***Learning Statistics w/ R - University of Adelaide***

***Part VI – Endings, alternatives and prospects***

**Chapter 17 - Bayesian statistics**

* *In our reasoning concerning “matter of fact”, there are all imaginable degrees of assurance, from the highest certainty to the lowest species of moral evidence. A wise man, therefore, proportions his belief to the evidence*. – David Hume
* The ideas presented prior describe **inferential statistics** from the **frequentist perspective**.
* Almost every text given to undergrad psych students presents the opinions of the frequentist statistician as *THE* theory of inferential statistics, the one true way to do things
* Frequentist statistics dominated the academic field of statistics for most of the 20th century, + this dominance is even more extreme among applied scientists.
* It was + is current practice among psychologists to use frequentist methods.
* B/c frequentist methods are ubiquitous in scientific papers, every student of statistics needs to understand those methods, otherwise they will be unable to make sense of what those papers are saying
* Unfortunately, the current practice in psychology is often misguided + the reliance on frequentist methods is partly to blame.

**17.1 Probabilistic reasoning by rational agents**

* From a **Bayesian perspective**, statistical inference is all about **belief revision**.
* Start out w/ a set of **candidate hypotheses** **h** about the world + don’t know which are *true*, but do have some **beliefs** about *which* are *plausible* + which are not.
* When I observe the data **d**, I have to **revise** those beliefs.
* If data are consistent w/ a hypothesis, my belief in hypothesis is strengthened + inconsistent w/ the hypothesis, my belief in that hypothesis is weakened.
* Key Ideas Ex: I’m carrying an umbrella. Do you think it will rain?
* Single piece of data **d = carrying the umbrella**
* Asking about your beliefs about whether it’s raining.
* 2 possible hypotheses, **h** 🡺 either it rains today or it does not.
* **Priors** = *what you believed before*
* 1st thing to do = ignore what I told you about the umbrella, + write down your pre-existing beliefs about rain
* This is important 🡪 **to be honest about how beliefs have been revised in the light of new evidence, you must say something about what you believed before those data appeared**
* You know I live in Australia + that much of Australia is hot + dry (Adelaide has a Mediterranean climate, very similar to southern California, southern Europe or northern Africa).
* Assume it’s the middle of summer + look on Wikipedia to discover Adelaide gets an average of 4.4 days of rain across the 31 days of January.
* W/out knowing anything else, you might conclude probability of January rain in Adelaide is about 15%, + probability of a dry day is 85%.
* If this is really what you believe about Adelaide rainfall, this is your **prior distribution**, written **P(h):**



* **Likelihoods** = *theories about the data*
* To solve the reasoning problem, you need a **theory** about behavior 🡪 When do I carry an umbrella?
* May assume I will try to carry umbrellas only on rainy days, but we also may know I have young kids + wouldn’t be surprised to know that I’m pretty forgetful about this sort of thing.
* Suppose on rainy days I remember my umbrella ~30% of the time + say on dry days I’m only about ~5% likely to be carrying an umbrella.



* It’s important to remember each cell in this table describes **your beliefs about what data d will be observed, given the truth of a particular hypothesis h.**
* This **conditional probability** is written **P(d | h)** = probability of d given h.
* In **Bayesian statistics**, this = **likelihood of data d given hypothesis h**.
* At this point, all elements are in place 🡪 w/ priors + the likelihood, you have all the info needed for Bayesian reasoning.
* How do we use this information? 🡪 there’s a very simple equation we can use
* Let’s start out w/ 1 of the rules of **Probability Theory** that talks about probability 2 things are true
* Might want to calculate probability today is rainy (hypothesis h = true) + I’m carrying an umbrella (data d is observed).
* The **joint probability** of the hypothesis + the data is written **(P(d, h)** + is calculated by multiplying the **prior** P(h) by the **likelihood** P( d| h) 🡪 Mathematically, say that:



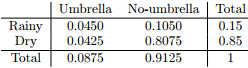
* Probability today = rainy + I remember to carry an umbrella 🡪 The **prior** says P(rain) = 15%, + the **likelihood** says probability of remembering my umbrella *on a rainy day* = 30%.
* Probability both of these things are true = calculated by multiplying the two



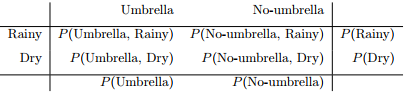
* In other words, before *being told anything* about what *actually happened*, you think there is a 4.5% probability today will be rainy + I will remember an umbrella.
* However, there are 4 four possible things that could happen 🡺 Repeat the exercise for all four.



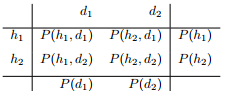
* This captures all the info about which of the 4 possibilities are likely.
* To get the full picture, it helps to add the row totals + column totals.



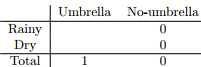
* Some statisticians would object to using the word **likelihood** here
* Problem = the word “likelihood” has a very specific meaning in frequentist statistics that is not quite the same in Bayesian statistics.
* Bayesians didn’t originally have any agreed upon name for “likelihood”, + so it became common practice for people to use frequentist terminology.
* This wouldn’t have been a problem, except the way Bayesians use the word turns out to be quite different to the way frequentists do.
* To put it crudely: when a *Bayesian* says a **likelihood function**, they’re usually referring 1 of the rows of the above table.
* When a *frequentist* says the same thing, they’re referring to the same table, but to them a likelihood function almost always refers to one of the columns.
* This distinction matters in some contexts, but it’s not important for our purposes.
* This is a very useful table, so it’s worth taking a moment to think about what these #’s are telling us.
* 1st, the row sums aren’t telling us anything new at all.
* 1st row = if we ignore all this umbrella business, the chance today will be rainy = 15% = **our prior**
* The important thing isn’t the number itself, but that it gives us some confidence that our calculations are sensible
* The column sums tell us something we haven’t explicitly stated yet.
* In the same way row sums tell us probability of rain, column sums tell us probability of me carrying an umbrella.
* Specifically, the 1st column tells us that, on average (ignoring whether it’s rainy or not), probability of me carrying an umbrella = 8.75%.
* Finally, when we sum across all 4 logically-possible events, everything adds up to 1.
* In other words, what we have written down is a proper **probability distribution** *defined over all possible combinations of data + hypothesis.*



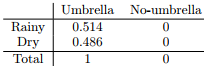
* Finally, use proper statistical notation 🡪 data corresponds to the observation that I do or do not have an umbrella 🡪 d1 = possibility you observe me carrying an umbrella + d2 = you observing me not carrying one.
* Similarly, h1 = your hypothesis that today is rainy + h2 = hypothesis that it is not.



* This table is a very powerful tool for solving the rainy day problem b/c it considers all 4 logical possibilities + states exactly how confident you are in each of them *before being given any data*.
* But, beliefs can change when actually given the data.
* In the rainy day problem, you are told I *really am* carrying an umbrella, which is surprising b/c according to our table, P(u) = 8.75%.
* No matter how unlikely you thought it was, you must now **adjust your beliefs** to accommodate the fact that *you now know that I have an umbrella*.
* If being a bit more sophisticated, could extend the example to accommodate the possibility I’m lying about the umbrella. But let’s keep things simple
* To reflect this new knowledge, our *revised* table must have the following numbers:



* In other words, the **facts** have eliminated *any possibility of no umbrella*, so we have to put 0’s into any cell that implies I’m not carrying an umbrella.
* Also, you know for a FACT I am carrying an umbrella, so the column sum on the left must = 1 to correctly describe the fact that P(u) = 1.
* *What 2 #’s should we put in the empty cells?*
* Don’t worry about the math + instead think about intuitions.
* In the 1st table, it turned out those 2 cells had almost identical numbers
* We worked out the joint probability of rain + umbrella = 4.5%, + the joint probability of dry + umbrella = 4.25%
* In other words, before we knew I am in fact carrying an umbrella, you’d have said these 2 events were *almost identical in probability*
* But notice *both of these possibilities are consistent w/ the fact I actually am carrying an umbrella*.
* *From the perspective of these 2 possibilities, very little has changed*.
* So, it’s still true that these 2 possibilities are equally plausible.
* What we expect to see in our final table is some #’s that preserve the fact that rain + umbrella is slightly more plausible than dry + umbrella, while still ensuring that #’s in the table add up.



* This says that, after being told I’m carrying an umbrella, we believe there’s a 51.4% chance today will be rainy + a 48.6% chance it won’t.
* The **posterior probability** of rain given I am carrying an umbrella **P(h | d)** = 51.4%
* To work out there was a 0.514 probability of rain, take P(rain \* u) = 0.045 + divide it by P(u) = 0.0875
* This produces a table that satisfies our need to have everything sum to 1 + our need not to interfere w/ the relative plausibility of the 2 events that are actually consistent w/ the data.
* To say the same thing using statistical jargon, what I’ve done here is **divide the joint probability of the hypothesis + the data P(d, h) by the marginal probability of the data P(d) to get the posterior probability of the hypothesis given we know the data have been observed.**



* Might notice this equation is actually a restatement of the same basic rule listed at the start of the last section.
* If you multiply both sides of the equation by P(d), you get **P(d)\*P(h | d) = P(d, h)**, which is the rule for how joint probabilities are calculated.
* So I’m not actually introducing any new rules here, it’s the same rule in a different way
* However, remember a joint probability P(d, h) = multiplying the prior P(h) by the likelihood P(d | h)
* This gives us the following formula for the posterior probability:



* This says Probability of hypothesis given observed data = result of the probability of observed data given the hypotheses multiplied by the probability of the hypothesis divided by the probability of the observed data
* This is as **Bayes’ rule =** describes how a learner starts out w/ prior beliefs about the plausibility of different hypotheses + tells you how those beliefs should be revised in the face of data.
* In the Bayesian paradigm, all statistical inference flows from this 1 simple rule.

**17.2 Bayesian hypothesis tests**

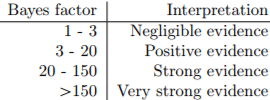
* Null hypothesis testing is a very elaborate contraption people find very hard to make sense of.
* In contrast, the Bayesian approach to hypothesis testing is incredibly simple.
* Pick a setting closely analogous to an orthodox scenario 🡪 2 hypotheses to compare: h0 + h1.
* Prior to running the experiment, we have some **beliefs** **P(h)** about *which hypotheses are true*.
* We run an experiment + obtain data **d**.
* Unlike *frequentist* statistics Bayesian statistics DOES allow one to talk about the probability the null is true.
* Better yet, it allows us to calculate **posterior probability** of the null, using Bayes’ rule
* Similarly, we can work out how much belief to place in the alternative using the same equation.

* This tells us exactly *how much belief we should have in the null after having observed the data d.*
* This is highly simplified 🡪all the complexity of real-life Bayesian hypothesis testing comes down to how you calculate the **likelihood P(d|h)** *when the hypothesis h is a complex + vague thing.*
* In practice, most Bayesian data analysts tend not to talk in terms of the *raw* posterior probabilities P(h0|d) + P(h1|d).
* Instead, they tend to talk in terms of the posterior **odds ratio**.
* Think of it like betting.
* Suppose posterior probability of the null = 25% + posterior probability of the alternative = 75%.
* The alternative is 3X as probable as the null, so we say the odds are 3:1 in favor of the alternative.
* To calculate **posterior odds** = divide 1 posterior probability by the other: P(h1|d) / P(h0|d)



* There are 3 different terms here to know.
* On the LHS = **posterior odds 🡪** tells you what you believe about the relative *plausibility* of the null + the alternative *after seeing the data*
* On the RHS: **prior odds =** what you thought before seeing the data.
* In the middle: the **Bayes factor =** describes the amount of evidence provided by the data:
* **Bayes factor** (**BF**) has a special place in Bayesian hypothesis testing b/c it serves a similar role to the p-value in orthodox hypothesis testing = **quantifies the strength of evidence provided by the data**
* BF is what’s usually reported when running a Bayesian hypothesis test.
* Reason for reporting BF’s rather than posterior odds = different researchers have different priors
* Some might have a strong bias to believe the null is true, others might have a strong bias to believe it is false.
* B/c of this, the polite thing for an applied researcher to do is report the BF
* That way, anyone reading a paper can multiply the BF by their own personal prior odds + can work out for themselves what the posterior odds would be.
* In any case, by convention we like to pretend we give equal consideration to both the null + alternative, in which case the prior odds = 1, + the posterior odds become the same as the BF.
* 1 of the really nice things about BF = *the numbers are inherently meaningful.*
* If in an experiment you compute a BF = 4, it means the evidence provided by your data corresponds to betting odds of 4:1 in favor of the alternative.
* However, there have been some attempts to quantify the standards of evidence that would be considered meaningful in a scientific context.
* The 2 most widely used are from Jeffreys (1961) + Kass + Raftery (1995).
* The Kass + Raftery (1995) table is a bit more conservative.



* Some even think Kass + Raftery standards are a bit charitable + call the “positive evidence” category “weak evidence” + think anything in the range 3:1 to 20:1 is weak/modest evidence at best.
* But there are no hard + fast rules
* What counts as strong or weak evidence depends entirely on how conservative you are + upon the standards your community insists upon before it is willing to label a finding as true.
* In any case, note that all numbers listed above make sense if BF > 1 (i.e., the evidence favors the alternative hypothesis).
* However, 1 big practical advantage of the Bayesian approach relative to the orthodox approach is it also allows you to *quantify evidence for the null*.
* When that happens, BF < 1.
* You can choose to report a BF < 1, but it can be confusing.
* For example, suppose the likelihood of the data under the null P(d|h0) = 0.2, + the corresponding likelihood under the alternative P(d|h0) = 0.1.
* Using the equations above, BF here:



* Read literally, this result tells us the **evidence in favor of the alternative is 0.5:1**
* This can be hard to understand + it can make a lot more sense to turn the equation upside down + report the amount of evidence in favor of the null.



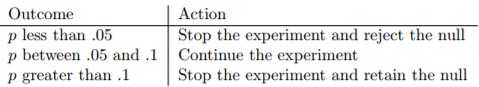
* We report is a BF of 2:1 in favor of the null 🡪 much easier to understand + can be interpreted using the table above

**17.3 Why be a Bayesian?**

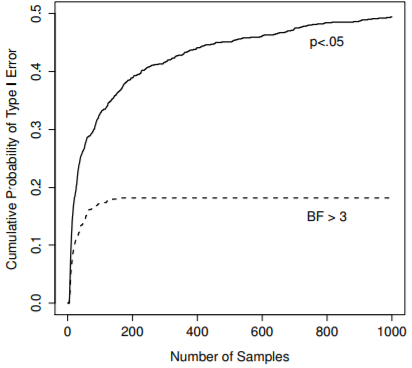
* Up to this point I’ve focused exclusively on logic underpinning Bayesian statistics + talked about the idea of probability as a degree of belief + it implies about how a **rational agent** should reason about the world.
* The question that you have to answer for yourself is this: *how do you want to do your statistics*?
* Do you want to be an orthodox statistician relying on sampling distributions + p-values to guide decisions?
* Or do you want to be a Bayesian, relying on BFs + the rules for rational belief revision?
* Ultimately it depends on what you think is right. It’s your call, + your call alone
* But if one would lean towards Bayesian, statistics mean what you think they mean
* 1 of the biggest advantages to the Bayesian approach is it answers the *right questions*.
* W/in the Bayesian framework, it is perfectly sensible + allowable to refer to the probability a hypothesis is true + can even try to calculate this probability.
* Ultimately, isn’t that what you want your statistical tests to tell you?
* To an actual human being, this would seem to be the whole point of doing statistics: to determine what is true + what isn’t.
* Any time you aren’t exactly sure about what the truth is, you should use the language of **probability theory** to say things like “there is an 80% chance Theory A is true, but a 20% chance Theory B is true instead”
* This seems so obvious to a human, *yet is explicitly forbidden w/in the orthodox framework.*
* To a frequentist, such statements are a nonsense b/c the theory is *true is not a repeatable event*.
* A theory is TRUE or it is NOT, + NO probabilistic statements are allowed, no matter how much you might want to make them.
* There’s a reason why we do not interpret the p-value as probability of the null is true.
* There’s a reason why almost every textbook on statistics is forced to repeat that warning 🡪 b/c people desperately want that to be the correct interpretation.
* Frequentist dogma not w/standing, most actual humans think that the probability a hypothesis is true is not only meaningful, but is the thing we care most about.
* It’s such an appealing idea that even trained statisticians fall prey to the mistake of trying to interpret a p-value this way.
* For example, here is a quote from an official Newspoll report in 2013, explaining how to interpret their (frequentist) data analysis:
* “Throughout the report, where relevant, statistically significant changes have been noted. All significance tests have been based on the 95% level of confidence. This means that if a change is noted as being statistically significant, there is a 95% probability that a real change has occurred, + is not simply due to chance variation.
* *No, that’s not what p < .05 means + is NOT what 95% confidence means to a frequentist statistician*
* It is just plain wrong: *Orthodox methods cannot tell you there is a 95% chance a real change has occurred, b/c this is not the kind of event to which frequentist probabilities may be assigned.*
* To an ideological frequentist, this sentence should be meaningless.
* Even if you’re a more pragmatic frequentist, it’s still the wrong definition of a p-value.
* It is simply not an allowed or correct thing to say if you want to rely on orthodox statistical tools.
* On the other hand, suppose you’re a Bayesian.
* Although the passage is still the wrong definition of a p-value, it’s pretty much exactly what a Bayesian means when they say the **posterior probability** of the alternative is > 95%.
* If the Bayesian posterior is the *actual thing* you want to report, why even try to use orthodox methods?
* If you want to make Bayesian claims, all you have to do is be a Bayesian + use Bayesian tools (maybe the most liberating thing about switching to the Bayesian view)
* Once you’ve made the jump, you no longer have to wrap your head around counterintuitive definitions of p-values + don’t have to bother remembering why you can’t say you’re 95% confident the true mean lies w/in some interval.
* All you have to do is be honest about what you believed before you ran the study + then report what you learned from doing it
* This is the big promise of the Bayesian approach 🡪 do the analysis you really want to do, + express what you really believe the data are telling you
* “If [p] is below .02 it is strongly indicated that the [null] hypothesis fails to account for the whole of the facts. We shall not often be astray if we draw a conventional line at .05 + consider that [smaller values of p] indicate a real discrepancy.” – Sir Ronald Fisher (1925)
* Consider the quote above by Sir Ronald Fisher, 1 of the founders of orthodox statistics.
* If anyone has ever been entitled to express an opinion about the intended function of p-values, it’s Fisher.
* He’s pretty clear about what it means to reject a null at p < .05.
* In his opinion, if we take p < .05 to mean there is a *real effect*, we shall not often be astray.
* This is hardly unusual 🡪 most practitioners express views very similar to Fisher’s.
* In essence, the p < .05 convention is assumed to represent a fairly stringent **evidentiary standard**.
* *Well, how true is that?*
* 1 way to approach this question is to try to *convert p-values to BFs* + see how the 2 compare.
* It’s not easy to do b/c *a p-value is a fundamentally different kind of calculation to a BF* + *they don’t measure the same thing*.
* However, there have been some attempts to work out the relationship between the 2, + it’s somewhat surprising
* For example, Johnson (2013) presents a pretty compelling case that (for t-tests at least) the p < .05 threshold corresponds roughly to a BF of somewhere between 3:1 + 5:1 in favor of the alternative.
* If that’s right, Fisher’s claim is a bit of a stretch.
* Suppose the null is true about ½ the time (i.e., prior probability of H0 = 0.5) + we use those numbers to work out that the posterior probability of the null given it has been rejected at p < .05.
* Using the data from Johnson (2013), we see that if you reject the null at p < .05, you’ll be correct about 80% of the time.
* An evidentiary standard that ensures you’ll be wrong on 20% of your decisions isn’t good enough.
* The fact remains that, quite contrary to Fisher’s claim, if you reject at p < .05 you shall quite often go astray
* It’s not a very stringent evidentiary threshold at all.
* At this point you might be thinking the real problem is not w/ orthodox statistics, just the p < .05 standard.
* In 1 sense, that’s true: The recommendation Johnson gives is not that everyone must be a Bayesian
* Instead, the suggestion is it would be wiser to shift the conventional standard to something like a p < .01 level.
* That’s not an unreasonable view to take, but the problem is a little more severe than that.
* There’s a fairly big problem built into the way most (not all) orthodox hypothesis tests are constructed
* They’re grossly naive about how humans actually do research + b/c of this, most p-values are wrong
* Consider the following scenario: You’ve come up w/ a really exciting research hypothesis + design a study to test it.
* You’re very diligent + run a power analysis to work out what your sample size should be + run the study.
* You run your hypothesis test + out pops a p-value = 0.072. Really annoying
* What should you do? Here are some possibilities:
* 1. Conclude there is no effect + try to publish it as a null result
* If you try to publish it as a null result, the paper will struggle to be published.
* Some reviewers will think that p = .072 is *not* really a null result + will argue it’s *borderline* significant.
* Others will agree it’s a null result, but will claim that even though some null results are publishable, *yours* isn’t.
* 1-2 reviewers might even be on your side, but you’ll be fighting an uphill battle to get it through
* 2. Guess that there *might* be an effect + try to publish it as a *borderline significant* result
* Some reviewers will claim it’s a null result + should not be published.
* Others will claim the evidence is ambiguous + that you should collect more data until you get a *clear* significant result.
* Again, the publication process does not favor you
* 3. Give up + try a new study
* Given the difficulties in publishing an ambiguous result like p = .072, this option might seem tempting: give up + do something else.
* But that’s a recipe for career suicide.
* If you give up + try a new project else every time you find yourself faced w/ ambiguity, your work will never be published
* If you’re in academia w/out a publication record you can lose your job.
* 4. Collect some more data to see if the p value goes up or (preferably) drops below the magic criterion of p < .05
* You don’t have conclusive results, so you decide to collect some more data + re-run the analysis.
* Seems sensible, but unfortunately, if you do this, ALL your p-values are now *incorrect*.
* Not *just* the p-values you calculated for this study. All of them. All p-values you calculated in the past + all p-values you will calculate in the future.
* No one will notice, you’ll get published, + *you’ll have lied*.
* 4) sounds like a perfectly reasonable strategy 🡪 You collected some data, results weren’t conclusive, so now you collect more data until the results ARE conclusive.
* There’s nothing “wrong” w/ it 🡪 It’s a reasonable, sensible + rational thing to do.
* In real life, this is exactly what every researcher does.
* Unfortunately, the theory of null hypothesis testing as (Chapter 11) *forbids* you from doing this
* This b/c the theory assumes the experiment is finished + all the data are in
* B/c it assumes the experiment is over, it only considers 2 possible decisions.
* If using the conventional p < .05 threshold, those decisions are:



* What *you’re* doing is adding a 3rd possible action to the decision-making problem 🡪 using the p-value itself as a reason to justify continuing the experiment + as a consequence transformed the decision-making procedure into one that looks more like this:



* \*\*\*Not all orthodox statistical tests rely on this silly assumption.
* There’re a # of sequential analysis tools sometimes used in clinical trials + the like.
* These methods are built on the assumption that data are analyzed as they arrive + these tests aren’t horribly broken in the way I’m complaining about here.
* However, sequential analysis methods are constructed in a *very different* fashion to the standard version of null hypothesis testing.
* They don’t make it into any intro texts + are not very widely used in psychological literature
* The concern I’m raising here is valid for every single orthodox test I’ve presented so far, + for almost every test reported in the papers
* The “basic” theory of null hypothesis testing isn’t built to handle this sort of thing, not in this form
* If you’re the kind of person who would choose to “collect more data” in real life, it implies you are not making decisions in accordance w/ the rules of null hypothesis testing.
* Even if you happen to arrive at the same decision as the hypothesis test, you aren’t following the decision process it implies, + it’s this failure to follow the process that is causing the problem
* Your p-values are a lie in a dangerous way b/c *they’re all too small*.
* Consider the following (worst-case) scenario: You’re an enthusiastic researcher on a tight budget who didn’t pay any attention to my warnings above + design a study comparing 2 groups
* You desperately want to see a significant result at the p < .05 level, but really don’t want to collect any more data than you have to (it’s expensive).
* In order to cut costs, you start collecting data, but every time a new observation arrives you run a t-test on your data.
* If the t-tests says p < .05, you stop the experiment and report a significant result.
* If not, you keep collecting data.
* You keep doing this until you reach your pre-defined spending limit for this experiment which kicks in at N = 1000 observations.
* As it turns out, the truth of the matter is that *there is no real effect to be found*: the null is true
* So, what’s the chance you’ll make it to the end of the experiment + (correctly) conclude there is no effect?
* In an ideal world, the answer here should be 95%.
* After all, the whole point of the p < .05 criterion is to control the Type I error rate at 5%, so we’d hope is there’s only a 5% chance of falsely rejecting the null in this situation.
* However, there’s no guarantee that will be true.
* *You’re breaking the rules 🡪* you’re running tests repeatedly, “peeking” at your data to see if you’ve gotten a significant result, and all bets are off



* Look at the solid black line 🡪 If you peek at your data *after every single observation*, there’s a 49% chance you will make a Type I error, quite a bit bigger than the 5% it’s supposed to be
* By comparison, imagine you used the following strategy: Start collecting data 🡺 Every single time an observation arrives, run a Bayesian t-test (Section 17.5) + look at the BF
* Assume Johnson (2013) is right + treat a BF of 3:1 as roughly equivalent to a p-value of .05
* Some readers might wonder why I picked 3:1 rather than 5:1, given Johnson suggests that p < .05 lies somewhere in that range.
* I did so in order to be charitable to the p-value.
* W/ a 5:1 BF instead, the results would look even better for the Bayesian approach
* This time around, our trigger-happy researcher uses the following procedure:
* If the BF is 3:1 or more in favor of the null, stop the experiment + retain the null.
* If it is 3:1 or more in favor of the alternative, stop the experiment + reject the null
* Otherwise continue testing.
* Now, just like last time, assume the null hypothesis is true 🡪 results = dashed line
* It turns out the Type I error rate is much, much lower than the 49% rate resulted from using the orthodox t-test.
* In some ways, this is remarkable.
* *The entire point of orthodox null hypothesis testing is to control the Type I error rate.*
* Bayesian methods aren’t *actually designed to do this at all*.
* Yet, as it turns out, when faced w/ a “trigger-happy” researcher who keeps running hypothesis tests as the data come in, the Bayesian approach is much more effective.
* Even the 3:1 standard, which most Bayesians consider unacceptably lax, is much safer than the p < .05 rule
* This is a pretty extreme situation + IRL people don’t run hypothesis tests every time a new observation arrives.
* So, it’s not fair to say the p < .05 threshold “really” corresponds to a 49% Type I error rate (i.e., p = .49)
* But the fact remains: If you want your p-values to be *honest*, you either have to switch to a completely different way of doing hypothesis tests, or you enforce a strict rule: no peeking
* *You’re not allowed to use the data to decide when to terminate the experiment + are not allowed to look at a “borderline” p-value + decide to collect more data.*
* You aren’t even allowed to change your data analysis strategy *after looking at data*.
* You’re *strictly* required to follow these rules, otherwise p-values calculated will be nonsense.
* And *yes*, these rules are *surprisingly* strict.
* Suppose you started running a study w/ the intention of collecting N = 80 people.
* When the study starts out, you follow the rules, refusing to look at the data or run any tests.
* But when you reach N = 50, your willpower gives in + you take a peek + you’ve got a significant result
* You said you’d keep running the study out to a sample size of N = 80, but it seems sort of pointless now as the result is significant w/ N = 50
* Wouldn’t it be wasteful + inefficient to keep collecting data? Aren’t you tempted to stop?
* Keep in mind that if you do, your Type I error rate at p < .05 just ballooned out to 8%.
* When you report p < .05 in your paper, you’re really saying p = .08.
* That’s how bad the consequences of “just one peek” can be.
* Now consider this: Scientific literature is filled w/ t-tests, ANOVAs, regressions + chi-square tests
* These 4 tools appear in most intro stats texts is b/c *these are the bread + butter tools of science.*
* NONE of these tools include a correction to deal w/ data peeking + they all assume you’re not doing it
* But how realistic is that assumption? IRL, how many people do you think have “peeked” at data before an experiment was finished + adapted subsequent behaviors after seeing what the data looked like?
* Except when a sampling procedure is fixed by an external constraint, most people have done it.
* If that has happened, you can infer that the reported p-values are wrong.
* Worse yet, b/c we don’t know *what decision process they actually followed*, we have no way to know what the p-values *should have been*.
* **You can’t compute a p-value when you don’t know the decision-making procedure a researcher used**
* And so, the reported p-value remains a lie.
* Given all the above the take home message is NOT that Bayesian methods are foolproof
* If a researcher is determined to cheat, they can always do so.
* Bayes’ rule cannot stop people from lying, nor from rigging an experiment.
* Key point 🡪 The reason we run statistical tests is to protect us from ourselves.
* The reason **data peeking** is such a concern is it’s so tempting, even for honest researchers.
* A theory for statistical inference HAS to acknowledge this.
* Yes, you might try to defend p-values by saying it’s the fault of the researcher for not using them properly, but that misses the point.
* A theory of statistical inference that is so completely naive about humans that it doesn’t even consider the possibility a researcher might look at their own data isn’t a theory worth having.
* In essence, my point is this: **Good laws have their origins in bad morals** – Ambrosius Macrobius
* **Good rules for statistical testing have to acknowledge human frailty**.
* None of us are without sin or are beyond temptation.
* A good system for statistical inference should still work even when it is used by actual humans.
* **Orthodox null hypothesis testing does not**

**17.4 Bayesian analysis of contingency tables**

* Time to change gears. Up to this point I’ve been talking about what Bayesian inference is and why you might consider using it. I now want to briefly describe how to do Bayesian versions of various statistical tests. The discussions in the next few sections are not as detailed as I’d like, but I hope they’re enough to help you get started. So let’s begin. The first kind of statistical inference problem I discussed in this book appeared in Chapter 12, in which we discussed categorical data analysis problems. In that chapter I talked about several different statistical problems that you might be interested in, but the one that appears most often in real life is the analysis of contingency tables. In this kind of data analysis situation, we have a cross-tabulation of one variable against another one, and the goal is to find out if there is some association between these variables. The data set I used to illustrate this problem is found in the chapek9.Rdata file, and it contains a single data frame chapek9 > load("chapek9.Rdata") > head(chapek9) species choice
* 14http://www.quotationspage.com/quotes/Ambrosius Macrobius/ 15Okay, I just know that some knowledgeable frequentists will read this + start complaining about this section. Look, I’m not dumb. I absolutely know that if you adopt a sequential analysis perspective you can avoid these errors w/in the orthodox framework. I also know that you can explictly design studies w/ interim analyses in mind. So yes, in one sense I’m attacking a straw man version of orthodox methods. However, the straw man that I’m attacking is the one that is used by almost every single practitioner. If it ever reaches the point where sequential methods become the norm among experimental psychologists + I’m no longer forced to read 20 extremely dubious ANOVAs a day, I promise I’ll rewrite this section + dial down the vitriol. But until that day arrives, I stand by my claim that default BF methods are much more robust in the face of data analysis practices as they exist in the real world. Default orthodox methods suck, + we all know it. - 568 - 1 robot flower 2 human data 3 human data 4 human data 5 robot data 6 human flower In this data set, we supposedly sampled 180 beings + measured two things. First, we checked whether they were humans or robots, as captured by the species variable. Second, we asked them to nominate whether they most preferred flowers, puppies, or data. When we produce the cross-tabulation, we get this as the results: > crosstab <- xtabs( ~ species + choice, chapek9 ) > crosstab choice species puppy flower data robot 13 30 44 human 15 13 65 Surprisingly, the humans seemed to show a much stronger preference for data than the robots did. At the time we speculated that this might have been b/c the questioner was a large robot carrying a gun, + the humans might have been scared. 17.4.1 The orthodox text Just to refresh your memory, here’s how we analysed these data back in Chapter 12. B/c we want to determine if there is some association between species + choice, we used the associationTest() function in the lsr package to run a chi-square test of association. The results looked like this: > library(lsr) > associationTest( ~species + choice, chapek9 ) BLAH BLAH BLAH Test results: X-squared statistic: 10.722 degrees of freedom: 2 p-value: 0.005 B/c we found a small p value (in this case p ă .01), we concluded that the data are inconsistent w/ the null hypothesis of no association, + we rejected it. 17.4.2 The Bayesian test How do we run an equivalent test as a Bayesian? Well, like every other bloody thing in statistics, there’s a lot of different ways you could do it. However, for the sake of everyone’s sanity, throughout this chapter I’ve decided to rely on one R package to do the work. Specifically, I’m going to use the BayesFactor package written by Jeff Rouder + Rich Morey, which as of this writing is in version 0.9.10. For the analysis of contingency tables, the BayesFactor package contains a function called contingencyTableBF(). The data that you need to give to this function is the contingency table itself (i.e., the crosstab variable above), so you might be expecting to use a command like this: - 569 - > library( BayesFactor ) # ...b/c we have to load the package > contingencyTableBF( crosstab ) # ...b/c that makes sense, right? However, if you try this you’ll get an error message. This is b/c the contingencyTestBF() function needs one other piece of information from you: it needs to know what sampling plan you used to run your experiment. You can specify the sampling plan using the sampleType argument. So I should probably tell you what your options are! The contingencyTableBF() function distinguishes between four different types of experiment: • Fixed sample size. Suppose that in our chapek9 example, our experiment was designed like this: we deliberately set out to test 180 people, but we didn’t try to control the number of humans or robots, nor did we try to control the choices they made. In this design, the total number of observations N is fixed, but everything else is random. This is referred to as joint multinomial sampling, + if that’s what you did you should specify sampleType = jointMulti. In the case of the chapek9 data, that’s actually what I had in mind when I invented the data set. • Fixed row (or column) totals. A different kind of design might work like this. We decide ahead of time that we want 180 people, but we try to be a little more systematic about it. Specifically, the experimenter constrains it so that we get a predetermined number of humans + robots (e.g., 90 of each). In this design, either the row totals or the column totals are fixed, but not both. This is referred to as independent multinomial sampling, + if that’s what you did you should specify sampleType = indepMulti. • Both row + column totals fixed. Another logical possibility is that you designed the experiment so that both the row totals + the column totals are fixed. This doesn’t make any sense at all in the chapek9 example, but there are other deisgns that can work this way. Suppose that I show you a collection of 20 toys, + then given them 10 stickers that say boy + another 10 that say girl. I then give them 10 blue stickers + 10 pink stickers. I then ask you to put the stickers on the 20 toys such that every toy has a colour + every toy has a gender. No matter how you assign the stickers, the total number of pink + blue toys will be 10, as will the number of boys + girls. In this design both the rows + columns of the contingency table are fixed. This is referred to as hypergeometric sampling, + if that’s what you’ve done you should specify sampleType = hypergeom. • Nothing is fixed. Finally, it might be the case that nothing is fixed. Not the row columns, not the column totals, + not the total sample size either. For instance, in the chapek9 scenario, suppose what I’d done is run the study for a fixed length of time. By chance, it turned out that I got 180 people to turn up to study, but it could easily have been something else. This is referred to as Poisson sampling, + if that’s what you’ve done you should specify sampleType=poisson. Okay, so now we have enough knowledge to actually run a test. For the chapek9 data, I implied that we designed the study such that the total sample size N was fixed, so we should set sampleType = jointMulti. The command that we need is, > contingencyTableBF( crosstab, sampleType = jointMulti ) + the output looks like this: BF analysis -------------- [1] Non-indep. (a=1) : 15.92684 ˘0% Against denominator: - 570 - Null, independence, a = 1 --- BF type: BFcontingencyTable, joint multinomial As w/ most R commands, the output initially looks suspiciously similar to utter gibberish. Fortunately, it’s actually pretty simple once you get past the initial impression. Firstly, note that the stuff at the top + bottom are irrelevant fluff. You already know that you’re doing a BF analysis. You already know that you’re analysing a contingency table, + you already know that you specified a joint multinomial sampling plan. So let’s strip that out + take a look at what’s left over: [1] Non-indep. (a=1) : 15.92684 ˘0% Against denominator: Null, independence, a = 1 Let’s also ignore those two a=1 bits, since they’re technical details that you don’t need to know about at this stage.16 The rest of the output is actually pretty straightforward. At the bottom, the output defines the null hypothesis for you: in this case, the null hypothesis is that there is no relationship between species + choice. Or, to put it another way, the null hypothesis is that these two variables are independent. Now if you look at the line above it, you might (correctly) guess that the Non-indep. part refers to the alternative hypothesis. In this case, the alternative is that there is a relationship between species + choice: that is, they are not independent. So the only thing left in the output is the bit that reads 15.92684 ˘0% The 15.9 part is the BF, + it’s telling you that the odds for the alternative hypothesis against the null are about 16:1. The ˘0% part is not very interesting: essentially, all it’s telling you is that R has calculated an exact BF, so the uncertainty about the BF is 0%.17 In any case, the data are telling us that we have moderate evidence for the alternative hypothesis. 17.4.3 Writing up the results When writing up the results, my experience has been that there aren’t quite so many rules for how you should report Bayesian hypothesis tests. That might change in the future if Bayesian methods become standard + some task force starts writing up style guides, but in the meantime I would suggest using some common sense. For example, I would avoid writing this: A Bayesian test of association found a significant result (BF=15.92) To my mind, this write up is unclear. Even assuming that you’ve already reported the relevant descriptive statistics, there are a number of things I am unhappy w/. First, the concept of statistical significance is pretty closely tied w/ p-values, so it reads slightly strangely. Second, the BF=15.92 part will only 16If you’re desperate to know, you can find all the gory details in Gunel + Dickey (1974). However, that’s a pretty technical paper. The help documentation to the contingencyTableBF() gives this explanation: the argument priorConcentration indexes the expected deviation from the null hypothesis under the alternative, + corresponds to Gunel + Dickey’s (1974) a parameter. As I write this I’m about halfway through the Gunel + Dickey paper, + I agree that setting a 1 is a pretty sensible default choice, since it corresponds to an assumption that you have very little a priori knowledge about the contingency table. 17In some of the later examples, you’ll see that this number is not always 0%. This is b/c the BayesFactor package often has to run some simulations to compute approximate BFs. So the answers you get won’t always be identical when you run the command a second time. That’s why the output of these functions tells you what the margin for error is. - 571 - make sense to people who already understand Bayesian methods, + not everyone does. Third, it is somewhat unclear exactly which test was run + what software was used to do so. On the other hand, unless precision is extremely important, I think that this is taking things a step too far: We ran a Bayesian test of association (see Gunel & Dickey, 1974) using version 0.9.10-1 of the BayesFactor package (Morey & Rouder, 2015) using default priors + a joint multinomial sampling plan. The resulting BF of 15.92 to 1 in favour of the alternative hypothesis indicates that there is moderately strong evidence for the non-independence of species + choice. Everything about that passage is correct, of course. Morey + Rouder (2015) built their Bayesian tests of association using the paper by Gunel + Dickey (1974), the specific test we used assumes that the experiment relied on a joint multinomial sampling plan, + indeed the BF of 15.92 is moderately strong evidence. It’s just far too wordy. In most situations you just don’t need that much information. My preference is usually to go for something a little briefer. First, if you’re reporting multiple BF analyses in your write up, then somewhere you only need to cite the software once, at the beginning of the results section. So you might have one sentence like this: All analyses were conducted using the BayesFactor package in R (Morey & Rouder, 2015), + unless otherwise stated default parameter values were used Notice that I don’t bother including the version number? That’s b/c the citation itself includes that information (go check my reference list if you don’t believe me). There’s no need to clutter up your results w/ redundant information that almost no-one will actually need. When you get to the actual test you can get away w/ this: A test of association produced a BF of 16:1 in favour of a relationship between species + choice. Short + sweet. I’ve rounded 15.92 to 16, b/c there’s not really any important difference between 15.92:1 + 16:1. I spelled out BF rather than truncating it to BF b/c not everyone knows the abbreviation. I indicated exactly what the effect is (i.e., a relationship between species + choice) + how strong the evidence was. I didn’t bother indicating whether this was moderate evidence or strong evidence, b/c the odds themselves tell you! There’s nothing stopping you from including that information, + I’ve done so myself on occasions, but you don’t strictly need it. Similarly, I didn’t bother to indicate that I ran the joint multinomial sampling plan, b/c I’m assuming that the method section of my write up would make clear how the experiment was designed. (I might change my mind about that if the method section was ambiguous.) Neither did I bother indicating that this was a Bayesian test of association: if your reader can’t work that out from the fact that you’re reporting a BF + the fact that you’re citing the BayesFactor package for all your analyses, then there’s no chance they’ll understand anything you’ve written. Besides, if you keep writing the word Bayes over + over again it starts to look stupid. Bayes Bayes Bayes Bayes Bayes. See? 17.4.4 Other sampling plans Up to this point all I’ve shown you is how to use the contingencyTableBF() function for the joint multinomial sampling plan (i.e., when the total sample size N is fixed, but nothing else is). For the - 572 - Poisson sampling plan (i.e., nothing fixed), the command you need is identical except for the sampleType argument: > contingencyTableBF(crosstab, sampleType = poisson ) BF analysis -------------- [1] Non-indep. (a=1) : 28.20757 ˘0% Against denominator: Null, independence, a = 1 --- BF type: BFcontingencyTable, poisson Notice that the BF of 28:1 here is not the identical to the BF of 16:1 that we obtained from the last test. The sampling plan actually does matter. What about the design in which the row columns (or column totals) are fixed? As I mentioned earlier, this corresponds to the independent multinomial sampling plan. Again, you need to specify the sampleType argument, but this time you need to specify whether you fixed the rows or the columns. For example, suppose I deliberately sampled 87 humans + 93 robots, then I would need to indicate that the fixedMargin of the contingency table is the rows. So the command I would use is: > contingencyTableBF(crosstab, sampleType = indepMulti, fixedMargin=rows) BF analysis -------------- [1] Non-indep. (a=1) : 8.605897 ˘0% Against denominator: Null, independence, a = 1 --- BF type: BFcontingencyTable, independent multinomial Again, the BF is different, w/ the evidence for the alternative dropping to a mere 9:1. As you might expect, the answers would be diffrent again if it were the columns of the contingency table that the experimental design fixed. Finally, if we turn to hypergeometric sampling in which everything is fixed, we get... > contingencyTableBF(crosstab, sampleType = hypergeom) Error in contingencyHypergeometric(as.matrix(data2), a) : hypergeometric contingency tables restricted to 2 x 2 tables; see help for contingencyTableBF() ... an error message. Okay, some quick reading through the help files hints that support for larger contingency tables is coming, but it’s not been implemented yet. In the meantime, let’s imagine we have data from the toy labelling experiment I described earlier in this section. Specifically, let’s say our data look like this: > toys pink blue girl 8 2 boy 2 8 The Bayesian test w/ hypergeometric sampling gives us this: > contingencyTableBF(toys, sampleType = hypergeom) BF analysis - 573 - -------------- [1] Non-indep. (a=1) : 8.294321 ˘0% Against denominator: Null, independence, a = 1 --- BF type: BFcontingencyTable, hypergeometric The BF of 8:1 provides modest evidence that the labels were being assigned in a way that correlates gender w/ colour, but it’s not conclusive. 17.5 Bayesian t-tests The second type of statistical inference problem discussed in this book is the comparison between two means, discussed in some detail in the chapter on t-tests (Chapter 13). If you can remember back that far, you’ll recall that there are several versions of the t-test. The BayesFactor package contains a function called ttestBF() that is flexible enough to run several different versions of the t-test. I’ll talk a little about Bayesian versions of the independent samples t-tests + the paired samples t-test in this section. 17.5.1 Independent samples t-test The most common type of t-test is the independent samples t-test, + it arises when you have data that look something like this: > load( harpo.Rdata ) > head(harpo) grade tutor 1 65 Anastasia 2 72 Bernadette 3 66 Bernadette 4 74 Anastasia 5 73 Anastasia 6 71 Bernadette In this data set, we have two groups of students, those who received lessons from Anastasia + those who took their classes w/ Bernadette. The question we want to answer is whether there’s any difference in the grades received by these two groups of student. Back in Chapter 13 I suggested you could analyse this kind of data using the independentSamplesTTest() function in the lsr package. For example, if you want to run a Student’s t-test, you’d use a command like this: > independentSamplesTTest( formula = grade ~ tutor, data = harpo, var.equal = TRUE ) Like most of the functions that I wrote for this book, the independentSamplesTTest() is very wordy. It prints out a bunch of descriptive statistics + a reminder of what the null + alternative hypotheses - 574 - are, before finally getting to the test results. I wrote it that way deliberately, in order to help make things a little clearer for people who are new to statistics. This time around, though, I’ll just jump straight to the test results: Test results: t-statistic: 2.115 degrees of freedom: 31 p-value: 0.043 Again, we obtain a p-value less than 0.05, so we reject the null hypothesis. What does the Bayesian version of the t-test look like? Using the ttestBF() function, we can obtain a Bayesian analog of Student’s independent samples t-test using the following command: > ttestBF( formula = grade ~ tutor, data = harpo ) Notice that format of this command is pretty standard. As usual we have a formula argument in which we specify the outcome variable on the left hand side + the grouping variable on the right. The data argument is used to specify the data frame containing the variables. However, notice that there’s no analog of the var.equal argument. This is b/c the BayesFactor package does not include an analog of the Welch test, only the Student test.18 In any case, when you run this command you get this as the output: BF analysis -------------- [1] Alt., r=0.707 : 1.754927 ˘0% Against denominator: Null, mu1-mu2 = 0 --- BF type: BFindepSample, JZS So what does all this mean? Just as we saw w/ the contingencyTableBF() function, the output is pretty dense. But, just like last time, there’s not a lot of information here that you actually need to process. Firstly, let’s examine the bottom line. The BFindepSample part just tells you that you ran an independent samples t-test, + the JZS part is technical information that is a little beyond the scope of this book.19 Clearly, there’s nothing to worry about in that part. In the line above, the text Null, mu1-mu2 = 0 is just telling you that the null hypothesis is that there are no differences between means. But you already knew that. So the only part that really matters is this line here: [1] Alt., r=0.707 : 1.754927 ˘0% Ignore the r=0.707 part: it refers to a technical detail that we won’t worry about in this chapter.20 Instead, you should focus on the part that reads 1.754927. This is the BF: the evidence provided by these data are about 1.8:1 in favour of the alternative. 18Apparently this omission is deliberate. I have this vague recollection that I spoke to Jeff Rouder about this once, + his opinion was that when homogeneity of variance is violated the results of a t-test are uninterpretable. I can see the argument for this, but I’ve never really held a strong opinion myself. (Jeff, if you never said that, I’m sorry) 19Just in case you’re interested: the JZS part of the output relates to how the Bayesian test expresses the prior uncertainty about the variance σ 2 , + it’s short for the names of three people: Jeffreys Zellner Siow. See Rouder, Speckman, Sun, Morey, + Iverson (2009) for details. 20Again, in case you care . . . the null hypothesis here specifies an effect size of 0, since the two means are identical. The alternative hypothesis states that there is an effect, but it doesn’t specify exactly how big the effect will be. The r value here relates to how big the effect is expected to be according to the alternative. You can type ?ttestBF to get more details. - 575 - Before moving on, it’s worth highlighting the difference between the orthodox test results + the Bayesian one. According to the orthodox test, we obtained a significant result, though only barely. Nevertheless, many people would happily accept p .043 as reasonably strong evidence for an effect. In contrast, notice that the Bayesian test doesn’t even reach 2:1 odds in favour of an effect, + would be considered very weak evidence at best. In my experience that’s a pretty typical outcome. Bayesian methods usually require more evidence before rejecting the null. 17.5.2 Paired samples t-test Back in Section 13.5 I discussed the chico data frame in which students grades were measured on two tests, + we were interested in finding out whether grades went up from test 1 to test 2. B/c every student did both tests, the tool we used to analyse the data was a paired samples t-test. To remind you of what the data look like, here’s the first few cases: > load(chico) > head(chico) id grade\_test1 grade\_test2 1 student1 42.9 44.6 2 student2 51.8 54.0 3 student3 71.7 72.3 4 student4 51.6 53.4 5 student5 63.5 63.8 6 student6 58.0 59.3 We originally analysed the data using the pairedSamplesTTest() function in the lsr package, but this time we’ll use the ttestBF() function from the BayesFactor package to do the same thing. The easiest way to do it w/ this data set is to use the x argument to specify one variable + the y argument to specify the other. All we need to do then is specify paired=TRUE to tell R that this is a paired samples test. So here’s our command: > ttestBF( + x = chico$grade\_test1, + y = chico$grade\_test2, + paired = TRUE + ) + here’s the output: BF analysis -------------- [1] Alt., r=0.707 : 5992.05 ˘0% Against denominator: Null, mu = 0 --- BF type: BFoneSample, JZS At this point, I hope you can read this output w/out any difficulty. The data provide evidence of about 6000:1 in favour of the alternative. We could probably reject the null w/ some confidence! - 576 - 17.6 Bayesian regression Okay, so now we’ve seen Bayesian equivalents to orthodox chi-square tests + t-tests. What’s next? If I were to follow the same progression that I used when developing the orthodox tests you’d expect to see ANOVA next, but I think it’s a little clearer if we start w/ regression. 17.6.1 A quick refresher In Chapter 15 I used the parenthood data to illustrate the basic ideas behind regression. To remind you of what that data set looks like, here’s the first six observations: > load(parenthood.Rdata) > head(parenthood) dan.sleep baby.sleep dan.grump day 1 7.59 10.18 56 1 2 7.91 11.66 60 2 3 5.14 7.92 82 3 4 7.71 9.61 55 4 5 6.68 9.75 67 5 6 5.99 5.04 72 6 Back in Chapter 15 I proposed a theory in which my grumpiness (dan.grump) on any given day is related to the amount of sleep I got the night before (dan.sleep), + possibly to the amount of sleep our baby got (baby.sleep), though probably not to the day on which we took the measurement. We tested this using a regression model. In order to estimate the regression model we used the lm() function, like so: > model <- lm( + formula = dan.grump ~ dan.sleep + day + baby.sleep, + data = parenthood + ) The hypothesis tests for each of the terms in the regression model were extracted using the summary() function, a (somewhat truncated) version of which is shown below: > summary(model) BLAH BLAH BLAH Coefficients: Estimate Std. Error t value Pr(>|t|) (Intercept) 126.278707 3.242492 38.945 regressionBF( + formula = dan.grump ~ dan.sleep + day + baby.sleep, + data = parenthood + ) So that’s pretty straightforward: it’s exactly what we’ve been doing throughout the book. The output, however, is a little different from what you get from lm(). Here’s what we get: BF analysis -------------- [1] dan.sleep : 1.622545e+34 ˘0% [2] day : 0.2724027 ˘0% [3] baby.sleep : 10018411 ˘0% [4] dan.sleep + day : 1.016578e+33 ˘0.01% [5] dan.sleep + baby.sleep : 9.770233e+32 ˘0.01% [6] day + baby.sleep : 2340755 ˘0% [7] dan.sleep + day + baby.sleep : 7.835625e+31 ˘0% Against denominator: Intercept only --- BF type: BFlinearModel, JZS The format of this is pretty familiar. At the bottom we have some techical rubbish, + at the top we have some information about the BFs. What’s new is the fact that we seem to have lots of BFs here. What’s all this about? The trick to understanding this output is to recognise that if we’re interested in working out which of the 3 predictor variables are related to dan.grump, there are actually 8 possible regression models that could be considered. One possibility is the intercept only model, in which none of the three variables have an effect. At the other end of the spectrum is the full model in which all three variables matter. So what regressionBF() does is treat the intercept only model as the null hypothesis, + print out the BFs for all other models when compared against that null. For example, if we look at line 4 in the table, we see that the evidence is about 1033 to 1 in favour of the claim that a model that includes both dan.sleep + day is better than the intercept only model. Or if we look at line 1, we can see that the odds are about 1.6 ˆ 1034 that a model containing the dan.sleep variable (but no others) is better than the intercept only model. 17.6.3 Finding the best model In practice, this isn’t super helpful. In most situations the intercept only model is one that you don’t really care about at all. What I find helpful is to start out by working out which model is the best one, + then seeing how well all the alternatives compare to it. Here’s how you do that. In this case, it’s easy enough to see that the best model is actually the one that contains dan.sleep only (line 1), b/c it has the largest BF. However, if you’ve got a lot of possible models in the output, it’s handy - 578 - to know that you can use the head() function to pick out the best few models. First, we have to go back + save the BF information to a variable: > models <- regressionBF( + formula = dan.grump ~ dan.sleep + day + baby.sleep, + data = parenthood + ) Let’s say I want to see the best three models. To do this, I use the head() function specifying n=3, + here’s what I get as the result: > head( models, n = 3) BF analysis -------------- [1] dan.sleep : 1.622545e+34 ˘0% [2] dan.sleep + day : 1.016578e+33 ˘0.01% [3] dan.sleep + baby.sleep : 9.770233e+32 ˘0.01% Against denominator: Intercept only --- BF type: BFlinearModel, JZS This is telling us that the model in line 1 (i.e., dan.grump ~ dan.sleep) is the best one. That’s almost what I’m looking for, but it’s still comparing all the models against the intercept only model. That seems silly. What I’d like to know is how big the difference is between the best model + the other good models. For that, there’s this trick: > head( models/max(models), n = 3) BF analysis -------------- [1] dan.sleep : 1 ˘0% [2] dan.sleep + day : 0.06265328 ˘0.01% [3] dan.sleep + baby.sleep : 0.06021549 ˘0.01% Against denominator: dan.grump ~ dan.sleep --- BF type: BFlinearModel, JZS Notice the bit at the bottom showing that the denominator has changed. What that means is that the BFs are now comparing each of those 3 models listed against the dan.grump ~ dan.sleep model. Obviously, the BF in the first line is exactly 1, since that’s just comparing the best model to itself. More to the point, the other two BFs are both less than 1, indicating that they’re all worse than that model. The BFs of 0.06 to 1 imply that the odds for the best model over the second best model are about 16:1. You can work this out by simple arithmetic (i.e., 0.06{1 « 16), but the other way to do it is to directly compare the models. To see what I mean, here’s the original output: > models BF analysis -------------- [1] dan.sleep : 1.622545e+34 ˘0% - 579 - [2] day : 0.2724027 ˘0% [3] baby.sleep : 10018411 ˘0% [4] dan.sleep + day : 1.016578e+33 ˘0.01% [5] dan.sleep + baby.sleep : 9.770233e+32 ˘0.01% [6] day + baby.sleep : 2340755 ˘0% [7] dan.sleep + day + baby.sleep : 7.835625e+31 ˘0% Against denominator: Intercept only --- BF type: BFlinearModel, JZS The best model corresponds to row 1 in this table, + the second best model corresponds to row 4. All you have to do to compare these two models is this: > models[1] / models[4] BF analysis -------------- [1] dan.sleep : 15.96086 ˘0.01% Against denominator: dan.grump ~ dan.sleep + day --- BF type: BFlinearModel, JZS + there you have it. You’ve found the regression model w/ the highest BF (i.e., dan.grump ~ dan.sleep), + you know that the evidence for that model over the next best alternative (i.e., dan.grump ~ dan.sleep + day) is about 16:1. 17.6.4 Extracting BFs for all included terms Okay, let’s say you’ve settled on a specific regression model. What BFs should you report? In this example, I’m going to pretend that you decided that dan.grump ~ dan.sleep + baby.sleep is the model you think is best. Sometimes it’s sensible to do this, even when it’s not the one w/ the highest BF. Usually this happens b/c you have a substantive theoretical reason to prefer one model over the other. However, in this case I’m doing it b/c I want to use a model w/ more than one predictor as my example! Having figured out which model you prefer, it can be really useful to call the regressionBF() function + specifying whichModels=top. You use your preferred model as the formula argument, + then the output will show you the BFs that result when you try to drop predictors from this model: > regressionBF( + formula = dan.grump ~ dan.sleep + baby.sleep, + data = parenthood, + whichModels = top + ) BF top-down analysis -------------- When effect is omitted from dan.sleep + baby.sleep , BF is... [1] Omit baby.sleep : 16.60702 ˘0.01% - 580 - [2] Omit dan.sleep : 1.025401e-26 ˘0.01% Against denominator: dan.grump ~ dan.sleep + baby.sleep --- BF type: BFlinearModel, JZS Okay, so now you can see the results a bit more clearly. The BF when you try to drop the dan.sleep predictor is about 10´26, which is very strong evidence that you shouldn’t drop it. On the other hand, the BF actually goes up to 17 if you drop baby.sleep, so you’d usually say that’s pretty strong evidence for dropping that one. 17.7 Bayesian ANOVA As you can tell, the BayesFactor package is pretty flexible, + it can do Bayesian versions of pretty much everything in this book. In fact, it can do a few other neat things that I haven’t covered in the book at all. However, I have to stop somewhere, + so there’s only one other topic I want to cover: Bayesian ANOVA. 17.7.1 A quick refresher As w/ the other examples, I think it’s useful to start w/ a reminder of how I discussed ANOVA earlier in the book. First, let’s remind ourselves of what the data were. The example I used originally is the clin.trial data frame, which looks like this > load(clinicaltrial.Rdata) > head(clin.trial) drug therapy mood.gain 1 placebo no.therapy 0.5 2 placebo no.therapy 0.3 3 placebo no.therapy 0.1 4 anxifree no.therapy 0.6 5 anxifree no.therapy 0.4 6 anxifree no.therapy 0.2 To run our orthodox analysis in earlier chapters we used the aov() function to do all the heavy lifting. In Chapter 16 I recommended using the Anova() function from the car package to produce the ANOVA table, b/c it uses Type II tests by default. If you’ve forgotten what Type II tests are, it might be a good idea to re-read Section 16.10, b/c it will become relevant again in a moment. In any case, here’s what our analysis looked like: > model <- aov( mood.gain ~ drug \* therapy, data = clin.trial ) > Anova(model) Anova Table (Type II tests) Response: mood.gain Sum Sq Df F value Pr(>F) drug 3.4533 2 31.7143 1.621e-05 \*\*\* - 581 - therapy 0.4672 1 8.5816 0.01262 \* drug:therapy 0.2711 2 2.4898 0.12460 That’s pretty clearly showing us evidence for a main effect of drug at p ă .001, an effect of therapy at p ă .05 + no interaction. 17.7.2 The Bayesian version How do we do the same thing using Bayesian methods? The BayesFactor package contains a function called anovaBF() that does this for you. It uses a pretty standard formula + data structure, so the command should look really familiar. Just like we did w/ regression, it will be useful to save the output to a variable: > models <- anovaBF( + formula = mood.gain ~ drug \* therapy, + data = clin.trial + ) The output is quite different to the traditional ANOVA, but it’s not too bad once you understand what you’re looking for. Let’s take a look: > models BF analysis -------------- [1] drug : 245.9026 ˘0% [2] therapy : 0.7316007 ˘0% [3] drug + therapy : 698.3343 ˘0.96% [4] drug + therapy + drug:therapy : 688.3077 ˘1.3% Against denominator: Intercept only --- BF type: BFlinearModel, JZS This looks very similar to the output we obtained from the regressionBF() function, + w/ good reason. Remember what I said back in Section 16.6: under the hood, ANOVA is no different to regression, + both are just different examples of a linear model. Becasue of this, the anovaBF() reports the output in much the same way. For instance, if we want to identify the best model we could use the same commands that we used in the last section. One variant that I find quite useful is this: > models/max(models) BF analysis -------------- [1] drug : 0.3521273 ˘0.96% [2] therapy : 0.001047637 ˘0.96% [3] drug + therapy : 1 ˘0% [4] drug + therapy + drug:therapy : 0.9856421 ˘1.62% Against denominator: mood.gain ~ drug + therapy --- BF type: BFlinearModel, JZS - 582 - By dividing the models output by the best model (i.e., max(models)), what R is doing is using the best model (which in this case is drugs + therapy) as the denominator, which gives you a pretty good sense of how close the competitors are. For instance, the model that contains the interaction term is almost as good as the model w/out the interaction, since the BF is 0.98. In other words, the data do not clearly indicate whether there is or is not an interaction. 17.7.3 Constructing Bayesian Type II tests Okay, that’s all well + good, you might be thinking, but what do I report as the alternative to the p-value? In the classical ANOVA table, you get a single p-value for every predictor in the model, so you can talk about the significance of each effect. What’s the Bayesian analog of this? It’s a good question, but the answer is tricky. Remember what I said in Section 16.10 about ANOVA being complicated. Even in the classical version of ANOVA there are several different things that ANOVA might correspond to. Specifically, I discussed how you get different p-values depending on whether you use Type I tests, Type II tests or Type III tests. To work out which BF is analogous to the p-value in a classical ANOVA, you need to work out which version of ANOVA you want an analog for. For the purposes of this section, I’ll assume you want Type II tests, b/c those are the ones I think are most sensible in general. As I discussed back in Section 16.10, Type II tests for a two-way ANOVA are reasonably straightforward, but if you have forgotten that section it wouldn’t be a bad idea to read it again before continuing. Assuming you’ve had a refresher on Type II tests, let’s have a look at how to pull them from the BF table. Suppose we want to test the main effect of drug. The null hypothesis for this test corresponds to a model that includes an effect of therapy, but no effect of drug. The alternative hypothesis is the model that includes both. In other words, what we want is the BF corresponding to this comparison: Null model: mood.gain ~ therapy Alternative model: mood.gain ~ therapy + drug As it happens, we can read the answer to this straight off the table b/c it corresponds to a comparison between the model in line 2 of the table + the model in line 3: the BF in this case represents evidence for the null of 0.001 to 1. Or, more helpfully, the odds are about 1000 to 1 against the null. The main effect of therapy can be calculated in much the same way. In this case, the null model is the one that contains only an effect of drug, + the alternative is the model that contains both. So the relevant comparison is between lines 2 + 1 in the table. The odds in favour of the null here are only 0.35 to 1. Again, I find it useful to frame things the other way around, so I’d refer to this as evidence of about 3 to 1 in favour of an effect of therapy. Finally, in order to test an interaction effect, the null model here is one that contains both main effects but no interaction. The alternative model adds the interaction. That is: Null model: mood.gain ~ drug + therapy Alternative model: mood.gain ~ drug + therapy + drug:therapy If we look those two models up in the table, we see that this comparison is between the models on lines 3 + 4 of the table. The odds of 0.98 to 1 imply that these two models are fairly evenly matched. You might be thinking that this is all pretty laborious, + I’ll concede that’s true. At some stage I might consider adding a function to the lsr package that would automate this process + construct something like a Bayesian Type II ANOVA table from the output of the anovaBF() function. However, I haven’t had time to do this yet, nor have I made up my mind about whether it’s really a good idea to do this. In the meantime, I thought I should show you the trick for how I do this in practice. The - 583 - command that I use when I want to grab the right BFs for a Type II ANOVA is this one: > max(models)/models denominator numerator drug therapy drug + therapy drug + therapy + drug:therapy drug + therapy 2.839882 954.5292 1 1.014567 The output isn’t quite so pretty as the last one, but the nice thing is that you can read off everything you need. The best model is drug + therapy, so all the other models are being compared to that. What’s the BF for the main effect of drug? The relevant null hypothesis is the one that contains only therapy, + the BF in question is 954:1. The main effect of therapy is weaker, + the evidence here is only 2.8:1. Finally, the evidence against an interaction is very weak, at 1.01:1. Reading the results off this table is sort of counterintuitive, b/c you have to read off the answers from the wrong part of the table. For instance, the evidence for an effect of drug can be read from the column labelled therapy, which is pretty damned weird. To be fair to the authors of the package, I don’t think they ever intended for the anovaBF() function to be used this way. My understanding21 is that their view is simply that you should find the best model + report that model: there’s no inherent reason why a Bayesian ANOVA should try to follow the exact same design as an orthodox ANOVA.22 In any case, if you know what you’re looking for, you can look at this table + then report the results of the Bayesian analysis in a way that is pretty closely analogous to how you’d report a regular Type II ANOVA. As I mentioned earlier, there’s still no convention on how to do that, but I usually go for something like this: A Bayesian Type II ANOVA found evidence for main effects of drug (BF: 954:1) + therapy (BF: 3:1), but no clear evidence for or against an interaction (BF: 1:1). 17.8 Summary The first half of this chapter was focused primarily on the theoretical underpinnings of Bayesian statistics. I introduced the mathematics for how Bayesian inference works (Section 17.1), + gave a very basic overview of how Bayesian hypothesis testing is typically done (Section 17.2). Finally, I devoted some space to talking about why I think Bayesian methods are worth using (Section 17.3). The second half of the chapter was a lot more practical, + focused on tools provided by the BayesFactor package. Specifically, I talked about using the contingencyTableBF() function to do Bayesian analogs of chi-square tests (Section 17.4), the ttestBF() function to do Bayesian t-tests, (Section 17.5), the regressionBF() function to do Bayesian regressions, + finally the anovaBF() function for Bayesian ANOVA. If you’re interested in learning more about the Bayesian approach, there are many good books you could look into. John Kruschke’s book Doing Bayesian Data Analysis is a pretty good place to start (Kruschke, 2011), + is a nice mix of theory + practice. His approach is a little different to the BF approach that I’ve discussed here, so you won’t be covering the same ground. If you’re a cognitive 21Again, guys, sorry if I’ve misread you. 22I don’t even disagree w/ them: it’s not at all obvious why a Bayesian ANOVA should reproduce (say) the same set of model comparisons that the Type II testing strategy uses. It’s precisely b/c of the fact that I haven’t really come to any strong conclusions that I haven’t added anything to the lsr package to make Bayesian Type II tests easier to produce. - 584 - psychologist, you might want to check out Michael Lee + E.J. Wagenmakers’ book Bayesian Cognitive Modeling (Lee & Wagenmakers, 2014). I picked these two b/c I think they’re especially useful for people in my discipline, but there’s a lot of good books out there, so look around!