Beyond One Proportion

STAT 341, Spring 2023

2023-02-01

Contents

What Came Before

So far, we have been fitting very simple models: we are only trying to estimate **one** parameter, and it has always been a proportion.

But our recent exploration of causal diagrams has reminded us that we are most often interested in *relationships* between two or more variables (and our one-parameter simple models were oversimplifying things quite a bit). For example, the probability of a patent being for a product is not really the same for every patent in the world. It surely depends on lots of things (the company filing the patent, tariffs, the state of the economy, as well as other shifts over time).

Now that we have a more solid grasp on what the prior, likelihood and posterior are and are a bit familiar with grid search and quadratic approximation as two ways of fitting Bayesian models, we are ready to branch out into slightly more complicated models!

Probability Distributions - Review (?)

So far, we've seen the Uniform, Beta, and Binomial distributions. There are A LOT MORE options out there! Each one is a function that matches possible *values* of a variable (usually depicted on the x-axis) with a measure of how often they occur (usually on the y-axis).

We can classify them into the categories of **discrete** and **continuous** distributions.

- Discrete distributions model variable that can take on a discrete set of values (for example, the number of successes in n trials, or the number of birds spotted in the forest). For these, the function values are a set of probabilities that sum to 1.
- Continuous distributions model variables that can take on any numeric value within a specified range (for example, proportion between 0 and 1, or height of people in cm). For these, the function values are in "density" or "likelihood" units – they are scaled such that the total area under the functions' curve is 1.

The **support** of a distribution describes the range of values a variable can have (it might be 0-1, 0-∞, $-\infty$ - ∞, or (for the Uniform!) values between some specified minimum and maximum).

Knowing all this, you can choose a distribution that is a good fit for a variable by matching its type and support.

Here are some flow charts that try to illustrate the process:

Distributions in R

In R, each distribution has 2 functions we may use often. (Fill in the `___` with the (often shortened) name of the distribution.)

- `d____()` returns the value(s) of the function corresponding to given variable value(s). The result are in *Likelihood* units: either probability for discrete distributions, or density for continuous ones. The first input is `x`, the variable value(s). Examples: `dunif()`, `dbinom()`, `dbeta()`...
- `r____()` returns random sample(s) drawn from the specified distribution. The first input is `n`, the number of samples to draw.

To get information about parameter/input names for a specific function, ask R for help...for example,

```
?dunif
```

```
## starting httpd help server ... done
```

Yikes

Don't get overwhelmed. We know there are lots of distributions out there. But we will ease into it.

Normal (Gaussian)

The only new one we'll really use for the moment is the **Normal or Gaussian distribution**, which you should have seen before. It has parameters μ, the `mean` or center; and σ, the `sd` (standard deviation) or spread.

Why the Normal?? The very short answer is that is comes up, and proves useful, often...both mathematically and with real-world data. For a much longer answer check out SR Chapter 4.1.

Question: Drawing on previous knowledge (and maybe the charts above), is the Normal distribution discrete or continuous? What is its support?

> Normal distribution is continuous, and its support is from $-\infty$ to ∞.

Question: If we somehow used a normal distribution in a model, what would be really different about our posterior? (*Hint: at least twice the fun...*)

> Mean and standard deviation.

```r
dnorm(0, mean = 0, sd = 0.1) # 1
```

Question: Try to explain in words what the R code, and output, below mean.

```
## [1] 3.989423
```

```r
dnorm(0, mean = 2, sd = 10) # 2
```

```
## [1] 0.03910427
```

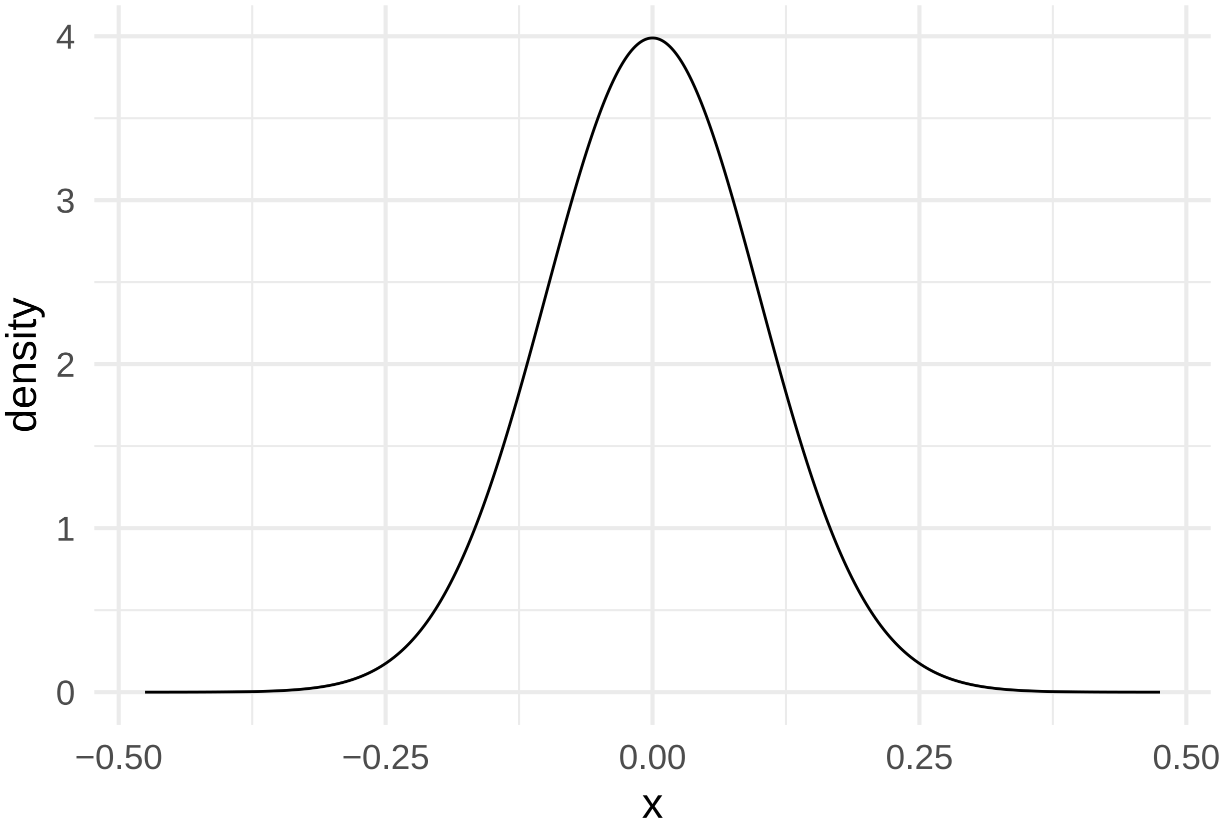
```r
rnorm(7, mean = 44, sd = 10) # 3
```

```
## [1] 53.21190 32.26411 57.82191 41.53370 50.20319 29.65691 35.52667
```

> 1. Calculates the density at 0 for normal distribution with mean = 0 and sd = 0.1.
> 2. Calculates the density at 0 for normal distribution with mean = 2 and sd = 10.
> 3. Generate 7 random samples from a normal distribution with mean = 44 and sd = 10.

Hint: remember you can use `gf_dist('dist_name', parameter_name = value, parameter2_name = value)` to draw a probability distribution if it helps you visualize!

```r
gf_dist('norm', mean = 0, sd = 0.1)
```



Language for Describing Models

Text reference: SR 4.2

As our models get more complex, we need notation to help us describe them in terms of **parameters** we want (or need) to estimate, and the probability distributions we're using to model them.

We will often use the symbol $\sim$ to mean "follows the distribution" or "is distributed."

Rearrange the items below and fill in the blanks to get a description of our blood model (you should end up with 2 lines, each of the form *parameter* $\sim$ distribution)

- Parameters and data variables: b (count of "blood"), n (number of trials), p (probability of blood), 0, 1
- Distributions: Binomial(. . . , . . .); Uniform(. . . ,. . .)

Question: A complete model description has to tell us all about the prior(s) and the likelihood. In your description above, which are which?

> Prior: σ ~ Uniform(0, 1)
>
> Likelihood: b ~ Binomial(n, p)

Practice Describing a Model

Your book considers a model for $i = 1, 2, 3, ...n$ observations of peoples' heights, so height$_i$ is the height given in the ith row of the dataset.

Eek, suddenly we're using a fair bit of mathematical notation in R. If you love it, LOVELY. If you need help, a brief reference guide is available. If you want to just use paper, that's ok too. Go back and forth from the Rmd to the the compiled version to get a sense for how the (LaTeX-y) notation works, too.

$$\text{height}_i \sim \text{Normal}(\mu, \sigma)$$

$$\mu \sim \text{Normal}(\text{mean} = 178, \text{sd} = 20)$$

$$\sigma \sim \text{Unif}(\text{min} = 0, \text{max} = 50)$$

Question: Can you explain in words what this model description is saying?

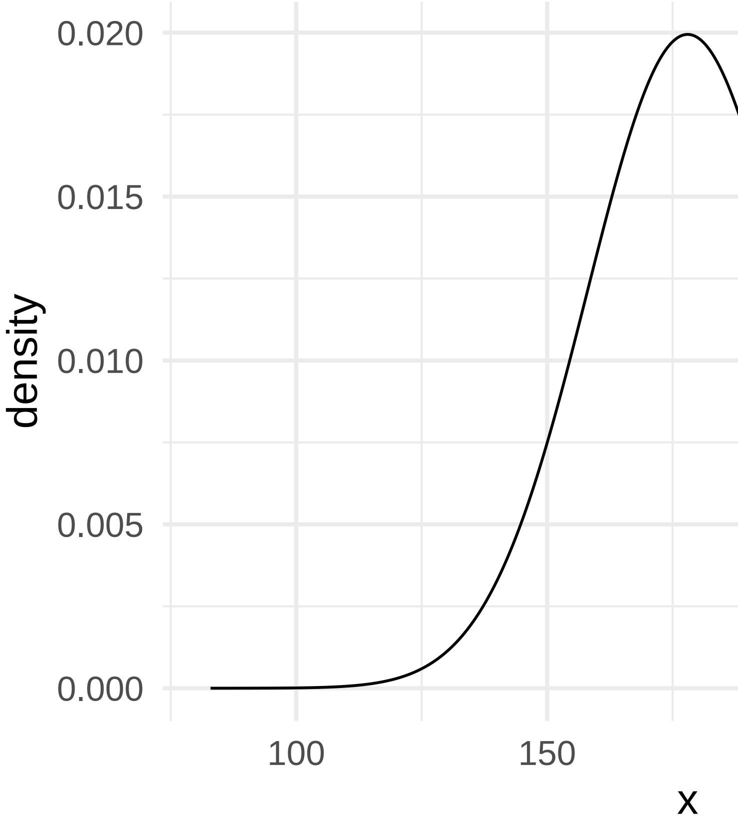
> A person is normally distributed with a mean of 178 cm and standard deviation of 20 cm.
>
> The sigma is uniformly distributed between 0 and 50.

Question: What part tells us about the prior? What part about the likelihood?

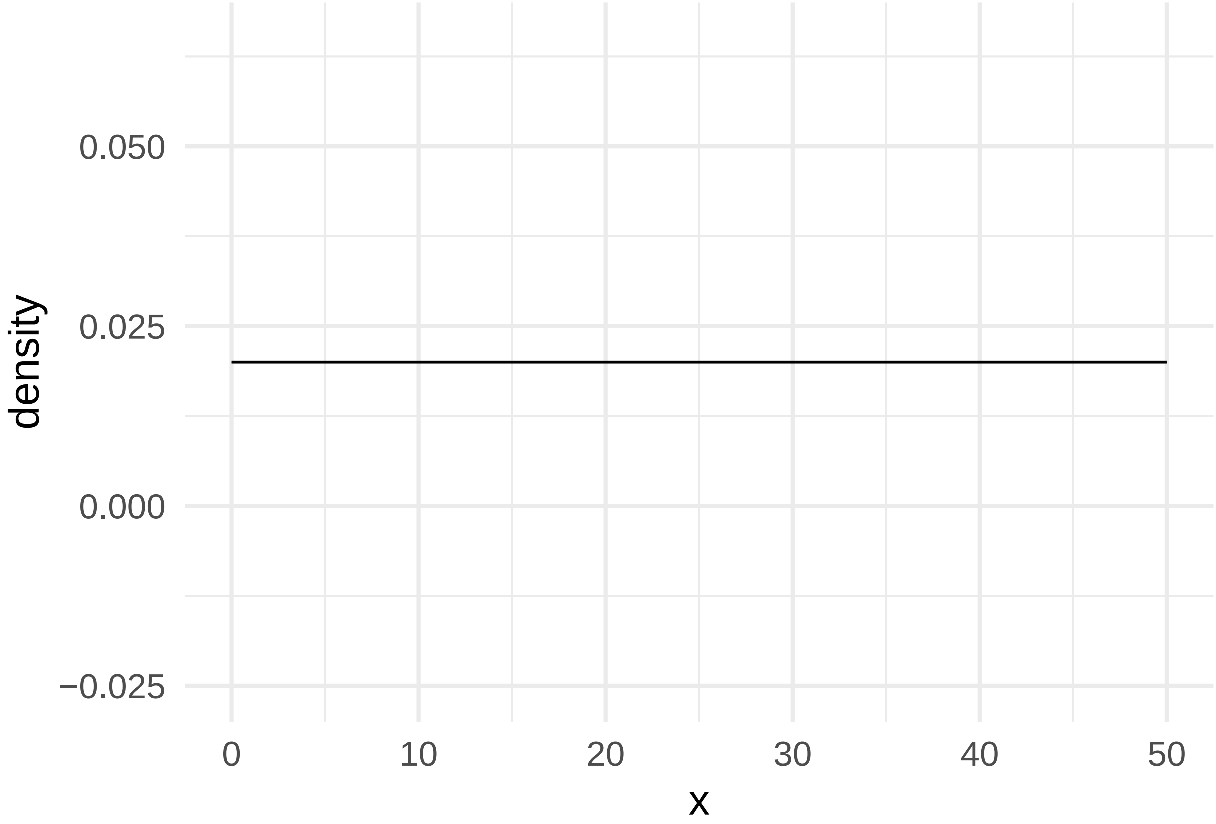
> likelihood is
> $$\text{height}_i \sim \text{Normal}(\mu, \sigma)$$
>
> priors are
>
> $$\mu \sim \text{Normal}(\text{mean} = 178, \text{sd} = 20)$$
>
> $$\sigma \sim \text{Unif}(\text{min} = 0, \text{max} = 50)$$

```r
gf_dist("norm", mean = 178, sd = 20)
```

Task: use `gf_dist()` to draw the two priors. ***Note: there are two now, in our ONE model, because we are estimating more than one parameter!***



```r
gf_dist("unif", min = 0, max = 50)
```



Question: Based on the priors given, what do you think the units of measure of the height data are?

Centimeters.

Question: Why were Normal and Uniform distributions chosen; would you choose the same? (*Advanced add-on: what is tricky about the support of* σ compared to μ? Do you have any related worry about the support of the height data?)

Normal distribution is chosen because the bell shape makes sense for people's heights.

Uniform distribution is chosen because we assume standard deviations are equally likely between 0 and 50.

I would chose normal distribution for μ, but I'm not sure about σ is equally likely between 0 and 50, there are some giants…

I worry that uniform distribution for σ is not accurate, 2 standard deviations is 1 meter.

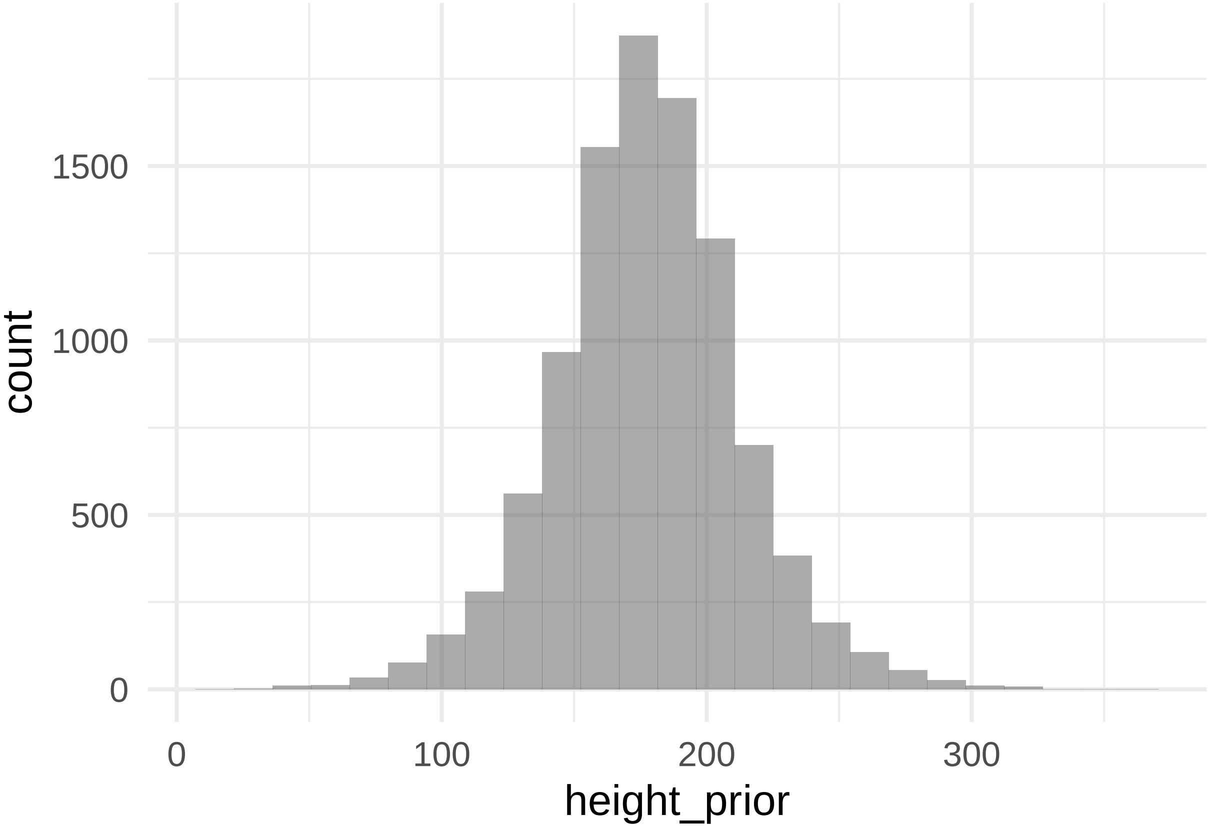
Question: How might you adjust the prior to better reflect your own knowledge?

Maybe change the uniform distribution to (0,15).

```r
mu_prior <- rnorm(10000, mean = 178, sd = 20)
sigma_prior <- runif(10000, min = 0, max = 50)
height_prior <- rnorm(10000, mean = mu_prior, sd = sigma_prior)

gf_histogram(~ height_prior)
```

Task: use `rnorm()` and `runif()` to *simulate* some fake data based on the priors stated above – or your new-and-improved ones! Use `gf_histogram()` to make a histogram of the resulting fake-height-data to check it out. (Remember, this is called a *prior* predictive distribution. . . why is it useful?)



> Prior predictive distribution is useful because we can see the assumptions we have put in to the priors for the parameters of the model and not rely on any observation.

Another Example

Consider a data set based on Tintle et al. 2019, "Evaluating the efficacy of point-of-use water filtration units in Fiji". In this study, water filters were distributed to households in Fiji to provide better access to safer drinking water, and data was collected on money spent on water, and health outcomes, before and after filter distribution.

```r
fiji_water <- read_csv('https://sldr.netlify.app/data/fiji-filters.csv',
                       show_col_types = FALSE)
glimpse(fiji_water)
```

```
## Rows: 1,006
## Columns: 17
## $ household_id            <dbl> 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14,~
## $ town                    <chr> "Nadi", "Nadi", "Nadi", "Nadi", "Nadi", "Nadi"~
## $ time_point              <chr> "Baseline", "Baseline", "Baseline", "Baseline"~
## $ water_source            <chr> "Catchment", "Borehole", "Borehole", "River/cr~
## $ season                  <chr> "Dry", "Rainy", "Rainy", "Dry", "Dry", "Rainy"~
## $ n_adults                <dbl> 2, 2, 2, 1, 5, 4, 1, 1, 2, 1, 3, 1, 4, 2, 2, 4~
## $ n_kids                  <dbl> 0, 3, 2, 1, 2, 1, 1, 0, 0, 0, 0, 0, 0, 2, 1, 2~
## $ n_total                 <dbl> 2, 5, 4, 2, 7, 5, 1, 0, 2, 0, 3, 1, 4, 4, 3, 6~
## $ household_annual_income <dbl> 16750.42273, 34132.59160, 2065.90881, 82.37176~
## $ severe_diarrhea_adults  <chr> "Absent", "Absent", "Absent", "Absent", "Absen~
## $ severe_diarrhea_kids    <chr> NA, "Absent", "Absent", "Absent", "Absent", "A~
## $ diarrhea_adults         <chr> "Absent", "Absent", "Absent", "Absent", "Absen~
## $ diarrhea_kids           <chr> NA, "Absent", "Absent", "Absent", "Absent", "A~
## $ medical_expenses_pp     <dbl> 0.0000, 0.0000, 26.4600, 0.0000, 0.0000, 0.002~
## $ water_expenses_pp       <dbl> 2.19, 5.50, 0.24, 9.45, 0.00, 0.01, 0.00, 0.00~
## $ water_expenses          <dbl> 4.38, 27.50, 0.96, 18.90, 0.00, 0.05, 0.00, 0.~
## $ medical_expenses        <dbl> 0.0000, 0.0000, 105.8400, 0.0000, 0.0000, 0.01~
```

Let's consider modeling the `water_expenses_pp` variable, which is water expenses per person in Fijian dollars.

Task: Without further peeking at the data first – but you can use a web search if you like – set up a model analogous to the height model above, but for the `water_expenses_pp` variable. Write it down in our new notation.

$$\text{waterExpensePpPdfNoUnderscore}_i \sim \text{Normal}(\mu, \sigma)$$

$$\mu \sim \text{Normal}(\text{mean} = 10, \text{sd} = 3)$$

$$\sigma \sim \text{Unif}(\text{min} = 0, \text{max} = 15)$$

More time?

- Consider the same questions I asked you about the height example, but now for the water-cost example.
- Right now we still have *only one* data variable, so none of the causal diagram stuff comes into play yet… can you imagine how it *will* soon come into play?

> We will have use variables like household income, number of adults/children… having more dimensions, more complex models?

What's next?

Clearly, we need to learn to *fit* these new models, which is what we will try next.

If you have extra time, make an attempt to fit the water-cost model using a grid search. Can you identify why it's tricky?

- You will need to know how to compute the **likelihood of one water-cost observation given the model**
- You'll need to determine how to *combine* the likelihoods of all the individual datapoints into a *joint* likelihood of the whole dataset
- You'll need R code that lets you keep track of a big set of candidate values for *more than one* parameter
- (sorry) You'll need to work on the *log*-likelihood scale to avoid numerical problems with the calculations

Help is coming next time!