Practical Applications of Bayesian Statistics

for Data Scientists in Business

Matt DiNauta
Principal Applied Scientist
Zillow Group

About Me

- Principal Applied Scientist at Zillow Group
- Work on models and data products supporting our internal teams: finance, business operations, product, etc.
- Run a Bayesian statistics interest group for data practitioners across Zillow Group

I also run a Bayesian statistics interest group for data practitioners across the company, where we work through text books, and act as a resource for one another when applying these methods to problems at Zillow. This talk comes out of my experience in running that group, where I have had data scientists drop into sessions out of curiosity about the subject and ask, “well, what are the practical applications of this?”.

Intended Audience

- Data scientists new to Bayesian Statistics
- Data scientists of the “analyst” variety: those whose work involves producing insights from data

“I am a busy data scientist. I’m aware of Bayesian statistics but it seems very theoretical and it’s not clear to be why should I add this to my toolbox over other the many other things I could spend my time learning.”

The audience I have in mind for this talk are data scientists new to Bayesian statistics, as this will serve as an introduction.

I also have in mind product data scientists whose work involves producing insights from data and guiding business strategy. For data scientists who are more focused on deep learning, computer vision, etc.,, this content may be less relevant to your day-to-day work. As we’ll see, this is because Bayesian statistics becomes most practically useful when you are working with smaller data, and when we’re interested in inference-type questions.

This quote here on the slides representments the sort of thing I have heard from data scientists who have joined my Bayesian statistics interest group at Zillow. It isn’t clear why they should spend their limited time learning this over the many other things they could be learning. And that’s what I hope to address with this talk.

Goals

- Introduce common terms and notation so you are well prepared for further reading
- Recognize the next time you run into a data science problem that is particularly well suited for applying Bayesian methods

To that end, the two goals I have for the talk are to first introduce some common terms and notation - that will be the first 10 minutes or so - and second, for you to recognize the next time you run into a data science problem that is particularly well suited for applying these methods.

Core Concepts

Bayes' Theorem

$$P(A \mid B) = \frac{P(B \mid A) \cdot P(A)}{P(B)}$$

A, B	= events
$P(A\|B)$	= probability of A given B is true
$P(B\|A)$	= probability of B given A is true
$P(A), P(B)$	= the independent probabilities of A and B

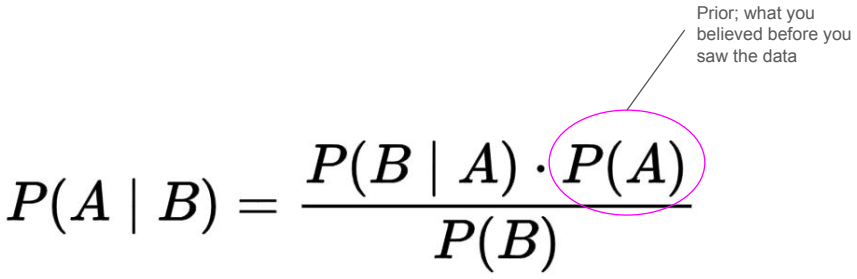
Starting with the most core concept, Bayes' theorem. Two things I'd like to highlight in this.

The first is that this a formula involving conditional probabilities. We're calculating the probability of A given B is true, using the reverse - the probability of B given A is true.

Second, I'd like to point out that we have P(A) in the right-hand side, and we end up calculating P(A|B). So, the difference between the value we're inputting into the formula, and what we're getting out, is that it's now conditioned on B. This is very central point that we'll touch on again and again.

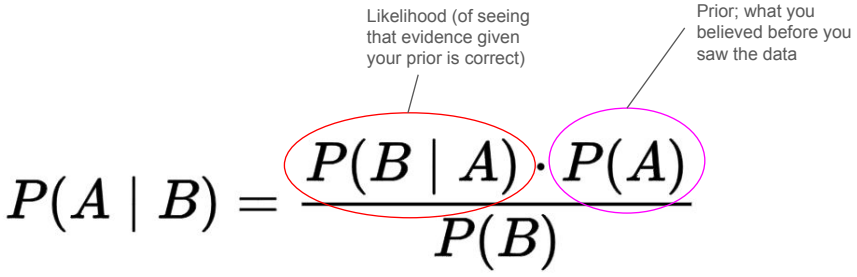
By looking at the individual parts of this formula…

Bayes Rule Annotated



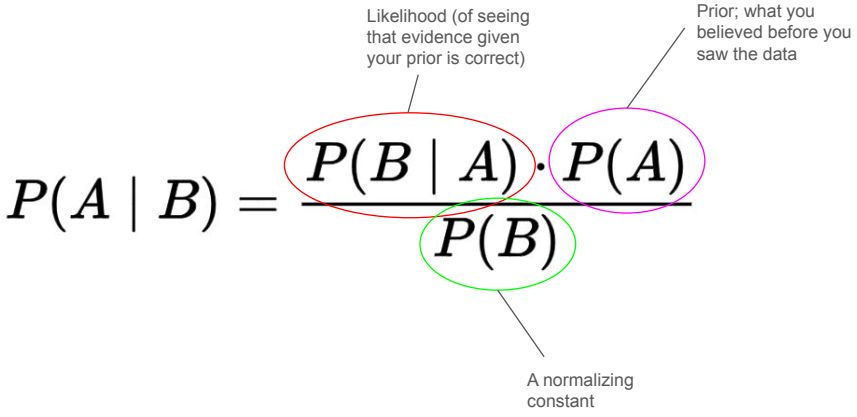
…we can introduce a few other core concepts. We have the probability of some event, P(A), which is known as our prior probability. What we believed before we saw the data.

Bayes Rule Annotated



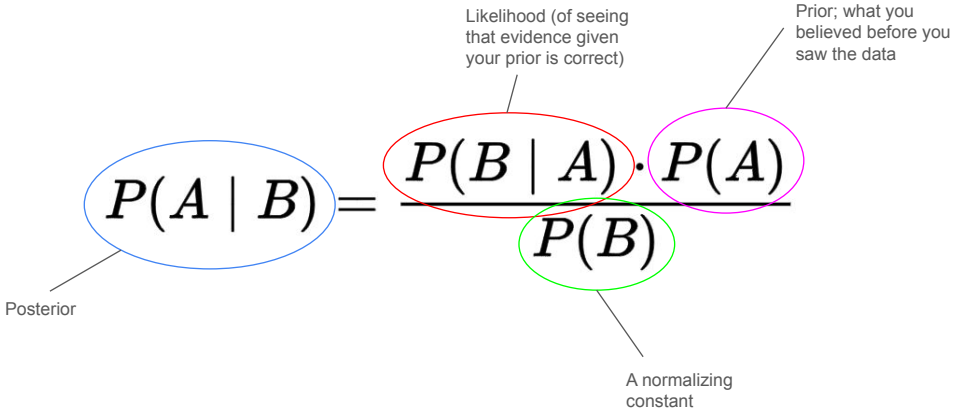
We combine that with the likelihood, the conditional probability of B given A, or the probability of seeing the evidence given the prior. What we mean by “evidence” here is the data that we’ve collected.

Bayes Rule Annotated



Lastly, we divide by the probability of the data, P(B). This is just normalizing to ensure that the probabilities we calculate add up to 1.

Bayes Rule Annotated



And finally we calculate what's known as the posterior probability, the probability of A given B. So the probability of the event given the data.

So, again note that the difference between the prior and the posterior is that, in the posterior, we're conditioning on the data. We start with the prior probability of event, and we calculate the probability of the event given some data. And we call this updating the prior.

Canonical example of applying Bayes rule

An individual tests positive for a rare disease. What is the probability they actually have the disease?

- Disease affects 1 in every 10,000 people in the population (P(D))
- Test has a 1% false positive rate (P(T|~D)); 1% false negative rate (P(~T|D))
- P(D|T)?

If you read any intro text book or tutorial on this topic, you'll probably find this "testing for a rare disease" situation used as a motivating example for Bayes rule. And we'll talk through as well, as it's a really nice example.

An individual tests positive for a rare disease. What is the probability they actually have the disease? It's not 100%, because the test isn't perfect - it has some error.

The disease is quite rare, affecting 1 in 10,000 people. That's our prior. We want to update the prior with the new evidence. The evidence being the positive test result.

The test has a 1% false positive rate. Expressed as a conditional probability, that is the probability of the test coming back positive given the individual that does not have the disease. And the false negative rate is also 1%.

We want to use all of this information to calculate the probability of the disease given a positive test…

Canonical example of applying Bayes rule

An individual tests positive for a rare disease. What is the probability they actually have the disease?

- Disease affects 1 in every 10,000 people in the population (P(D))
- Test has a 1% false positive rate (P(T|~D)); 1% false negative rate (P(~T|D))
- P(D|T)?

- Bayes rule: P(D|T) = [P(T|D) * P(D)] / [P(T|D) * P(D) + P(T|~D) * P(~D)]
  - P(D|T) = [0.99 * 0.0001] / [0.99 * 0.0001 + 0.01 * 0.9999]
- P(D|T) = ~1%

…so we we plug this all into the Bayes theorem formula we saw in the previous slide. It's not necessary to track all that, just note that we're simply plugging into the formula.

And we find that the probability is very low, only 1%. So, this is illustrating why we don't test everyone for all potential diseases routinely. For example we wait until certain ages, when the disease prevalence has increased, to run certain cancer screenings. This is because, even if the test is pretty good - only a 1% false positive rate - given it's such a rare disease, the chances of having it would remain low even after the positive test result.

But also note that, before the test, the probability that the individual had the disease was 1/10,000, and now it is 1/100 after updating with the positive test result. The evidence has increased our probability substantially. Again, we have updated the prior with the new evidence.

Introducing some notation

- Regression equation we are all familiar with: $Y = \beta_0 + X_1\beta_1 + \epsilon$
- Bayesian approach specifies priors for β
- These priors are *probability distributions*

That is the fundamental operation: take a prior probability and update it with some data to arrive at a posterior probability.
When we talk about Bayesian statistics more generally, we're usually talking about incorporating priors into our analysis. Let's look at what that means in the context of a regression model.

In a regression model, we want to estimate some parameters of interest. These estimates are the Betas in the linear regression equation we are all familiar with.

We can apply priors for these estimates. Rather than a single number - like in the disease example, where we had 1/10,000 - we'll use probability distributions as our priors.

In notation,

Introducing some notation

$$\text{data:} \qquad Y_i|\beta_0,\beta_1,\sigma \overset{ind}{\sim} N\left(\mu_i,\sigma^2\right) \quad \text{with} \quad \mu_i = \beta_0 + \beta_1 X_i$$

…the first line is just re-writing the standard linear regression equation. Here is it specifying that Y, given the Betas and Sigma, is normally distributed with mean mu, and mu is the regression equation.

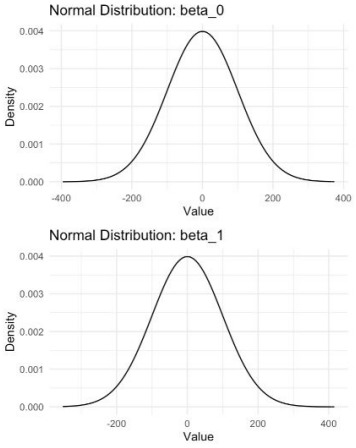
Introducing some notation

$$
\begin{aligned}
\text{data:} \quad & Y_i|\beta_0,\beta_1,\sigma \overset{ind}{\sim} N\left(\mu_i,\sigma^2\right) \quad \text{with} \quad \mu_i = \beta_0 + \beta_1 X_i \\
\text{priors:} \quad & \beta_0 \sim N\left(m_0, s_0^2\right) \\
& \beta_1 \sim N\left(m_1, s_1^2\right) \\
& \sigma \sim \text{Exp}(l).
\end{aligned}
$$

…and where it starts to get interesting is here, where we specify the priors. We say that beta naught and beta 1 each follow a normal distribution with some mean and standard deviation.

Introducing some notation

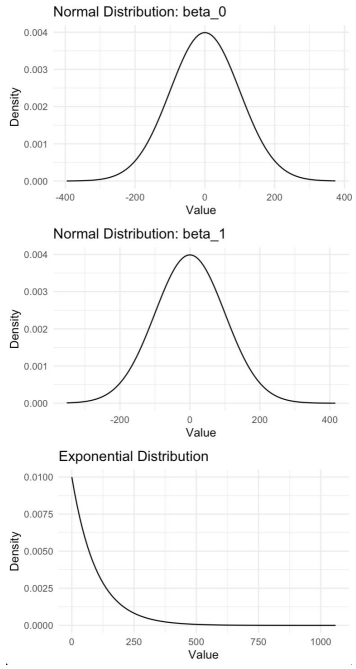
$$\begin{aligned}
&\text{data:} && Y_i|\beta_0,\beta_1,\sigma \overset{ind}{\sim} N\left(\mu_i,\sigma^2\right) \quad \text{with} \quad \mu_i = \beta_0 + \beta_1 X_i \\
&\text{priors:} && \beta_0 \sim N\left(m_0, s_0^2\right) \\
& && \beta_1 \sim N\left(m_1, s_1^2\right) \\
& && \sigma \sim \text{Exp}(l).
\end{aligned}$$



We can make that a little more clear by adding a couple of plots.

Introducing some notation

$$
\begin{aligned}
\text{data:} \quad & Y_i|\beta_0, \beta_1, \sigma \overset{ind}{\sim} N\left(\mu_i, \sigma^2\right) \quad \text{with} \quad \mu_i = \beta_0 + \beta_1 X_i \\
\text{priors:} \quad & \beta_0 \sim N\left(m_0, s_0^2\right) \\
& \beta_1 \sim N\left(m_1, s_1^2\right) \\
& \sigma \sim \text{Exp}(l).
\end{aligned}
$$



And sigma is also given a prior, and that prior is a also a probability distribution. Sigma, the variance, can't be negative, so we give it a prior that reflects that. We'll give it an exponential distribution.

That is the full specification for a simple Bayesian linear regression. To recap, to turn it into a Bayesian model, we supply priors for each of the unknown regression parameters, and those priors are probability distributions.

We might give these Betas priors with a mean of 0 and a large standard deviation, like these plots are showing. This is appropriate in a situation where we don't have very strong prior beliefs on what beta should be. We call that an uninformative prior. We could take that even further and supply a uniform distribution as the prior.

Alternatively, we may have a lot prior information, and so the prior could be something more specific, which will we see in moment.

Practical Examples

Moving on from the core concepts to some examples.

Comparing many proportions

Imagine we are analyzing…

- Sports statistics: baseball batting averages, basketball free-throw percentages
- Real estate agents: conversion rate (deals closed / leads)

We need to somehow account for the varying number of observations in each dimension.

The first is to imagine we're in situation where we need to compare many proportions, or rates. Maybe across lots of different levels of a dimension.

To illustrate what I mean, I like examples from sports. Think about batting averages in baseball, or free-throw percentages in basketball. We need to somehow account for the varying number of attempts; in particular, we need to deal with players who have had very few attempts.

If we sorted major league baseball players by best batting average, we'd likely see a multi-way tie for 100%, a perfect batting average, at the top. This would probably consist of bench players who had a handful of at-bats over their career, and happened to get hits at those few at-bats. Or, brand-new players.

We could set some minimum threshold for inclusion. In fact, this is what the official major league baseball statistic for best career batting average does. They apply a minimum of 3,000 at-bats to be eligible for the "best career batting average" list. But this isn't a satisfying solution. It is an arbitrary cutoff. Why not 2500 at-bats? It is both arbitrary and could make a meaningful difference in our results. Not good.

Instead of a threshold, we can supply a prior. Taking our baseball example, we could look at the full distribution of batting averages. We'd find that the mean is around 25%, with the all-time bests being in the 30%-40% range. That is very useful prior information that we can use to inform our individual-player level estimates. That information implies that it is exceeding unlikely that a new player will have a

something like a 60% batting average, because that would imply that they are twice as good as the best baseball players ever. We can assign a probability very close to 0 there.

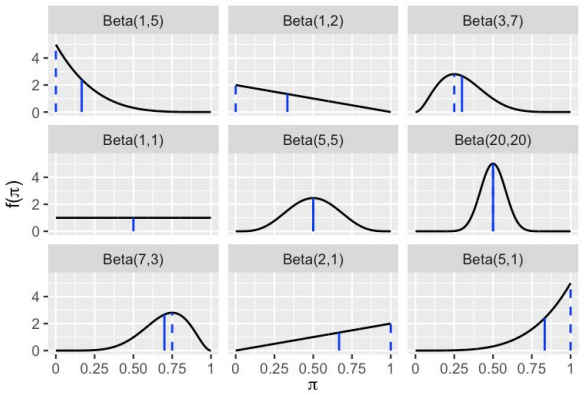
To use an example from my work at Zillow, and we'll use this the next few slides, is that we're interested in real estate agents, who are our customers. One thing we'd like to do is measure their conversion rates on the sales leads they get from Zillow. There are many thousands of real estate agents in US, so we have lots of data in aggregate, but we have little data on any individual real estate agent. We can apply a prior distribution of real estate agent conversion rates, like we did with the baseball player batting averages, and use that to produce better individual-level estimates.

About the Beta distribution

- Often used for proportions
- Described by two parameters, α and β

$$E(\theta) = \frac{\alpha}{\alpha+\beta}$$

- α is the number of successes
- α + β is the number of trials



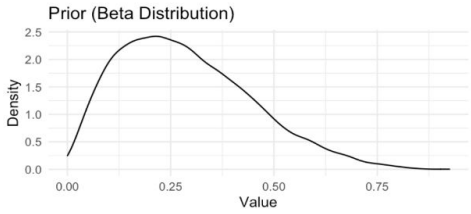
In this case, where we are talking about proportions, it wouldn't appropriate to supply a normal distribution as a prior because proportions can only be positive. More specifically, they are bounded between 0 and 1.

It turns out that the beta distribution is commonly used to model this type of data.

Support for the beta distribution is over the interval 0 to 1. It's also very flexible, as you see here in these example. It can take on a lot of different shapes. So, it's often used to model proportions.
It's described by two parameters, alpha and beta, and it's mean is α divided by α plus β.
You can start to see how, in the example plots, as α and β increases, we get a more peaked distribution.

The beta-binomial model

$$\pi \sim Beta(\alpha, \beta)$$

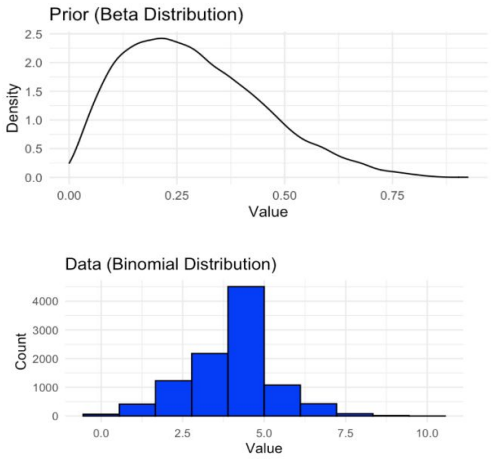


Our prior is a beta distribution, and the data we have, is “number of successes in some number of trials”, which is binomial data….

The beta-binomial model

$$\pi \sim Beta(\alpha, \beta)$$

$$Y \sim Binomial(n, \pi)$$



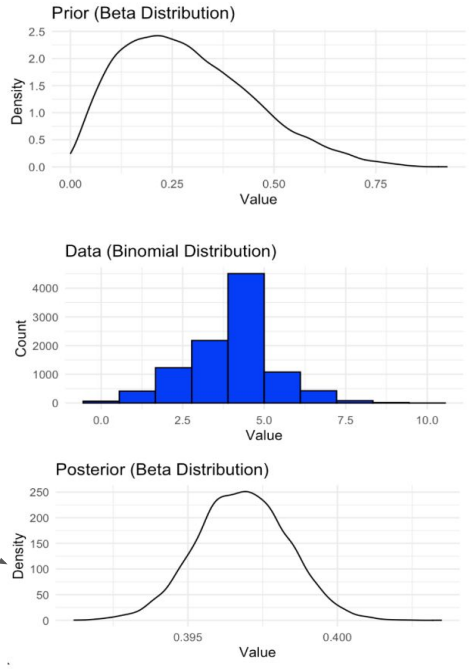
So, back to the fundamental Bayes theorem concept slide from a few minutes ago, we have our prior information encapsulated in a probability distribution, and we want to update that prior with our data to get a posterior.

The beta-binomial model

$$\pi \sim Beta(\alpha, \beta)$$

$$Y \sim Binomial(n, \pi)$$

$$\pi|Y \sim Beta(\alpha + Y, \beta + n - Y)$$



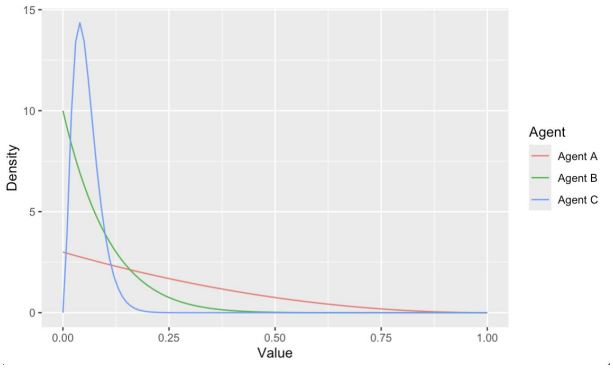
It turns out that it's simply a matter of adding the data and parameters of the beta prior, and we get another beta distribution as a result, as our posterior.

I need to skip past the details of why the math works out this way, why combing the beta prior with the binomial likelihood results in another beta distribution, because it would take a little too long for this talk. I'll note that it's not always as straightforward as summing things up. Often, we need to use computational algorithms to get our posterior. You may have heard of libraries like PyMC in Python, or Stan in R; that is where those come into play.

But, in terms of the concepts, the takeaway is that we are performing the same operation that we have been performing all along - updating the prior with the data to get the posterior.

Ranking performance

- Agent A: 1 sale from 3 leads (30%)
- Agent B: 1 sales from 10 leads (10%)
- Agent C: 3 sales from 50 leads (6%)



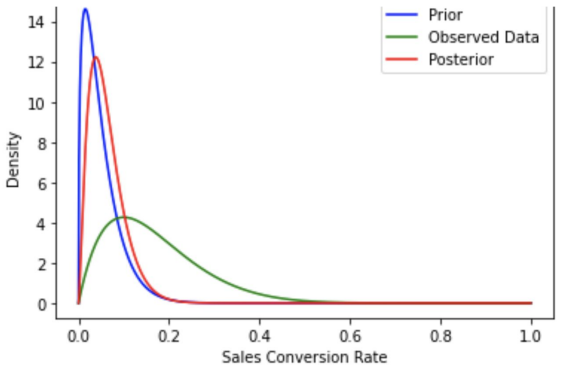
So, again, we are interested in comparing conversion rates of real estate agents - what proportion of their sales leads turn into transactions.

I made up a small amount of data. Let's say we are comparing these three agents, and they have 3, 10, and 50 leads respectively. If we were to rank the agents by their conversion rate, Agent A would rank as #1 with a 30% conversion rate. This doesn't seem reasonable though as that 30% conversion rate is based on only three leads.

We can plot these values as beta distributions and visually see that we are not at all confident in Agent A's conversion rate of 30%. In general, these distributions overlap a lot. This reflects that we cannot, with much confidence at all, conclude that the conversion rate of one Agent is greater than the conversion rate of any other.

Add a prior

- Prior = mean 5% and sd 4%.
- Beta(α ≅1.4, β ≅27.3)



Let's say we have a lot of prior information on conversion rates; we'll imagine they typically have a mean of 5% with a 4% standard deviation.

We can get the parameters for the beta distribution that has that particular mean and standard deviation. These turn out to be roughly alpha equals 1.4 and beta equals 27. And we will combine this prior with our data to get three posterior distributions, one for each of the agents.

Here is a plot of the prior, the data, and the posterior for one of the agents. One thing to note is that the posterior, the red line, is in between the prior and the data. In this case, we've applied a strong prior, so we've "shrunk' the data toward the prior quite a bit.

And this posterior shows that we think it is very unlikely that that the agent's conversion rate is above 10%, and that we assign essentially 0% probability that it's over 20%.

Rank our sales people again

- Agent 1: (1 + 1.4) / (3 + 1.4 + 27.3) = 7.6%
- Agent 2: (1 + 1.4) / (10 + 1.4 + 27.3) = 6.2%
- Agent 3: (3 + 1.4) / (50 + 1.4 + 27.3) = 5.6%

Intuition:

1. “We are ‘starting off’ all agents with 1.4 sales and 28.7 attempts, to reflect that the average agent has a 5% CVR.”
2. “As we collect more data on their performance, the influence of the prior gets smaller.”

Applying this to all of the agents, we get posterior estimates of their conversion rates that have all been “shrunk” down towards our prior in this way.

The 30% conversion rate has been shrunk down to 7.6%. So, we still think that agent may be better than average, but our estimate is reasonable; they are probably not 5 times better than average, they are probably within one standard deviation. So, you can begin to see how this can be useful.

It can also be very intuitive. We can leave aside the details about the probability distributions and such, and explain what we’re doing in a way that is understandable to non-technical stakeholders. One way to look at it is that we are “starting off” all agents with roughly 1 sale and 27 leads, to reflect that the average agent has about a 5% conversion rate. As we collect more data, this will matter less, and the agent’s performance will be reflect more and more the data we’ve collected on them specifically and less the prior.

Extending to hierarchical models

- Rather than a single prior, real estate agents in a particular geographic area probably look more other agents in that geographic area than they do agents from other geographic regions
- We want a different prior for each geographic region. The geographic regions themselves may be considered as belonging to another, global distribution of regions.

This type of situation is the motivation for hierarchical models

We saw that making comparisons based on the observed data alone was not great, because there may be very few data points. Applying a cutoff threshold is also not so great - a cutoff has to be selected, and we throw away data. Applying a prior is an improvement.

But we can improve it further.

Continuing with the example from Zillow, the real estate market is very local. Conversion rates for real estate agents in the Boston area probably look more other agents in Boston than they do agents from Kansas City. To account for this, what we'd really like is a different prior for each geographic region.

We have individuals within geographic regions, let's say a zip code. Those zip codes could themselves be within larger geographic regions, like a city. We have a hierarchical structure to our data that we can exploit to get better priors. This is the motivation for hierarchical models.

The Key Idea: Partial Pooling

- Complete pooling - ignore groups
- No pooling - analyze each group separately
- Partial pooling - middle ground. All groups are connected and thus might contain valuable information about one another.

The key concept in hierarchical models is “partial pooling”.

We can consider the beta binomial model we’ve looked at already as complete pooling. It’s lumping all data together in one big group, and assuming all agents are independent. It’s ignoring any grouping structure that may exist.

The other extreme is “no pooling”, which would be a fitting a completely separate model for each region. This under-utilizes the data, and probably does not generalize well. It probably overfits when we have a group with a small number of data points.

The middle ground that we want is partial pooling, where the groups are independent but connected and we use information from one group to inform others.

A Hierarchical Model Example

- Example from Bayes Rules! An Introduction to Applied Bayesian Modeling
- Spotify: modeling the popularity of songs
  - What's the typical popularity of a Spotify song?
  - To what extent does popularity vary **from artist to artist**?
  - **For any single artist**, how much might popularity vary from song to song?

For this last section, where we will look at a particular hierarchical model, I'm going to introduce a new example. I'm going to take an example from this text book I like called Bayes Rules.

In this example, we are data scientists at Spotify modeling the popularity of songs, and are interested in questions such as:

- What's the typical popularity of a song?
- To what extent does popularity vary from artist to artist?
- For any single artist, how much might popularity vary from song to song?

A Hierarchical Model Example

- Example from Bayes Rules! An Introduction to Applied Bayesian Modeling
- Spotify: modeling the popularity of songs
  - What's the typical popularity of a Spotify song?
  - To what extent does popularity vary from artist to artist?
  - For any single artist, how much might popularity vary from song to song?

- Complete pooling: ignore artists and lump all songs together
- No pooling: separately analyze each artist
- Partial pooling: even though artists differ in popularity, they might share information about each other

Complete pooling will ignore the fact that songs come from different artists.
No pooling will build a completely separate model for each artist.
And partial pooling provides a middle ground, where although artists differ in popularity, they might share information about each other.

Hierarchical Model Specification

Layer 1: $Y_{ij}|\mu_j, \sigma_y \sim N(\mu_j, \sigma_y^2)$ model of individual songs within artist j

Layer 2: $\mu_j|\mu, \sigma_\mu \overset{ind}{\sim} N(\mu, \sigma_\mu^2)$ model of variability between artists

Layer 3:
$\mu \sim N(50, 52^2)$ prior models on global parameters
$\sigma_y \sim \text{Exp}(0.048)$
$\sigma_\mu \sim \text{Exp}(1)$

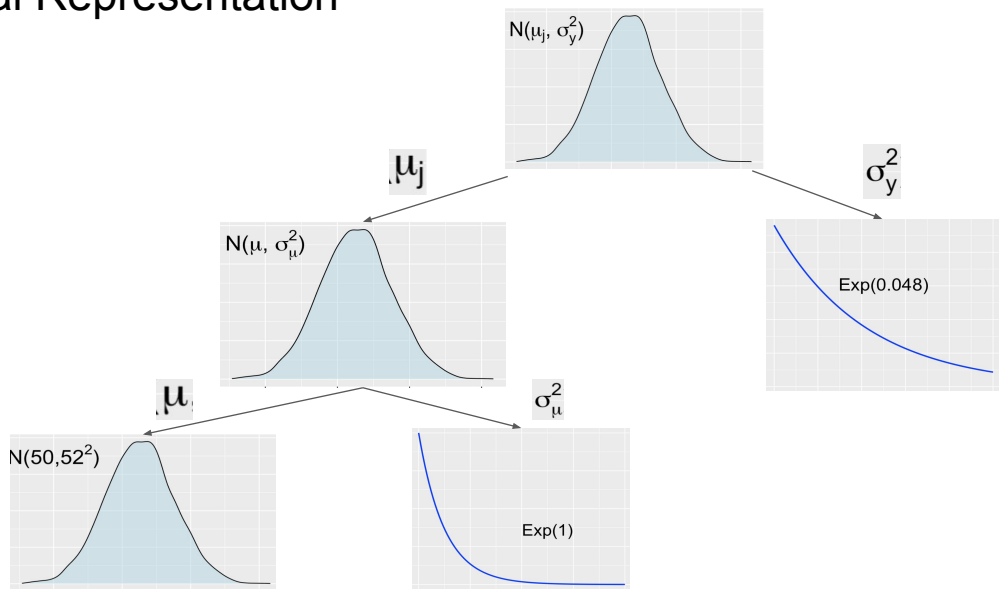
How exactly do we achieve partial pooling?
We move to a hierarchical model, and introduce a layer, layer 2.

To specify this model, it's the same idea as with the earlier examples, except now we're paying close attention to these subscripts.
So, we say that the popularity of song i, which is associated with artist j, is drawn from a normal distribution with mean mu sub j, which is the mean *for that particular artist*. So, this first layer is modeling songs *within artists*. In turn, the artist-level means follow a normal distribution with mean mu. This normal distribution has variance sigma mu, representing the variance *between artists*.

So, we have a model for songs within artists, and then for the variability between artists.

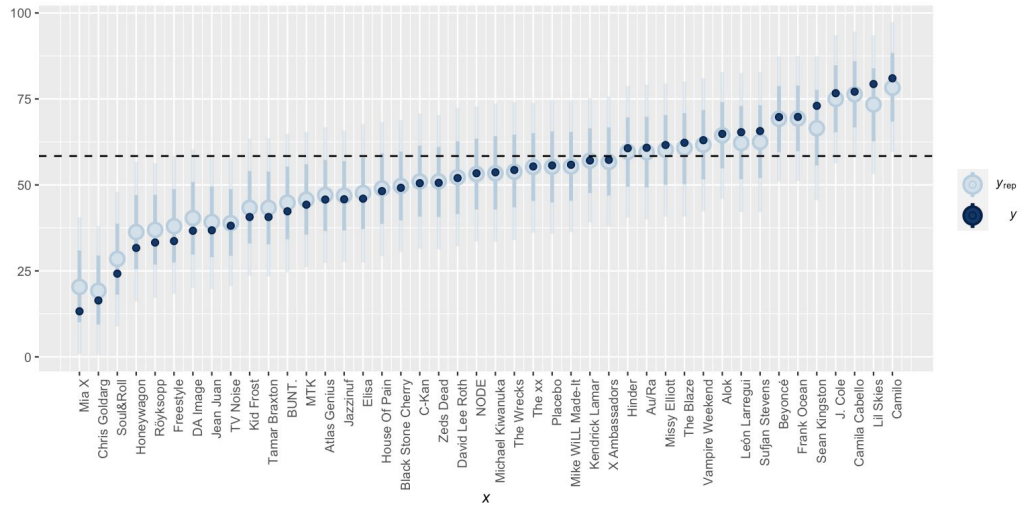
As in the other examples, the global, or highest level parameters, get priors also. And those might be uninformative, or flat priors, or something more informative.

Visual Representation



It is a lot of notation and so can be helpful to have a visual representation as well. Again, we have the distribution of the popularity of songs up top. This has a mean, representing the artist-level mean, and a standard deviation. This is the middle layer. And *that* artist-level mean is drawn from a global distribution which has a standard deviation representing the variability between artists.

Estimates “shrunk” to global mean



In the end, we get a result that involves the same core concept as the baseball players example, and as the real estate agent example.
The estimates are shrunk toward the mean. The more uncertain we are about a given artist, the more their estimate will be shrunk toward the global mean.

In the plot, the dashed line represents the global mean of all songs, and the black dots dots represent the artist mean, measured independently. So, not pooled. The blue dots are posterior means from the hierarchical model. Those have been shrunk toward the global mean.

There are a few connections from this to other concepts you may be familiar with. From a bias/variance trade-off perspective, we’ve reduced the variance and added bias. This is also closely connected to the concept of regularization.

Recap

- Priors are useful when we have small amounts of data
- A situation where they can be very useful is when we have a lot of information globally, but little information in the dimensions we are interested in
- Take this idea further by exploiting hierarchical structure

This talk is a short introduction to a deep topic, so let's recap what we've covered so far.

Priors are useful when we have small amounts of data. And we very often have prior information. A particular situation that I have found comes up often is when I want to make lots of estimates, across levels of some dimension. I may have a lot of information in total, but little information in the level I am interested in.

So, in my work at Zillow, we have lots of data on real estate transactions in US, but few involving an individual real estate agent that I'm interested in.
Spotify has many songs in their catalog, but fewer for any given artist, and very few songs for some artists.

We can take this general idea of applying a prior a little further through exploiting a hierarchical structure in the data.
I can analyze data on real estate agents within a specific geographic area, and if that geographic area happens to be small, pull it more towards the global average.
I can analyze the popularity of songs on Spotify, and if some particular song has few ratings, pull it towards the mean for that artist. If, in turn, if that artist also has few ratings, pull both the song and the artist's estimates toward the global mean.

Last Step: Making Comparisons

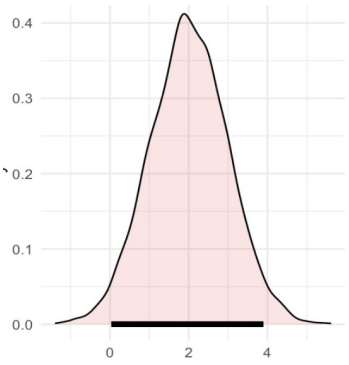
- We have a bunch of posterior distributions. How do we compare them?
- Frequentist approach will use p-values, confidence intervals
- Bayesian approach is arguably more intuitive

One final concept to close out. Let's say we've fit one of these models, and we now have a bunch of posterior distributions. What is the final step - how can we compare those individual posterior distributions? We know the tools from the frequentist approach: p-values, confidence intervals. The Bayesian approach is arguably more intuitive.

The key takeaway here is that the posterior distribution is a full probability distribution for the thing we are interested in. We can summarize or compare those distributions in any way we want, and we can make direct, probabilistic statements about what we believe the value of interest to be. This is as opposed to the maneuvering we do with "assuming the null is true, the probability that we'd see data as extreme.." etc. Or with frequentist confidence intervals, which have a particularly nuanced definition. The statements we can make from our posterior distributions are much more direct.

Last Step: Making Comparisons

- HDI → highest density interval. Smallest possible interval that covers n% (e.g. 95%) of the probability density

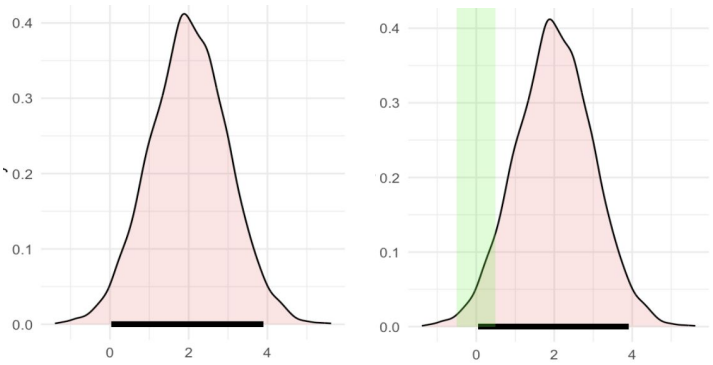


A heuristic that is useful is the Highest Density Interval. The highest density interval is the smallest possible interval that covers some percent, maybe 95%, of the probability density. In the plot on the bottom left, the highest density interval is shown by the black bar at the bottom. This is a Bayesian analogue to a confidence interval.

If this were the posterior distribution for some parameter in a regression, and we wanted to know if the value that parameter is greater than 0, 0 is outside of the 95% HDI, so we may say yes, we're quite confident that it is.

Last Step: Making Comparisons

- HDI → highest density interval. Smallest possible interval that covers n% (e.g. 95%) of the probability density
- ROPE → “region of practical equivalence”

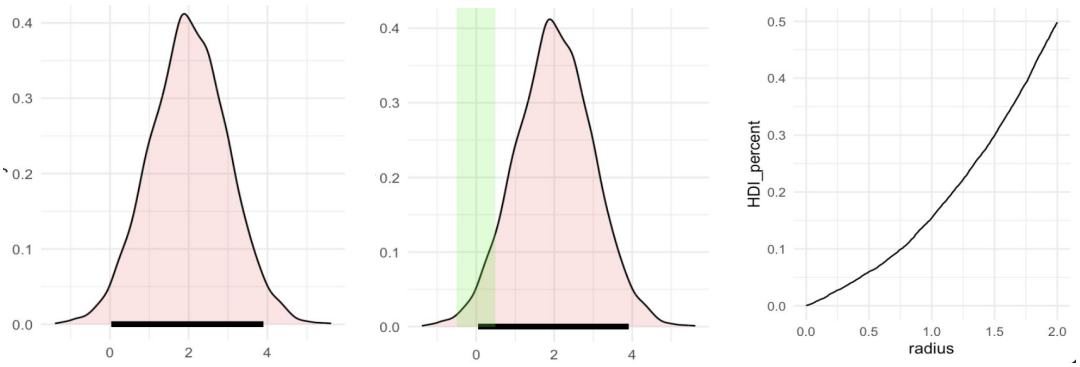


We can also talk about a ROPE, or region of practical equivalence. Again, we are interested in if our parameter is greater than 0, but for practical purposes some number very close to 0 is effectively 0. Maybe we’ve determined that plus or minus 0.05 is practically equivalent to 0 for our purposes. This would be the area covered by the green bar. In that case, it our region begins to overlap with our highest density interval - roughly 6% of the highest density interval is in the ROPE.

We’re still pretty confident that our parameter is greater than 0. Whatever decision we are going to make based on “is the parameter greater than 0”, we probably want to make it, unless we’re very risk averse.

Last Step: Making Comparisons

- HDI → highest density interval. Smallest possible interval that covers n% (e.g. 95%) of the probability density
- ROPE → “region of practical equivalence”



To that end, we can plot a curve that shows a us, as our ROPE increases, how much does it overlap with our HDI.

The takeaway I’d like folks to get from this is that we’ve moving beyond a binary idea of statistical significance, and instead directly looking our posterior probability density and making statements based on that.

I find that, in a business situation, I’m doing data science because the business needs to make some decision. My role to inform the decision, or make a recommendation. Heuristics like this region of practical equivalence are nice because they get us thinking very directly in terms of the cost/benefit of the decision we’ll make. And get us beyond the “statistically significant or not” thinking, which has its place, but we can’t do cost/benefit analysis based on that alone.

Goals: Revisited

- Introduce common terms and notation so you are well prepared for further reading
  - Bayes' theorem: prior, likelihood, and posterior
  - The beta-binomial model
  - Hierarchical models: partial pooling
- Recognize the next time you run into a problem that is particular well suited for applying Bayesian methods
  - You have small data, but some prior beliefs
  - Hierarchical structure to data
    - Real estate agents within geographic regions
    - Songs within artists (within genres?)
  - Quantifying cost/benefit is important - the *direct probabilistic* interpretations we get from Bayesian approach lends itself well to this

Let's revisit our two goals from the top.

We wanted to introduce common terms and notation so you are well prepared for further reading.
To that end, we talked about Bayes' theorem, and the prior, likelihood, and posterior.
We also covered the beta-binomial model, and introduced hierarchical models.

We also wanted to recognize the next time we run into a problem that is particularly well suited for applying Bayesian methods.
I find the key is small data, but some prior information or beliefs.
I suggest that, in particular, look for situations where you want to produce estimates for many individual levels of some dimension, and you have small data at those levels but a large amount of data overall.

Another thing to look out for is hierarchical structure in your data. I run into this all of the time with real estate data because location is super important.
In the Spotify example, it was songs within artists. Perhaps artists then within genres.

Lastly, when you are concerned with making a cost/benefit calculation, the direct probabilistic interpretations we get from Bayesian approach lends itself well to this.

Appendix

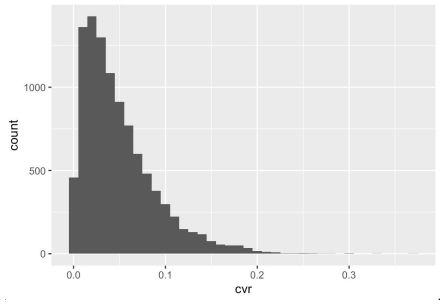
Recommended Further Reading

- Bayes Rules!
  - Freely available online
  - Succinct, lots of practical examples
- Doing Bayesian Statistics
  - Lots of direct comparisons between the frequentist way of doing things vs. Bayesian
  - E.g. “what is the Bayesian analogue for a t test”?
- Rethinking Statistics
  - More of a complete introduction
  - Ties together Bayesian statistics and causal inference

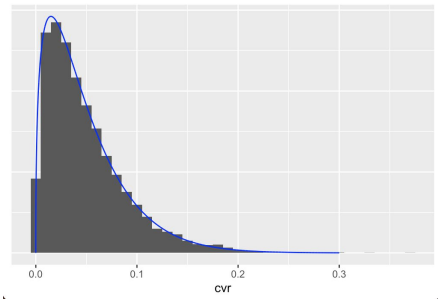
How to determine a prior?

- Priors can come from previous research, domain experts, or the data itself (empirical Bayes)
  - I find that the empirical Bayes approach is particularly useful for practical data science applications
- Consider an empirical distribution of salesperson-level CVRs (a)
- Fit a Beta distribution to the data - we get the two parameters of the Beta distribution that best fits (b)
- This Beta distribution is used as our prior for estimating the CVR of individual sales people

(a)



(b)



How do we actually fit these models

- Terms to know: Probabilistic programming, MCMC
- Probabilistic programming
  - Python: pymc
  - R: Stan, JAGS
- Declarative: you describe the model; the libraries fit
- Practical benefit: do not need to find, or write, an implementation for different methods. E.g. a library for time-series modeling, a library for hierarchical/mixed models. If you can describe it, pymc will (try to) fit it

“Shrinkage” Intuition