STATS 331

Introduction to Bayesian Statistics

Brendon J. Brewer

Contents

Chapter 1

Prologue

This course was originally developed by Dr Wayne Stewart (formerly of The University of Auckland) and was first offered in 2009 (Figure 1.1). I joined the Department of Statistics in July 2012 and took over the course from him. It was good fortune for me that Wayne left the university as I arrived. If I had been able to choose which undergraduate course I would most like to teach, it would have been this one!

Wayne is a passionate Bayesian[1] and advocate for the inclusion of Bayesian statistics in the undergraduate statistics curriculum. I also consider myself a Bayesian and agree that this approach to statistics should form a greater part of statistics education than it does today. While this edition of the course differs from Wayne's in some ways[2], I hope I am able to do the topic justice in an accessible way.

In this course we will use the following software:

- R (`http://www.r-project.org/`)
- JAGS (`http://mcmc-jags.sourceforge.net/`)
- The `rjags` package in R
- RStudio (`http://www.rstudio.com/`)

You will probably have used R, at least a little bit, in previous statistics courses. RStudio is just a nice program for editing R code, and if you don't like it, you're welcome to use any other text editor. JAGS is in a different category and you probably won't have seen it before. JAGS is used to implement Bayesian methods in a straightforward way, and `rjags` allows us to use JAGS from within R. Don't worry, it's not too difficult to learn and use JAGS! We will have a lot of practice using it in the labs.

These programs are all free and open source software. That is, they are free to use, share and modify. They should work on virtually any operating system including the three

[1] Bayesian statistics has a way of creating extreme enthusiasm among its users. I don't just *use* Bayesian methods, I *am a Bayesian*.

[2] The differences are mostly cosmetic. 90% of the content is the same.

Freaky meets Tom
as paradigms
clash!

Introductory Bayesian
Data Analysis
STATS 390

©Department of Statistics – Bayesian Statistics

STATS 390 , S8

Figure 1.1: *An ad for the original version of this course (then called STATS 390), showing Wayne Stewart with two ventriloquist dolls (Tom Bayes and Freaky Frequentist), who would have debates about which approach to statistics is best.*

most popular: Microsoft Windows, Mac OS X and GNU/Linux. In previous editions of the course, another program called WinBUGS was used instead of JAGS. Unfortunately, WinBUGS has not been updated for several years, and only works on Microsoft Windows. Therefore I switched over to JAGS in 2013. The differences between JAGS and WinBUGS are fairly minor, but JAGS has the advantage of being open source and cross-platform. All of this software is already installed on the lab computers, but if you would like to install it on your own computer, instructions are provided on the Course Information Sheet.

1.1 Bayesian and Classical Statistics

Throughout this course we will see many examples of Bayesian analysis, and we will sometimes compare our results with what you would get from *classical* or *frequentist* statistics, which is the other way of doing things. You will have seen some classical statistics methods in STATS 10X and 20X (or BioSci 209), and possibly other courses as well. You may have seen and used Bayes' rule before in courses such as STATS 125 or 210. Bayes' rule can sometimes be used in classical statistics, but in Bayesian stats it is used *all the time*).

Many people have differing views on the status of these two different ways of doing statistics. In the past, Bayesian statistics was controversial, and you had to be very brave to admit to using it. Many people were **anti**-Bayesian! These days, instead of

Bayesians and anti-Bayesians, it would be more realistic to say there are Bayesians and *non*-Bayesians, and many of the non-Bayesians would be happy to use Bayesian statistics in some circumstances. The non-Bayesians would say that Bayesian statistics is *one way* of doing things, and it is a matter of choice which one you prefer to use. Most Bayesian statisticians think Bayesian statistics is *the right way* to do things, and non-Bayesian methods are best thought of as either approximations (sometimes very good ones!) or alternative methods that are only to be used when the Bayesian solution would be too hard to calculate.

Sometimes I may give strongly worded opinions on this issue, but there is one important point that you should keep in mind throughout this course:

You do not have to agree with me in order to do well in STATS 331!

1.2 This Version of the Notes

Wayne Stewart taught STATS 331 with his own course notes. When I took over the course, I found that our styles were very different, even though we teach the same ideas. Unfortunately, it was challenging for the students to reconcile my explanations with Wayne's. Therefore I thought it would be better to have my own version of the notes. These lecture notes are a work in progress, and do not contain everything we cover in the course. There are many things that are important and examinable, and will be only discussed in lectures, labs and assignments!

The plots in these notes were not produced using R, but using a different plotting package where I am more familiar with the advanced plotting features. This means that when I give an R command for a plot, it will not produce a plot that looks exactly like the plot that follows. However, it will give approximately the same plot, conveying the same information. I apologise if you find this inconsistency distracting.

At this stage, the course notes contain the basic material of the course. Some more advanced topics will be introduced and discussed in lectures, labs and assignments.

I appreciate any feedback you may have about these notes.

1.3 Assessment

The assessment for this course is broken down as follows:

- 20% Assignments. There will be four assignments, worth 5% each. The assignments are not small, so please **do not** leave them until the last minute.

- 20% Midterm test (50 minutes, calculators permitted). This will be held in class, in place of a lecture, some time just after mid semester break.
- 60% Final exam (two hours, calculators permitted).

Chapter 2

Introduction

Every day, throughout our lives, we are required to believe certain things and not to believe other things. This applies not only to the "big questions" of life, but also to trivial matters, and everything in between. For example, this morning I boarded the bus to university, sure that it would actually take me here and not to Wellington. How did I know the bus would not take me to Wellington? Well, for starters I have taken the same bus many times before and it has always taken me to the university. Another clue was that the bus said "Midtown" on it, and a bus to Wellington probably would have said Wellington, and would not have stopped at a minor bus stop in suburban Auckland. None of this evidence *proves* that the bus would take me to university, but it does makes it very *plausible*. Given all these pieces of information, I feel quite certain that the bus will take me to the city. I feel so certain about this that the possibility of an unplanned trip to Wellington never even entered my mind until I decided to write this paragraph.

Somehow, our brains are very often able to accurately predict the correct answer to many questions (e.g. the destination of a bus), even though we don't have all the available information that we would need to be 100% certain. We do this using our experience of the world and our intuition, usually without much conscious attention or problem solving. However, there are areas of study where we can't just use our intuition to make judgments like this. For example, most of science involves such situations. Does a new treatment work better than an old one? Is the expansion of the universe really accelerating? People tend to be interested in trying to answer questions that haven't been answered yet, so our attention is always on the questions where we're not sure of the answer. This is where statistics comes in as a tool to help us in this grey area, when we can't be 100% certain about things, but we still want to do the best we can with our incomplete information.

2.1 Certainty, Uncertainty and Probability

In the above example, I said things like "I couldn't be 100% certain". The idea of using a number to describe how certain you are is quite natural. For example, contestants on the TV show "Who Wants to be a Millionaire" often say things like "I'm 75% sure the answer is A”[1].

There are some interesting things to notice about this statement. Firstly, it is a subjective statement. If someone else were in the seat trying to answer the question, she might say the probability that A is correct is 100%, because she knows the answer! A third person faced with the same question might say the probability is 25%, because he has no idea and only knows that one of the four answers must be correct.

In Bayesian statistics, the interpretation of what *probability* means is that it is a description of *how certain you are that some statement, or proposition, is true.* If the probability is 1, you are sure that the statement is true. So sure, in fact, that nothing could ever change your mind (we will demonstrate this in class). If the probability is 0, you are sure that the proposition is false. If the probability is 0.5, then you are as uncertain as you would be about a fair coin flip. If the probability is 0.95, then you’re quite sure the statement is true, but it wouldn’t be *too* surprising to you if you found out the statement was false. See Figure 2.1 for a graphical depiction of probabilities as degrees of certainty or plausibility.

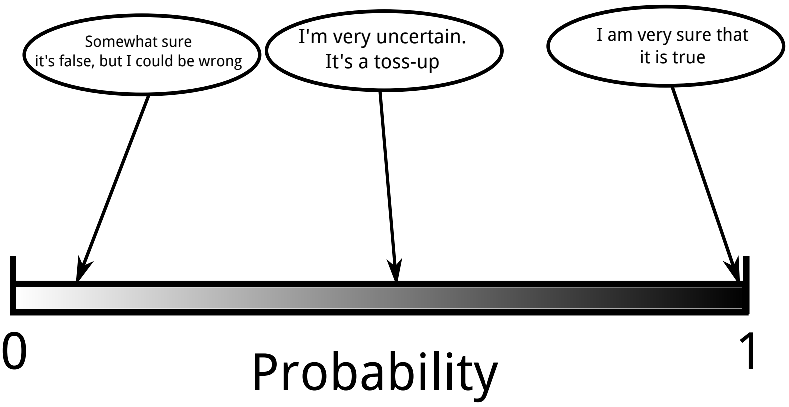

Figure 2.1: *Probability can be used to describe degrees of certainty, or how plausible some statement is. 0 and 1 are the two extremes of the scale and correspond to complete certainty. However, probabilities are not static quantities. When you get more information, your probabilities can change.*

In Bayesian statistics, probabilities are in the mind, not in the world.

It might sound like there is nothing more to Bayesian statistics than just thinking about a question and then blurting out a probability that feels appropriate. Fortunately for us, there’s more to it than that! To see why, think about how you change your mind when new evidence (such as a data set) becomes available. For example, you may be on “Who Wants to be a Millionaire?” and not know the answer to a question, so you might think the probability that it is A is 25%. But if you call your friend using “phone a friend”, and your friend says, “It’s definitely A”, then you would be much more confident that it is A! Your probability probably wouldn’t go all the way to 100% though, because there is

[1] This reminds me of an amusing exchange from the TV show *Monk*. **Captain Stottlemeyer**: [about someone electrocuting her husband] Monk, are you sure? I mean, are you really sure? And don’t give me any of that “95 percent” crap. **Monk**: Captain, I am 100% sure... that she *probably* killed him.

always the small possibility that your friend is mistaken.

When we get new information, we should ***update*** **our probabilities to take the new information into account. Bayesian methods tell us exactly how to do this.**

In this course, we will learn how to do *data analysis* from a Bayesian point of view. So while the discussion in this chapter might sound a bit like philosophy, we will see that using this kind of thinking can give us new and powerful ways of solving practical data analysis problems. The methods we will use will all have a common structure, so if you are faced with a completely new data analysis problem one day, you will be able to design your own analysis methods by using the Bayesian framework. Best of all, the methods make sense and perform extremely well in practice!

Chapter 3

First Examples

We will now look at a simple example to demonstrate the basics of how Bayesian statistics works. We start with some probabilities at the beginning of the problem (these are called *prior probabilities*), and how exactly these get updated when we get more information (these updated probabilities are called *posterior probabilities*). To help make things more clear, we will use a table that we will call a *Bayes' Box* to help us calculate the posterior probabilities easily.

Suppose there are two balls in a bag. We know in advance that at least one of them is black, but we're not sure whether they're both black, or whether one is black and one is white. These are the only two possibilities we will consider. To keep things concise, we can label our two competing hypotheses. We could call them whatever we want, but I will call them `BB` and `BW`. So, at the beginning of the problem, we know that *one and only one* of the following statements/hypotheses is true:

`BB`: Both balls are black
`BW`: One ball is black and the other is white.

Suppose an experiment is performed to help us determine which of these two hypotheses is true. The experimenter reaches into the bag, pulls out one of the balls, and observes its colour. The result of this experiment is (drumroll please!):

D: The ball that was removed from the bag was black.

We will now do a Bayesian analysis of this result.

3.1 The Bayes' Box

A Bayesian analysis starts by choosing some values for the prior probabilities. We have our two competing hypotheses `BB` and `BW`, and we need to choose some probability values to describe how sure we are that each of these is true. Since we are talking about two hypotheses, there will be two prior probabilities, one for `BB` and one for `BW`. For simplicity, we will assume that we don't have much of an idea which is true, and so we will use the following prior probabilities:

$$P(\texttt{BB}) = 0.5 \tag{3.1}$$
$$P(\texttt{BW}) = 0.5. \tag{3.2}$$

Pay attention to the notation. The upper case P stands for probability, and if we just write $P(\texttt{whatever})$, that means we are talking about the prior probability of `whatever`. We will see the notation for the posterior probability shortly. Note also that since the two hypotheses are mutually exclusive (they can't both be true) and exhaustive (one of these is true, it can't be some undefined third option). We will almost always consider mutually exclusive and exhaustive hypotheses in this course[1].

The choice of 0.5 for the two prior probabilities describes the fact that, before we did the experiment, we were very uncertain about which of the two hypotheses was true. I will now present a *Bayes' Box*, which lists all the hypotheses (in this case two) that might be true, and the prior probabilities. There are some extra columns which we haven't discussed yet, and will be needed in order to figure out the posterior probabilities in the final column. The first column of a Bayes' Box is just the list of hypotheses we are considering. In this case there are just two. If you need to construct a Bayes' box for a new problem, just think about what the possible answers to the problem are, and list them in the first column. The second column lists the prior probabilities for each of the hypotheses. Above, before we did the experiment, we decided to say that there was a 50% probability that BB is true and a 50% probability that BW is true, hence the 0.5 values in this column. The prior column should always sum to 1. Remember, the prior probabilities only describe our initial uncertainty, before taking the data into account. Hopefully the data will help by changing these probabilities to something a bit more decisive.

Hypotheses	prior	likelihood	prior × likelihood	posterior
BB	0.5			
BW	0.5			
Totals:	1			

3.1.1 Likelihood

The third column is called *likelihood*, and this is a really important column where the action happens. The likelihood is a quantity that will be used for calculating the posterior probabilities. In colloquial language, likelihood is synonymous with probability. It means the same thing. However, in statistics, likelihood is a very specific kind of probability. To fill in the third column of the Bayes' Box, we need to calculate two likelihoods, so you can tell from this that the likelihood is something different for each hypothesis. But what is it exactly?

[1] If this does not appear to be true in a particular problem, it is usually possible to redefine the various hypotheses into a set that of hypotheses that *are* mutually exclusive and exhaustive.

The likelihood for a hypothesis is the probability that you would have observed the data, if that hypothesis were true. The values can be found by going through each hypothesis in turn, imagining it is true, and asking, "What is the probability of getting the data that I observed?".

Here is the Bayes' Box with the likelihood column filled in. I will explain how these numbers were calculated in a bit more detail in the next subsection. If you have taken STATS 210 and used the maximum likelihood method, where you find the value of a parameter that maximises the likelihood function, that is the same as the likelihood we use in this course! So you have a head start in understanding this concept.

Hypotheses	prior	likelihood	h = prior $\times$ likelihood	posterior
BB	0.5	1		
BW	0.5	0.5		
Totals:	1			

3.1.2 Finding the Likelihood Values

We will first calculate the value of the likelihood for the BB hypothesis. Remember, the data we are analysing here is that we chose one of the balls in the bag "at random", and it was black. The likelihood for the BB hypothesis is therefore the probability that we would get a black ball if BB is true.

Imagine that BB is true. That means both balls are black. What is the probability that the experiment would result in a black ball? That's easy – it's 100%! So we put the number 1 in the Bayes Box as the likelihood for the BB hypothesis.

Now imagine instead that BW is true. That would mean one ball is black and the other is white. If this were the case and we did the experiment, what would be the probability of getting the black ball in the experiment? Since one of the two balls is black, the chance of choosing this one is 50%. Therefore, the likelihood for the BW hypothesis is 0.5, and that's why I put 0.5 in the Bayes' Box for the likelihood for BW.

In general, the likelihood is the *probability of the data that you actually got, assuming a particular hypothesis is true.* In this example it was fairly easy to get the likelihoods directly by asking "if this hypothesis is true, what is the probability of getting the black ball when we do the experiment?". Sometimes this is not so easy, and it can be helpful to think about ALL possible experimental outcomes/data you might have seen – even though ultimately, you just need to select the one that actually occurred. Table 3.1 shows an example of this process.

The fact that only the blue probabilities in Table 3.1 enter the Bayes' Box calculation is related to the *likelihood principle*, which we will discuss in lectures. Note also that in Table 3.1, the probabilities for the different possible data sets add to 1 within each hypothesis, but the sum of the blue "selected" likelihood values is not 1 (it is, in fact, meaningless).

Hypotheses	Possible Data	Probability
BB	Black Ball	1
	White Ball	0
BW	Black Ball	0.5
	White Ball	0.5

Table 3.1: *This table demonstrates a method for calculating the likelihood values, by considering not just the data that actually occurred, but all data that might have occurred. Ultimately, it is only the probability of the data which actually occurred that matters, so this is highlighted in blue.*

When we come to parameter estimation in later chapters, we will usually set up our problems in this way, by considering what data sets are possible, and assigning probabilities to them.

3.1.3 The Mechanical Part

The third column of the Bayes' Box is the product of the prior probabilities and the likelihoods, calculated by simple multiplication. The result will be called "prior times likelihood", but occasionally we will use the letter h for these quantities. This is the *unnormalised* posterior. It does not sum to 1 as the posterior probabilities should, but it is at least proportional to the actual posterior probabilities.

To find the posterior probabilities, we take the `prior` × `likelihood` column and divide it by its sum, producing numbers that do sum to 1. This gives us the final posterior probabilities, which were the goal all along. The completed Bayes' Box is shown below:

Hypotheses	prior	likelihood	h = prior × likelihood	posterior
BB	0.5	1	0.5	0.667
BW	0.5	0.5	0.25	0.333
Totals:	1		0.75	1

We can see that the posterior probabilities are not the same as the prior probabilities, because we have more information now! The experimental result made BB a little bit more plausible than it was before. Its probability has increased from 1/2 to 2/3.

3.1.4 Interpretation

The posterior probabilities of the hypotheses are proportional to the prior probabilies and the likelihoods. A high prior probability will help a hypothesis have a high posterior probability. A high likelihood value also helps. To understand what this means about reasoning, consider the meanings of the prior and the likelihood. There are two things that can contribute to a hypothesis being plausible:

- If the prior probability is high. That is, the hypothesis was *already* plausible, before we got the data.
- If the hypothesis *predicted the data* well. That is, the data was what we would have expected to occur if the hypothesis had been true.

I hope you agree that this is all very sensible.

In class we will also study variations on this problem, considering different assumptions about the prior probabilities and how they affect the results, and also considering what happens when we get more and/or different data.

3.2 Bayes' Rule

Bayes' rule is an equation from probability theory, shown in Figure 3.1. The various terms in Bayes' rule are all probabilities, but notice that there are conditional probabilities in there. For example, the left hand side of the equation is $P(A|B)$ and that means the probability of A **given** B. That is, it's the probability of A after taking into account the information B. In other words, $P(A|B)$ is a posterior probability, and Bayes' rule tells us how to calculate it from other probabilities. Bayes' rule is true for *any* statements A and B. If you took the equation in Figure 3.1 and replaced A with "Kākāpō will survive beyond 2050" and B with "I had coffee this morning", the resulting equation would still be true[2].

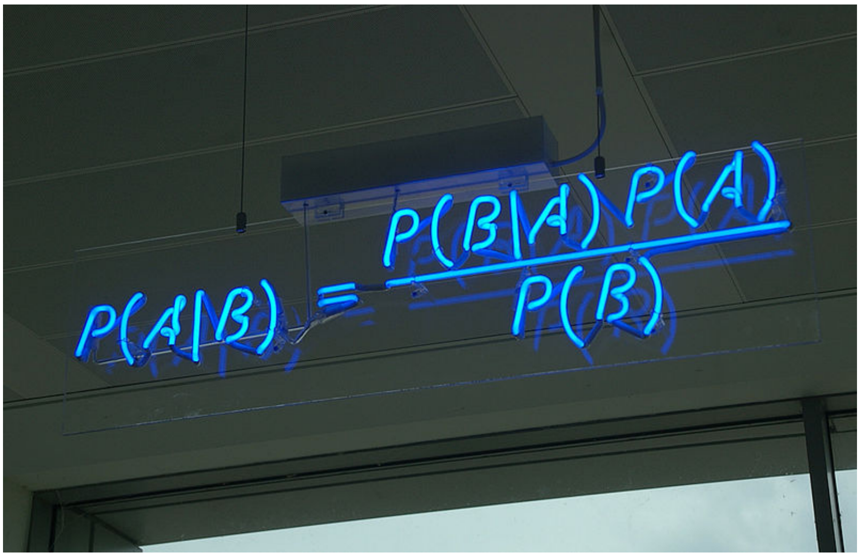

Figure 3.1: *A blue neon sign displaying Bayes' rule. You can use it to calculate the probability of A given B, if you know the values of some other probabilities on the right hand side. Image credit: Matt Buck. Obtained from Wikimedia Commons.*

It is helpful to relabel A and B in Bayes' rule to give a more clear interpretation of how the equation is to be used. In this version of Bayes' rule (which is one you should commit to memory), A has been replaced by H, and B has been replaced by D. The reason for these letters is that you should interpret H as *hypothesis* and D as *data.* Then you can interpret Bayes' rule as telling you the probability of a hypothesis given some data, in

[2] It would still be true, but it would not very interesting, because whether or not I had coffee doesn't tell you much about the survival prospects of endangered New Zealand parrots.

other words, a posterior probability.

$$P(H|D) = \frac{P(H)P(D|H)}{P(D)} \tag{3.3}$$

In Bayesian statistics, most of the terms in Bayes' rule have special names. Some of them even have more than one name, with different scientific communities preferring different terminology. Here is a list of the various terms and the names we will use for them:

- $P(H|D)$ is the **posterior probability**. It describes how certain or confident we are that hypothesis H is true, given that we have observed data D. Calculating posterior probabilities is the main goal of Bayesian statistics!

- $P(H)$ is the **prior probability**, which describes how sure we were that H was true, before we observed the data D.

- $P(D|H)$ is the **likelihood**. If you were to assume that H is true, this is the probability that you would have observed data D.

- $P(D)$ is the **marginal likelihood**. This is the probability that you would have observed data D, *whether H is true or not.*

Since you may encounter Bayesian methods outside of STATS 331, I have included an Appendix called "Rosetta Stone" that lists some common alternative terminology.

In the above example, we did some calculations to work out the numbers in the Bayes' Box, particularly the posterior probabilities, which are the ultimate goal of the calculation. *What we were actually doing in these calculations was applying Bayes' rule.* We actually applied Bayes' rule twice, once to compute $P(\texttt{BB}|D)$ and a second time to calculate $P(\texttt{BW}|D)$.

When you use a Bayes' Box to calculate posterior probabilities, you are really just applying Bayes' rule a lot of times: once for each hypothesis listed in the first column.

3.3 Phone Example

This example is based on Question 1 from the 2012 final exam. I got the idea for this question from an example in David MacKay's wonderful book "Information Theory, Inference and Learning Algorithms" (available online as a free PDF download. You're welcome to check it out, but it is a large book and only about 20% of the content is relevant to this course!).

You move into a new house which has a phone installed. You can't remember the phone number, but you suspect it might be `555-3226` (some of you may recognise this as being the phone number for Homer Simpson's "Mr Plow" business). To test this hypothesis, you carry out an experiment by picking up the phone and dialing `555-3226`.

If you are correct about the phone number, you will definitely hear a busy signal because you are calling yourself. If you are incorrect, the probability of hearing a busy signal is 1/100. However, all of that is only true if you assume the phone is working, and it might be broken! If the phone is broken, it will always give a busy signal.

When you do the experiment, the outcome (the data) is that you do actually get the busy signal. The question asked us to consider the following four hypotheses, and to calculate their posterior probabilities: Note that the four hypotheses are mutually exclusive and exhaustive. If you were to come up with hypotheses yourself, “phone is working” and “555-3226 is correct” might spring to mind. They wouldn’t be mutually exclusive so you couldn’t do a Bayes’ Box with just those two, but it is possible to put these together (using “**and**”) to make the four mutually exclusive options in the table.

Hypothesis	Description	Prior Probability
H_1	Phone is working and 555-3226 is correct	0.4
H_2	Phone is working and 555-3226 is incorrect	0.4
H_3	Phone is broken and 555-3226 is correct	0.1
H_4	Phone is broken and 555-3226 is incorrect	0.1

Table 3.2: *The four hypotheses about the state of the phone and the phone number. The prior probabilities are also given.*

3.3.1 Solution

We will go through the solution using a Bayes’ Box. The four hypotheses listed in Table 3.2 and their prior probabilities are given, so we can fill out the first two columns of a Bayes’ Box right away: The next thing we need is the likelihoods. The outcome of the experiment (the data) was the busy signal, so we need to work out $P(\text{busy signal}|H)$ for each H in the problem (there are four of them). Let’s start (naturally!) with H_1.

Hypotheses	prior	likelihood	prior × likelihood	posterior
H_1	0.4			
H_2	0.4			
H_3	0.1			
H_4	0.1			
Totals:	1			

If we assume H_1 is true, then the phone is working and 555-3226 is the correct phone number. In that case, we would definitely get a busy signal because we are calling ourselves. Therefore $P(\text{busy signal}|H_1) = 1$ is our first likelihood value.

Next, let’s imagine that H_2 is true, so the phone is working, but 555-3226 is not the right phone number. In this case, it is given in the question that the probability of getting a busy signal is 1/100 or 0.01 (in reality, this would be based on some other data, or perhaps be a totally subjective judgement). Therefore $P(\text{busy signal}|H_2) = 0.01$, and that’s our second likelihood value.

The likelihoods for H_3 and H_4 are quite straightforward because they both imply the phone is broken, and that means a busy signal is certain. Therefore $P(\text{busy signal}|H_3) = P(\text{busy signal}|H_4) = 1$. We have our four likelihoods, and can proceed to work out everything in the Bayes’ Box, including the main goal – the posterior probabilities! Here it is:

Hypotheses	`prior`	`likelihood`	`prior` × `likelihood`	`posterior`
H_1	0.4	1	0.4	0.662
H_2	0.4	0.01	0.004	0.00662
H_3	0.1	1	0.1	0.166
H_4	0.1	1	0.1	0.166
Totals:	1		0.604	1

To conclude this phone problem, I should admit that I actually calculated the numbers in the Bayes' Box using R. My code is shown below. A lot of the code we write in labs will look like this. Obviously in the 2012 exam the students had to use their calculators instead.

```
prior = c(0.4, 0.4, 0.1, 0.1) # Vector of prior probs
lik = c(1, 0.01, 1, 1)        # Vector of likelihoods
h = prior*lik
Z = sum(h)                    # Sum of prior times likelihood
post = prior*lik/Z            # Normalise to get posterior
# Look at all the results
print(prior)
print(lik)
print(h)
print(Z)
print(post)
```

Now let's try to see if this makes sense. There are many things we could think about, but let's just consider the question of whether the phone is working or not. The first two hypotheses correspond to the phone being in a working state. If you want to calculate the probability of A **or** B, then you can just add the probabilities if they are mutually exclusive. The prior probability that the phone is working is therefore:

$$\begin{aligned}
P(\text{phone working}) &= P(H_1 \vee H_2) && (3.4)\\
&= P(H_1) + P(H_2) && (3.5)\\
&= 0.4 + 0.4 && (3.6)\\
&= 0.8. && (3.7)
\end{aligned}$$

Here, I have introduced the notation $\vee$, meaning "logical or": For any two propositions A, B, the proposition $(A \vee B)$ is true if either one of A or B is true (or both).

The posterior probability is worked out in a similar way, but using the posterior probabilities instead of the prior ones:

$$\begin{aligned}
P(\text{phone working}|\text{busy signal}) &= P(H_1 \vee H_2|\text{busy signal}) && (3.8)\\
&= P(H_1|\text{busy signal}) + P(H_2|\text{busy signal}) && (3.9)\\
&= 0.662 + 0.00662 && (3.10)\\
&= 0.6689. && (3.11)
\end{aligned}$$

Our probability that the phone is working has gone down a little bit as a result of this evidence! That makes sense to me. A busy signal is what you would expect to happen if the phone was broken. This data doesn't *prove* the phone is broken, but it does point in that direction a little bit, and hence the probability that the phone is working has been reduced from 0.8 to 0.6689.

3.4 Important Equations

Posterior probabilities are calculated using Bayes' rule. For a single hypothesis H given data D, Bayes' rule is:

$$P(H|D) = \frac{P(H)P(D|H)}{P(D)} \tag{3.12}$$

This gives the posterior probability $P(H|D)$ in terms of the prior probability $P(H)$, the likelihood $P(D|H)$ and the marginal likelihood $P(D)$ in the denominator. To obtain $P(H)$, think about your prior beliefs (which may indicate a large amount of uncertainty, or may already be well informed based on previous data sets). To obtain $P(D|H)$, think about what the experiment is doing: If H is true, what data would you expect to see and with what probabilities?

The denominator is the probability of obtaining the data D but without assuming that H is either true or false. This is obtained using the sum rule. There are two ways that the data D could occur, either via the route of H being true (this has probability $P(H)P(D|H)$), or via the route of H being false (this has probability $P(\bar{H})P(D|\bar{H})$). These two ways are mutually exclusive, so we can add their probabilities:

$$P(D) = P(H)P(D|H) + P(\bar{H})P(D|\bar{H}). \tag{3.13}$$

Bayes' rule can be applied to a whole set of hypotheses (that are mutually exclusive and exhaustive) simultaneously. This is a more common way of using it, and it is the way we use it when we use a Bayes' Box. If we applied Equation 3.12 to N hypotheses $H_1, H_2, ..., H_N$, given data D, we would get the following for the posterior probability of each hypothesis H_i (for $i = 1, 2, ..., N$):

$$P(H_i|D) \quad = \quad \frac{P(H_i)P(D|H_i)}{P(D)} \tag{3.14}$$

The denominator $P(D)$ is a single number. It does not depend on the index i. It can again be obtained using the sum rule. There are N mutually exclusive ways that the data D could have occurred: via H_1 being true, or via H_2 being true, etc. Adding the probabilities of these gives:

$$P(D) \quad = \quad \sum_{i=1}^{N} P(H_i)P(D|H_i). \tag{3.15}$$

which just happens to be the sum of the prior times likelihood values. If you don't find equations particularly easy to read, just remember that following the steps for making a Bayes' Box is equivalent to applying Bayes' rule in this form! The $P(H_i)$ values are the prior probability column, the $P(D|H_i)$ values are the likelihood column, and the denominator is the sum of the prior times likelihood column. For example, the posterior probability for H_1 (the top right entry in a Bayes' Box) is given by the prior probability for H_1 times the likelihood for H_1, divided by the sum of prior times likelihood values. That is, $P(H_1|D) = P(H_1)P(D|H_1)/P(D)$. The correspondence between the probabilities that go in a Bayes' Box (in general) and the terms in the Equations are given in Table 3.3.

Hypotheses	`prior`	`likelihood`	`prior` × `likelihood`	`posterior`
H_1	$P(H_1)$	$P(D\|H_1)$	$P(H_1) \times P(D\|H_1)$	$P(H_1\|D)$
H_2	$P(H_2)$	$P(D\|H_2)$	$P(H_2) \times P(D\|H_2)$	$P(H_2\|D)$
...	...	...	...	...
Totals:	1		$P(D)$	1

Table 3.3: *A general Bayes' Box. Using Bayes' rule or making a Bayes' Box are actually the same thing, and this table can be used to identify the terms.*

Chapter 4

Parameter Estimation I: Bayes' Box

One of the most important times to use Bayes' rule is when you want to do *parameter estimation*. Parameter estimation is a fairly common situation in statistics. In fact, it is possible to interpret almost any problem in statistics as a parameter estimation problem and approach it in this way!

Firstly, what is a parameter? One way to think of a parameter is that it is just a fancy term for a quantity or a number that is unknown[1]. For example, how many people are currently in New Zealand? Well, a Google search suggests 4.405 million. But that does not mean there are **exactly** 4,405,000 people. It could be a bit more or a bit less. Maybe it is 4,405,323, or maybe it is 4,403,886. We don't really know. We could call the true number of people in New Zealand right now θ, or we could use some other letter or symbol if we want. When talking about parameter estimation in general we often call the unknown parameter(s) θ, but in specific applications we will call the parameter(s) something else more appropriate for that application.

The key is to realise that we can use the Bayes' Box, like in previous chapters. But now, our list of possible hypotheses is a list of possible values for the unknown parameter. For example, a Bayes' Box for the precise number of people in New Zealand might look something like the one in Table 4.1.

There are a few things to note about this Bayes' box. Firstly, it is big, which is why I just put a bunch of "..."s in there instead of making up numbers. There are lots of possible hypotheses, each one corresponding to a possible value for θ. The prior probabilities I have put in the second column were for illustrative purposes. They needn't necessarily all be equal (although that is often a convenient assumption). All the stuff we've seen in smaller examples of Bayes' rule and/or use of a Bayes' Box still applies here. The likelihoods will still be calculated by seeing how the probability of the data depends on the value of the unknown parameter. You still go through all the same steps, multiplying prior times likelihood and then normalising that to get the posterior probabilities for all of the possibilities listed. Note that a set of possible values together with the probabilities is what is commonly termed a *probability distribution*. In basic Bayesian problems, like in the introductory chapters, we start with some prior probabilities and update them to get

[1] Another use for the term parameter is any quantity that something else depends on. For example, a normal distribution has a mean μ and a standard deviation σ that defines which normal distribution we are talking about. μ and σ are then said to be parameters of the normal distribution.

Possible Hypotheses	`prior`	`likelihood`	`prior` × `likelihood`	`posterior`
...	...	...	...	...
$\theta = 4404999$	0.000001	...	...	...
$\theta = 4405000$	0.000001	...	...	...
$\theta = 4405001$	0.000001	...	...	...
$\theta = 4405002$	0.000001	...	...	...
$\theta = 4405003$	0.000001	...	...	...
$\theta = 4405004$	0.000001	...	...	...
$\theta = 4405005$	0.000001	...	...	...
$\theta = 4405006$	0.000001	...	...	...
...	...	...	...	...
Totals:	1		...	1

Table 4.1: *An example of how a Bayes' Box may be used in a parameter estimation situation.*

posterior probabilities. In parameter estimation, we start with a prior *distribution* for the unknown parameter(s) and update that to get a posterior *distribution* for the unknown parameter(s).

A quantity which has a probability associated with each possible value is traditionally called a "random variable". Random variables have probability distributions associated with them. In Bayesian stats, an unknown parameter looks mathematically like a "random variable", but I try to avoid the word random itself because it usually has connotations about something that fluctuates or varies. In Bayesian statistics, the prior distribution and posterior distribution only describe our uncertainty. The actual parameter is a single fixed number.

4.1 Parameter Estimation: Bus Example

This is a beginning example of parameter estimation from a Bayesian point of view. It shows the various features that are always present in a Bayesian parameter estimation problem. There will be a prior distribution, the likelihood, and the posterior distribution. We will spend a lot of time on this problem but keep in mind that this is just a single example, and certain things about this example (such as the choice of the prior and the likelihood) are specific to this example only, while other things about it are very general and will apply in all parameter estimation problems. You will see and gain experience with different problems in lectures, labs, and assignments.

After moving to Auckland, I decided that I would take the bus to work each day. However, I wasn't very confident with the bus system in my new city, so for the first week I just took the first bus that came along and was heading in the right direction, towards the city. In the first week, I caught 5 morning buses. Of these 5 buses, two of them took me to the right place, while three of them took me far from work, leaving me with an extra 20 minute walk. Given this information, I would like to try to infer the proportion of the buses that are "good", that would take me right to campus. Let us call this fraction θ and we will infer θ using the Bayesian framework. We will start with a prior distribution that describes initial uncertainty about θ and update this to get the posterior distribution, using the data that 2/5 buses I took were "good".

First we must think about the meaning of the parameter θ in our particular problem so we can choose a sensible prior distribution. Since θ is, in this example, a proportion, we know it cannot be less than 0 or greater than 1. In principle, θ could be any real value between 0 and 1. To keep things simple *to begin with*, we shall make an approximation and assume that the set of possible values for θ is:

$$\{0, 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9, 1\}.$$

This discrete approximation means that we can use a Bayes' Box. The first things to fill out in the Bayes' Box are the possible values and the prior probabilities (the prior distribution). For starters, let us assume that before we got the data (two successes out of 5 trials), we were very uncertain about the value of θ, and this can be modelled by using a uniform prior distribution. There are 11 possible values for θ that are being considered with our discrete approximation, so the probability of each is $1/11 = 0.0909$. The partially complete Bayes' Box is given in Table 4.2. Note the new notation that I have put in the column titles. We will use this notation in all of our parameter estimation examples (although the parameter(s) and data may have different symbols when θ and x respectively are not appropriate).

possible values θ	prior $p(\theta)$	likelihood $p(x\|\theta)$	prior × likelihood $p(\theta)p(x\|\theta)$	posterior $p(\theta\|x)$
0	0.0909			
0.1	0.0909			
0.2	0.0909			
0.3	0.0909			
0.4	0.0909			
0.5	0.0909			
0.6	0.0909			
0.7	0.0909			
0.8	0.0909			
0.9	0.0909			
1	0.0909			
Totals	1			1

Table 4.2: *Starting to make a Bayes' Box for the bus problem. This one just has the possible parameter values and the prior distribution.*

To get the likelihoods, we need to think about the properties of our experiment. In particular, we should imagine that we knew the value of θ and were trying to predict what experimental outcome (data) would occur. Ultimately, we want to find the probability of our actual data set (2 out of the 5 buses were "good"), for all of our possible θ values.

Recall that, if there are N repetitions of a "random experiment" and the "success" probability is θ at each repetition, then the number of "successes" x has a binomial distribution:

$$p(x|\theta) = \left(\begin{array}{c} N \\ x \end{array}\right) \theta^x (1-\theta)^{N-x}. \tag{4.1}$$

where $\left(\begin{array}{c} N \\ x \end{array}\right) = \frac{N!}{x!(N-x)!}$. This is the probability mass function for x (if we imagine θ to be known), hence the notation $p(x|\theta)$, read as "the probability distribution for x given θ". Since there are five trials ($N = 5$) in the bus problem, the number of successes x must be one of 0, 1, 2, 3, 4, or 5. If θ is a high number close to 1, then we would expect the resulting value of the data (number of successes) x to be something high like 4 or 5. Low values for x would still be possible but they would have a small probability. If θ is a small number, we would expect the data to be 0, 1, or 2, with less probability for more than 2 successes. This is just saying in words what is written precisely in Equation 5.1. The probability distribution for the data x is plotted in Figure 4.1 for three illustrative values of the parameter θ. To obtain the actual likelihood values that go into the Bayes' Box, we can simply substitute in the known values $N = 5$ and $x = 2$:

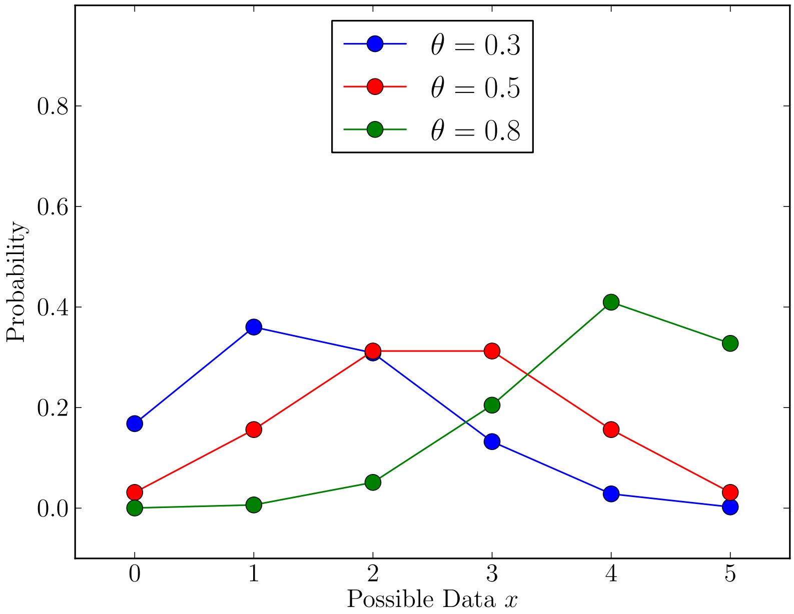

Figure 4.1: *The binomial probability distribution for the data x, for three different values of the parameter θ. If θ is low then we would expect to see lower values for the data. If θ is high then high values are more probable (but all values from 0 to 5 inclusive are still possible). The actual observed value of the data was $x = 2$. If we focus only on the values of the curves at $x = 2$, then the heights of the curves give the likelihood values for these three illustrative values of θ.*

$$P(x=2|\theta) = \left(\begin{array}{c} 5 \\ 2 \end{array}\right) \theta^2 (1-\theta)^{5-2} \tag{4.2}$$

$$= 10 \times \theta^2 (1-\theta)^3. \tag{4.3}$$

The resulting equation depends on θ only! We can go through the list of θ values and get a numerical answer for the likelihood $P(x = 2|\theta)$, which is what we need for the Bayes' Box.

possible values θ	prior $p(\theta)$	likelihood $p(x\|\theta)$	prior × likelihood $p(\theta)p(x\|\theta)$	posterior $p(\theta\|x)$
0	0.0909	0	0	0
0.1	0.0909	0.0729	0.0066	0.0437
0.2	0.0909	0.2048	0.0186	0.1229
0.3	0.0909	0.3087	0.0281	0.1852
0.4	0.0909	0.3456	0.0314	0.2074
0.5	0.0909	0.3125	0.0284	0.1875
0.6	0.0909	0.2304	0.0209	0.1383
0.7	0.0909	0.1323	0.0120	0.0794
0.8	0.0909	0.0512	0.0047	0.0307
0.9	0.0909	0.0081	0.0007	0.0049
1	0.0909	0	0	0
Totals	1		0.1515	1

Table 4.3: *The completed Bayes' Box for the bus problem (using a binomial distribution to obtain the likelihood).*

The final steps are, as usual, to multiply the prior by the likelihood and then normalise that to get the posterior distribution. The completed Bayes' Box is given in Table 4.3.

There are a few interesting values in the likelihood column that should help you to understand the concept of likelihood a bit better. Look at the likelihood for $\theta = 0$: it is zero. What does this mean? It means that if we imagine $\theta = 0$ is the true solution, the probability of obtaining the data that we got ($x = 2$ successes) would be zero. That makes sense! If $\theta = 0$, it means none of the buses are the "good" buses, so how could I have caught a good bus twice? The probability of that is zero.

The likelihood for $\theta = 1$ is also zero for similar reasons. If all of the buses are good, then having 2/5 successes is impossible. You would get 5/5 with 100% certainty. So $P(x = 2|\theta = 1) = 0$. The likelihood is highest for $\theta = 0.4$, which just so happens to equal 2/5. This $\theta = 0.4$ predicted the data best. It does not necessarily mean that $\theta = 0.4$ is the most probable value. That depends on the prior as well (but with a uniform prior, it does end up being that way. As you can see in the posterior distribution column, $\theta = 0.4$ has the highest probability in this case).

4.1.1 Sampling Distribution and Likelihood

As we study more examples of parameter estimation, you might notice that we always find the likelihood by specifying a probability distribution for the data given the parameters $p(x|\theta)$, and then we substituting in the actual observed data (Equations 4.1 and 4.3). Technically, only the version with the actual data set substituted in is called the likelihood. The probability distribution $p(x|\theta)$, which gives the probability of other data sets that did not occur (as well as the one that did), is sometimes called the *sampling distribution.* At times, I will distinguish between the sampling distribution and the likelihood, and at other times I might just use the word likelihood for both concepts.

4.1.2 What is the "Data"?

Even though this example is meant to be introductory, there is a subtlety that has been swept under the rug. Notice that our data consisted of the fact that we got 2/5 successes in the experiment. When we worked out the likelihood, we were considering the probability of getting $x = 2$, but we didn't have a probability for $N = 5$. In principle, we could treat x and N as two separate data sets. We could first update from the prior to the posterior given $N = 5$, and then update again to take into account x as well as N. However, the first update would be a bit weird. Why would knowing the number of trials tell you anything about the success probability? Effectively, what we have done in our analysis is assume that $N = 5$ is prior information that lurks in the background the whole time. Therefore our uniform prior for θ already "knows" that $N = 5$, so we didn't have to consider $P(N = 5|\theta)$ in the likelihood. This subtlety usually doesn't matter much.

4.2 Prediction in the Bus Problem

We have now seen how to use information (data) to update from a prior distribution to a posterior distribution when the set of possible parameter values is discrete. The posterior distribution is the complete answer to the problem. It tells us exactly how strongly we should believe in the various possible solutions (possible values for the unknown parameter). However, there are other things we might want to do with this information. Predicting the future is one! It's fun, but risky. Here we will look at how prediction is done using the Bayesian framework, continuing with the bus example. To be concrete, we are interested in the following question: *what is the probability that I will catch the right bus tomorrow?*. This is like trying to predict the result of a future experiment.

In the Bayesian framework, our predictions are always in the form of probabilities or (later) probability distributions. They are usually calculated in three stages. First, you pretend you *actually know* the true value of the parameters, and calculate the probability based on that assumption. Then, you do this for all possible values of the parameter θ (alternatively, you can calculate the probability as a function of θ). Finally, you combine all of these probabilities in a particular way to get one final probability which tells you how confident you are of your prediction.

Suppose we knew the true value of θ was 0.3. Then, we would know the probability of catching the right bus tomorrow is 0.3. If we knew the true value of θ was 0.4, we would say the probability of catching the right bus tomorrow is 0.4. The problem is, we don't know what the true value is. We only have the posterior distribution. Luckily, the sum rule of probability (combined with the product rule) can help us out. We are interested in whether I will get the good bus tomorrow. There are 11 different ways that can happen. Either $\theta = 0$ and I get the good bus, or $\theta = 0.1$ and I get the good bus, or $\theta = 0.2$ and I get the good bus, and so on. These 11 ways are all mutually exclusive. That is, only one of them can be true (since θ is actually just a single number). Mathematically, we can obtain the posterior probability of catching the good bus tomorrow using the sum rule:

$$\begin{aligned} P(\text{good bus tomorrow}|x) &= \sum_{\theta} p(\theta|x) P(\text{good bus tomorrow}|\theta, x) \qquad (4.4) \\ &= \sum_{\theta} p(\theta|x)\theta \qquad (4.5) \end{aligned}$$

This says that the total probability for a good bus tomorrow (given the data, i.e. *using the posterior distribution* and not the prior distribution) is given by going through each possible θ value, working out the probability *assuming the θ value you are considering is true*, multiplying by the probability (given the data) this θ value is actually true, and summing. In this particular problem, because $P(\text{good bus tomorrow}|\theta, x) = \theta$, it just so happens that the probability for tomorrow is the expectation value of θ using the posterior distribution. To three decimal places, the result for the probability tomorrow is 0.429. Interestingly, this is not equal to $2/5 = 0.4$.

In practice, these kinds of calculations are usually done in a computer. The R code for computing the Bayes' Box and the probability for tomorrow is given below. This is very much like many of the problems we will work on in labs.

```
# Make a vector of possibilities (first column of the Bayes' Box)
theta = seq(0, 1, by=0.1)

# Corresponding vector of prior probabilities
# (second column of the Bayes' Box)
prior = rep(1/11,11)

# Likelihood. Notice use of dbinom() rather than formula
# because R conveniently knows a lot of
# standard probability distributions already
lik = dbinom(2,5,theta)

# Prior times likelihood, then normalise to get posterior
h = prior*lik
post = h/sum(h)

# Probability for good bus tomorrow (prediction!)
# This happens to be the same as the posterior expectation of theta
# *in this particular problem* because the probability of a
# good bus tomorrow GIVEN theta is just theta.
prob_tomorrow = sum(theta*post)
```

4.3 Bayes' Rule, Parameter Estimation Version

Mathematically, what we did to calculate the posterior distribution was to take the prior distribution as a whole (the whole second column) and multiply it by the likelihood (the whole third column) to get the unnormalised posterior, then normalise to get the final posterior distribution. This can be written as follows, which we will call the "parameter estimation" version of Bayes' rule. There are three ways to write it:

$$\begin{aligned} p(\theta|x) &= \frac{p(\theta)p(x|\theta)}{p(x)} && (4.6) \\ p(\theta|x) &\propto p(\theta)p(x|\theta) && (4.7) \\ \texttt{posterior} &\propto \texttt{prior} \times \texttt{likelihood}. && (4.8) \end{aligned}$$

Writing the equations in these ways is most useful when you can write the prior $p(\theta)$ and the likelihood $p(x|\theta)$ as formulas (telling you how the values depend on θ as you go through the rows). Then you can get the equation for the posterior distribution (whether it is a discrete distribution, or a continuous one, in which case $p(\theta)$ and $p(\theta|x)$ are probability densities. We will do this in the next chapter.

The notation in Equation 4.8 is very simplified and concise, but is a popular kind of notation in Bayesian work. For an explanation of the relationship between this notation and other common choices (such as $P(X = x)$ for a discrete distribution or $f(x)$ for a density), see Appendix A.

Chapter 5

Parameter Estimation: Analytical Methods

Analytical methods are those which can be carried out with a pen and paper, or the "old school" way before we all started using computers. There are some problems in Bayesian statistics that can be solved in this way, and we will see a few of them in this course. For an analytical solution to be possible, the maths usually has to work out nicely, and that doesn't always happen, so the techniques shown here don't *always* work. When they do – great! When they don't, that's what MCMC (and JAGS) is for!

Let's look at the *binomial likelihood* problem again, with the familiar bus example. Out of $N = 5$ attempts at a "repeatable" experiment, there were $x = 2$ successes. From this, we want to infer the value of θ, the success probability that applied on each trial, or the overall fraction of buses that are good. Because of its meaning, we know with 100% certainty that θ must be between 0 and 1 (inclusive).

Recall that, if we knew the value of θ and wanted to predict the data x (regarding N as being known in advance), then we would use the binomial distribution:

$$p(x|\theta) = \left(\begin{array}{c} N \\ x \end{array}\right) \theta^x \left(1-\theta\right)^{N-x}. \tag{5.1}$$

Let's use a uniform prior for θ, but instead of making the discrete approximation and using a Bayes' Box, let's keep the continuous set of possibilities, that θ can be any real number between 0 and 1. Because the set of possibilities is continuous, the prior and the posterior for θ will both be probability *densities*. If we tried to do a Bayes' Box now, it would have infinitely many rows! The equation for our prior, a uniform probability density between 0 and 1, is:

$$p(\theta) = \left\{\begin{array}{ll} 1, & 0 \leq \theta \leq 1 \\ 0, & \text{otherwise} \end{array}\right. \tag{5.2}$$

If we keep in mind that θ is between 0 and 1, and therefore remember at all times that we are restricting our attention to $\theta \in [0, 1]$, we can write the uniform prior much more simply as:

$$p(\theta) = 1. \tag{5.3}$$

If you find the Bayes' Box way of thinking easier to follow than the mathematics here, you can imagine we are making a Bayes' Box like in Table 4.2, but with an "infinite" number of rows, and the equation for the prior tells us how the prior probability varies as a function of θ as we go down through the rows (since the prior is uniform, the probabilities don't vary at all).

To find the posterior probability density for θ, we use the "parameter estimation" form of Bayes' rule:

$$\begin{aligned} \text{posterior} &\propto \text{prior} \times \text{likelihood} && (5.4)\\ p(\theta|x) &\propto p(\theta)p(x|\theta). && (5.5)\end{aligned}$$

We already wrote down the equations for the prior and the likelihood, so we just need to multiply them.

$$\begin{aligned} p(\theta|x) &\propto p(\theta)p(x|\theta) && (5.6)\\ &\propto 1 \times \left(\begin{array}{c} N \\ x \end{array}\right) \theta^x (1-\theta)^{N-x} && (5.7)\end{aligned}$$

Since we are using the abbreviated form of the prior, we must remember this equation only applies for $\theta \in [0, 1]$. To simplify the maths, there are some useful tricks you can use a lot of the time when working things out analytically. Notice that the "parameter estimation" form of Bayes' rule has a proportional sign in it, not an equals sign. That's because the prior times the likelihood can't actually be the posterior distribution because it is not normalised. The sum or integral is not 1. However, the equation still gives the correct shape of the posterior probability density function (the way it varies as a function of θ). This is helpful because you can save ink. If there are some constant factors in your expression for the posterior that don't involve the parameter (in this case, θ), you can ignore them. The proportional sign will take care of them. In this case, it means we can forget about the pesky "N choose x" term, and just write:

$$\begin{aligned} p(\theta|x) &\propto \theta^x (1-\theta)^{N-x} && (5.8)\\ &\propto \theta^2 (1-\theta)^3. && (5.9)\end{aligned}$$

The final step I included was to substitute in the actual values of N and x instead of leaving the symbols there. That's it! We have the correct shape of the posterior distribution. We can use this to plot the posterior, as you can see in Figure 5.1.

5.1 "$\sim$" Notation

While it is very helpful to know the full equations for different kinds of probability distributions (both discrete and continuous), it is useful to be able to communicate about probability distributions in an easier manner. There is a good notation for this which we will sometimes use in STATS 331. If we want to communicate about our above analysis, and someone wanted to know what prior distribution we used, we could do several things. We could say "the prior for θ was uniform between 0 and 1", or we could give the formula for the prior distribution (Equation 5.2). However, a convenient shorthand in common use is to simply write:

$$\theta \sim \text{Uniform}(0, 1) \qquad (5.10)$$

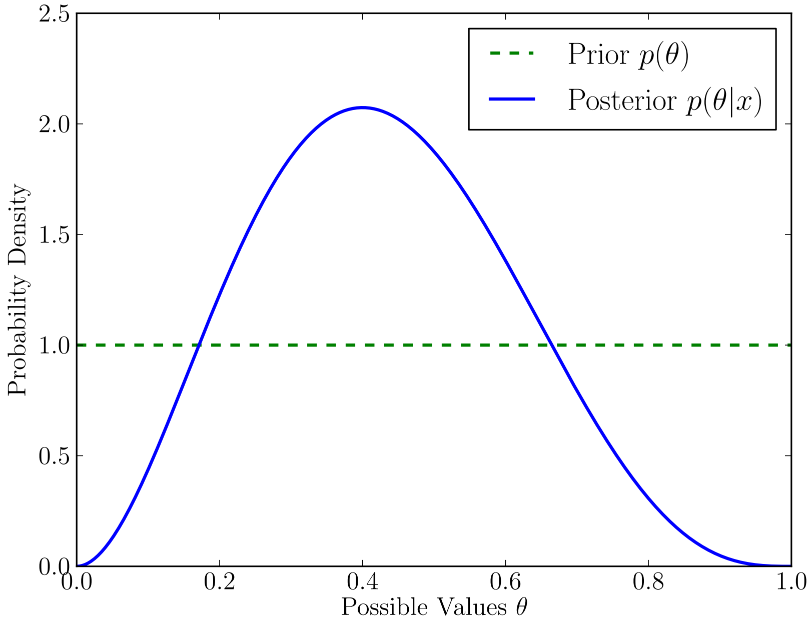

Figure 5.1: *The prior and the posterior for θ in the bus problem, given that we had 2/5 successes. The prior is just a uniform density and this is plotted as a flat line, describing the fact that θ can be anywhere between 0 and 1 and we don't have much of an idea. After getting the data, the distribution changes to the posterior which is peaked at 0.4, although there is still a pretty wide range of uncertainty.*

or, even more concisely:

$$\theta \sim U(0,1). \tag{5.11}$$

This notation conserves ink, and is good for quick communication. It is also very similar to the notation used in JAGS, which will be introduced in later chapters.

We can also write the binomial likelihood (which we used for the bus problem) in this notation, instead of writing out the full equation (Equation 5.1). We can write:

$$x|\theta \sim \text{Binomial}(N,\theta) \tag{5.12}$$

This says that if we knew the value of θ, x would have a binomial distribution with N trials and success probability θ. We can also make this one more concise:

$$x \sim \text{Bin}(N,\theta) \tag{5.13}$$

The differences here are that "Binomial" has been shortened to "Bin" and the "given θ" part has been left out. However, we see that there is a θ present on the right hand side, so the "given θ" must be understood implicitly.

5.2 The Effect of Different Priors

We decided to do this problem with a uniform prior, because it is the obvious first choice to describe "prior ignorance". However, in principle, the prior could be different. This will change the posterior distribution, and hence the conclusions. This isn't a problem of Bayesian analysis, but a feature. Data on its own doesn't tell us exactly what should believe. We must combine the data with all our other prior knowledge (i.e. put the data in context) to arrive at reasoned conclusions.

In this section we will look at the effect of different priors on the results, again focusing on the bus problem for continuity. Specifically, we will look at three different priors: the uniform one that we already used, and two other priors discussed below.

5.2.1 Prior 2: Emphasising the Extremes

One possible criticism of the uniform prior is that there is not much probability given to extreme solutions. For example, according to the Uniform(0, 1) prior, the prior probability that θ is between 0 and 0.1 is only $\int_0^{0.1} 1\, d\theta = 0.1$. But, depending on the situation, we might think values near zero should be more plausible[1]. One possible choice of prior distribution that assigns more probability to the extreme values (close to 0 or 1) is:

$$p(\theta) \propto \theta^{-\frac{1}{2}}(1-\theta)^{-\frac{1}{2}}. \tag{5.14}$$

5.2.2 Prior 3: Already Being Well Informed

Here's another scenario that we might want to describe in our prior. Suppose that, before getting this data, you weren't ignorant at all, but already had a lot of information about the value of the parameter. Say that we already had a lot of information which suggested the value of θ was probably close to 0.5. This could be modelled by the following choice of prior:

$$p(\theta) \propto \theta^{100}(1-\theta)^{100}. \tag{5.15}$$

The three priors are plotted in Figure 5.2 as dotted lines. The three corresponding posterior distributions are plotted as solid lines. The posteriors were computed by multiplying the three priors by the likelihood and normalising. The blue curves correspond to the uniform prior we used before, the red curves use the "emphasising the extremes" prior, and the green curves use the "informative" prior which assumes that θ is known to be close to 0.5.

There are a few interesting things to notice about this plot. Firstly, the posterior distributions are basically the same for the red and blue priors (the uniform prior and the "emphasising the extremes" prior). The main difference in the posterior is, as you would expect, that the extremes are emphasised a little more. If something is more plausible before you get the data, it's more plausible afterwards as well.

The big difference is with the informative prior. Here, we were already pretty confident that θ was close to 0.5, and the data (since it's not very much data) hasn't given us any

[1] Here's another parameter that is between 0 and 1: the proportion of households in New Zealand that keep a Macaw as a pet (call that ϕ). I hope this number is low (it is very difficult to take responsible care of such a smart bird). I also think it probably is low. I would definitely object to a prior that implied $P(\phi < 0.1) = 0.1$. I would want a prior that implied something like $P(\phi < 0.1) = 0.999999$.

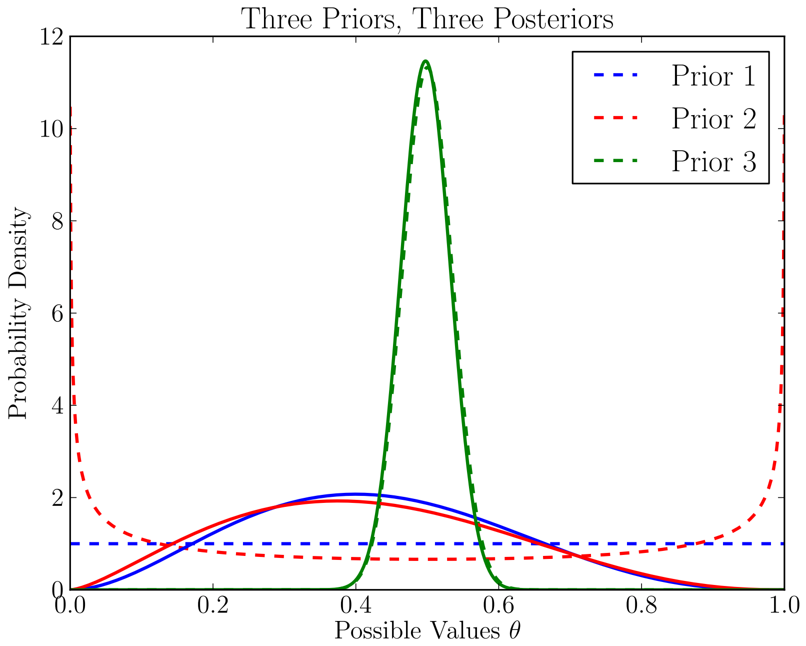

Figure 5.2: *Three different priors (dotted lines) and the corresponding three posteriors (solid lines) given the bus example data. See the text for discussion of these results.*

reason to doubt that, so we still think θ is close to 0.5. Since we already knew that θ was close to 0.5, the data are acting only to increase the precision of our estimate (i.e. make the posterior distribution narrower). But since we had so much prior information, the data aren't providing much "extra" information, and the posterior looks basically the same as the prior.

5.2.3 The Beta Distribution

The three priors we have used are all examples of *beta* distributions. The beta distributions are a family of probability distributions (like the normal, Poisson, binomial, and so on) which can be applied to continuous random variables known to be between 0 and 1. The general form of a beta distribution (here written for a variable x) is:

$$p(x|\alpha,\beta) \propto x^{\alpha-1}(1-x)^{\beta-1}. \tag{5.16}$$

The quantities α and β are two parameters that control the shape of the beta distribution. Since we know the variable x is between 0 and 1 with probability 1, the normalisation constant could be found by doing an integral. Then you could write the probability distribution with an equals sign instead of a proportional sign,

$$\begin{aligned} p(x|\alpha,\beta) &= \frac{x^{\alpha-1}(1-x)^{\beta-1}}{\int_0^1 x^{\alpha-1}(1-x)^{\beta-1}\,dx} & (5.17)\\ &= \frac{x^{\alpha-1}(1-x)^{\beta-1}}{B(\alpha,\beta)}. & (5.18)\end{aligned}$$

where $B(\alpha,\beta)$ (called the "beta function") is defined (usefully...) as the result of doing that very integral (it can be related to factorials too, if you're interested). Thankfully, we can get away with the "proportional" version most of the time. In "∼" notation the beta distribution is written as:

$$x|\alpha, \beta \quad \sim \quad \text{Beta}(\alpha, \beta). \tag{5.19}$$

Again, the "given α and β" can be dropped. It is implicit because they appear on the right hand side. By identifying the terms of Equation 5.16 with the form of our three priors (Equations 5.2, 5.14 and 5.15)), we see that our three priors can be written in "∼" notation like this:

$$\begin{array}{ll} \text{Prior 1:} & \theta \sim \text{Beta}(1, 1) \\ \text{Prior 2:} & \theta \sim \text{Beta}\left(\frac{1}{2}, \frac{1}{2}\right) \\ \text{Prior 3:} & \theta \sim \text{Beta}(101, 101) \end{array}$$

When you work out the posterior distributions analytically and then compare them to the formula for the beta distribution, you can see that the three posteriors are also beta distributions! Specifically, you get:

$$\begin{array}{ll} \text{Posterior 1:} & \theta \sim \text{Beta}(3, 4) \\ \text{Posterior 2:} & \theta \sim \text{Beta}(2.5, 3.5) \\ \text{Posterior 3:} & \theta \sim \text{Beta}(103, 104) \end{array}$$

This is "magic" that is made possible by the mathematical form of the beta prior and the binomial likelihood[2]. It is not always possible to do this.

We can also derive the general solution for the posterior for θ when the prior is a Beta(α, β) distribution, the likelihood is a binomial distribution, and x successes were observed out of N trials. The posterior is:

$$\begin{aligned} p(\theta|x) &\propto p(\theta)p(x|\theta) && (5.20) \\ &\propto \theta^{\alpha-1}(1-\theta)^{\beta-1} \times \theta^{x}(1-\theta)^{N-x} && (5.21) \\ &= \theta^{\alpha+x-1}(1-\theta)^{\beta+N-x-1} && (5.22) \end{aligned}$$

which can be recognised as a Beta$(\alpha + x, \beta + N - x)$ distribution.

Remember that in this particular problem, the probability of a success tomorrow is simply the expectation value (mean) of the posterior distribution for θ. We can look up (or derive) the formula for the mean of a beta distribution and find that if $x \sim \text{Beta}(\alpha, \beta)$ then $\mathbb{E}(x) = \alpha/(\alpha + \beta)$. Applying this to the three posterior distributions gives:

$$\begin{array}{lll} P(\text{good bus tomorrow}|x) = 3/7 & \approx 0.429 & \text{(using prior 1)} \\ P(\text{good bus tomorrow}|x) = 2.5/6 & \approx 0.417 & \text{(using prior 2)} \\ P(\text{good bus tomorrow}|x) = 103/207 & \approx 0.498 & \text{(using prior 3)} \end{array}$$

The result for Prior 1 is Laplace's infamous "rule of succession" which I will discuss a little bit in lectures.

[2] The technical term for this magic is that the beta distribution is a *conjugate prior* for the binomial likelihood.

5.2.4 A Lot of Data

As shown above, the choice of prior distribution has an impact on the conclusions. Sometimes it has a big impact (the results using prior 3 were pretty different to the results from priors 1 and 2), and sometimes not much impact (e.g. the results from priors 1 and 2 were pretty similar). There is a common phenomenon that happens when there is a lot of data: the prior tends not to matter so much. Imagine we did a much bigger version of the bus experiment with $N = 1000$ trials, which resulted in $x = 500$ successes. Then the posterior distributions corresponding to the three different priors are all very similar (Figure 5.3).

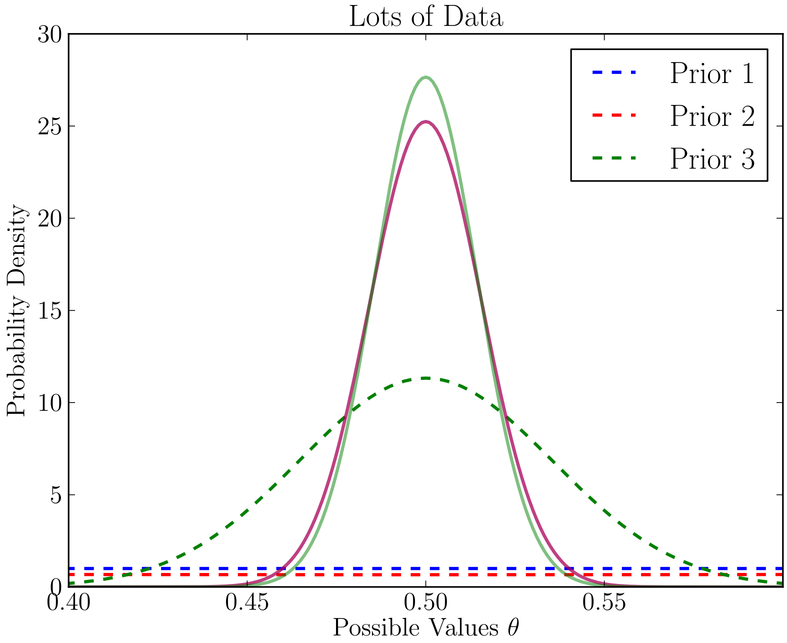

Figure 5.3: *When you have a lot of data, the results are less sensitive to the choice of prior distribution. Note that we have zoomed in and are only looking around $\theta = 0.5$: these posterior distributions are quite narrow because there is now a lot more information about θ. The red and blue posteriors (based on priors 1 and 2) are so similar that they overlap and look like one purple curve.*

This is reassuring. Note, however, that this only occurs because the three analyses used the same likelihood. If three people have different prior distributions for something *and* they can't agree on what the experiment even means, there is no guarantee they will end up agreeing, even if there's a large amount of data!

Remember though, that when the results *are* sensitive to the choice of prior, that is not a problem with the Bayesian approach, but rather an important warning message: the data aren't very informative! Then, the options are: i) think really hard about your prior distribution and be careful when deciding what it should be, and ii) get more or better data!

Chapter 6

Summarising the Posterior Distribution

The posterior distribution is the full answer to any Bayesian problem. It gives a complete description of our state of knowledge and our uncertainty about the value(s) of unknown parameters. From the posterior distribution, we can calculate any probability we want. For example, if we had a posterior distribution $p(\theta|x)$ and we wanted to know the probability that θ is greater than or equal to 100, we could do:

$$P(\theta \geq 100|x) = \int_{100}^{\infty} p(\theta|x)\, d\theta \tag{6.1}$$

or

$$P(\theta \geq 100|x) = \sum_{100}^{\infty} p(\theta|x) \tag{6.2}$$

depending on whether the set of possible θ values is continuous or discrete. We could also work out the probability of anything else. However, the posterior distribution is sometimes too much information for us to think about easily. Maybe a giant list of θ values and probabilities isn't easy to digest. Sometimes, we need to *summarise* the posterior distribution to help us communicate our results with others. A giant Bayes' Box (or a million MCMC samples of the parameter, we'll see that later), might technically contain everything we want, but it's not easy to talk about.

For example, say you were trying to estimate a parameter, and a colleague asked you to state your uncertainty about the parameter. Well, your posterior distribution might be complicated. It might have bumps and wiggles in it, or some other kind of structure. If there were two or more unknown parameters, there might be dependence in the posterior distribution. In some cases there might even me multiple separate peaks! Figure 6.1 shows an example of what a complicated posterior distribution might look like. If this was your result, your colleague might not care about all the little wiggles in this plot. They just want to know the "big picture" of your results.

The idea of summarising the posterior distribution is very closely related to the idea of summarising a data set, which you probably encountered when you studied descriptive statistics.

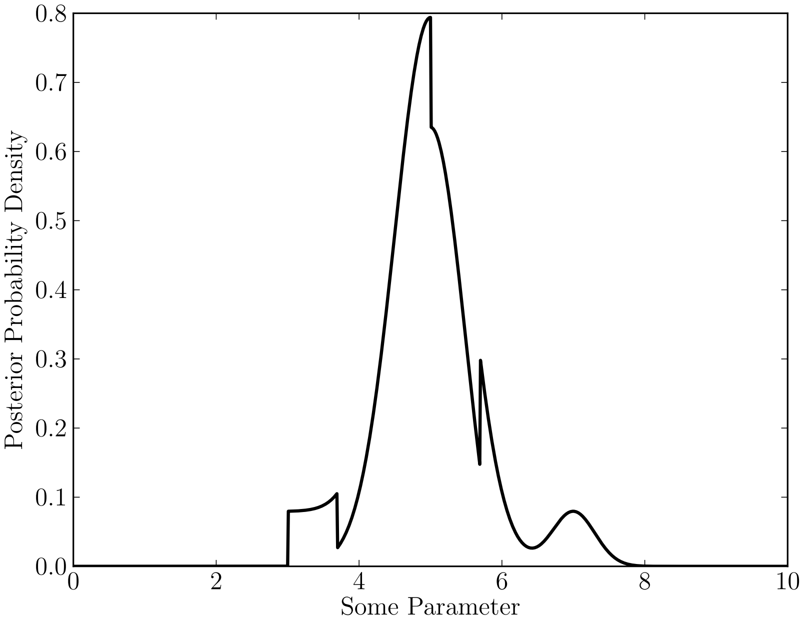

Figure 6.1: *A complicated posterior distribution. When communicating with others, it is often useful to summarise the posterior distribution with a few numbers. In this case, something like "the parameter* $= 5 \pm 1$*" might be a useful summary.*

In descriptive statistics, you often make summaries of a complex data set (e.g. the mean and the standard deviation) so that you can communicate about the data set in a concise way. In Bayesian statistics, you often do a similar thing, but instead of giving a concise description of the *data*, **you give a concise description of the** *posterior distribution*.

6.1 Point Estimates

A "point estimate" refers to a single number guess for the value of a parameter. If you have several parameters, a point estimate would be a single guess for the value of each parameter (like a single point in a multidimensional space). If you look at the posterior distribution plotted in Figure 6.1, you can see that the true value of the parameter is probably somewhere around 5, but with some uncertainty. If you were to provide a single number as a guess of the parameter, you would probably say something close to 5. In classical statistics, a single number guess is called an "estimate", and a rule for generating such guesses is called an "estimator". Estimates are usually written by putting a hat over the name of the parameter. So, by looking at the plot of the posterior, you could give an estimate like this:

$$\hat{\theta} = 5. \tag{6.3}$$

But there are better things you could do than just looking at the plot, and you've probably learnt some of them in previous statistics courses. Here are three methods you could use to choose a point estimate using the posterior distribution: the posterior mean (the expectation value of the parameter), the posterior median (the value that divides the probability in half), and the posterior mode (the value where the posterior distribution has its peak). In our illustrative example, the values of these three point estimates are:

$$\hat{\theta} = 4.988 \text{ (the posterior mean)} \tag{6.4}$$

$$\hat{\theta} = 4.924 \text{ (the posterior median)} \tag{6.5}$$

$$\hat{\theta} = 4.996 \text{ (the posterior mode)} \tag{6.6}$$

In this example, there's not much of a difference between these three methods. But in other situations, they can be quite different (this usually happens if the posterior distribution is skewed, or has multiple modes; you may notice a strong analogy between this topic and descriptive statistics). Is there a way to say which one is the *best*? It turns out there is, but that depends on what you mean by "best".

Before we move on to the formal ways of deciding what constitutes a good estimate, I would like to mention a very common method that is easy to use. If the posterior distribution looks even vaguely like a normal distribution, it is common to summarise it like this:

$$\theta = \text{posterior mean } \pm \text{posterior standard deviation.} \tag{6.7}$$

I use this kind of summary frequently in my own research.

6.1.1 A Very Brief Introduction to Decision Theory

Decision theory is a very important topic. In this course we will use a *tiny* amount of it, just enough to solve the problem of "which point estimate is best?". If you think about it, this is a bit of a weird question. Obviously, the best point estimate is the true value. Of course it is, how could it be otherwise? Our only problem is that we can't actually implement this suggestion. We don't know the true value. We only have the posterior distribution (which is based on all the evidence we have), and we have to do the best we can with our incomplete information. To think about which decision is best, the first thing we should think about is which decisions are *possible*. For estimating a single parameter, any real number is a possible guess.

The key idea in decision theory is the concept of *utility*, and the related concept of *loss* (loss is just negative utility). Utility is a numerical measure of how good it would be if a certain outcome came true. Conversely, loss is a measure of how bad it would be if a certain outcome came true. Utilities are often subjective (not unlike prior probabilities), but in some applications utility can be more concrete. For example, in betting or investment decisions the utility can be measured in dollars. The problem with utility is that we have uncertainty about what is going to happen, or about what is true, so we can't just choose the decision that gives us the greatest utility. Instead we will use our posterior probabilities and choose the decision that gives us the maximum possible *expected value* of the utility.

Imagine we were estimating a parameter θ and we wanted to give a point estimate $\hat{\theta}$. One idea for what the utility or loss might be is the *quadratic* loss function, which is given by

$$L(\theta, \hat{\theta}) = \left(\hat{\theta} - \theta\right)^2 . \tag{6.8}$$

This expression inside the parentheses is the difference between our point estimate and the true value. This formula says that if our point estimate is off by 2, that is four times worse than if we were off by 1. If we were off by 10, that is 100 times worse than if we were off by 1, due to the squaring in the quadratic loss function formula.

It turns out (we will prove this below) that *if the loss function is quadratic, the best estimate you can give is the posterior mean.* Here is the proof. The expected value of the loss is

$$\mathbb{E}\left[L(\theta,\hat{\theta})\right] = \int p(\theta|x)(\hat{\theta}-\theta)^2\,d\theta \tag{6.9}$$

Since we are summing (integrating) over all possible true θ values, the expected loss is only a function of our estimate $\hat{\theta}$. To minimise a function of one variable, you differentiate it and then set the derivative to zero. The derivative is

$$\begin{aligned}\frac{d}{d\hat{\theta}}\mathbb{E}\left[L(\theta,\hat{\theta})\right] &= \int p(\theta|x)\frac{d}{d\hat{\theta}}(\hat{\theta}-\theta)^2\,d\theta & (6.10)\\ &= \int p(\theta|x)2(\hat{\theta}-\theta)\,d\theta & (6.11)\end{aligned}$$

Setting this equal to zero and then solving for $\hat{\theta}$ gives the final result:

$$\hat{\theta} = \int \theta p(\theta|x)\,d\theta. \tag{6.12}$$

which is the posterior mean. Some people call the posterior mean the “Bayes Estimate” for this reason. I don’t like that term because I don’t think point estimates are really Bayesian. The actual output of a Bayesian analysis is the posterior distribution.

Note that I didn’t verify that $\hat{\theta}$ actually minimises the expected loss , because setting the derivative to zero would also find a maximum. To make sure it really does minimise the expected loss, you can calculate the second derivative and verify that it is positive. But that’s not really needed. It would be pretty bizarre if the posterior mean was the *worst* estimate!

6.1.2 Absolute Loss

Sometimes, the quadratic loss/utility is not a reasonable model for the consequences of an incorrect estimate. Another plausible form for the loss function is the *absolute* loss. This looks like:

$$L(\hat{\theta},\theta) = |\hat{\theta}-\theta|. \tag{6.13}$$

With this assumption, the “badness” of an incorrect estimate is proportional to how far the estimate is from the true value. If the estimate is twice as far from the true value, it is twice as bad. We will not prove it (although you are welcome to derive this yourself), but for this loss function the best estimate is the posterior median, which is the value of $\hat{\theta}$ for which $P(\theta \leq \hat{\theta}) = P(\theta > \hat{\theta}) = 0.5$.

6.1.3 All-or-nothing Loss

The third kind of loss function we will look at is the "all-or-nothing" loss, also sometimes called *0-1 loss*. Sometimes, you may need your estimate to be completely correct, and if it isn't correct, then it is irrelevant how far your estimate was from the true value. All incorrect estimates are equally bad. The all-or-nothing loss looks like:

$$L(\hat{\theta}, \theta) = \begin{cases} 0 & \hat{\theta} = \theta \\ 1, & \text{otherwise.} \end{cases} \tag{6.14}$$

If you were in this situation you would want to make your chances as high as possible, which implies you should simply choose the most probable value of θ as your point estimate $\hat{\theta}$. That is, the appropriate estimate is the posterior mode. This intuition is correct. With all-or-nothing loss, the best estimate is the posterior mode. The three loss functions we consider in STATS 331 are shown in Figure 6.2.

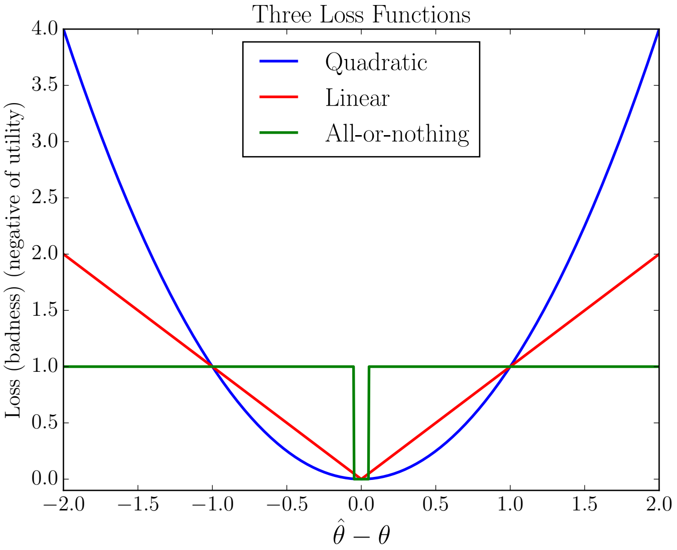

Figure 6.2: *Three kinds of loss function, which measure how bad it is for our point estimate $\hat{\theta}$ to be different from the true value of the parameter θ. Note that the all-or-nothing loss has a small amount of width in this plot, just so that we can clearly see the spike at $\hat{\theta} - \theta$ = 0.*

6.1.4 Invariance of Decisions

You may be wondering about the definitions of our loss functions. For example, we defined the quadratic loss as $(\hat{\theta} - \theta)^2$, but what if we defined it as $3(\hat{\theta} - \theta)^2 + 5$ instead? Would our best decision change? Luckily, the answer is no. The decision (estimate) which minimises the expected value of a loss function L also minimises the expected value of a different loss function $aL + b$, where a is any positive number and b is any other number. For the mathematicians, the optimal decision is invariant under positive affine transformations of the utility or loss function. Phew!

6.1.5 Computing Point Estimates from a Bayes' Box

We have just discussed three different point estimates, and under what circumstances we can consider them to be the best possible estimate we could make. Now, we will look at how to actually obtain the point estimates. The posterior mean is straightforward. It's the expectation value of the parameter using the posterior distribution. In R, the code is:

```
post_mean = sum(theta*post)
```

You should also know how to compute this manually from a Bayes' Box, using a calculator.

The posterior mode is also fairly straightforward. First, we can find the highest probability in the Bayes' Box. Then we find the corresponding parameter value.

```
highest_probability = max(post)
post_mode = theta[post == highest_probability]
```

In the case of a tie, `post_mode` might be a vector, indicating that there isn't a single mode.

The posterior median is a little harder. We need to find the θ value which has 50% of the probability to the left and 50% of the probability to the right. Note that this isn't precisely defined in some cases, particularly with discrete distributions. For example, if θ could be 1, 2, or 3, and the probabilities of these were 0.3, 0.6, and 0.1, then what is the median? It is not entirely clear. However, if there are a large number of possibilities then the definition becomes more clear.

To calculate the posterior median in R, we need to use the cumulative distribution which is defined as $F(t) = P(\theta \leq t)$. If we then find the value of t where $F(t) = 0.5$, we have found the posterior median. This isn't always possible but we can always find the value of t which makes $F(t)$ very close to 0.5. To obtain the cumulative distribution in R you can use the `cumsum` function, which calculates the cumulative sum of a vector. The posterior vector contains the probabilities of θ equalling certain values. If we want the probability that θ is less than or equal to a certain value, we sum all the probabilities up to and including that value. The cumulative sum function achieves this. Here is the code for calculating the posterior median:

```
F = cumsum(post)
dist = abs(F - 0.5) # Distance of the F-values from 0.5
post_median = theta[dist == min(dist)]
```

Note that this may also produce more than one result. Like the mode, the posterior median is not always uniquely defined.

6.1.6 Computing Point Estimates from Samples

When we use a Bayes' Box (or the equivalent R commands which represent the columns of a Bayes' Box as vectors), we end up with a vector of possible parameter values and another vector containing the posterior distribution. When we use MCMC and JAGS, the output is different. We will only have a vector of parameter values, without corresponding probabilities. The vector of parameter values is meant to be a random sample of values drawn from the posterior distribution. It's like saying "here are a bunch of guesses for the parameter", and any region where there are a lot of guesses is considered to be a region with high probability.

When we have samples instead of an exhaustive list of parameter values and probabilities, the methods for computing the summaries are different. For a parameter called θ, the methods for computing the summaries are given below.

```
# Posterior mean using samples
post_mean = mean(theta)
# Posterior mode using samples
# post_mode = ??? (this can't be done easily with samples!)
# If you have a really large number of samples,
# visually finding the peak of
# a histogram can work.
# Posterior median using samples
sorted = sort(theta)
post_median = sorted[0.5*length(theta)]
```

6.2 Credible Intervals

Credible intervals are another useful kind of summary. They are used to make statements like "There is a 95% probability the parameter is between 100 and 150". The basic idea is to use the posterior distribution to find an interval $[a, b]$ such that

$$P(a \leq \theta \leq b|x) \quad = \quad \alpha \tag{6.15}$$

where α is some pre-defined probability. 95% seems to be the most popular choice. An example of a 95% credible interval is given in Figure 6.3.

Note that the interval shown in Figure 6.3 is not the only possible interval that would contain 95% of the probability. However, to make the notion of a credible interval precise, we usually use a *central* credible interval. A central credible interval containing an amount of probability α will leave $(1 - \alpha)/2$ of the probability to its left and the same amount $(1 - \alpha)/2$ of the probability to its right.

6.2.1 Computing Credible Intervals from a Bayes' Box

The method for computing credible intervals is closely related to the method for computing the posterior median. With the median, we found the value of θ which has 50% of the posterior probability to its left and 50% to its right. To find the lower end of a 95% credible interval, we find the θ value that has 2.5% of the probability to its left. To find the upper end we find the value of θ that has 2.5% of the posterior probability to its right, or 97.5% to the left.

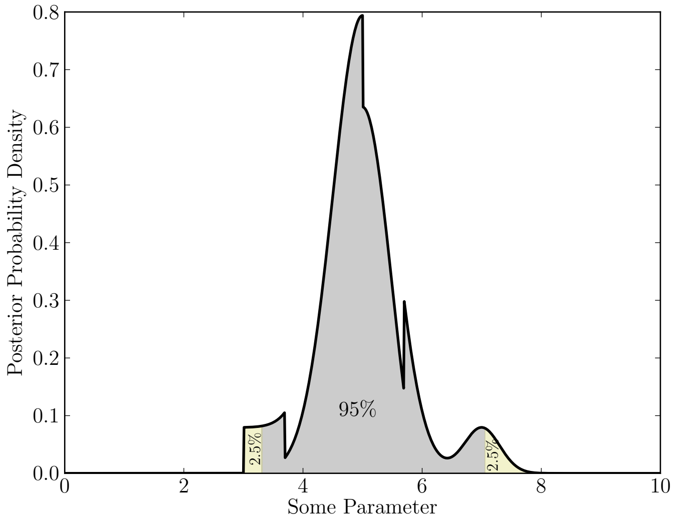

Figure 6.3: *A central 95% credible interval is defined as an interval that contains 95% of the posterior probability, while having 2.5% of the probability above the upper limit and 2.5% of the probability below the lower limit. The credible interval is formed by finding the edges of the grey region. In this case the credible interval is* $[3.310, 7.056]$.

6.2.2 Computing Credible Intervals from Samples

If you have used MCMC and have obtained random samples from the posterior distribution, you can find a credible interval in a similar way to how you would find the posterior median. Again, instead of finding the 0.5 quantile of the posterior distribution you would find the 0.025 quantile and the 0.975 quantile (if you wanted a central 95% credible interval).

6.3 Confidence Intervals

In previous stats courses you have probably come across the concept of a *confidence interval*. A confidence interval is a concept in classical statistics that is somewhat similar to a credible interval in Bayesian statistics. When people calculate confidence intervals, they usually want to say they are 95% sure that the parameter is in that interval, given the data. This is what Bayesian credible intervals do, but *it is not what classical confidence intervals do*!

Luckily, a lot of the time, the classical and the Bayesian methods for making intervals will actually give the same interval. But this isn't always the case! In lectures we will study an example (taken from an Ed Jaynes paper from the 70s) where the Bayesian credible interval and the classical confidence interval give completely different results. The result is shown in Figure 6.4. The key thing to note is the classical confidence interval lies entirely in a region where we are certain (from the data) that θ cannot possibly be!

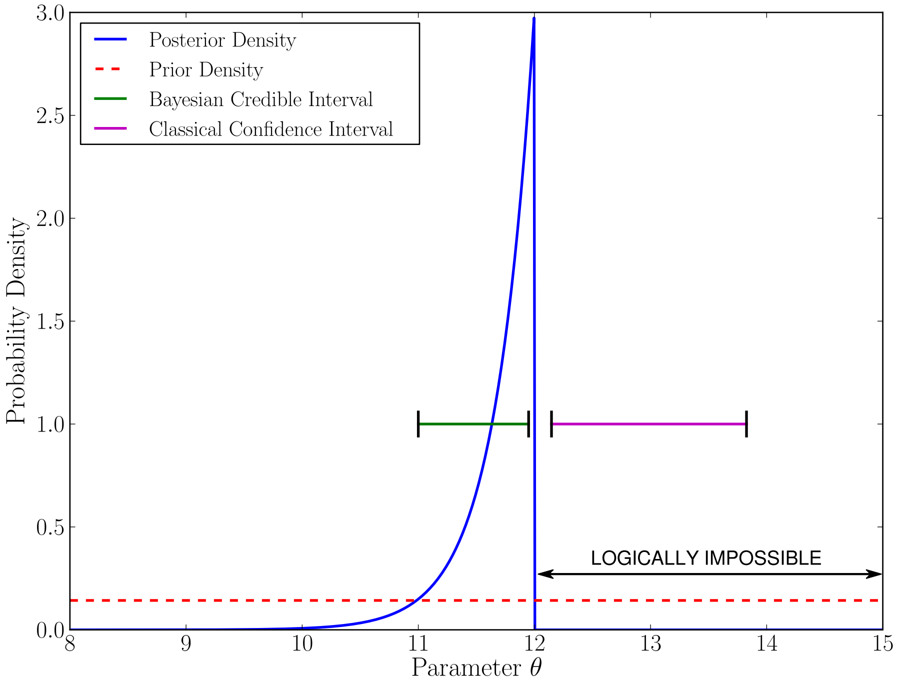

Figure 6.4: *An example of a Bayesian credible interval and a frequentist confidence interval applied to a particular problem where they give different answers. The posterior distribution in blue shows that the parameter θ is probably somewhere between 11 and 12, and values above 12 are completely impossible. However, the entire frequentist confidence interval lies above* $\theta = 12$.

Chapter 7

Hypothesis Testing and Model Selection

Hypothesis testing (also known as model selection, particularly when it is done using the method in Section 7.4) is a very important topic that is traditionally considered a different topic from parameter estimation. However, in Bayesian statistics we will see that hypothesis testing is basically the same thing as parameter estimation! The one difference, for us, will be that we will sometimes change the prior distribution a little bit.

One big advantage of Bayesian statistics is that it *unifies* parameter estimation and hypothesis testing[1]. That's good news, because instead of having to understand two different topics, we only have to understand one!

To see why hypothesis testing is fundamentally the same as parameter estimation, you only need to understand how parameter estimation works from a Bayesian point of view, which we have already studied. Parameter estimation is nothing more than testing a bunch of hypotheses about the value of the parameter. For example, $\theta = 1$ vs. $\theta = 2$ vs. $\theta = 3$ and so on. If we have their posterior probabilities, then we've tested them.

7.1 An Example Hypothesis Test

Suppose we were performing a Bayesian parameter estimation analysis using a Bayes' Box. Here is an example Bayes' Box with made up numbers:

possible values θ	prior $p(\theta)$	likelihood $p(x\|\theta)$	prior × likelihood $p(\theta)p(x\|\theta)$	posterior $p(\theta\|x)$
1.5	0.25	0.2	0.05	0.1
2.0	0.25	0.4	0.1	0.2
2.5	0.25	0.6	0.15	0.3
3.0	0.25	0.8	0.2	0.4
Totals	1		0.5	1

[1] "Unifies" is a popular word for physicists. It means that two seemingly different topics are fundamentally the same, or at least closely related.

Suppose we wanted to test the following two hypotheses about the parameter θ. The first hypothesis H_0 is a "null hypothesis", and the second hypothesis, H_1, is an "alternative hypothesis".

$$\begin{aligned} H_0: &\quad \theta = 2 && (7.1)\\ H_1: &\quad \theta \neq 2 && (7.2) \end{aligned}$$

In classical statistics, if you saw a question phrased in this way, you would need to come up with a *test statistic* and then calculate a *p-value*, which tries to say something about whether the value of the test statistic would be considered extreme, under the assumption that H_0 is true. In Bayesian statistics, the only thing we need to do is calculate the posterior probability of H_0 and the posterior probability of H_1. The posterior probability of H_0 is given by:

$$\begin{aligned} P(H_0|x) &= P(\theta = 2|x) && (7.3)\\ &= 0.2 && (7.4) \end{aligned}$$

All we did here was look up the appropriate number in the Bayes' Box! The posterior probability of H_1 is only slightly harder (but still easy) to calculate: H_1 will be true if θ takes any value other than 2. Therefore, the posterior probability of H_1 is

$$\begin{aligned} P(H_1|x) &= P(\theta = 1.5 \vee \theta = 2.5 \vee \theta = 3|x) && (7.5)\\ &= P(\theta = 1.5|x) + P(\theta = 2.5|x) + P(\theta = 3|x) && (7.6)\\ &= 0.1 + 0.3 + 0.4 && (7.7)\\ &= 0.8. && (7.8) \end{aligned}$$

Here we used the fact that everything in a Bayes' Box is mutually exclusive (only one of the hypotheses is true) so we could add the probabilities. Alternatively, you could have just noticed that H_1 is true if H_0 is false. So $P(H_0|x) + P(H_1|x) = 1$, which implies $P(H_1|x) = 1 - P(H_0|x)$.

7.2 The "Testing" Prior

Here we will study a hypothesis testing example that involves a null and an alternative hypothesis. Since the bus example has been used a lot, we will now switch over to a different example.

Suppose it is known that the mean systolic blood pressure in the general population is 120 mm Hg, with a standard deviation of 15 mm Hg (millimetres of mercury is an old fashioned unit for pressure, even though it sounds like a unit of length). A new drug is developed that may be helpful in reducing blood pressure. A sample of $N = 100$ people (who can be considered representative of the general population) are given the drug, and their systolic blood pressure is measured. This results in 100 blood pressure measurements $\{x_1, x_2, ..., x_N\}$, which will be our data. As a shorthand, I'll sometimes write $\boldsymbol{x}$ (a bold vector) to denote the data collectively, instead of $\{x_1, x_2, ..., x_N\}$.

We are interested in whether the drug works. Let μ be the mean systolic blood pressure that would apply in the general population if everyone was taking the drug. Our goal is to infer the value of μ from the data. In classical statistics, this is sometimes phrased as a hypothesis test between the two competing hypotheses. We will not be concerned with the possibility that the drug has the opposite effect to what is intended.

$$\begin{array}{ll} H_0: & \mu = 120 \text{ (the drug does nothing)} \\ H_1: & \mu < 120 \text{ (the drug reduces blood pressure)} \end{array} \tag{7.9}$$

Suppose the mean of all the data values was

$$\bar{x} = \frac{1}{N}\sum_{i=1}^{100} x_i \tag{7.10}$$

$$= 115.9. \tag{7.11}$$

Does this data provide evidence against H_0 and in favour of H_1? In classical statistics this question would be addressed using a *p-value*. The p-value would be the probability of getting a result this extreme or a result more extreme than what is observed, assuming that the "null hypothesis" is true. That is,

$$\text{p-value} = P(\bar{x} \leq 115.9|H_0). \tag{7.12}$$

In case you're curious, the p-value in this case is 0.0031, which is usually taken to mean that there is fairly strong evidence against H_0 and in favour of H_1. To calculate the p-value, I assumed that the probability distribution for the data values $\{x_1, x_2, ..., x_{100}\}$ was a normal distribution with a known standard deviation of $\sigma = 15$, and that they were independent:

$$x_i \sim \mathcal{N}(\mu, \sigma^2). \tag{7.13}$$

In Bayesian statistics, p-values are not used. Instead, we should think of this as a parameter estimation problem. We can state a set of hypotheses about the value of μ, and then choose a prior distribution, update to a posterior distribution, etc. Then our result will be the posterior probability of the null hypothesis, $P(H_0|\boldsymbol{x}) = P(\mu = 120|\boldsymbol{x})$. This is helpful because the posterior probability of the null hypothesis is exactly what we want. It is a description of how plausible the null hypothesis is given the data. It is not some other probability that isn't really relevant. We can also get the posterior probability of H_1 by summing the posterior probabilities for all other values of μ apart from 120, or by using $P(H_1|\boldsymbol{x}) = 1 - P(H_0|\boldsymbol{x})$.

There is only one minor tweak we need to make to make Bayesian inference an appropriate framework for solving this problem. When the null and alternative hypotheses are written like we wrote them above, it implies that the value of μ that we are calling the "null hypothesis" is a *special value that is especially plausible*. To take this into account in our Bayesian analysis we need to make sure the prior distribution recognises there is a special value of the parameter that we think is extra plausible. When we do this, we will call it a *testing prior*. An example of a testing prior and the resulting Bayes' Box for the blood pressure problem is given in Table 7.1. The R code for calculating these results is given below.

```r
# Parameter values
mu = seq(110, 120)
```

```
# Make the testing prior
prior = rep(0.5/10, 11)
prior[11] = 0.5

# Compute the likelihood for the 100 data points.
# The numbers get close to 0, so let's use logs
log_lik = rep(0, 11)

# Use a for loop to loop over all data values
# and multiply the likelihoods
for(i in 1:100)
{
  log_lik = log_lik + dnorm(x[i], mean=mu, sd=15, log=TRUE)
}

# Rescale the likelihood for readability
lik = exp(log_lik - max(log_lik))*1000
#lik = lik/max(lik)*1000

# Calculate the posterior
h = prior*lik
post = h/sum(h)

# The null hypothesis
post[11]
```

possible values μ	prior $p(\mu)$	likelihood $p(\boldsymbol{x}\|\mu)$	prior × likelihood $p(\mu)p(\boldsymbol{x}\|\mu)$	posterior $p(\mu\|\boldsymbol{x})$
110	0.05	0.44	0.02	0.0001
111	0.05	4.83	0.24	0.0012
112	0.05	34.12	1.71	0.0086
113	0.05	154.64	7.73	0.0389
114	0.05	449.33	22.47	0.1129
115	0.05	837.13	41.86	0.2103
116	0.05	1000.00	50.00	0.2512
117	0.05	756.93	38.30	0.1924
118	0.05	376.15	18.81	0.0945
119	0.05	118.44	5.92	0.0298
120	**0.5**	**23.91**	**11.96**	**0.0601**
Totals	1		199.01	1

Table 7.1: *An example of a testing prior for the blood pressure problem. We give more prior probability to the special value $\mu = 120$ because it is particularly plausible. For readability I have rescaled the likelihoods so that the maximum is 1000. Note that the posterior probability of H_0 can simply be read off the table.*

The conclusion of our Bayesian hypothesis test is that the posterior probability of H_0 is 0.0601. Recall that the classical p-value was 0.0031. These numbers are very different, and *there is no reason why they should be similar*. The p-value might imply that the evidence is overwhelming (if you are not experienced at interpreting p-values), but the posterior probability still says there's a 6% chance the drug does nothing.

Note that the calculation of the posterior distribution uses *all* of the data values, rather than reducing the whole data set down to a single number (the sample mean $\bar{x}$). In this particular example, reducing the whole dataset to a single number is harmless[2]. But in different situations (e.g. if your sampling distribution or likelihood was based on the heavy-tailed Cauchy distribution instead of a normal distribution), reducing an entire data set to a single "test statistic" can be extremely wasteful!

Note also that there were some fairly arbitrary decisions made in choosing our testing prior. We decided not to allow $\mu > 120$, but the analysis could also have allowed for that. The discrete approximation was fairly coarse. Finally, we assumed μ couldn't be lower than 110, and had a uniform prior for all μ values apart from 120. Some of these assumptions can and should be questioned when applying Bayesian hypothesis testing in practice. In Figure 7.1, there are three possible ideas for what the prior should be in the blood pressure question. They may all seem somewhat reasonable in this situation, but could lead to different conclusions.

Prior 1 is basically the same as the prior in our Bayes' Box, although it goes down to $\mu = 100$ and divides the possible μ values more finely. This prior says the null has a 50% probability, and if μ is not equal to 120, then it could be anything. Prior 2 is similar, but has only 30% of the prior probability on the null hypothesis, instead of 50%, and the shape of the prior is non-uniform for the lower values of μ. This is like saying "μ could be precisely 120, and if it's not precisely 120, then it is probably at least *close* to 120". In a lot of hypothesis testing situations this would be a more accurate description of our prior beliefs than Prior 1. Prior 3 isn't really a testing prior at all (it doesn't have a spike), but is just a bell-shaped prior. This is like saying "Alright, I would never believe μ is *exactly* 120, but I think there's a reasonable chance it's *close* to 120. Often, it would be nonsense to think the null hypothesis is perfectly true, to an arbitrary level of accuracy. Something like Prior 3 would be more appropriate. These three priors would all give different results, and the appropriate choice depends on the individual problem you are solving.

Remember, if the conclusions depend sensitively on the choice of the prior distribution, that is an important finding. You either need to be really careful about choosing your prior, or you need more data.

7.3 Some Terminology

There is some alternative terminology that is widely used and is particularly popular in Bayesian hypothesis testing (aka model selection) problems. Suppose there were two hypotheses H_1 and H_2, and some data x. Now, H_1 and H_2 might be a null and alternative hypothesis, or they might be two particular values of the parameter, or something else.

[2] In this problem, the sample mean is a "sufficient statistic": deleting all of the data and using just the mean has no consequences!

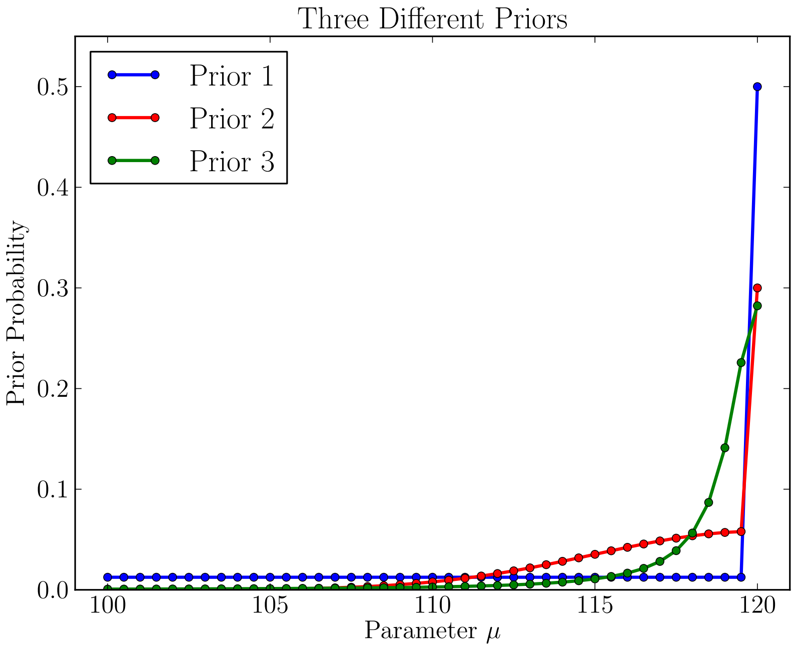

Figure 7.1: *Three possible priors we could use for the blood pressure question. All may seem "reasonable", and the choice can affect the results (quite significantly in some cases). Care should be taken when choosing the prior in hypothesis testing problems.*

Two repetitions of Bayes' rule for these two hypotheses are:

$$P(H_1|x) = \frac{P(H_1)p(x|H_1)}{p(x)} \tag{7.14}$$

$$P(H_2|x) = \frac{P(H_2)p(x|H_2)}{p(x)} \tag{7.15}$$

These could also be written in words:

$$\text{posterior} = \frac{\text{prior} \times \text{likelihood}}{\text{marginal likelihood}} \tag{7.16}$$

Dividing these two equations gives the *odds form* of Bayes' rule, which deals with ratios of probabilities instead of probabilities themselves.

$$\frac{P(H_1|x)}{P(H_2|x)} = \frac{P(H_1)}{P(H_2)} \times \frac{p(x|H_1)}{p(x|H_2)} \tag{7.17}$$

In words, this can be written as:

$$\text{posterior odds} = \text{prior odds} \times \text{bayes factor} \tag{7.18}$$

Sometimes people talk about odds (or odds ratios, which are the same thing) and Bayes Factors instead of about prior and posterior probabilities. The odds tell us how plausible H_1 is compared to H_2. For example, a posterior odds ratio of 5 means H_1 is 5 times as plausible as H_2. Of course, odds can be greater than 1 even though probabilities cannot be. The Bayes Factor is the ratio of the likelihoods. Results are often quoted as Bayes Factors because that's the part of the equation where the data is important. If you say "The Bayes Factor for H_1 over H_2 was 10", then whoever you're talking to is free to apply whatever prior odds they like, whereas if you state the posterior odds then people may wonder what your prior odds were.

7.4 Hypothesis Testing and the Marginal Likelihood

The Bayes' factor in Equation 7.17 is the ratio of likelihoods for two hypotheses H_1 and H_2. If we wanted to calculate the Bayes Factor for H_0 and H_1 in the blood pressure example, we could easily get the likelihood for H_0 (it's right there in the Bayes' Box). But how would we get $p(x|H_1)$, which needs to be a single number? H_1 is the statement $\mu = 100$ **or** $\mu = 111$ **or** $\mu = 112$ and so on up to $\mu = 119$.

Imagine we had left $\mu = 120$ out of our Bayes' Box and just done parameter estimation within the context of H_1 (i.e. assuming, for argument's sake, that H_1 is true). This would involve a *reduced* Bayes' Box with one less row in it. We would end up getting some marginal likelihood $p(x) = \sum p(\theta)p(x|\theta)$. The key is to realise that since we are assuming H_1 throughout, the marginal likelihood is really $p(x|H_1) = \sum p(\theta|H_1)p(x|\theta, H_1)$, which is exactly the thing we need to calculate the Bayes Factor!

All of this implies there are two mathematically equivalent ways of doing Bayesian hypothesis testing, or model selection. One is to make a big model that includes both the null and the alternative hypothesis. The Bayes' Box with a testing prior accomplishes this. In most cases this is the most convenient way to do the calculations.

The other way is to do the two analyses separately. First, do parameter estimation within the context of H_1. Then, do parameter estimation within the context of H_2. Then, use the marginal likelihoods as if they were likelihoods, to compare H_1 vs. H_2. This second way of calculating Bayes Factors is most useful when the two analyses were actually done separately by different people.

Chapter 8

Markov Chain Monte Carlo

8.1 Monte Carlo

Monte Carlo is a general term for computational techniques that use random numbers. Monte Carlo can be used in classical and Bayesian statistics. A special kind of Monte Carlo called Markov Chain Monte Carlo (MCMC) was one of the main reasons for the revival of Bayesian statistics in the second half of the 20th century. Before MCMC became popular, one of the major drawbacks of the Bayesian approach was that some of the calculations were too hard to do. MCMC enables us to solve a wide range of Bayesian problems which cannot be solved using analytical methods.

8.1.1 Summaries

So far, we have represented our probability distributions (prior and posterior) in a computer by using a vector of possible parameter values and a corresponding vector of probabilities. For example, suppose we have a single parameter θ and we have worked out the posterior distribution by using a Bayes' Box. This will give us a vector `theta` of possible θ values and a corresponding vector `post` containing the posterior probabilities. Well, one thing we could do is plot the posterior distribution, resulting in a plot like the one in Figure 8.1.

```
plot(theta, post, xlab="Theta", ylab="Posterior Probability")
```

If we want to obtain some summaries, we could do it like so:

```
post_mean = sum(theta*post)
post_sd = sqrt(sum(theta^2*post) - post_mean^2)
```

However, there is an alternative way of representing this posterior distribution in a computer. It may not be immediately obvious why this is a good idea, because there is nothing wrong with the tried and true method we have used so far. But this second method has the advantage that it continues to work well on much bigger problems, such as when we have more than one parameter. With more than one parameter, the 'vector of possible solutions" approach can fail very dramatically.

Our new way of representing a probability distribution in a computer will be via *Monte*

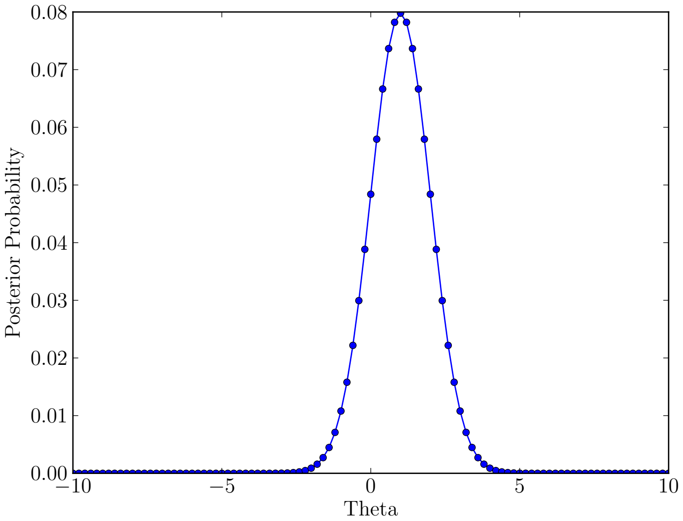

Figure 8.1: *A posterior distribution can be represented in a computer by a discrete set of possible parameter values, and the corresponding probabilities.*

Carlo samples. Instead of having two vectors (one of θ values and one of the corresponding probabilities), imagine we had some method to compute a random sample of θ values, drawn from the posterior distribution in Figure 8.1. There would only be one vector `theta`. So how would we know there is greater probability around $\theta = 1$? Well, *more elements of the* `theta` *vector would be near 1.* Instead of carrying around a second vector of probabilities, we understand that more probable regions will simply contain more points. Say our vector of random samples is called `theta`. Then we can look at the posterior distribution by plotting a histogram of samples:

```
hist(theta, breaks=100)
```

The histogram looks something like the one in Figure 8.2. We can also get our summaries, but the code looks different (it's actually easier than before!):

```
post_mean = mean(theta)
post_sd = sd(theta)
```

Because of the randomness involved in generating the `theta` values, the summaries aren't exact. For example, I know the actual posterior mean and standard deviation in this example were both 1, but the values obtained from the Monte Carlo samples were 0.9604 and 1.0008, respectively. However, this doesn't matter much because the results indicate θ is probably somewhere around 1, with an uncertainty of about 1. The error introduced by using random samples is much smaller than the amount of uncertainty inherent in the posterior distribution itself. For example, if I summarised the posterior distribution by saying $\theta = 0.9604 \pm 1.0008$, for almost all practical purposes the conclusion is exactly the same as the true version of the summaries $\theta = 1 \pm 1$.

In this discussion, we haven't answered the question of how to actually generate random samples of θ from the posterior distribution. This is the job of Markov Chain Monte Carlo.

The purpose of Markov Chain Monte Carlo is to generate random samples of parameter values drawn from the posterior distribution. This makes it

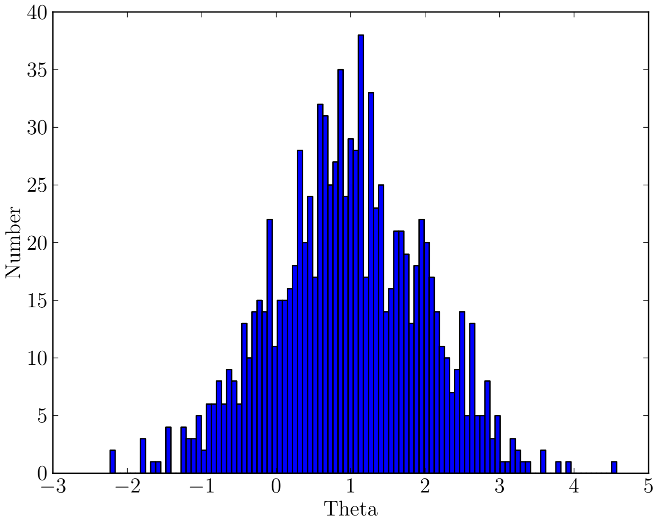

Figure 8.2: *The posterior distribution for a parameter θ can also be represented by a random sample of θ values drawn from the posterior distribution. Some θ values are more probable than others, which is encoded by certain values appearing more frequently in the sample.*

very easy to compute summaries even if you have more than one unknown parameter.

8.2 Multiple Parameters

MCMC becomes extremely useful when we begin to look at Bayesian models involving more than one unknown parameter. Having posterior samples makes the process of *marginalisation* much easier. Imagine we wanted to infer two parameters, called a and b, from data x. Bayes' rule (parameter estimation version) would give us the posterior distribution:

$$p(a,b|x) \propto p(a,b)p(x|a,b) \tag{8.1}$$

However, what if you didn't really care about the value of b but only really wanted to measure a? The terminology for this is that b is a *nuisance parameter*: you need it to define the model, but ultimately you are not really interested in knowing its value. What you need in this case is the *marginal* posterior distribution for a (that is, the posterior distribution for a on its own, not the joint distribution with b). This can be obtained using the sum rule. The result is:

$$p(a|x) = \int_b p(a,b|x)\,db \tag{8.2}$$

or

$$p(a|x) = \sum_b p(a,b|x) \tag{8.3}$$

depending on whether the set of possible b values is continuous or discrete. Before MCMC, these integrals or sums usually couldn't be done without making certain choices purely for mathematical convenience (e.g. choosing the prior to be a certain kind of distribution only because it will make the maths work out, rather than it being a good description of your prior beliefs).

Samples of parameter values drawn from the posterior distribution (achieved using MCMC) make this hard problem much easier. We no longer need to worry about mathematical convenience. See Figure 8.3 for an example showing how Monte Carlo sampling makes marginalisation trivial.

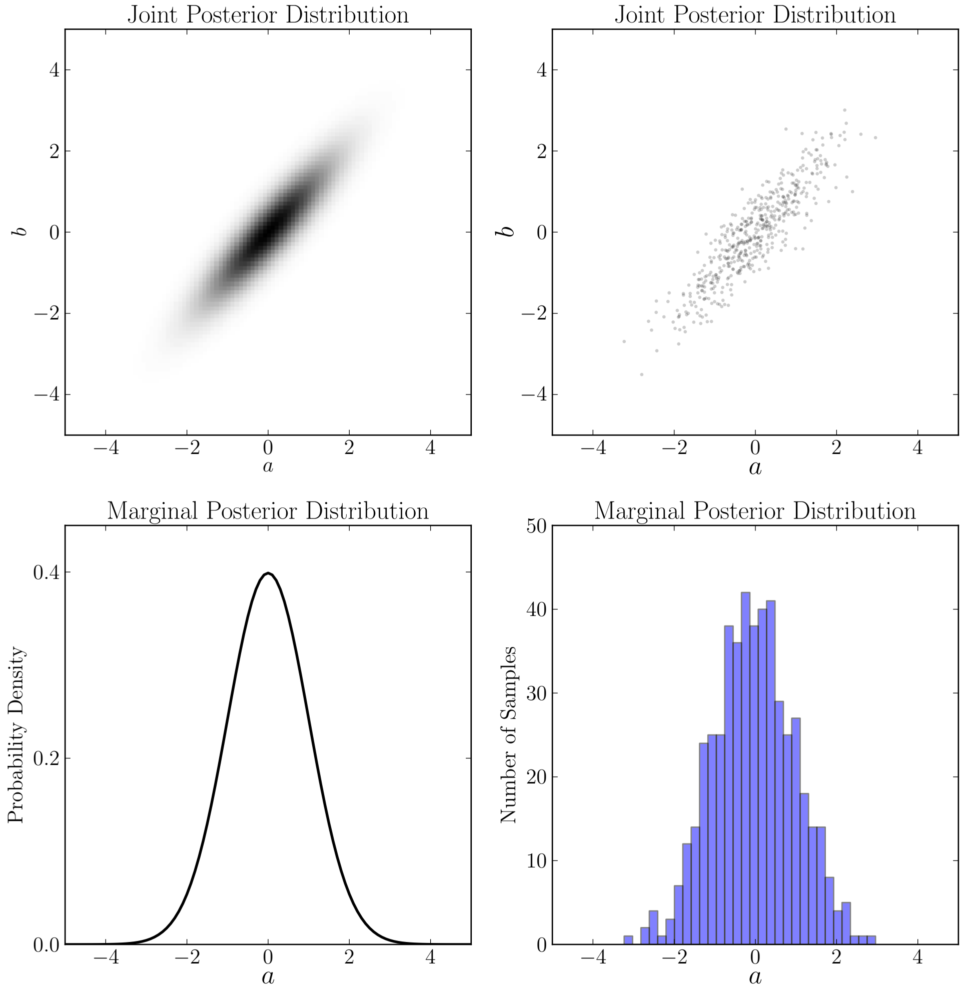

Figure 8.3: *An example of a posterior distribution for two parameters, a and b. The left panels show the joint posterior distribution (which has a correlation) and the marginal posterior distribution for a, obtained by integrating over all the possible b values. The top right panel contains random samples (points) drawn from the posterior distribution for a and b. The only step needed to get the marginal distribution for a (lower right panel) is to ignore the b-values of the points!*

8.3 The Metropolis Algorithm

The Metropolis algorithm is the most basic MCMC method. The ideas behind it are fairly simple, yet Metropolis forms the basis of a large number of more advanced MCMC methods. In STATS 331 we will study the basic ideas behind how the Metropolis algorithm works, which will involve a small amount of Markov chain theory. We will also look at a small amount of R code which implements the Metropolis algorithm, but for solving practical problems it is more convenient to use the JAGS program[1].

The Metropolis algorithm was invented in the 1950s by physicists (including Nicholas Metropolis, for whom the algorithm is named), who used it to do calculations in the field of statistical mechanics. This intriguing field focuses on calculating the macroscopic (large scale) properties of matter from knowledge of the small-scale properties. For example, if you know water is composed of H_2O molecules, you can use statistical mechanics to figure out what will happen if you have a lot of water molecules. For example, it will freeze at 0 degrees Celsius and boil at 100 degrees Celsius.

It took many decades before people started to realise the Metropolis algorithm was useful in Bayesian statistics as well as statistical mechanics. The Bayesian approach seemed very elegant and useful to many people, but it could always be criticised as unworkable, because you usually had to do difficult or impossible integrals when solving practical problems (to summarise the posterior, or to get rid of nuisance parameters). MCMC changed all that, and is one of the reasons for the explosion in the popularity of Bayesian statistics beginning in the 1990s.

The basic idea of MCMC is that we want a method which will travel between different possible states (such as the possible hypotheses/parameter values in a Bayesian analysis). We want the amount of time spent in any particular state to be proportional to the posterior probability of the state. The computer "explores" the set of possible parameter values, spending a lot of time in the regions with high posterior probability, and only rarely visiting regions of low posterior probability. Figure 8.4 shows an example of MCMC applied to a problem with only two possible hypotheses or parameter values. Nobody would actually use MCMC on such a small problem, but it is helpful for explaining how MCMC works.

8.3.1 Metropolis, Stated

The Metropolis algorithm is given below. The first thing to do is start *somewhere* in the "parameter space" (set of possible parameter values). You then *propose* to move somewhere else. There is an acceptance probability α that determines whether to accept the proposal. If the proposal is better (h, the prior times likelihood value is higher), then you accept it, and the proposed state becomes the new state of the algorithm. If the proposal is worse, you can also accept it, but the probability of accepting is given by the ratio of the unnormalised posterior probabilities, i.e. h'/h. For example, if the proposed point is 1/3 as good as the current point, the acceptance probability is 1/3. If the proposed point is rejected, the original point remains the state of the algorithm, and gets counted again in

[1] JAGS uses a number of MCMC methods internally, including Metropolis, "Gibbs Sampling", and "Slice Sampling", which we will not study in this course.

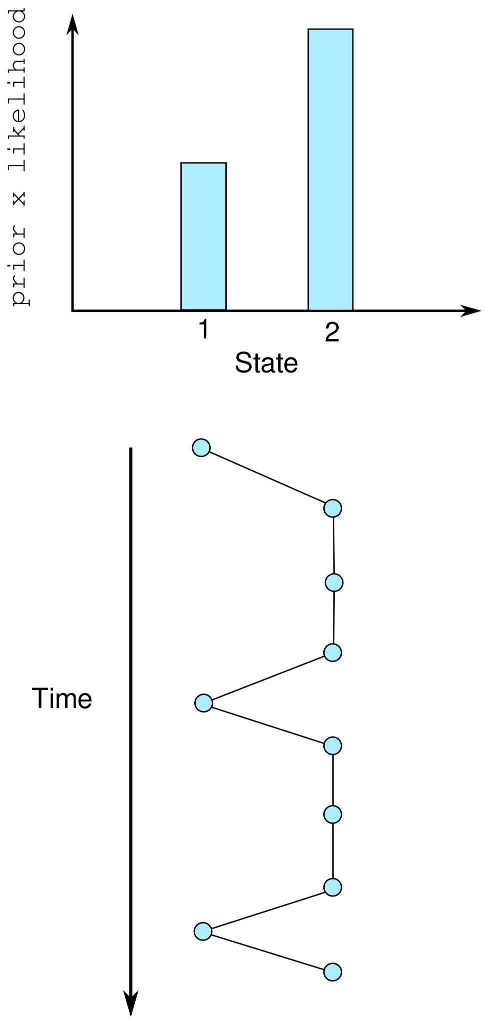

Figure 8.4: *An illustration of the basic idea behind MCMC. Imagine we had a Bayes' Box with two possible hypotheses, and we knew the* `prior` × `likelihood` *values. The amount of time the MCMC program will spend in each state is proportional to the posterior probability of the state. In this example, the MCMC algorithm was in State 1 three times and in State 2 seven times. Using this, we could estimate the posterior probability of State 2 as being 0.7. This estimate would become more accurate if we ran the MCMC for more iterations.*

the results.

The Metropolis algorithm works because it makes transitions towards low probability states rare, while transitions towards high probability states are common, because of the acceptance probability. It is hard to move into an improbable state, so not much time will be spent there. Over time, the fraction of time spent in any given state is equal to the posterior probability of the corresponding hypothesis.

- Start in some state θ
- Generate a "proposal" state θ' from a *proposal distribution* q (assumed symmetric so that $q(\theta'|\theta) = q(\theta|\theta')$)

- With probability $\alpha = \min(1, h'/h)$, replace the current state with the proposed state
- Repeat

8.4 A Two State Problem

We will now study how the Metropolis algorithm works on a very simple example, namely the two-ball problem from the beginning of the notes. There are two hypotheses and the posterior probabilities are 1/3 and 2/3. It is important to note that MCMC is not actually needed for a problem this simple, but it is a good test case to see precisely how an MCMC algorithm works. When we solve real data analysis problems with JAGS, we won't have to think too much about how the MCMC works, but can concentrate instead on the Bayesian statistics problem at hand.

Let's call the less probable hypothesis "State 1" and the more probable hypothesis "State 2" for the purposes of this section. What we need is a Markov process that will spend 1/3 of the time in State 1 and 2/3 of the time in State 2. The Metropolis algorithm described above will do what we need. The main thing we need to compute is the acceptance probability α_{ij} for a proposed transition from state i to state j where $i, j \in \{1, 2\}$. The acceptance probability α for a proposed move from state i to state j is given by

$$\alpha_{ij} = \min\left(1, \frac{h_j}{h_i}\right) \tag{8.4}$$

where h_i and h_j are proportional to the posterior probabilities of states i and j respectively. If the proposal is to move to an equal or better (higher posterior probability) state ($h_j \geq h_i$) then the acceptance probability is 1. If the proposal is to move to a less probable state then the acceptance probability is h_j/h_i, the ratio of the two posterior probabilities[2].

The transition probability is the probability of being in state j at the next iteration given that you are in state i at the current iteration. The transition probability is given by the product rule:

$$p_{ij} = q_j \alpha_{ij} \tag{8.5}$$

for $i \neq j$. The transition matrix of the Markov chain is a matrix with all the different p_{ij} values in it:

$$\mathbf{P} = \begin{bmatrix} p_{11} & p_{12} \\ p_{21} & p_{22} \end{bmatrix} \tag{8.6}$$

In our particular case, we can work out the off-diagonal elements of $\mathbf{P}$ using Equation 8.5:

$$\mathbf{P} = \begin{bmatrix} & \frac{1}{2} \times 1 \\ \frac{1}{2} \times \frac{1}{2} & \end{bmatrix} \tag{8.7}$$

[2] Note that this algorithm can be used even if the marginal likelihood is unknown, because only ratios of posterior probabilities are needed. This is useful because the marginal likelihood is sometimes very hard to calculate in multi-parameter problems.

The diagonal elements can be found by knowing the rows of $\mathbf{P}$ must sum to 1. We must be in *some* state at the next iteration. Therefore the transition matrix of our Markov chain in this two-state problem is:

$$\mathbf{P} = \begin{bmatrix} \frac{1}{2} & \frac{1}{2} \\ \frac{1}{4} & \frac{3}{4} \end{bmatrix} \tag{8.8}$$

A Markov chain with a small number of possible states can be represented graphically using a *transition diagram*, as in Figure 8.5.

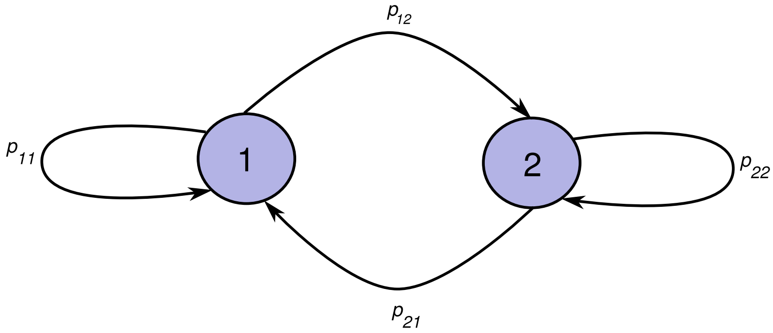

Figure 8.5: *A transition diagram for a Markov Chain with two possible states. The states (which correspond to hypotheses in Bayesian inference) are drawn as circles labelled "1" and "2". When the algorithm is in a particular state (e.g. State 1), and we apply one step of the Metropolis algorithm, the probability it will be in State 1 is p_{11} and the probability it will be in State 2 is p_{12}. MCMC works by making it easy to move into states with high posterior probability, and hard to move out of them.*

8.5 The Steady-State Distribution of a Markov Chain

Once we have the transition matrix, we can work out the *steady state distribution* of the Markov chain. Imagine, instead of starting the MCMC from an arbitrary initial state, you have a probability distribution for the initial state. After applying one iteration of the Metropolis algorithm, the probability distribution for the updated state will usually be different from the probability distribution for the initial state.

However, one special probability distribution, called the *steady state distribution*, does not change after you apply an MCMC update. If your initial point was drawn from the steady state distribution, then subsequent points will also be drawn from the steady state distribution. In MCMC the steady state distribution should be the same as the posterior distribution.

Imagine we are using MCMC on our two-state problem, and our initial position is State 1 with probability v_1 and State 2 with probability v_2. What is the probability of being in State 1 at the next iteration? There are two ways for that to happen: by starting in State 1 and then making a transition from 1 → 1, or by starting in State 2 and making a transition from 2 → 1. The total probability is then:

$$P(\text{State 1 after iteration}) = v_1 p_{11} + v_2 p_{21}. \tag{8.9}$$

Similarly for State 2:

$$P(\text{State 2 after iteration}) = v_1p_{12} + v_2p_{22}. \tag{8.10}$$

If v_1 and v_2 happened to be the steady state distribution, then these probabilities would also be v_1 and v_2. This gives us two simultaneous equations

$$\begin{aligned} v_1 &= v_1p_{11} + v_2p_{21} && (8.11) \\ v_2 &= v_1p_{12} + v_2p_{22} && (8.12) \end{aligned}$$

which can also be written in matrix form as $\mathbf{vP} = \mathbf{v}$ where $\mathbf{v} = (v_1, v_2)$ and $\mathbf{P}$ is the transition matrix[3]. These two simultaneous equations can be solved for v_1 and v_2. However there is a third constraint, that is $v_1 + v_2 = 1$. Since there are three equations but two unknowns, it seems like the problem might be "over-determined", but that is not actually the case because the transition matrix rows are not all linearly independent. It is left as an exercise for the reader to show the steady state distribution for our Markov chain (Equation 8.8) is in fact equal to the posterior distribution, so $\mathbf{v} = \left(\frac{1}{3}, \frac{2}{3}\right)$.

8.6 Tactile MCMC

In class we will do "Tactile MCMC", which is an implementation of the Metropolis algorithm using coins and dice instead of the random number generators provided in computer software such as R. This is a good way to get a feel for the flow of MCMC algorithms.

In STATS 331, when we want to use MCMC in practice, we will use the JAGS program rather than using Metropolis directly. JAGS is a general purpose MCMC program which allows you to solve fairly complex problems without a large amount of programming.

[3] Mathematics students might recognise this equation, which says $\mathbf{v}$ is the left-eigenvector of $\mathbf{P}$, with eigenvalue 1. An alternative is to write the stationary distribution as a column vector and use $\mathbf{P}^T\mathbf{v} = \mathbf{v}$.

Chapter 9

Using JAGS

JAGS stands for "Just Another Gibbs Sampler". JAGS is a computer program that allows the user to implement Markov Chain Monte Carlo (MCMC) on fairly complicated problems very quickly. The "Gibbs Sampling" mentioned in the name of JAGS is a particular MCMC technique that is beyond the scope of this course: however, it is not so different from the Metropolis-Hastings method that we do study in this course. If you are faced with a statistics problem which you would like to solve in a Bayesian way, JAGS makes it very straightforward to implement a Bayesian model. Essentially, you just need to tell JAGS the following things:

- The names of your unknown parameter(s), and their prior distributions
- The likelihood

Then we simply load in the data and let it go! JAGS will run MCMC, automatically choosing an appropriate starting point based on your prior, and moving the parameter values around so the posterior distribution is well sampled. In this chapter we will see how to use JAGS with a simple example, and we will see some more features of JAGS when we come to study more complex examples. There are some more advanced features of JAGS that we will not use in this course.

JAGS is not the first program of its kind, and it is related to many other available programs. Starting in the late 1980s, a program called BUGS (Bayesian inference Using Gibbs Sampling) was developed, and this evolved into the program WinBUGS. These programs had a huge effect on the uptake of Bayesian statistics, by dramatically reducing the amount of work it takes to get an MCMC simulation to run. Instead of having to code up your own implementation of the Metropolis algorithm or an alternative MCMC method, all you had to do was tell WinBUGS what your prior and likelihood were, and it would automatically use appropriate and sophisticated MCMC methods.

Up until 2012, STATS 331 was taught using WinBUGS. However, there are a number of disadvantages to WinBUGS, so I decided to switch over to JAGS in 2013. The main advantages of JAGS over WinBUGS are: i) it is open source software and works on all operating systems, ii) it allows a more concise version of notation that can save a lot of space and make the code easier to read and write, and iii) WinBUGS is not actively developed or maintained any more. In addition, a lot of time in previous iterations of 331 was spent teaching different ways of using WinBUGS (i.e. calling it from R, vs. using the graphical interface, vs. writing a script). In 2013 we will use JAGS in just one way (by calling it from R), which frees up time for us to concentrate on the stats! There is another up to date BUGS program called OpenBUGS, but it too is only a Windows program. Most of what you learn about JAGS can be transferred to WinBUGS or OpenBUGS quite easily, if necessary, with only minor changes.

9.1 Basic JAGS Example

Since we have used it a lot already, it makes sense to look at how the bus problem looks in JAGS. Recall we had a single unknown parameter θ, with a uniform prior between 0 and 1. We also had a binomial likelihood, which we could write as $x \sim \text{Binomial}(N, \theta)$. To implement this model in JAGS, the code looks like this:

```
model
{
    # Parameters and the priors
    theta ~ dunif(0, 1)

    # Likelihood
    x ~ dbin(theta, N)
}
```

As in R, comments (statements that have no effect but help to annotate the code) can be written with a # symbol. The names of the distributions in JAGS are very similar to (but not always exactly the same as) the names of the R functions for evaluating the probability densities or mass functions. In this example, `dunif` is the uniform distribution and `dbin` is the binomial distribution. One final thing to note is the order of the parameters in the binomial distribution. In JAGS the success probability comes first and the number of trials comes second. There are some quirks with the other distributions as well, such as the normal distribution, which we will see in later chapters.

The code for implementing a Bayesian model in JAGS belongs inside a `model{ }` block. In our example, the first statement inside the model is `theta ~ dunif(0, 1)`. As you can probably guess, `theta` is simply the name of our parameter. We are free to name it as we wish, and in different situations we will give the parameters different names. The tilde sign is like the "$\sim$" notation for probability distributions. We are about to specify a probability distribution that applies to `theta`. Finally, the actual distribution is given, which is a uniform distribution between 0 and 1. Note the command used for a uniform distribution is `dunif` and not `uniform`. *Our line of code* `theta ~ dunif(0, 1)` *tells JAGS there is a parameter called* `theta` *and we want to use a uniform prior between 0 and 1.*

The notation for the likelihood is very similar. We write the name of the data followed by "~" and then the distribution, in this case the binomial distribution. One annoying this about the binomial distribution is the ordering of the parameters. Usually people write the number of trials first and the success probability second, but in JAGS it's the other way around.

Interestingly, the likelihood part of the code looks exactly like the prior part. So how does

JAGS know that `x` is data and not just another parameter? Well, when we call JAGS from R, we will pass to it an R list containing the data. There will be a value for `x` in this list, which tells JAGS that it is a fixed and known quantity, and not another unknown parameter like `theta`.

Above, we specified the JAGS model, but this isn't the only thing we need. We also need a way to actually run JAGS! The most convenient way to use JAGS is to call it from R, using the R library `rjags`. This way, the output (samples from the posterior distribution for the parameters) is available in R for postprocessing such as plots and smmaries. The `rjags` library has many features and options, and it can be a bit overwhelming to figure out all of them using the documentation. Therefore, I have written a template R script called `use_jags.R` where you can specify the data, the JAGS model, and some options at the top of the file, and you do not have to worry about all the functions for calling JAGS from R.

The first part of `use_jags.R` is given below. This is the part you can modify to load different data, change the model assumptions, and decide how long (for how many iterations or steps) you would like the MCMC to run for.

The *burn-in* is an initial part of the MCMC run where results are not saved. This is beneficial because sometimes it can take a while for the MCMC to locate the regions of high posterior probability, and if you include the initial parts of the run in you results, you can get incorrect answers. For most of our models in STATS 331, we do not need a long burn-in period.

```
model = "model
{
  theta ~ dunif(0, 1)
  x ~ dbin(theta, N)
}
"

# The data (use NA for no data)
data = list(x=2, N=5)

# Variables to monitor
variable_names = c('theta')

# How many burn-in steps?
burn_in = 1000

# How many proper steps?
steps = 10000

# Thinning?
thin = 1
```

The second part of `use_jags.R` actually runs JAGS. You won't need to edit this or know much about it, but for completeness, here it is:

```
# NO NEED TO EDIT PAST HERE!!!
# Just run it all and use the results list.

library('rjags')

# Write model out to file
fileConn=file("model.temp")
writeLines(model, fileConn)
close(fileConn)

if(all(is.na(data)))
{
    m = jags.model(file="model.temp")
} else
{
    m = jags.model(file="model.temp", data=data)
}
update(m, burn_in)
draw = jags.samples(m, steps, thin=thin, variable.names = variable_names)
# Convert to a list
make_list <- function(draw)
{
  results = list()
  for(name in names(draw))
  {
    # Extract "chain 1"
    results[[name]] = as.array(draw[[name]][,,1])

    # Transpose 2D arrays
    if(length(dim(results[[name]])) == 2)
      results[[name]] = t(results[[name]])
  }
  return(results)
}
results = make_list(draw)
```

When this code is executed, it creates an R list called `results`. Inside `results`, there is a vector for each variable that you chose to “monitor” by listing its name in the `variable_names` vector. In this example there is only one parameter, so it seems obvious we would like to monitor it. In more complex situations there may be many parameters, and only some of them are actually interesting, the others are “nuisance parameters”. The `variable_names` vector allows you to choose just the parameters you really care about. Notice also the various options such as the number of steps, and the `thin` option. If `thin` is set to 10, for example, only every 10th iteration of the MCMC will appear in the results. This is useful for keeping the size of the `results` list manageable, even if you run the MCMC for a very long time.

One of the most important things to check after running JAGS is a *trace plot* of the parameters. A trace plot is a plot of the value of the parameter over time, as the MCMC was running. To plot a trace plot of the MCMC run, we can simply use the following code, for a parameter called `theta`. If the parameter has a different name, replace `theta` with the actual name of the parameter.

```
plot(results$theta, type='l')
```

You could look at the posterior distribution using a histogram, and you can compute summaries using the methods discussed in Chapter 6. The code for the histogram for a parameter `theta` is given below.

```
hist(results$theta, breaks=100)
```

Examples of a trace plot and a histogram are given in Figure 9.1. Trace plots are the best diagnostic tool for seeing whether MCMC is working properly. Ideally, trace plots should look like "noise", without strong correlations between one point and the next. If there are strong correlations in the trace plot, the MCMC will need to be run for a longer period of time to obtain effectively independent samples from the posterior distribution.

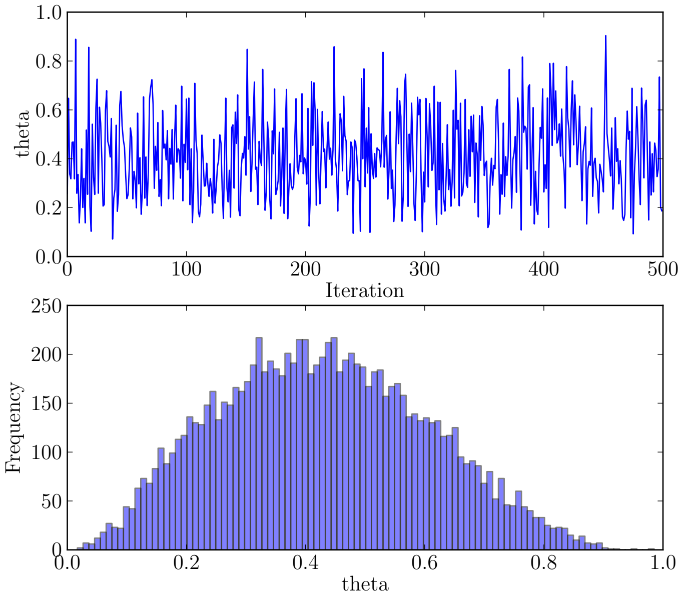

Figure 9.1: *The trace plot and histogram of the MCMC samples returned by JAGS. The trace plot (top) is zoomed in on the first 500 samples, and shows the* `theta` *value moving around as the MCMC proceeds. The histogram of* `theta` *values, sampled from the posterior distribution, is given in the lower panel. You can compare this with Figure 5.1.*

9.2 Checklist for Using JAGS

When running JAGS using the `use_jags.R` script provided, there are several things you need to ensure. These are listed below.

- The data you want to analyse must be contained in an R list called `data`. Inside this list you should also include variables such as the size of the data set (which I usually call `N`), or any other known constants that are referred to in the model.

- The JAGS model must be correctly written inside the R string called `model`. In the likelihood part of the JAGS model, you must ensure that the names of the data variables match the names in your `data` list. In the example of this chapter, the number of successes is called `x` in both the data list and in the JAGS model.
- The `variable_names` vector, which lists the parameters you are interested in, can only list parameters that actually exist in the JAGS model. If you try to monitor a variable that isn't in your model, you will get an error message.

Chapter 10

Regression

Regression is a very important topic in statistics that is applied extremely frequently. There are many different kinds of regression, but in STATS 331 we will mostly focus on linear regression. This gives us a nice familiar example example to demonstrate how Bayesian statistics works and how it is different from classical or frequentist statistics. Here we will study an example of a simple linear regression problem taken from STATS 20X.

10.1 A Simple Linear Regression Problem

Data were collected from a sample of 30 drivers. The age of the driver and the maximum distance at which they could read a newly designed road sign were recorded. It is of interest to build a simple model that can be used to predict the maximum distance at which the sign is legible, using the age of the driver. Figure 10.1 shows the data. The purpose of simple linear regression is to find a straight line that goes throught the data points. The slope and intercept of the straight line are then helpful for understanding the relationship between the variables. Also, the straight line can be used to predict future data, such as the maximum distance at which a 90-year-old person could read the sign. The most common method used to obtain the straight line is to find the line (i.e. the slope and intercept values) which fits best by the criterion of "least squares".

10.2 Interpretation as a Bayesian Question

From what you now know about Bayesian statistics, you might be able to come up with some reasons why the standard least squares solution is unsatisfactory. One glaring issue is that the data are hardly ever going to be good enough to tell us with certainty that a particular slope and intercept are correct (the exception would be if three or more points lie perfectly on a straight line, with no scatter). In principle, we will almost always have uncertainty about the slope and the intercept. From a Bayesian perspective, our goal is not to find a point estimate for the slope and the intercept. Instead we should calculate the *posterior distribution for the slope and the intercept, given the data.* The posterior

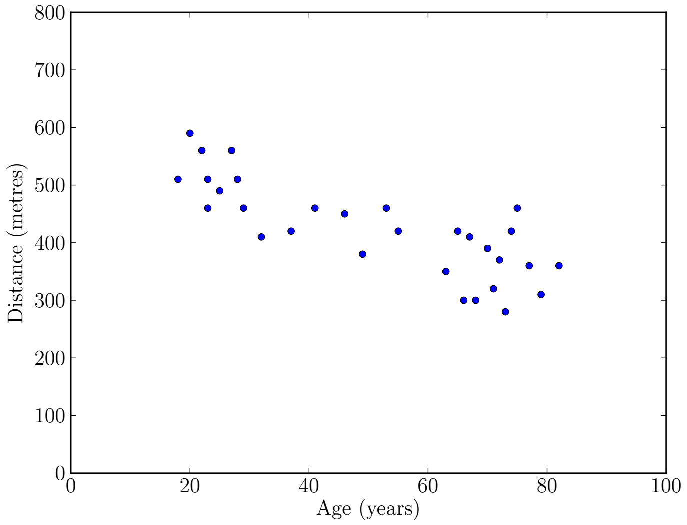

Figure 10.1: *The maximum distance at which a person can read a road sign vs. the age of the person. There are $N = 30$ data points. You can clearly see that older people have, on average, worse eyesight. Simple linear regression can be thought of as "fitting a straight line" to the data.*

distribution will tell us exactly how much uncertainty we have. If we do want summaries for convenience, we can use the posterior distribution to create the summaries, as discussed in Chapter 6.

The equation for a straight line is usually written as $y = mx + b$ where m is the gradient/slope and b is the intercept. However, for consistency with later, more complex regression models, we will write the equation as:

$$y = \beta_0 + \beta_1 x. \tag{10.1}$$

Here, β_0 is the y-intercept and β_1 is the slope. Our goal is to calculate the posterior distribution for β_0 and β_1 given the data.

10.3 Analytical Solution With Known Variance

Bayes' rule (parameter estimation version) tells us how to calculate the posterior distribution:

$$p(\theta|x) \propto p(\theta)p(x|\theta) \tag{10.2}$$

This is the generic form for parameters θ and data x. In our particular case, the unknown parameters are β_0 and β_1, and the data are the y values of the data points. The data also consist of a number N of points and the x-values, but we shall assume that these on their own provide no information about the slope and intercept (it would be a bit strange if they did). So the x-values and the number of points N act like prior information that lurks "in the background" of this entire analysis. The y-values are our data in the sense that we will obtain our likelihood by writing down a probability distribution for the y-values given the parameters.

Therefore, Bayes' rule *for this problem* (i.e. with the actual names of our parameters and data, rather than generic names) reads:

$$p(\beta_0, \beta_1 | y_1, y_2, ..., y_N) \propto p(\beta_0, \beta_1) p(y_1, y_2, ..., y_N | \beta_0, \beta_1) \tag{10.3}$$

We can now say some things about Bayesian linear regression by working analytically. For starters, let's assume uniform priors for both β_0 and β_1, and that the prior for these two parameters are independent. The probability density for a uniform prior distribution can be written simply as:

$$p(\beta_0, \beta_1) \propto 1. \tag{10.4}$$

Note that we have written proportional instead of equals. If we decided to place the limits at -500 and 500 (say) then the actual value of the density would be 10^{-6}. But this is just a number and in the end, when we normalise the posterior distribution, it won't matter. We can even imagine making our prior "infinitely wide", which is called an improper prior. In many cases simply writing $p(\beta_0, \beta_1) \propto 1$ will not cause any problems. We are assuming the prior probability density is uniform over a very wide range which we will not specify.

Now, on to the likelihood. There are N data points and so there are N y-values in the dataset, called $\{y_1, y_2, ..., y_N\}$. We can obtain the likelihood by writing down a probability distribution for the data given the parameters, sometimes called a "sampling distribution". This describes our beliefs about the connection between the data and the parameters, without which it would be impossible to learn anything from data. If we knew the true values of β_0 and β_1, then we would predict the y-values to be scattered around the straight line. Specifically we will assume that each point departs from the straight line by an amount ϵ_i which has a $\mathcal{N}(0, \sigma^2)$ probability distribution. For now, we will assume σ, the standard deviation of the scatter, is known. In "$\sim$" notation, this can be written as:

$$y_i \sim \mathcal{N}(\beta_0 + \beta_1 x_i, \sigma^2). \tag{10.5}$$

It is implied that all of the data values are independent (given the parameters). Therefore the likelihood can be written as a product of N normal densities, one for each data point:

$$p(\{y_1, y_2, ..., y_N\} | \beta_0, \beta_1) \quad = \quad \prod_{i=1}^{N} \frac{1}{\sigma\sqrt{2\pi}} \exp\left[-\frac{1}{2\sigma^2}\left(y_i - (\beta_0 + \beta_1 x_i)\right)^2\right]. \tag{10.6}$$

Remember, when we combine the likelihood with the prior using Bayes' rule, we can usually ignore any constant factors which do not depend on the parameters. This allows us to ignore the first part of the product, outside the exponential (since we are assuming σ is known).

$$\begin{aligned} p(\beta_0, \beta_1 | y_1, y_2, ..., y_N) &\propto p(\beta_0, \beta_1) p(y_1, y_2, ..., y_N | \beta_0, \beta_1) && (10.7) \\ &\propto 1 \times \prod_{i=1}^{N} \exp\left[-\frac{1}{2\sigma^2}\left(y_i - (\beta_0 + \beta_1 x_i)\right)^2\right] && (10.8) \\ &\propto \exp\left[-\frac{1}{2\sigma^2}\sum_{i=1}^{N}\left(y_i - (\beta_0 + \beta_1 x_i)\right)^2\right]. && (10.9) \end{aligned}$$

We have just found the expression for the posterior distribution for β_0 and β_1. This is a distribution for two parameters (i.e. it is bivariate). It is not easy to interpret this equation just by looking at it, but we could use it to work out the value of the posterior probability density for any possible values of β_0 and β_1

There are a few things you may notice about the posterior distribution in Equation 10.9. Firstly, the way it depends on the parameters is an exponential of something involving β_0 and β_1 in linear and second-order ways (if you were to expand the square, you would get terms like $\beta_0\beta_1$ and β_0^2). Mathematicians would call the expression inside the exponential a *quadratic form.* When a probability density can be written as the exponential of a quadratic form, it is a normal density. Therefore, the posterior distribution for β_0 and β_1 is a (bivariate) normal distribution.

We can obtain some more insight about this problem by inspecting the sum term inside the exponential in the posterior distribution (Equation 10.9). The sum is over all the data points, and what is being summed is the difference between the data value y_i and the straight line prediction $\beta_0 + \beta_1 x_i$, all squared. The sum term is just the sum of squared residuals that is minimised when solving this problem by "least squares". In classical least squares linear regression, β_0 and β_1 are estimated by minimising this sum of squared residuals. Because of the exp and the minus sign, the posterior distribution is telling us that the choice of β_0 and β_1 that minimises the sum of squared residuals, maximises the posterior probability density. Other values of the parameters that don't quite minimise the sum of squared residuals are somewhat plausible, and the form of the posterior density tells us exactly how much less plausible they are. The take home message is summarised below.

Doing a linear regression by least squares is equivalent to having a uniform prior and a normal likelihood, and finding the posterior mode. If you think this is appropriate in a particular application, and you are happy with just a point estimate, then classical least squares fitting will be fine. Otherwise, you'd better do a Bayesian analysis.

While classical regression results may come with "standard errors", these are not the same as a posterior distribution. A posterior distribution describes the uncertainty about the parameters given the specific data set you actually have. Standard errors describe how different your point estimate would be if your data set was different.

10.4 Solution With JAGS

The above analytical results made the unrealistic assumption that the standard deviation σ, of the scatter, was known. In practice, σ usually needs to be estimated from the data as well. Therefore, in the Bayesian framework, we should include it as an extra unknown parameter. Now we have three unknown parameters instead of two. Our parameters are now β_0, β_1, and σ. One major advantage of MCMC is that we can increase the number of unknown parameters without having to worry about the fact that the posterior distribution might be hard to interpret or plot.

The data is the same as before, $\{y_1, y_2, ..., y_N\}$. The likelihood is also the same as before:

$$y_i \sim \mathcal{N}(\beta_0 + \beta_1 x_i, \sigma^2). \tag{10.10}$$

Our three parameters will need priors. JAGS requires proper priors (i.e. we can't have a uniform prior over an infinite range), but we can still make our priors very wide. Instead of using uniform distributions this time, we will use normal distributions with a mean of 0 and a large standard deviation of 1000.

For the standard deviation parameter σ, we know firstly that this cannot be negative. We will use a "log-uniform" prior with generous lower and upper limits, to express uncertainty about the order of magnitude of σ. This prior implies things like $P(1 < \sigma < 10) = P(10 < \sigma < 100) = P(100 < \sigma < 1000)$, which is sometimes a good description of a large amount of uncertainty about a positive parameter. The easiest way to implement this in JAGS is to actually use $\log(\sigma)$ as the parameter, with a uniform prior, and then define $\sigma = \exp\left(\log(\sigma)\right)$. Notice that "deterministic nodes" (quantities that are defined in terms of other variables) in JAGS are defined using "`<-`" instead of "`~`".

In JAGS, the model looks like this:

```
model
{
  # Prior for all the parameters
  beta0 ~ dnorm(0, 1/1000^2)
  beta1 ~ dnorm(0, 1/1000^2)
  log_sigma ~ dunif(-10, 10)
  sigma <- exp(log_sigma)

  # Likelihood
  for(i in 1:N)
  {
    y[i] ~ dnorm(beta0 + beta1*x[i], 1/sigma^2)
  }
}
```

The first part defines the priors for the parameters. For β_0 and β_1, we have just chosen very broad priors that describe vague prior knowledge. Note that the standard deviations of the priors are 1000, so we should be careful to only apply this code in situtations where we don't expect the intercept or slope to have an extreme value (either positive or negative).

For the standard deviation parameter σ, which describes how much we expect the data points to be scattered around the straight line, we have assigned a log-uniform prior. The limits of -10 and 10 for `log_sigma` imply limits of 4.5×10^{-5} to 22,000 for `sigma`, a generous range. If we thought σ might actually be outside this range, we should change the prior to something else or risk getting strange answers.

The likelihood part of the code involves a `for` loop, because our data is more than just a single number. The code inside the loop is effectively duplicated N times (with $i = 1$, then with $i = 2$, etc), once for each data point. Since the loop refers to a quantity `N`, this must be specified in the `data` list if you are using my `use_jags.r` template code.

Another new feature of this JAGS model is the normal distribution, which is called `dnorm` in JAGS. Usually a normal distribution is written $\mathcal{N}(\mu, \sigma^2)$ where μ is the mean and σ is the standard deviation (and σ^2 is the variance). Unfortunately, in JAGS there is a quirk: the first argument to `dnorm` is indeed the mean, but the second argument must be one over the variance, or one over the standard deviation squared.

10.5 Results for “Road” Data

Our JAGS output will contain samples from the posterior distribution for β_0, β_1 and σ. The first thing we should do is make trace plots and check that everything converged properly. Then we can make histograms of each parameter to visually inspect the (marginal) posterior distribution for each parameter. We can also plot one parameter vs. another to look at the *joint* posterior distribution for the parameters. R code for all of these is given below.

```r
# Plot trace plots
plot(results$beta0, type='l', xlab='Iteration', ylab='beta0')
plot(results$beta1, type='l', xlab='Iteration', ylab='beta1')
plot(results$sigma, type='l', xlab='Iteration', ylab='sigma')

# Plot histograms
hist(results$beta0, breaks=20, xlab='beta0')
hist(results$beta1, breaks=20, xlab='beta1')
hist(results$sigma, breaks=20, xlab='sigma')

# Plot joint posterior distribution of beta0 and beta1
plot(results$beta0, results$beta1, cex=0.1, xlab='beta0', ylab='beta1')
```

All of the plots for the road data are shown in Figure 10.2.

With classical linear regression it is usually helpful to plot the best fitting line through the data. In Bayesian linear regression our output is posterior samples for what the parameters might be (and therefore what the line might be). A common way of displaying the posterior is to plot many lines through the data, with the lines produced using the posterior samples. Some R code for doing this is given below. The plot produced looks like the one in Figure 10.3.

```r
# Plot the data
plot(data$x, data$y)

# Make some x-values for plotting lines
x = c(0, 100)
# Plot the first 30 lines from the posterior distribution
for(i in 1:30)
{
    lines(x, results$beta0[i] + results$beta1[i]*x)
}
```

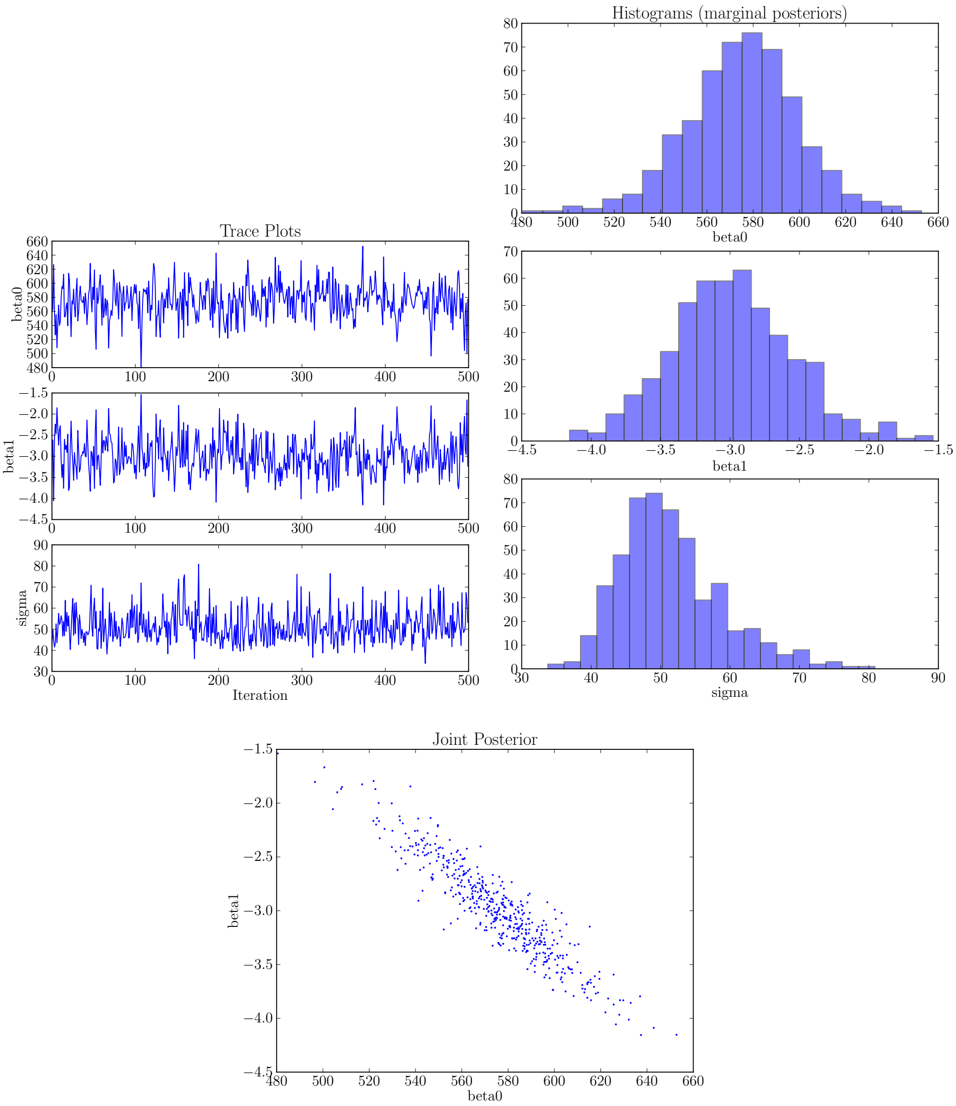

Figure 10.2: *Results (posterior samples) from the simple linear regression model applied to the road data.* **Top Left***: Trace plots showing the parameters moving around over time (as the MCMC progressed).* **Top Right***: Histograms of the posterior samples, showing the marginal posterior distributions for the parameters.* **Bottom***: A plot of β_1 vs. β_0, showing samples from the joint posterior for these two parameters. These parameters had independent priors, but the posterior shows a correlation. All this means is that if β_0 is a high value then β_1 must be low, and vice versa.*

10.6 Predicting New Data

One of the most important uses of regression models is for prediction. Given this data, what can we say about the value of the output variable y at some new value of the input variable, x_{new}? It's unknown, but let's call it y_{new}. With our road example, we will try to predict the maximum reading distance of a person who is $x_{\text{new}} = 90$ years old. If we knew the parameters of the straight line (or had a good point estimate), we could extend it out to $x = 90$, and compute our predicted value:

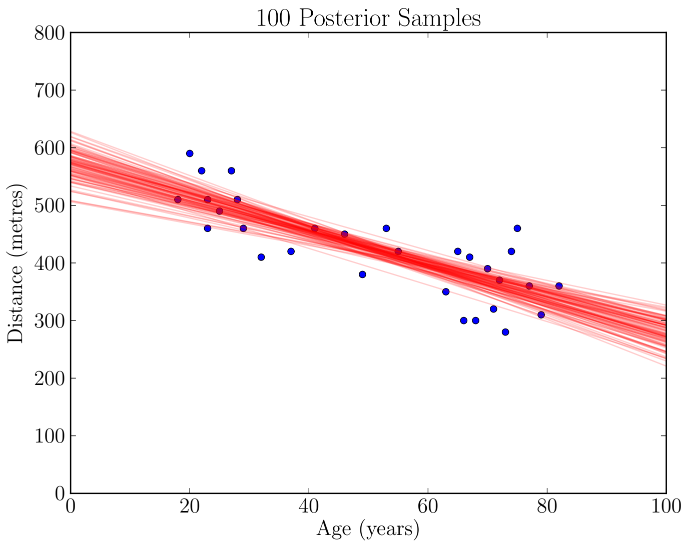

Figure 10.3: *The road data with credible regression lines (sampled from the posterior distribution) overplotted.*

$$
\begin{aligned}
y_{\text{new}} &= \beta_0 + \beta_1 x_{\text{new}} && (10.11)\\
&= \beta_0 + \beta_1 \times 90 && (10.12)
\end{aligned}
$$

but that's not quite in the Bayesian spirit. Firstly, we don't know the value of the parameters, we only have the posterior distribution. Secondly, we would like not just a point estimate for y_{new} but a whole probability distribution, describing all of the uncertainty in the prediction. One simple improvement would be to state that our uncertainty about y_{new} should be described by a normal distribution around the straight line, with standard deviation σ. This still isn't quite right though, because we'd need to know the true values of the parameters to actually do it.

In general, Bayesian prediction works like so. With parameters θ and data x, we can predict "new data" x' by calculating the "posterior predictive distribution" which is just the probability distribution for x' given x:

$$
\begin{aligned}
p(x'|x) &= \int p(\theta, x'|x)\, d\theta && (10.13)\\
&= \int p(\theta|x)p(x'|x,\theta)\, d\theta && (10.14)
\end{aligned}
$$

We first saw this in Section 4.2. The first term inside the integral is the posterior, and therefore the whole integral is an expectation value, with respect to the posterior distribution. The second term is the probability distribution for x' given the parameters, i.e. imagining that we knew the parameters.

This equation is telling us to imagine that we know the true parameter values, make a prediction (in terms of a probability distribution), repeat this for all possible parameter values and then average together all the probability distributions into one "final" distribution. In the averaging process we should give more weight to the probability distributions that were based on plausible values of θ.

Thankfully, actually doing this is much easier than it sounds, thanks to MCMC. Remarkably, we can accomplish all this, and obtain our probability distribution for y_{new}, by adding just a single line to the JAGS model:

```
y_new ~ dnorm(beta0 + beta1*90, 1/sigma^2)
```

Of course, to look at the posterior samples for `y_new`, you'll need to monitor it. The samples of `y_new` will be drawn from the posterior predictive distribution for the new data. Internally, JAGS will simulate a new value of `y_new` at every iteration, from the specified distribution, and using its current estimates of the parameters. Since the parameters are not fixed, but explore the posterior distribution, the distribution of `y_new` values will take into account all of the uncertainty we have about the parameters values.

As a general rule for predicting new data in JAGS, the extra line(s) you'll add to the JAGS model will usually resemble the likelihood, but the variable will have a different name. The results for the road data prediction are shown in Figure 10.4.

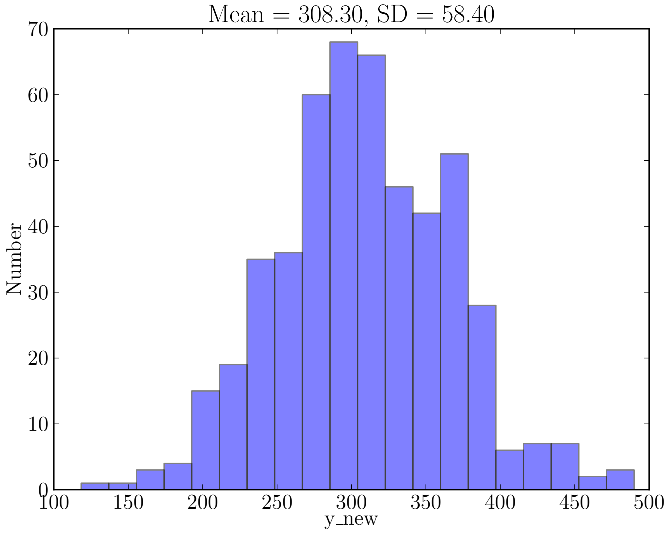

Figure 10.4: *Samples from the posterior predictive distribution, answering the question "what do we know about* y *at* $x = 90$*?". Summaries are shown in the title. If we simply assumed the best fit line was true and applied a point estimate of* σ *to get our uncertainty, we would have obtained a prediction of* 306.07 ± 49.76.

10.7 Simple Linear Regression With Outliers

One complication that is common in linear regression (or other model-fitting) problems is the existence of outliers. These are points that do not fit in with the general trend assumed by the model. Many methods exist for deciding how to "detect" and "remove" outliers. From a Bayesian point of view, there is not a one-size-fits-all approach to outliers. If you really believe in your model assumptions, an "outlier" might be your most important data point. If you aren't really sure of your model assumptions and are just using a convenient default model, then your results may be misleading. In lectures and labs we will study an extension to the simple linear regression model that allows for outliers by using a heavier-tailed sampling distribution, the Student-t distribution.

10.8 Multiple Linear Regression and Logistic Regression

We will study an example of multiple linear regression, and nonlinear regression (involving an "interaction term") in lectures. We will not study a logistic regression explicitly, but the special lecture on predicting sports matches is very closely related to logistic regression. The main difference between linear regression is that the output variable can only be 0 or 1, so the likelihood will usually involve the "Bernoulli" distribution (a Binomial distribution with one trial).

Chapter 11

Replacements for t-tests and ANOVA

ANOVA is a common procedure in classical statistics, and is related to the simpler idea of a t-test. These classical tests were designed for particular kinds of problems, and in this chapter we will study similar problems but solve them from a Bayesian point of view. We will also use these examples to discuss some issues about the choice of prior distributions when there are more than a few parameters. When there are only a few parameters it is usually safe to assign a vague, wide prior to describe your initial uncertainty (unless, of course, you have more information than that). In higher dimensions, problems can arise if you do this. One way of getting around these problems is to use a *hierarchical model.*

11.1 A T-Test Example

This example is based on one given in a 1976 article by physicist E. T. Jaynes, called "Confidence Intervals vs. Bayesian Intervals". This is a very strongly worded paper and might be an interesting read for those who are interested in the battle between frequentist and Bayesian statistics when the latter was making its comeback in the second half of the 20th century. It's also where I got the crazy confidence interval example from.

Two manufacturers, 1 and 2, both make "widgets", and we are interested in figuring out which manufacturer makes the best widgets (on average), as measured by their lifetime. To determine this, we obtain 9 widgets from manufacturer 1 and 4 widgets from manufacturer 2, and measure their lifetimes, in days. The results are given below:

$$x^1 = \{41.26, 35.81, 36.01, 43.59, 37.50, 52.70, 42.43, 32.52, 56.20\} \quad (11.1)$$

$$x^2 = \{54.97, 47.07, 57.12, 40.84\} \quad (11.2)$$

These measurements can be summarised by the means and standard deviations, which are 42 ± 7.48 for group 1 and 50 ± 6.48 for group 2. The question is: given this data, is there evidence that one of the manufacturers is better than the other, and if so, by how much? In classical statistics the standard procedure for this situation would be a two sample t0-test. However, before we do anything I'd like you to consider the numbers and use your intuition: what do *you* think about what the evidence says?

An underlying assumption of a classical t-test is that the data are normally distributed around the mean values for each group[1]. We may as well adopt this assumption for our Bayesian model. If we call the group 1 data points $\{x_1^1, x_2^1, ..., x_{N_1}^1\}$ and the group 2 data points $\{x_1^2, x_2^2, ..., x_{N_1}^2\}$, then the likelihood is:

$$\begin{aligned} x_i^1 &\sim \mathcal{N}\left(\mu_1, \sigma^2\right) \\ x_i^2 &\sim \mathcal{N}\left(\mu_2, \sigma^2\right) \end{aligned} \tag{11.3}$$

Where all the data points are independent given the parameters. Note the assumption that the two groups have the same underlying ("population") standard deviation σ. This is a popular assumption in this kind of analysis but it is not necessarily well justified! We will build our Bayesian models using this assumption, but it is not that difficult to relax it if you want to. You could just include multiple σ parameters in the model, just like how we will include the multiple μ parameters.

Instead of just one model for this situation, we will study three different versions. Each model will have the same likelihood as given above in Equation 11.3, and the same prior for σ. However, the models will all have different priors for μ_1 and μ_2. We will be able to see that the choice of prior does influence the results (of course), but in ways that make sense. Which of these models is more appropriate in a practical situation would depend on the exact situation. There is no "one size fits all" model.

11.1.1 Likelihood

To implement our model in JAGS, we can begin by specifying the likelihood part like so:

```
# Sampling distribution/likelihood
for(i in 1:N1)
{
  x1[i] ~ dnorm(mu1, 1/sigma^2)
}
for(i in 1:N2)
{
  x2[i] ~ dnorm(mu2, 1/sigma^2)
}
```

We have called our data arrays `x1` and `x2`, and we have also assumed that the sample sizes `N1` and `N2` are defined, so our `data` list will need to be consistent with these choices. The parameters we will be estimating are `mu1`, `mu2`, and `sigma`, so we will need to specify prior distributions for them. In the following sections, we'll use the same prior for `sigma`, so we may as well specify that now. Let's use a log-uniform prior where σ is between e^{-10} and e^{10}.

```
# Prior for sigma
log_sigma ~ dunif(-10, 10)
sigma <- exp(log_sigma)
```

[1] Strictly speaking, it's the probability distribution for the data given the parameters that is normal, the data may or may not look normally distributed.

11.1.2 Prior 1: Very Vague

The last missing ingredients to finish the JAGS model are the priors for mu1 and mu2. For our first model, let's be really naive and assign super-wide uniform priors.

```
# Prior 1: Very Vague
mu1 ~ dnorm(0, 1/1000^2)
mu2 ~ dnorm(0, 1/1000^2)
```

At first glance, this might seem like a fairly reasonable thing to do. In many problems, it doesn't make much difference if we just use vague priors and get on with the calculation (as opposed to thinking really hard about the prior, and what is actually known about the parameters).

However, this prior has a number of properties that suggest it might not be quite right: firstly, what is the probability that $\mu_1 = \mu_2$? In classical t-tests, the whole point is to test the hypothesis that the two "population means" (parameters) are equal. However, our prior actually implies that the probability they are equal is 0! Therefore, no matter what data we get, the posterior probability of $\mu_1 = \mu_2$ will always be zero.

11.1.3 Prior 2: They might be equal!

The problem with Prior 1 is that we may think μ_1 might exactly equal μ_2, and Prior 1 doesn't allow for this. So here's another way we might set up the prior. We'll start by defining the prior for μ_1 as we did before. Then, when we consider μ_2, we need a way of giving it a 50% probability of equalling μ_1, and if not, then it should have a "bi-exponential" distribution centered around μ_1. Here is our solution. Read it carefully and make sure you understand what this prior does.

```
# First mean
mu1 ~ dnorm(0, 1/1000^2)

# Prior for difference, mu2 - mu1
u ~ dunif(-1, 1)

# Length of exponential prior given difference != 0
L <- 5
size_of_difference <- step(u)*(-L*log(1 - u))

# To make the difference positive or negative
C ~ dbin(0.5, 1)
difference <- (2*C - 1)*size_of_difference

# Second mean
mu2 <- mu1 + difference
```

11.1.4 Prior 3: Alright, they're not equal, but they might be *close*

Prior 2 is also a little bit strange, if you think about it. If we're comparing these two manufacturers of widgets, why would we think it is possible that the two manufacturers are `exactly` equal? Maybe we just think the parameters μ_1 and μ_2 are likely to be *similar* in value. In other words, we shouldn't worry so much about the prior probablity of $\mu_1 = \mu_2$, but we should at least make sure there's a moderate prior probability that $\mu_1 \approx \mu_2$.

One way we could do this is by applying a normal prior to both μ_1 and μ_2 with some mean (let's call it the "grand mean") and some standard deviation (let's call it the "diversity"). That way, μ_1 and μ_2 would both be likely to be somewhere around the grand mean, and they would likely be different by roughly the size of the diversity. The challenge now seems to be the choice of appropriate values for the grand mean and the diversity. Fortunately, we don't actually have to! What we can do instead is apply priors for them instead.

This is our first example of a *hierarchical model*. In a hierarchical model, instead of directly assigning priors to our parameters, we imagine that we knew the values of some other parameters (called "hyperparameters"), and assign our prior for the parameters *given* the hyperparameters. Then we assign a prior for they hyperparameters as well, to complete the model.

```
# Hierarchical prior for the means
# Hyperparameters
grand_mean ~ dnorm(0, 1/1000^2)
log_diversity ~ dunif(-10, 10)
diversity <- exp(log_diversity)

# Prior for the parameters given the hyperparameters
mu1 ~ dnorm(grand_mean, 1/diversity^2)
mu2 ~ dnorm(grand_mean, 1/diversity^2)
```

Samples (obtained using JAGS) of the three priors are shown in Figure 11.1.

The posteriors are shown in Figure 11.2. The inferences are different, as you would expect, and that's entirely down to the choice of the prior. Any summaries we make will therefore depend on which prior we want to use.

The original question was whether manufacturer two was better, equal, or worse than manufacturer one. We can answer that question by calculating the posterior probabilities of $\mu_1 = \mu_2$, $\mu_1 < \mu_2$, and $\mu_1 > \mu_2$. The results are shown in Table 11.1.

Remember that Prior 1 did not assign any probability to the possibility of the two parameters being equal. Therefore, no possible evidence can increase make the posterior probability nonzero. However, according to this model, there is quite strong evidence that $\mu_1 < \mu_2$, as the probability changed from 0.5 to 0.946.

Prior 2 did allow the two parameters to be equal, and if we use Prior 2, we seem to have found very weak evidence that they are not in fact equal. The probability decreased from 0.5 to 0.424. According to Prior 2, if $\mu_1 \neq \mu_2$, then $\mu_1 < \mu_2$ is the next most likely scenario. However, Prior 2 has an issue associated with it. Our prior says that if μ_2 is not equal to

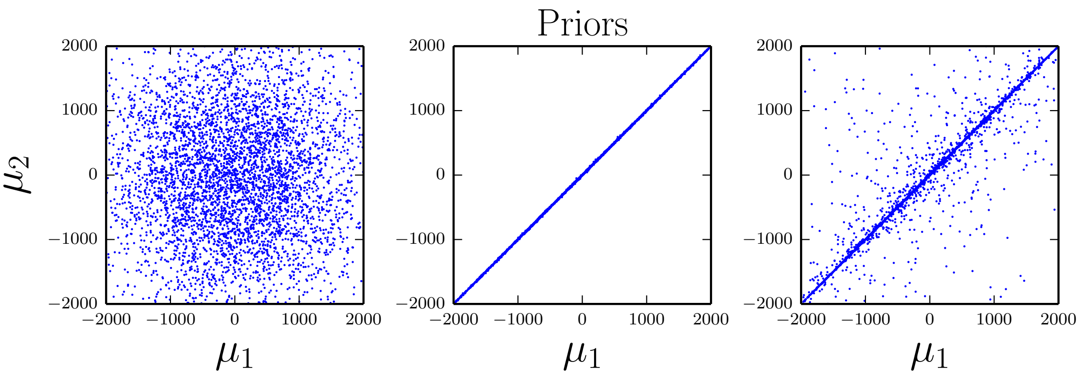

Figure 11.1: *The three different priors we are trying for our Bayesian equivalent of a t-test. The first prior simply asserts a large amount of prior ignorance about the value of the two parameters μ_1 and μ_2. The second is similar but applies 50% probability to the proposition $\mu_1 = \mu_2$. The third prior does not allow the two parameters to be exactly equal, but enhances the probability that they are quite similar in value.*

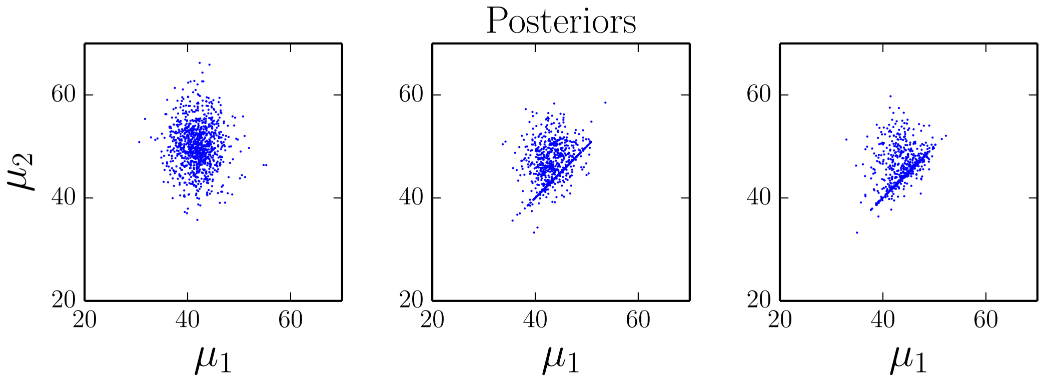

Figure 11.2: *The posterior distributions, given the widget data, based on the three different priors for μ_1 and μ_2.*

μ_1, then it is likely to be close to μ_1. Exactly how close we expect it to be is set by the variable `L` in the model. If we were to make `L` very large, then the data would go from weak evidence against $\mu_1 = \mu_2$ to strong evidence for it! Why does this happen? Well, if we increased `L`, the prior probability that μ_2 and μ_1 are close given that they're different is decreased. Then, the hypothesis that the μs are different does not predict our data as well, since our data looks like the μs are close together. Since it doesn't predict the data as well as before, its posterior probability will be lower. Some people think this sensitivity to the prior is a danger of Bayesian inference (if you want, you can do a web search for the "Jeffreys-Lindley paradox"), but it is behaving logically: the wider we make the prior, the lower we make the prior probability that μ_1 and μ_2 are close but not equal, giving the model no choice but to believe that they're equal. If the results are sensitive to the prior, that's important, and you should think about the logic of the problem to understand why.

Prior 3 seems like it's what we might want in general. It's often silly to think two parameters might be *exactly* equal. What we really think is that there is a difference, and it might be very small, or moderate or large.

Prior Probabilities:

Prior	$\mu_1 < \mu_2$	$\mu_1 = \mu_2$	$\mu_1 > \mu_2$
1	0.5	0	0.5
2	0.25	0.5	0.25
3	0.5	0	0.5

Posterior Probabilities:

Prior	$\mu_1 < \mu_2$	$\mu_1 = \mu_2$	$\mu_1 > \mu_2$
1	0.945	0	0.055
2	0.491	0.430	0.079
3	0.629	0	0.372

Table 11.1: *Prior and posterior probabilities for three different hypotheses about the two manufacturers, based on the models with the three different priors. As you can see, the conclusions are quite sensitive to the choice of prior in this case.*

11.2 One Way Anova

One-way ANOVA can be considered as a generalisation of a t-test to more than two groups. The question is usually phrased as a test of the hypothesis that the group means are the same, versus the alternative that there is some difference. As we saw in the Bayesian "t-test", it is possible (using clever tricks) to make a model that has some prior probability that the group means are equal. However, this gets more tricky with multiple groups. Therefore we will build our one-way ANOVA model in a similar way to the "hierarchical model" version of the t-test model. There will be one other major difference, but it is a difference in the way the model is coded, not a conceptual difference.

In the t-test section our data set was composed of measurements in two groups and our data list contained two vectors of measurements, called `x1` and `x2`. The sampling distribution/likelihood part of our JAGS model also needed two `for` loops, one for each group. If we have many groups (in the following example we will have four), it can get awkward having to write all those loops. Therefore, when we develop our "one-way ANOVA" model, we will format the data differently by putting all measurements into a single vector `x`. To make this work, we'll need an extra vector in the dataset, which tells us which group each data point belongs to.

We'll use an example dataset on the masses of starlings (a type of bird). The masses of some starlings were measured at four locations. We are interested in the differences between the locations. How similar are they in terms of the average weight of starlings? Are they basically the same, radically different, or something in between? A boxplot of the data is shown in Figure 11.3, which seems to show substantial differences between the locations. However, only ten starlings were measured at each location, so we can't be absolutely sure of this, and our goal is to investigate how sure we should be.

To solve this problem in a Bayesian way, we will treat it as a parameter estimation problem with four μ parameters, one for each of the locations. We will also need at least one parameter describing the standard deviation of the starling masses at each location. For convenience we'll assume that's the same across all locations, but it is straightforward to relax this assumption later.

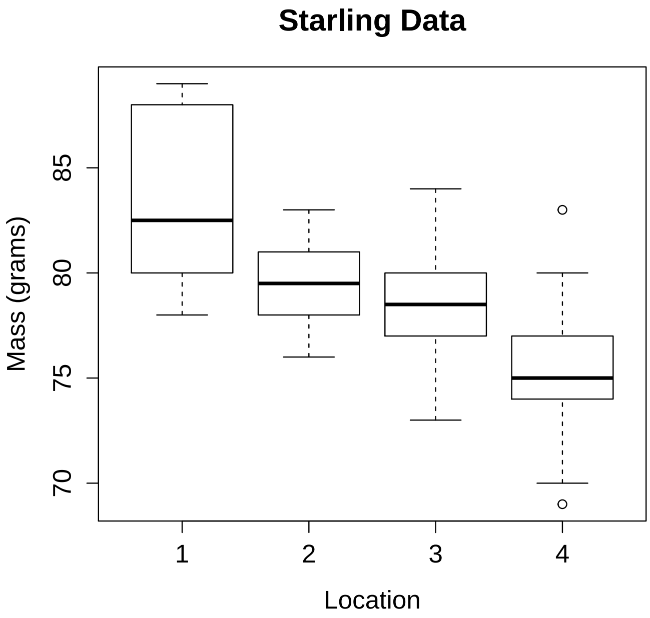

Figure 11.3: *The masses of starlings as measured at four different locations. It seems as though the mean mass varies somewhat according to the location, and our results will tell us how plausible this is.*

11.2.1 Hierarchical Model

Our "one-way ANOVA" model is very much the same as our final "t-test" model, except for the format of the dataset. The main advantage of this model is that it generalises to more than two groups in a very straightforward way; we no longer need to write separate for loops for each group. As with the third "t-test" model, we are not seriously considering the hypothesis that all of the group means (i.e. the μ parameters) are exactly equal, but we are allowing them to be quite close together, or quite distinct in value, by using the hierarchical model structure with the diversity parameter.

```
model
{
    # Log-uniform prior for the scatter
    log_sigma ~ dunif(-10, 10)
    sigma <- exp(log_sigma)

    # Hierarchical prior for the means
    # Hyperparameters
    grand_mean ~ dnorm(0, 1/1000^2)
    log_diversity ~ dunif(-10, 10)
    diversity <- exp(log_diversity)

    # Parameters
    for(i in 1:N)
    {
      mu[i] ~ dnorm(grand_mean, 1/diversity^2)
    }

    # Sampling distribution/likelihood
    for(i in 1:N)
    {
        x[i] ~ dnorm(mu[group[i]], 1/sigma^2)
    }
}
```

After running this model on the starling data, we can plot any results we wish. In Figure 11.4, I have plotted a trace plot of μ_1, the parameter for the mean weight of starlings at location 1. This is a healthy trace plot, although there is a strange feature near iteration 1000 which we will discuss in the next section. Figure 11.5 shows the posterior distribution for the `log_diversity` hyperparameter, which quantifies how different the groups really are. Our prior for this parameter was U(-10, 10), and the posterior peaks at around 1.5, which corresponds to `diversity` $\approx$ 4.5, although there is a fair bit of uncertainty. Notice also the long tail of the posterior on the left hand side. Although we never allowed the μs to be exactly the same, we did allow them to be close (and this corresponds to the diversity being low). The fact that some posterior samples landed between -10 and 0 suggests there is a small probability that the differences between groups are very small, despite the fact that the data doesn't look that way.

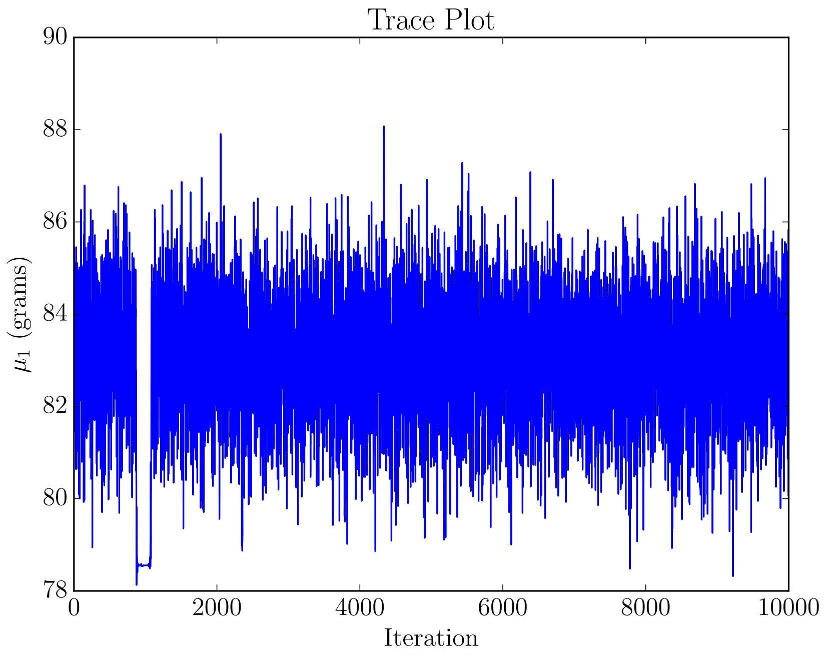

Figure 11.4: *A trace plot of μ_1 from a JAGS run on the starling data. Things appear to be mixing well, except for an odd feature near iteration 1000.*

As usual, we can use our posterior samples to calculate the posterior probability of any hypothesis that we can think of based on the parameters. Here are a couple of interesting examples:

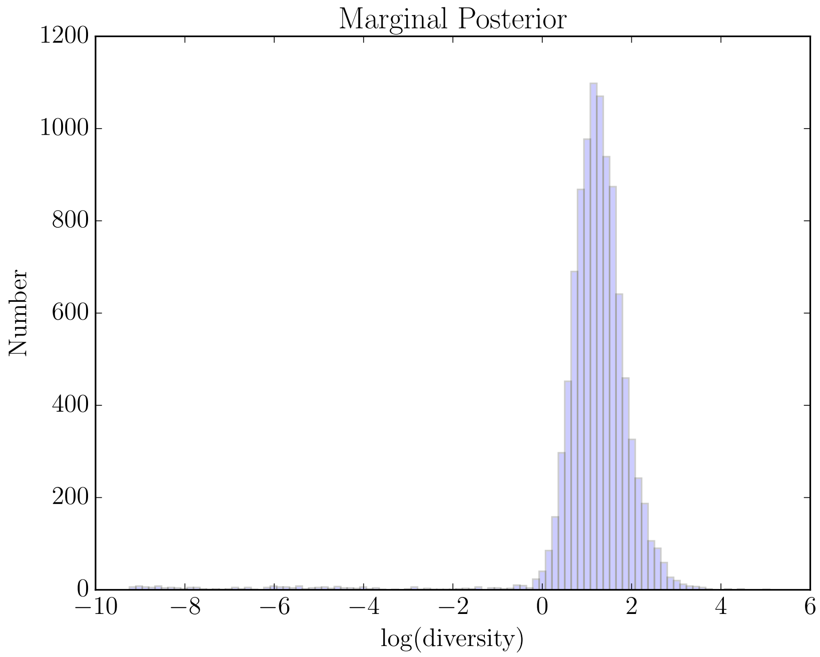

Figure 11.5: *The posterior distribution for* `log_diversity`.

```
# Is mu2 really greater than mu3?
> mean(results$mu[,2] > results$mu[,3])
[1] 0.672
# Is the diversity really less than 1 (log diversity less than 0)?
> mean(results$diversity < 0)
[1] 0.0272
```

11.2.2 MCMC Efficiency

The hierarchical “one-way ANOVA” model given above works, but a quick look at the trace plot suggests the mixing (how easily the MCMC algorithm is able to move around) did have some difficulties (see Figure 11.4). In this particular example the problem wasn’t fatal, but this problem could be more severe with a different data set. In some models, it makes sense to consider the *parameterisation* of the model. There are actually different ways to implement exactly the same model assumptions, but in a way that helps the efficiency of the MCMC sampling. This is done by changing which parameters are defined by “~” and which are defined by “<-”, in a way that keeps the meaning of the model intact, but forces JAGS to do the exploration differently.

Let’s look at a small subset of the above model: just the hierarchical prior for the μs. Here it is:

```
# Hierarchical prior for the means
# Hyperparameters
grand_mean ~ dnorm(0, 1/1000^2)
log_diversity ~ dunif(-10, 10)
```

```
diversity <- exp(log_diversity)

# Parameters
for(i in 1:N)
{
  mu[i] ~ dnorm(grand_mean, 1/diversity^2)
}
```

To understand why this causes problems, we need to understand a little about how JAGS works internally. JAGS uses two main MCMC methods, known as *Gibbs Sampling* and *Slice Sampling*. Both of these methods usually work by updating a single parameter or hyperparameter at a time, while keeping all of the others fixed. Because JAGS is sampling the posterior, each parameter will tend to move about as far as it can without the new value becoming inconsistent with the data or the (joint) prior distribution. But the above model doesn't just have problems exploring the posterior efficiently, but would also have problems exploring the prior!

For example, if the current values of grand_mean and diversity are 50 and 5, and JAGS is moving the parameter mu[3], it will probably move it to somewhere within the range 50 ± 5, roughly speaking, since the prior for mu[3] given grand_mean=50 and diversity=5 is Normal$(50, 5^2)$. But another possibility that is (speaking loosely again) compatible with the prior is to have grand_mean=-1500, diversity=10000, and mu[3]=5600. How would the sampler move from having mu[3]=5 to having mu[3]=5600? It certainly couldn't do this while diversity was still 5. Somehow, diversity would have to be much greater than 5. Yet when the sampler tries to increase the value of diversity, it won't be able to move very far, because that would make it inconsistent with the values of the other mu parameters!

Many MCMC methods (and importantly for us, the ones used by JAGS) are inefficient when the posterior distribution has strong *dependence* between different parameters. Unfortunately, in our one-way ANOVA model, it's not just the posterior that has strong dependence, but even the prior has strong dependence!

11.2.3 An Alternative Parameterisation

We will now look at an alternative way of implementing the hierarchical model, that entails exactly the same assumptions (the same prior distributions and sampling distribution), yet has computational advantages. The alternative parameterisation is given below.

```
# Hierarchical prior for the means
# Hyperparameters
grand_mean ~ dnorm(0, 1/1000^2)
log_diversity ~ dunif(-10, 10)
diversity <- exp(log_diversity)

# Parameters
for(i in 1:N)
{
  n[i] ~ dnorm(0, 1)
  mu[i] <- grand_mean + diversity*n[i]
}
```

The only difference between this implementation and the original is the part within the loop. Instead of defining the prior for the μs directly, we have defined different parameters called `n`, with standard normal priors. We then compute the `mus` deterministically from the `ns`. In this alternative parameterisation, the prior for the `ns` is completely independent of `grand_mean` and `diversity`, so sampling from the prior would be extremely efficient, yet the implied prior for the `mus` is exactly the same as before. Of course, the posterior (what we actually want to sample) will still probably have dependence, but hopefully less.

Running this new version of the model on the starling data gives the trace plot in Figure 11.6, which doesn't have any strange features.

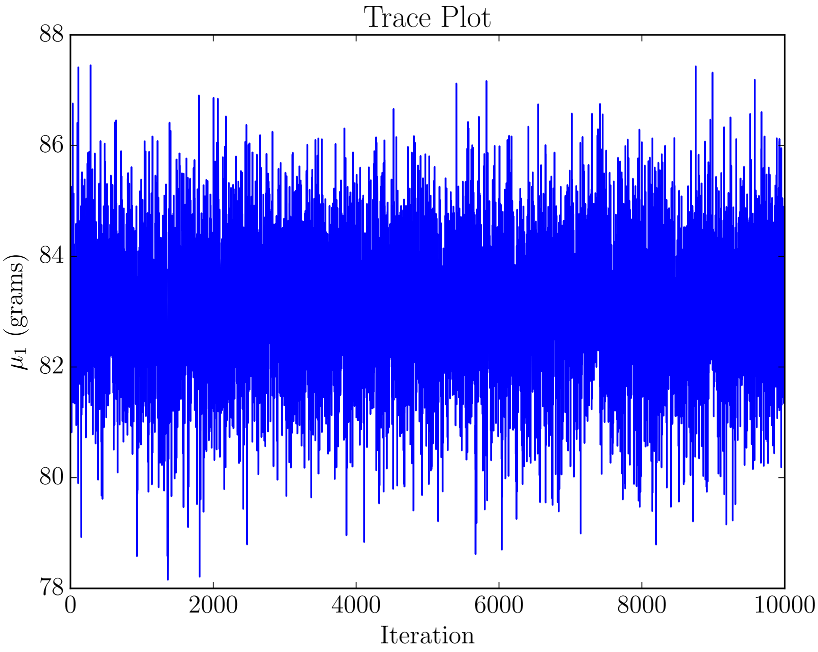

Figure 11.6: A trace plot of the parameter μ_1 using the revised model.

Chapter 12

Acknowledgements

I would like to thank Michael Williams (Columbia), Renate Meyer (Auckland), and Geraint Lewis (Sydney) for their comments. Thanks also to Sampath Fernando (Auckland) for finding several errors, and the other students who noticed an error.

Appendix A

R Background

This chapter describes the R programming techniques that we will need in order to do Bayesian statistics both within R and also by using JAGS.

A.1 Vectors

We will use vectors frequently throughout the course. A vector can be thought of as a list of numbers. In R, you can create a vector using the function `c`, which stands for "concatenate".

```
my_vector = c(12.4, -6.2, 4.04)
```

You can then examine the contents of the vector by typing its name.

```
> my_vector
[1] 12.40 -6.20  4.04
```

There are various helpful things you can do with vectors. You can get the length of a vector (the number of elements in it) like so:

```
> length(my_vector)
[1] 3
```

You can do arithmetic with vectors too, if they're the same length. For example:

```
> c(1,2,3)*c(2,5,3)
[1]  2 10 9
```

If you wanted, you could assign the output to a new variable:

```
> x = c(1,2,3)
> y = c(2,5,3)
> z = x*y
> z
[1]  2 10 9
```

It is important to understand how to access subsets (or "slices") of vectors based on which elements satisfy a certain condition. Here is an example:

```
> x = c(1, 2, 3, 0, 10)
> test = x > 3
> test
[1] FALSE FALSE FALSE FALSE  TRUE
> y = x[x <= 2]
> y
[1] 1 2 0
> z = sum(x[x <= 2])
> z
[1] 3
```

This kind of thing will be used frequently in the computational parts of the course.

A.2 Lists

Lists are a bit like vectors, in that they can contain a lot of information in a single variable. But instead of just being numbers, lists can contain all sorts of things. For example, suppose I want a variable/object in an R program to represent a person. A person can have a name and an age. Here is how to make a list:

```
a_person = list(name="Nicole", age=21)
```

If you needed to extract certain elements from a list, you can do it using the $ operator in R. For example suppose I wanted to extract the `name` variable from within `a_person`. I could do this:

```
> a_person$name
[1] "Nicole"
```

and voila. When we use JAGS, a data set will be represented using a list. So will our JAGS output.

If you have ever learned C, C++, or Matlab, a list in R is basically the same thing as a "struct" in these languages. If you have learned C++, Java, or Python, a list is like a "class" but with just variables and no functions. In Python, "dictionaries" are also very similar to R lists.

A.3 Functions

In this course you'll need to be able to read and understand simple R functions, and perhaps write a few. A function is like a machine that takes an input, does something, and returns an output. You have probably used many built-in R functions already, like `sum()`.

```
# Defining a function called my_function
my_function = function(x)
{
  # Do some stuff
  result = 3*x + 0.5
  return(result)
}
```

A.4 For Loops

For loops are mostly used to repeat an action many times. Here is an example.

```
N = 100
for(i in 1:N)
{
  print(i)
}
```

A.5 Useful Probability Distributions

Since R is a statistics program, it knows about a lot of probability distributions already. So, if I wanted to use the probability density function of a normal distribution, instead of having to code something like this:

```
f = exp(-0.5*((x - mu)/sigma)**2)/(sigma*sqrt(2*pi))
```

I can just use the built-in function `dnorm`.

```
f = dnorm(x, mean=mu, sd=sigma)
```

Much easier! If, instead of wanting to evaluate the PDF, I wanted to generate random samples from a normal distribution, I could use `rnorm`.

```
# Generate 1000 samples
samples = rnorm(1000, mean=50., sd=10.)
```

Appendix A

Probability

This course will use probability theory quite a lot, but we will often use a fairly informal notation. Bayesian statistics is really just probability theory used for a particular purpose, to describe uncertainty. The two most important rules of probability are given below, for reference.

A.1 The Product Rule

The first important rule of probability is the product rule. This tells us how to calculate the probability that any two propositions or hypotheses, A and B, are **both** true. The probability of A **and** B, will be denoted $P(A, B)$. This can be calculated by first finding the probability that A is true, and then multiplying by the probability that B is true *given* that A is true.

$$P(A, B) = P(A)P(B|A) \tag{A.1}$$

We could also have done this the other way around: first finding the probability that B is true and then the probability that A is true given that B is true:

$$P(A, B) = P(B)P(A|B). \tag{A.2}$$

When using the product rule (or any rule of probability, for that matter), you must ensure that the statements to the right of the "given" sign (or the absence of any statements) are consistent throughout. For example, $P(A, B|C) = P(A|C)P(B|A, C)$ is a valid use of the product rule, since "given C" is part of the background information in all of the terms.

You may be familiar with the idea of a *tree diagram* from earlier statistics courses or maybe even high school. A tree diagram is a helpful way to work with the product rule. If you find tree diagrams helpful, feel free to use them, although tree diagrams themselves will not be examinable. An example tree diagram is given in Figure A.1.

The product rule can also be applied to more than two propositions, like so:

$$P(A, B, C) = P(A)P(B|A)P(C|B, A). \tag{A.3}$$

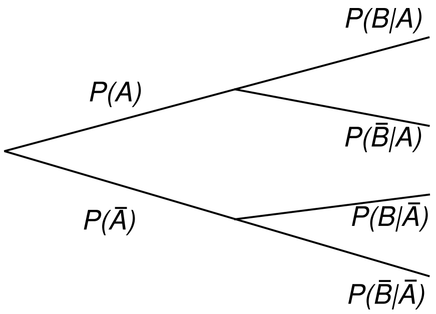

Figure A.1: *A tree diagram.*

You can also apply the product rule in a situation where there is a common statement in the "given part" of all probabilities in the expression. For example, the following is also valid:

$$P(A, B, C|D) \quad = \quad P(A|D)P(B|A, D)P(C|B, A, D). \tag{A.4}$$

In fact, it's best to regard even "unconditional" probabilities such as $P(A)$ as being conditional on some prior information I, which is just left out to keep the notation simple.

A.1.1 Bayes' Rule

Looking at Equations A.1 and A.2, they are both equations for the same thing, $P(A, B)$. Therefore we can equate the right hand sides. Doing this gives a result known as Bayes' rule:

$$P(A|B) \quad = \quad \frac{P(A)P(B|A)}{P(B)}. \tag{A.5}$$

Bayes' rule will be used extensively throughout this course. You will need to know it and know how to use it!

A.2 The Sum Rule

The sum rule is the second important rule of probability. A general statement of the sum rule is

$$P(A \vee B) = P(A) + P(B) - P(A, B). \tag{A.6}$$

where $\vee$ means logical *or*.

The sum rule is often used to calculate the probability of some statement A when we only know the probability of A conditional on some some other statement B. Then we can use the sum rule like this:

$$\begin{aligned} P(A) &= P(A, B) + P(A, \neg B) && \text{(A.7)} \\ &= P(B)P(A|B) + P(\neg B)P(A|\neg B). && \text{(A.8)} \end{aligned}$$

where the $\neg$ symbol means "not", i.e. $\neg B$ is the statement that B is false. To understand this formula, imagine we want to know the probability of A. There are two mutually exclusive ways that could happen: via B being true, or via B being false. The first way has probability $P(B)P(A|B)$, and the second way has probability $P(\neg B)P(A|\neg B)$.

If, instead of just two mutually exclusive and exhaustive pathways B and $\neg B$, there are many, such as $B_1, ..., B_n$. Then the sum rule takes the form

$$P(A) = \sum_{i=1}^{n} P(B_i)P(A|B_i). \tag{A.9}$$

As an exercise, you can try proving this version of the sum rule starting from the simpler version of Equation A.6.

In Bayesian statistics the sum rule is most often used to calculate the marginal likelihood $P(D)$, and to marginalise out "nuisance parameters" from the posterior distribution. In STATS 331 we will mostly use MCMC to do the latter.

A.3 Random Variables

Throughout this course I will use the term "probability distribution" to refer to both the probability mass function for a discrete random variable, and the probability density function for a continuous random variable. I will also use a common shorthand notation.

A.3.1 Discrete Random Variables

We will also see quite a lot of random variables in this course (although without using that terminology very much, as I consider the word "random" to be worse than useless). A discrete random variable is just a quantity X that has a countable number of possible values. A discrete random variable has a *probability mass function* that tells you the probability as a function of the *possible* values x. For example, the equation for the Poisson distribution (a useful discrete distribution) is:

$$P(X = x) = \frac{\lambda^x e^{-\lambda}}{x!} \tag{A.10}$$

for $x \in \{0, 1, 2, 3, ...\}$. The actual random variable is named X, and x is just used so we can write the probabilities $(P(X = 0), P(X = 1), ...)$ as a formula.

A.3.2 Continuous Random Variables

Continuous random variables are those where the set of possibilities is continuous. For example, with a normal distribution, technically any real value is possible. Therefore it doesn't make sense to ask, for example, the probability that $X = 1.32$. The answer is zero because the total probability of 1 has to be spread among an infinite number of possibilities. Instead, we can ask the probability that X is in some region that has a nonzero size. In general, if X has a probability density function $f(x)$, then the probability that $X \in [a, b]$ is:

$$P(a \leq X \leq b) \;=\; \int_a^b f(x)\,dx. \tag{A.11}$$

Note the lower case x in the probability density function. This is analogous to the lower case x in the probability mass function of a discrete random variable. Note that I won't often get you to do an integral analytically. One of the major reasons why MCMC is so awesome is that you can get away without having to do hard integrals!

A.3.3 Shorthand Notation

The notation of random variables can be cumbersome. For example, consider inferring a (discrete) parameter Y from (discrete) data X. Bayes' rule gives us:

$$P(Y = y|X = x) \;=\; \frac{P(Y = y)P(X = x|Y = y)}{P(X = x)}. \tag{A.12}$$

That's very verbose, so instead we use the shorthand:

$$p(y|x) \;=\; \frac{p(y)p(x|y)}{p(x)}. \tag{A.13}$$

In this notation we don't distinguish between the symbol for the random variable and the dummy variables that allow us to write the probability distribution as a formula, we just use the lower case for everything. Despite this simplification, everything still works. Just read $p(y|x)$ as "the probability distribution for y given x" and everything will be fine. Astute readers may have noticed that when I gave the Poisson formula (Equation A.10), "given λ" was implicit throughout.

To be clear when we are talking about a simple probability and when we are talking about the probability distribution for a variable, I will use upper case P for the former and lower-case p for the latter.

A.4 Useful Probability Distributions

In the course we will study Bayesian models which involve the following discrete probability distributions: general (all probabilities given explicitly, such as in a Bayes' Box), binomial, Poisson, discrete uniform, negative binomial, multinomial. We may also use the following continuous distributions: uniform, normal, Cauchy, student-t, beta, log-uniform, exponential, gamma, Dirichlet.

Wikipedia is an excellent resource for looking up all the properties of these distributions, but I will also give various properties (e.g. the mean of a beta distribution) and describe the distributions when we need them.

Appendix A

Rosetta Stone

Bayesian statistics has a lot of terminology floating around, and sometimes different communities use different terms for the same concept. This appendix lists various common terms that are basically synonymous (they mean the same thing). I may even throw in the different terms from time to time, intentionally or unintentionally!

Event, Hypothesis, Proposition, Statement

These are all basically the same thing. A proposition is something that can be either true or false, such as “my age is greater than 35” or “the number of Aardvarks in my room is either zero or one”. These are the things that go in our probability statements: If we write $P(A|B)$, A and B are both propositions, that is, statements that may be true or false. Event is the preferred term in classical statistics.

Sampling Distribution, Probability Model for the Data, Generative Model, Likelihood Function, Likelihood

This is the thing that we write as $p(x|\theta)$. Sometimes it is called the sampling distribution or a generative model because you can sometimes think of the data as having been “drawn from” $p(x|\theta)$ but using the true value of θ, which you don’t actually know. There is a subtlety here, and that is that the word likelihood can be used to mean either $p(x|\theta)$ before x is known (in which case it is the thing you would use to predict possible data) or after x is known. In the latter case $p(x|\theta)$ is only a function of θ because x is fixed at the observed value. The term likelihood function is often used at this point.

Probability Distribution

The term probability distribution is used to refer to either a probability density function (in the continuous case) or a probability mass function (in the discrete case). The latter gives the probability of each particular value, whereas the former only gives a probability if you integrate it within some region.

Marginal Likelihood, Evidence, Prior Predictive Probability, Normalising Constant

This is the $p(x)$ or $P(x)$ term in the denominator of Bayes' rule, and it is also the total of the `prior` × `likelihood` column of a Bayes' Box. This is the probability of getting the data that you actually got, before you observed it: hence the terminology "prior predictive probability". It is also the thing you use to normalise the posterior distribution (make it sum or integrate to 1), hence the term normalising constant. Marginal likelihood makes sense because it is a probability of data (like the regular likelihood) but "marginalised" (i.e. not caring about) the value of the parameter(s). It is also called "evidence" because it can be used to compare different models, or to "patch together" two Bayes' Boxes after the fact (see the hypothesis testing chapter).