225 -1.21324543  
## 04 -5.4853563 1.399412 -8.26773679 -2.80563555  
## 05 0.2766460 1.364600 -2.38936835 2.98866625  
## 06 1.9620211 1.397258 -0.80944735 4.74651976  
## 07 2.6837029 1.417091 -0.01980965 5.48380263  
## 08 -0.6775710 1.394401 -3.48431980 2.07749542  
## 09 1.3843760 1.398717 -1.32978697 4.15290581  
## 10 6.5341588 1.414080 3.73165414 9.40907991  
## 11 -3.3391433 1.410366 -6.10731787 -0.61254988  
## 12 1.9169167 1.380566 -0.86234583 4.62592284  
## 13 3.8045962 1.406219 1.07431669 6.69654921  
## 14 -0.6911198 1.384535 -3.38696132 2.01938046  
## 15 -3.5850982 1.390240 -6.36290615 -0.91573390

Notice that the listener averages vary around 0, and some are even negative. That is because the listener effects (and all random effects) are represented as deviations from the intercept, as noted above. We can verify this by calculating the average reported height for each listener and centering these values. We can then compare these centered means to our model random effects and see that they are very similar.

listener\_means = tapply (men$height, men$L, mean)  
listener\_means\_centered = listener\_means - mean(listener\_means)  
  
round (rbind (listener\_means\_centered, listener\_random\_effects = listener\_intercepts[,1]), 1)

## 01 02 03 04 05 06 07 08 09 10 11 12 13 14 15  
## listener\_means\_centered 1.6 -2.9 -4.1 -6.0 0.3 2.2 2.9 -0.7 1.5 7.1 -3.6 2.1 4.2 -0.8 -3.9  
## listener\_random\_effects 1.5 -2.7 -3.8 -5.5 0.3 2.0 2.7 -0.7 1.4 6.5 -3.3 1.9 3.8 -0.7 -3.6

We can use a simple plotting function included in the bmmb package (brmplot) to look at distributions of effects. The function takes in the summary matrices made by brms and plots a point for each parameter mean/median, and lines indicating the credible intervals calculated by brm (usually 95% credible intervals). These matrices all have a standard form where parameters vary across rows, the first column is the posterior mean, the second is the posterior standard deviation (the estimated error), and the third and fourth columns are the 2.5% and 97.5% credible intervals. If you have a matrix like this you can easily plot it with brmplot as shown below.

brmplot(listener\_intercepts)

We can compare the estimates of our by-speaker effects to the distribution of centered height responses arranged by subject. We can see that the arrangement of the means and raw data shows a close correspondence.

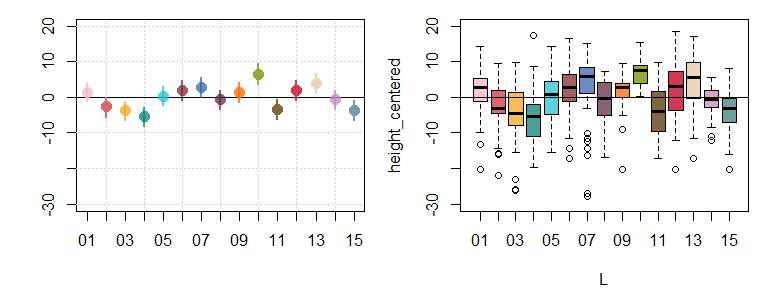


Figure 4.4: Comarisons of random speaker effects to the distribution of productions for the same speakers, for girls (cyan) and women (red).

## 4.5 Simulating data using our model parameters

One way to think about what all the numbers in our model mean is to simulate data that has the same characteristics by building it from the individual component parts. First, we set the intercept to 174 cm. Then, we create a random sample of 15 effects representing simulated listeners. The listener population has a mean of 0 and a standard deviation equal to (3.8 according to our model). These effects are stored in a vector called L\_, and they represent the values of for our different speakers. We want each of these simulated listeners to provide 45 responses just like in our data, so we need an L vector with 15 values that repeat 45 times each. We can use this to index the L\_ vector containing the effects (as in the example involving listener\_means[L] above). We also draw our error (i.e. ). It’s important to note that the error is just 15x45 random draws from this population. There is no distinction between one listener and another when it comes to the magnitude of the error. This property is called **homoskedasticity**, and the error is called **homoskedastic** when it is constant across all conditions/groups.

## don't run this line if you want a new simulated dataset.   
set.seed(1)  
## this is the value of our intercept  
Intercept = 174  
## this is a vector of 15 listener effects  
L\_ = rnorm (15, 0, 3.8 )  
## this is a vector indicating which speaker produced which utterance  
L = rep (1:15, each = 45)  
## this vector contains the error  
error = rnorm (45 \* 15, 0, 7)

After creating our components, we add the Intercept, listener effects and random error to make our fake ‘replicated’ data. Since this data has the same statistical properties as our real data, it should look a lot like it.

# the sum of an intercept, listener effects and random error makes our fake data  
height\_rep = Intercept + L\_[L] + error

Figure 4.5 we compare the results of our simulation to our real data. The are reasonably similar, which tells us that our model is a good reflection of the data.

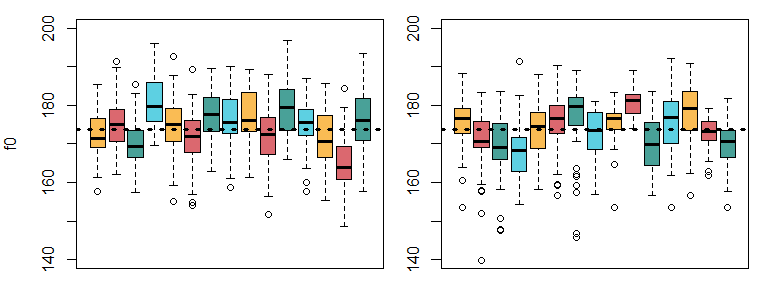


Figure 4.5: Comparison of real and simulated f0 production data.

Below we make two datasets that are ‘incomplete’ with respect to the variation in our data. The first contains the intercept and noise only, the second contains the intercept and speaker effects only.

# this fake data is missing between speaker variation  
height\_rep\_1 = Intercept + error  
# this fake data is missing within speaker variation  
height\_rep\_2 = Intercept + L\_[L]

In figure 4.6 we compare these ‘incomplete’ datasets to the full simulated data. The top row contains only error (). As a result, height judgments vary around the intercept, but there is no listener to listener variation. This is what that data would look like if there is no random variation in means across listeners. However, notice that each little box is not centered at zero. Although the error distribution is centered at zero, the small number of errors added to the true speaker mean are **extremely** unlikely to add up to exactly zero. So, error contributes some unknown amount to each listener’s sample mean estimate. This makes estimations of the *real* speaker effects impossible in practice. In the middle plot, the figure shows only between-listener variation () but no within-speaker variation (i.e., no ). Now listener vary from the intercept, but they do not vary within themselves. The final plot is the combination of the variation in the top two plots and show the final simulated data: The sum of the intercept, the within-listener variation and the between-listener variation.

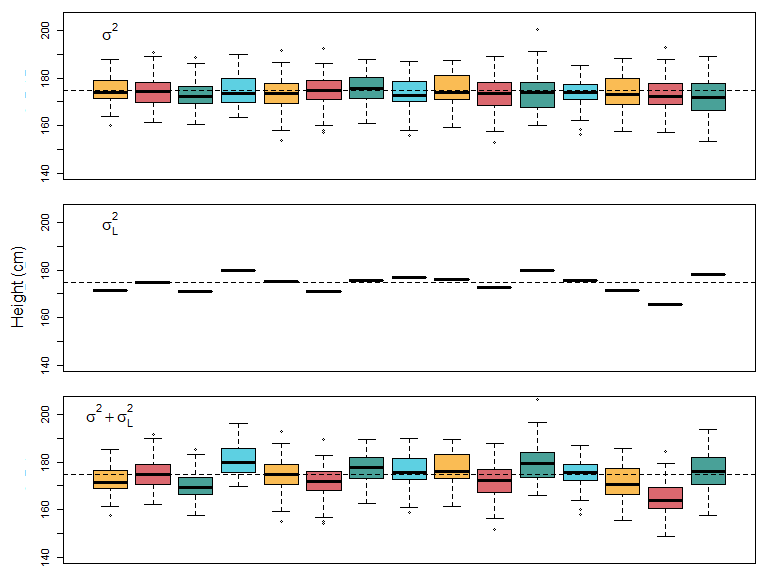


Figure 4.6: (top) Simulated error variation around the intercept. (middle) Simulated between-speaker variation, but no production error. (bottom) Simulated data containing both within and between-speaker variation in f0.

Note that the top plot in figure 4.6 basically represents the complete pooling model, a model that includes but not . As seen in the figure such a model is unlikely to generate the substantially-different listener boxplots we see in our data. This is because this data has no mechanism by which to make the listener-dependent boxplot *be* different, since there is no between-listener variation.

We can try another simulation, identical in all respects except we set the between-listener variance to zero, and set the noise variance to 7.8 (as in our complete pooling model). Below we generate 10,000 simulated data sets and record the standard deviation of the listener means for each one. These values will be stored in a vector called sigma\_L\_rep since we are simulating values of for our complete pooling model.

set.seed(1)  
reps = 10000  
sigma\_L\_rep = rep(0,reps)  
for ( i in 1:reps){  
 Intercept = 173.8  
 L\_L = rnorm (15, 0, 0) # zero between-listener variance  
 L = rep (1:15, each = 45)  
 epsilon = rnorm (45 \* 15, 0, 7.78)  
 height\_rep = Intercept + L\_L[L] + epsilon  
 sigma\_L\_rep[i] = sd (tapply(height\_rep, L, mean))  
}

Below, we see that even in 10,000 simulation we do not see a standard deviation greater than 2.1, and that most are smaller than 1.3. The standard deviation of the listener means in our data was 3.6 and our model estimate of was 3.8. This reinforces the idea that between-listener variation () is a ‘real’ aspect of our data and that our models benefit from taking it into account.

quantile(sigma\_L\_rep)

## 0% 25% 50% 75% 100%   
## 0.4429460 0.9862375 1.1336760 1.2809929 2.0812768

## 4.6 Adding a second random effect

Our model is looking better, but is still not ‘right’. This is because in addition to repeatedly sampling from only 15 listeners, our model also features repeated sampling from 45 different speakers. In figure 4.7 we see that we can recreate figure 4.1, save for with speakers instead of listeners. So, we are interested in the average male from the US but have only sampled 45 of them, and we need to build this information into our model.

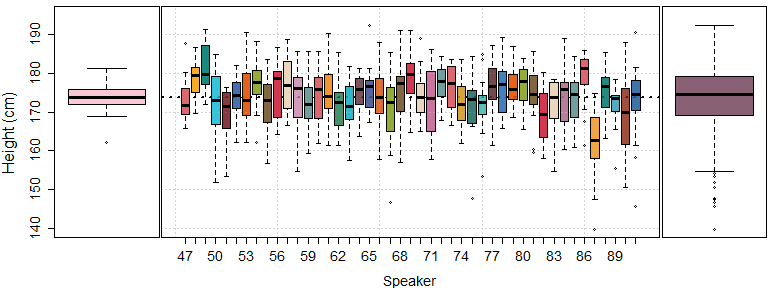


Figure 4.7: (left) Boxplot of distribution of average height judgments associated with each adult male speaker in the experiment. (middle) Individual boxplots of height judgments, for each adult male speaker. (right) The marginal distribution of the height judgments shown in the middle plot.

### 4.6.1 Updating the model

In order to model the effects of the different speakers in our data, our model formula must be updated to include our S (speaker) predictor in parentheses, like this:

height ~ 1 + ( 1 | L) + ( 1 | S)

The new term in parenthesis indicates a second predictor whose levels we want to estimate using partial pooling. Again, we include only a one on the left hand side of the pipe indicating that we are only estimating speaker-specific intercepts. So, this formula says ‘predict height using only an overall intercept, but also estimate a different intercept for each level of listener and speaker’. This regression model corresponding to the formula above now looks like this:

Just as with the listener coefficients (), the speaker coefficients () represent deviations in values from the intercept and so the speaker average is set to zero. In plain English, the model description above says:

We expect height judgments to be normally distributed around the expected value for any given trial, , with some unknown standard deviation . The expected value for a trial is equal to the sum of a fixed overall average (Intercept), some value associated with the individual listener () who judged the trial, and some value associated with the individual speaker () who produced the trial. The listener and speaker coefficients ( and ) were both modelled as coming from normal distributions whose standard deviations, and , were estimated from the data. Prior distributions for , , , and the intercept were all specified a priori.

The model we fit last chapter only included variation due to error. The first model we fit in this chapter divided variation into systematic between-listener variation and random within-listener error. Now, our model attempts to break up the total variance in our data into three components: Between-listener variation, between-speaker variation, and random error, as seen below.

^2\_{total} = ^2\_{L} + ^2\_{S} + ^2 (#eq:210)

If the error was the within-listener variation before, what does it represent now? Our error is now the difference between the value we expect, given the speaker *and* listener, and the value we observe. So, it is the variation *controlling for* the listener and listener. In other words, if you know who the speaker is and who the listener is, and you control for this, how much random variability do you still have left over in your model? The answer is .

### 4.6.2 Fitting and interpreting the model

Below, we fit the new model using exactly the same code as for multilevel\_model, save for the modification to the model formula. We don’t need to specify a hyperprior for because this falls under the sd category (see section @ref({c3-fitting-1})). As long as we can use the same hyperprior for each factor-specific standard deviation, i.e.  for factor , we don’t need to specify a prior for each parameter.

# Fit the model yourself  
set.seed (1)  
multilevel\_L\_S =   
 brms::brm (height ~ 1 + (1|L) + (1|S), data = men, chains = 4, cores = 4,  
 warmup = 1000, iter = 3500, thin = 2,  
 prior = c(brms::set\_prior("normal(176, 15)", class = "Intercept"),  
 brms::set\_prior("normal(0, 15)", class = "sd"),  
 brms::set\_prior("normal(0, 15)", class = "sigma")))

After fitting the model we inspect the output:

bmmb::short\_summary (multilevel\_L\_S)

The model print statement should look familiar except it contains one new chunk representing information about the speaker predictor in our model. This chunk reminds us that our S predictor had 45 levels, and provides us information about the estimate of the parameter (sd(Intercept)), which is 2.84 according to our model.

## 4.7 Investigating ‘shrinkage’

We’ve mentioned that multilevel models can result in *shrinkage*, that is, in coefficient estimates that are smaller in magnitude (i.e. closer to the mean) than their maximum likelihood counterparts. In the code below we extract the posterior mean estimates of our listener and speaker random effects (i.e., and ). We also calculate the maximum likelihood estimates for the listener and speaker averages, and center these so that they will resemble their corresponding effects.

# these are the Bayesian 'shrinkage' estimates  
L\_hat = brms::ranef(multilevel\_L\_S)$L[,1,'Intercept']  
S\_hat = brms::ranef(multilevel\_L\_S)$S[,1,'Intercept']  
  
# these are the maximum likelihood estimates  
L\_ml = tapply (men$height, men$L, mean)  
S\_ml = tapply (men$height, men$S, mean)  
# and now they are centered  
L\_ml = L\_ml - mean (L\_ml)  
S\_ml = S\_ml - mean (S\_ml)

In the top row of figure 4.8 we see a comparison of the Bayesian estimates of the speaker and listener effects compared to their maximum likelihood estimates. We can see that the Bayesian listener effects are nearly identical to their maximum likelihood counterparts. In the bottom row we see the difference between our two estimates plotted against the maximum likelihood estimates. In this row we see that the degree to which an estimate is shrunk towards the mean depends on how much it deviates from the mean.

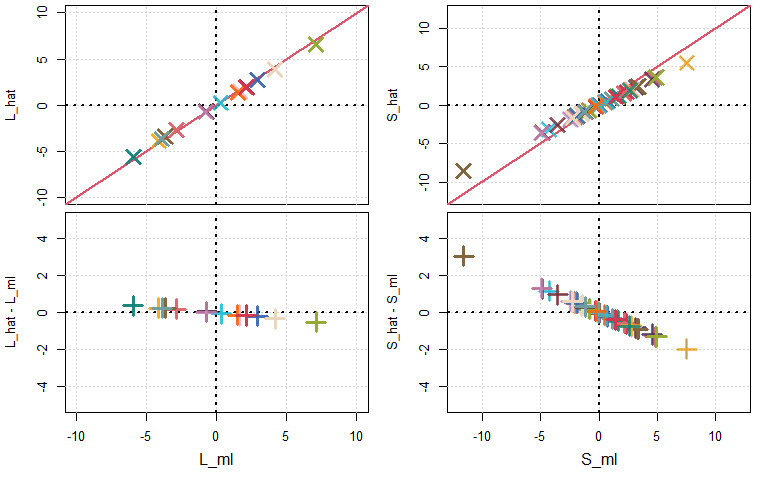


Figure 4.8: (top row) Maximum likelihood (ml) estimates of listener and speaker effects plotted against the Bayesian estimates of the same effects (hat). (bottom row) The y-axis now represents the difference between the Bayesian and maximum likelihood estimates. Positive values indicate that the Bayesian estimate was greater than the maximum likelihood one, and vice versa.

Figure 4.9 shows the effect of shrinkage for the listener and speaker effects: Individual effects are ‘pulled’ towards the overall mean (represented by zero). The reason the listener effects are shrunk less than the speaker effects is because we have 45 observations for each listener but only 15 for each speaker. As a result the likelihood functions for our listener effects will be much narrower than those of our speaker effects. This has the effect of making our listener parameters more resistant to the effects of the prior, and therefore more resistant to shrinkage (see sections 3.2.0.2 and 2.5.1 for more information).

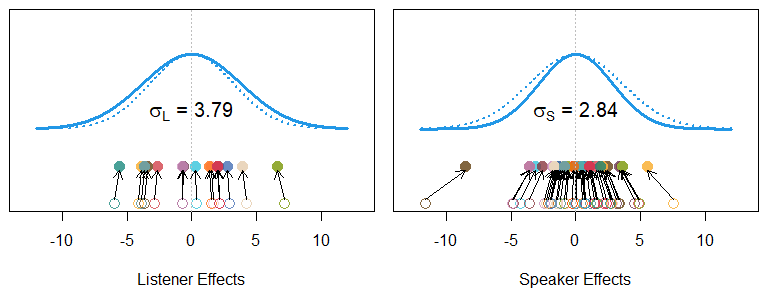


Figure 4.9: Empty points and dotted lines indicate the maximum likelihood estimates for listener and speaker effects, and the densities corresponding to the standard deviations of these effects. The filled points and solid lines indicate the Bayesian estimates and corresponding density estimates.

## 4.8 Answering our research questions

Let’s return to the research questions we posed in section 4.3.1:

Q1) How tall does the average adult male from the US sound?

Q2) Can we set limits on credible average apparent heights based on the data we collected?

We can consider the answers provided to this question by the complete pooling model we fit at the end of last chapter:

bmmb::short\_summary (model\_priors)  
## Formula: height ~ 1  
## Population-Level Effects:  
## Estimate Est.Error l-95% CI u-95% CI  
## Intercept 173.8 0.3 173.21 174.39  
##   
## Family Specific Parameters:  
## Estimate Est.Error l-95% CI u-95% CI  
## sigma 7.78 0.21 7.37 8.22

And compare this to the final model we considered above, which contained information about speakers and appropriate priors:

bmmb::short\_summary (multilevel\_L\_S)  
## Formula: height ~ 1 + (1 | L) + (1 | S)  
##   
## Group-Level Effects:  
## ~L (Number of levels: 15)  
## Estimate Est.Error l-95% CI u-95% CI  
## sd(Intercept) 3.79 0.86 2.51 5.83  
##   
## ~S (Number of levels: 45)  
## Estimate Est.Error l-95% CI u-95% CI  
## sd(Intercept) 2.84 0.43 2.1 3.77  
##   
## Population-Level Effects:  
## Estimate Est.Error l-95% CI u-95% CI  
## Intercept 173.83 1.12 171.64 176.07  
##   
## Family Specific Parameters:  
## Estimate Est.Error l-95% CI u-95% CI  
## sigma 6.47 0.18 6.12 6.84

Our complete pooling model from the previous chapter, model\_priors, and our partial pooling model from this chapter, multilevel\_L\_S, agree on the average apparent height as reflected by their respecive intercept terms. However, they disagree on a credible interval for that parameter with the complete pooling model having a credible interval that is nearly four times narrower (1.2 cm vs 4.4 cm). This is because the complete pooling model did not specify information about repeated measures, which caused our model to think that it had more independent observations than it did. This led the initial model to return an overly-precise estimate. Another difference is that the final model has a smaller sigma parameter (7.8 vs 6.5), which indicates that the error is smaller in the final model than in the initial model. Keep in mind that ‘error’ is just what your model can’t explain. Our final model explains more and so there is less error. The reduced error is a direct consequence of the fact that the final model splits variation in the data into several components (i.e., ). Since is a fixed value (given the data), obviously, the larger your between-speaker and between-listener variation is, the smaller your random error () *must* be.

Usually, parameters should be reported with *at least* the mean/median and standard deviations of the posterior distribution, in addition to some useful credible interval (i.e., 50%, 95%) around that parameter. Based on the result of our final model, a thorough answer to our research questions might go something like:

"Based on our model the average apparent height of adult males is likely to be 174.8 cm (s.d. = 1.1, 95% CI = 171.6, 176.0). The estimated magnitude of the random error was 6.5 cm (s.d. = 0.2, 95% CI = 6.1, 6.8). Consistent between-listeners variation averages about 3.8 cm (s.d. = 0.9, 95% CI = 2.5, 5.8), while consistent between-listeners variation averages about 2.8 cm (s.d. = 0.4, 95% CI = 2.1, 3.8).

## 4.9 Frequentist corner

In frequentist corner, we’re going to compare the output of brms to some more ‘traditional’, often called *frequentist* approaches. We’re not going to talk about the traditional models in much detail since there are hundreds of other sources for this, and this book is long enough as it is (see section @@ for suggestions). The focus of these sections is simply to highlight the similarities between different approaches, and to point out where to find equivalent information in the different models. If you are already familiar with these approaches, these sections may be helpful. If not, feel free to skip these sections of the book, although they may still be helpful.

### 4.9.1 Bayesian multilevel modesl vs. lmer

Here we compare the output of brms to the output of the lmer (“linear mixed-effects regression”) function, a very popular function for fitting multilevel models in the lme4 R package. The lmer function uses partial pooling to estimate its ‘random effects’ and no pooling to estimate its ‘fixed effects’. We are not going to talk about lmer in very much detail. The focus of this section is simply to highlight the similarities between different approaches, and to point out where to find equivalent information in both kinds of models. Below we use lmer to fit a model that is analogous to our multilevel\_L\_S model. Notice that the formulas are identical in both cases, except we do not specify priors for this model.

set.seed (1)  
lmer\_model = lme4::lmer (height ~ 1 + (1|L) + (1|S), data = men)

We can consider the short print statement for our Bayesian model, which we have annotated with numbers.

bmmb::short\_summary(multilevel\_L\_S)  
  
# (1) Formula: height ~ 1 + (1 | L) + (1 | S)  
#   
# (2) Group-Level Effects:  
# ~L (Number of levels: 15)  
# Estimate Est.Error l-95% CI u-95% CI  
# (3) sd(Intercept) 3.79 0.86 2.51 5.83  
#   
# ~S (Number of levels: 45)  
# Estimate Est.Error l-95% CI u-95% CI  
# (4) sd(Intercept) 2.84 0.43 2.1 3.77  
#   
# Population-Level Effects:  
# Estimate Est.Error l-95% CI u-95% CI  
# (5) Intercept 173.83 1.12 171.64 176.07  
#   
# Family Specific Parameters:  
# Estimate Est.Error l-95% CI u-95% CI  
# (6) sigma 6.47 0.18 6.12 6.84

And compare this to the print statement of our lmer model, annotated with the same numbers. We see that our models provide very similar information, although it is organized (and named) in somewhat different ways. We also see that the two approaches provide us with very similar parameter estimates for our models.

summary (lmer\_model)  
# Linear mixed model fit by REML ['lmerMod']  
# (1) Formula: height ~ 1 + (1 | L) + (1 | S)  
# Data: men  
#   
# REML criterion at convergence: 4527.4  
#   
# Scaled residuals:  
# Min 1Q Median 3Q Max  
# -4.6205 -0.4868 0.0722 0.5700 2.7179  
#   
# (2) Random effects:  
# Groups Name Variance Std.Dev.  
# (3) S (Intercept) 7.593 2.756  
# (4) L (Intercept) 11.990 3.463  
# (6) Residual 41.630 6.452  
# Number of obs: 675, groups: S, 45; L, 15  
#   
# Fixed effects:  
# Estimate Std. Error t value  
# (5) (Intercept) 173.788 1.015 171.3

We can get the random effects out of our brm model and compare them to the random effects we get from lmer. As seen below, the random effects estimates we get from brm also include credible intervals. As a result, we have some idea regarding the uncertainty in these estimates.

brms\_ranefs = brms::ranef (multilevel\_L\_S)$L[,,"Intercept"]  
head (brms\_ranefs)

In contrast, lmer gives you what are called **point estimates**. These are single estimates of parameter values with no intervals indicating uncertainty. Because of this, it is difficult to compare estimates for different speakers/participants in the data.

lmer\_ranefs = lme4::ranef (lmer\_model)[["L"]]  
head (lmer\_ranefs)

Importantly, the values we get from both approaches are nearly identical, as seen in 4.10 The *largest* difference between the two sets of coefficients is 0.05 cm for the listener effects and 0.06 cm for the speaker effects, with the average absolute difference for both being around 0.03 cm. So, as a practical matter, analyzing this data using a Bayesian multilevel model provides some advantages (e.g. intervals around random effects), while still providing effectively the same ‘answers’ as a ‘frequentist’ approach to the data.

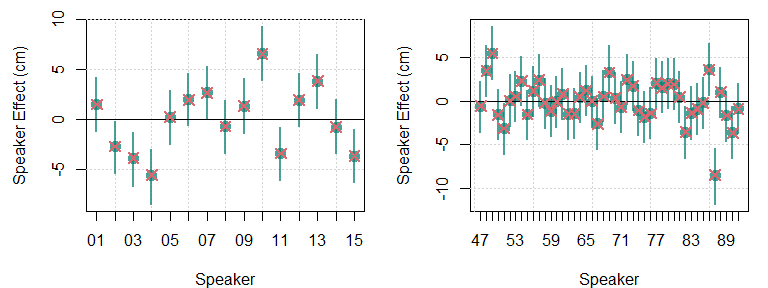


Figure 4.10: (left) In green, the random speaker intercept estimates provided by brm. The red arrows indicate the estimates of the same provided by lmer.