

Choosing priors Part

Choosing priors Part I

We're going to cover priors in two parts. In this part, we're going to cover some general implications of different prior choices and we're going to talk about priors specifically in the context of Bayes factor. In Part II, we cover prior choice in the context of **Bayesian estimation**. The reasons for making particular choices and the implications of our choices in these two contexts are a little different, so it's best to deal with them separately.

Priors with specific mathematical properties

The first prior choice strategy we'll cover is choosing priors that have particular mathematical properties. One property we might be interested in is choosing priors that are from the same **family** as the posterior. When we do this, the prior is known as a **conjugate prior**.

Ordinarily there aren't **closed form** solutions for calculating posteriors. However, the beauty of using a **conjugate prior** is that this allows us to derive a closed form solution for the posterior. One example of a *conjugate prior* is the **Beta** distribution, which is a conjugate prior when we have a **Binomial** or **Negative binomial** likelihood (like our coin flip example). The resulting posterior would then take the form of a **Beta** distribution.

```
## Warning in grid.Call(C_textBounds, as.graphicsAnnot(x$label), x$x, x$y, :  
## conversion failure on ' ' in 'mbcsToSbcs': dot substituted for <ce>  
  
## Warning in grid.Call(C_textBounds, as.graphicsAnnot(x$label), x$x, x$y, :  
## conversion failure on ' ' in 'mbcsToSbcs': dot substituted for <b8>  
  
## Warning in grid.Call(C_textBounds, as.graphicsAnnot(x$label), x$x, x$y, :  
## conversion failure on ' ' in 'mbcsToSbcs': dot substituted for <ce>  
  
## Warning in grid.Call(C_textBounds, as.graphicsAnnot(x$label), x$x, x$y, :  
## conversion failure on ' ' in 'mbcsToSbcs': dot substituted for <b8>  
  
## Warning in grid.Call(C_textBounds, as.graphicsAnnot(x$label), x$x, x$y, :  
## conversion failure on ' ' in 'mbcsToSbcs': dot substituted for <ce>  
  
## Warning in grid.Call(C_textBounds, as.graphicsAnnot(x$label), x$x, x$y, :  
## conversion failure on ' ' in 'mbcsToSbcs': dot substituted for <b8>  
  
## Warning in grid.Call(C_textBounds, as.graphicsAnnot(x$label), x$x, x$y, :  
## conversion failure on ' ' in 'mbcsToSbcs': dot substituted for <ce>  
  
## Warning in grid.Call(C_textBounds, as.graphicsAnnot(x$label), x$x, x$y, :  
## conversion failure on ' ' in 'mbcsToSbcs': dot substituted for <b8>  
  
## Warning in grid.Call(C_textBounds, as.graphicsAnnot(x$label), x$x, x$y, :  
## conversion failure on ' ' in 'mbcsToSbcs': dot substituted for <ce>
```

```
## Warning in grid.Call(C_textBounds, as.graphicsAnnot(x$label), x$x, x$y, :
## conversion failure on ' ' in 'mbcsToSbcs': dot substituted for <b8>

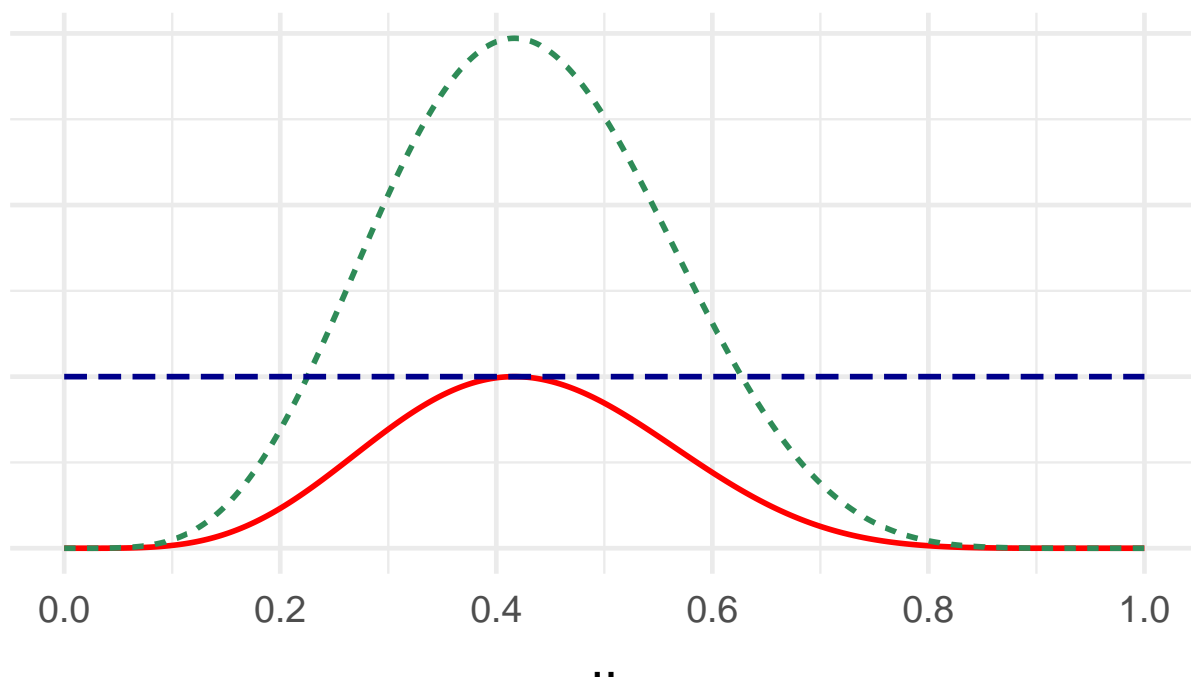
## Warning in grid.Call(C_textBounds, as.graphicsAnnot(x$label), x$x, x$y, :
## conversion failure on ' ' in 'mbcsToSbcs': dot substituted for <ce>

## Warning in grid.Call(C_textBounds, as.graphicsAnnot(x$label), x$x, x$y, :
## conversion failure on ' ' in 'mbcsToSbcs': dot substituted for <b8>

## Warning in grid.Call.graphics(C_text, as.graphicsAnnot(x$label), x$x, x$y, :
## conversion failure on ' ' in 'mbcsToSbcs': dot substituted for <ce>

## Warning in grid.Call.graphics(C_text, as.graphicsAnnot(x$label), x$x, x$y, :
## conversion failure on ' ' in 'mbcsToSbcs': dot substituted for <b8>
```

— posterior — prior — likelihood



Priors that reflect ignorance

The next thing that we might want to take into account when choosing a prior is choosing a prior that doesn't commit us to any particular value of the parameter. But doing this might not be as straightforward as it initially seems. We might think that we can just weight each possible value of the parameter equally. However, it turns out that this isn't always the best strategy. To see why, we'll turn to an example.

For our coin flipping example we can calculate the maximum likelihood estimate for the coin bias given our data. The formula for this is given as:

$$\hat{\theta}_{\text{mle}} = \frac{x}{n},$$

where x is the number of heads in n flips. This value is the same as the **mean** of our likelihood. We can think of this as our estimate of the parameter without taking account any prior information.

Next we can incorporate a prior and calculate a posterior. We'll use a **Beta** prior, because it'll allow us to easily calculate the posterior. For a prior of the form $Beta(\alpha, \beta)$, the posterior for x heads in n flips is given as $Beta(\alpha + x, \beta + n - x)$. From this, we can now work out the **mean of the posterior** as:

$$\hat{\theta}_{\text{bayes}} = \frac{\alpha + x}{\alpha + \beta + n}.$$

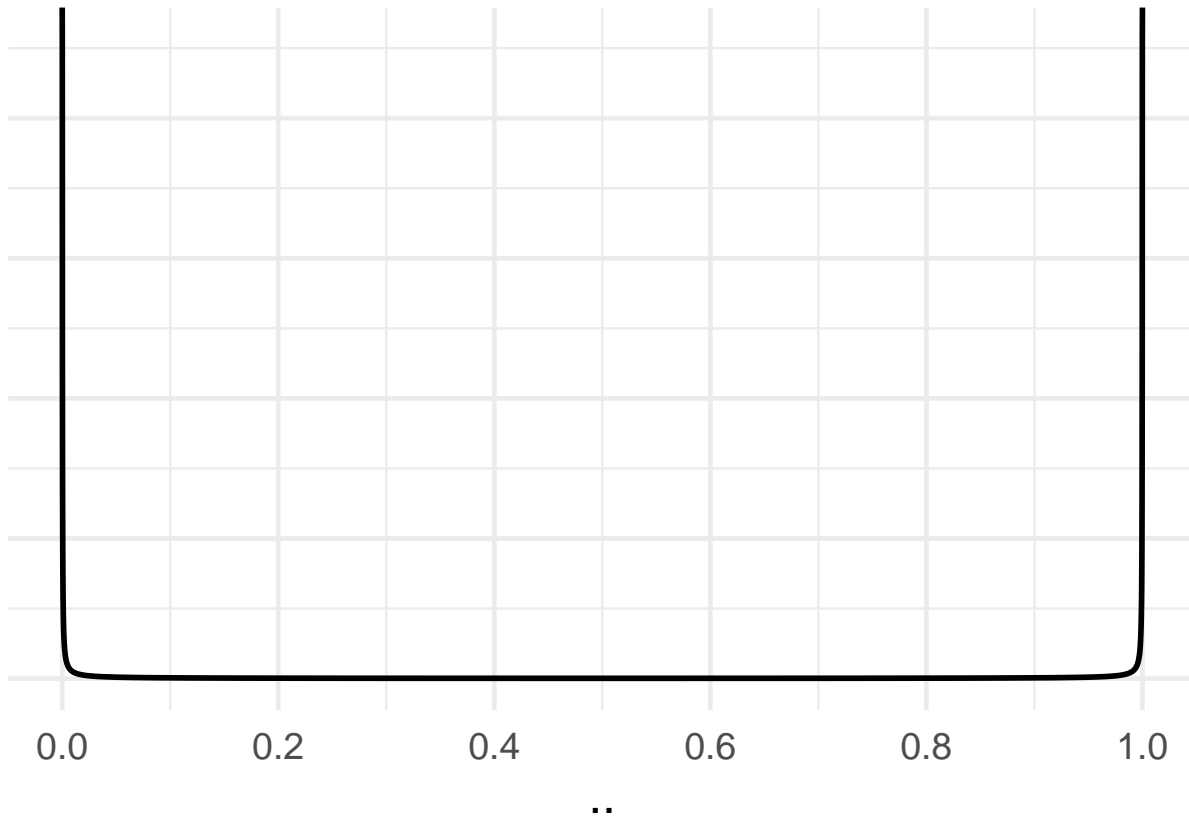
From this we can see that for any value of $\alpha > 0 < \beta$ the value of $\hat{\theta}_{\text{bayes}}$ and $\hat{\theta}_{\text{mle}}$ will be different.

This results in a rather unintuitive conclusion. The uniform prior $Beta(1,1)$ has a bigger influence (is in a sense more informative) than the prior $Beta(0,0)$ (this prior is called Haldane’s prior). Technically, $Beta(0,0)$ is undefined, but we can examine the prior $Beta(\epsilon,\epsilon)$, where ϵ is an arbitrarily small number.

The mean of the likelihood is 0.42

The mean of the posterior is 0.42

```
## Warning in grid.Call(C_textBounds, as.graphicsAnnot(x$label), x$x, x$y, :  
## conversion failure on ' ' in 'mbcsToSbcs': dot substituted for <ce>  
  
## Warning in grid.Call(C_textBounds, as.graphicsAnnot(x$label), x$x, x$y, :  
## conversion failure on ' ' in 'mbcsToSbcs': dot substituted for <b8>  
  
## Warning in grid.Call(C_textBounds, as.graphicsAnnot(x$label), x$x, x$y, :  
## conversion failure on ' ' in 'mbcsToSbcs': dot substituted for <ce>  
  
## Warning in grid.Call(C_textBounds, as.graphicsAnnot(x$label), x$x, x$y, :  
## conversion failure on ' ' in 'mbcsToSbcs': dot substituted for <b8>  
  
## Warning in grid.Call(C_textBounds, as.graphicsAnnot(x$label), x$x, x$y, :  
## conversion failure on ' ' in 'mbcsToSbcs': dot substituted for <ce>  
  
## Warning in grid.Call(C_textBounds, as.graphicsAnnot(x$label), x$x, x$y, :  
## conversion failure on ' ' in 'mbcsToSbcs': dot substituted for <b8>  
  
## Warning in grid.Call(C_textBounds, as.graphicsAnnot(x$label), x$x, x$y, :  
## conversion failure on ' ' in 'mbcsToSbcs': dot substituted for <ce>  
  
## Warning in grid.Call(C_textBounds, as.graphicsAnnot(x$label), x$x, x$y, :  
## conversion failure on ' ' in 'mbcsToSbcs': dot substituted for <b8>  
  
## Warning in grid.Call(C_textBounds, as.graphicsAnnot(x$label), x$x, x$y, :  
## conversion failure on ' ' in 'mbcsToSbcs': dot substituted for <ce>  
  
## Warning in grid.Call(C_textBounds, as.graphicsAnnot(x$label), x$x, x$y, :  
## conversion failure on ' ' in 'mbcsToSbcs': dot substituted for <b8>  
  
## Warning in grid.Call(C_textBounds, as.graphicsAnnot(x$label), x$x, x$y, :  
## conversion failure on ' ' in 'mbcsToSbcs': dot substituted for <ce>  
  
## Warning in grid.Call(C_textBounds, as.graphicsAnnot(x$label), x$x, x$y, :  
## conversion failure on ' ' in 'mbcsToSbcs': dot substituted for <b8>
```



From the plot we can see that this prior splits nearly all of its weight on the extreme of $\theta = 0$ and $\theta = 1$. In short, a prior that seems to make the least commitments has a bigger influence than the prior that seemingly makes the strongest commitments.

If we think about it, we'll also see that a prior like this makes a lot of sense if we want to *quantify* our uncertainty. To understand this, let's plot the likelihood functions for various observations. We'll pick two extreme observations (1 head in 100 flips and 99 heads in 100 flips) and one middle of the road observation (50 heads in 100 flips). The plots will allow us to see the range of plausible values for the coin bias given the observation.

```
## Warning in grid.Call(C_textBounds, as.graphicsAnnot(x$label), x$x, x$y, :
## conversion failure on ' ' in 'mbcsToSbcs': dot substituted for <ce>

## Warning in grid.Call(C_textBounds, as.graphicsAnnot(x$label), x$x, x$y, :
## conversion failure on ' ' in 'mbcsToSbcs': dot substituted for <b8>

## Warning in grid.Call(C_textBounds, as.graphicsAnnot(x$label), x$x, x$y, :
## conversion failure on ' ' in 'mbcsToSbcs': dot substituted for <ce>

## Warning in grid.Call(C_textBounds, as.graphicsAnnot(x$label), x$x, x$y, :
## conversion failure on ' ' in 'mbcsToSbcs': dot substituted for <b8>

## Warning in grid.Call(C_textBounds, as.graphicsAnnot(x$label), x$x, x$y, :
## conversion failure on ' ' in 'mbcsToSbcs': dot substituted for <ce>

## Warning in grid.Call(C_textBounds, as.graphicsAnnot(x$label), x$x, x$y, :
## conversion failure on ' ' in 'mbcsToSbcs': dot substituted for <b8>

## Warning in grid.Call(C_textBounds, as.graphicsAnnot(x$label), x$x, x$y, :
## conversion failure on ' ' in 'mbcsToSbcs': dot substituted for <ce>

## Warning in grid.Call(C_textBounds, as.graphicsAnnot(x$label), x$x, x$y, :
```

[illegible]

```
## Warning in grid.Call.graphics(C_text, as.graphicsAnnot(x$label), x$x, x$y, :
## conversion failure on ' ' in 'mbcsToSbcs': dot substituted for <b8>

## Warning in grid.Call(C_textBounds, as.graphicsAnnot(x$label), x$x, x$y, :
## conversion failure on ' ' in 'mbcsToSbcs': dot substituted for <ce>

## Warning in grid.Call(C_textBounds, as.graphicsAnnot(x$label), x$x, x$y, :
## conversion failure on ' ' in 'mbcsToSbcs': dot substituted for <b8>

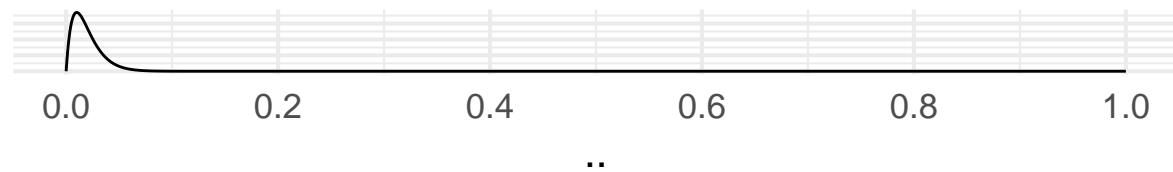
## Warning in grid.Call(C_textBounds, as.graphicsAnnot(x$label), x$x, x$y, :
## conversion failure on ' ' in 'mbcsToSbcs': dot substituted for <ce>

## Warning in grid.Call(C_textBounds, as.graphicsAnnot(x$label), x$x, x$y, :
## conversion failure on ' ' in 'mbcsToSbcs': dot substituted for <b8>

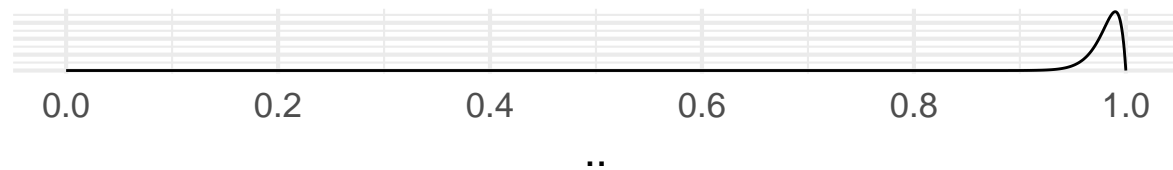
## Warning in grid.Call.graphics(C_text, as.graphicsAnnot(x$label), x$x, x$y, :
## conversion failure on ' ' in 'mbcsToSbcs': dot substituted for <ce>

## Warning in grid.Call.graphics(C_text, as.graphicsAnnot(x$label), x$x, x$y, :
## conversion failure on ' ' in 'mbcsToSbcs': dot substituted for <b8>
```

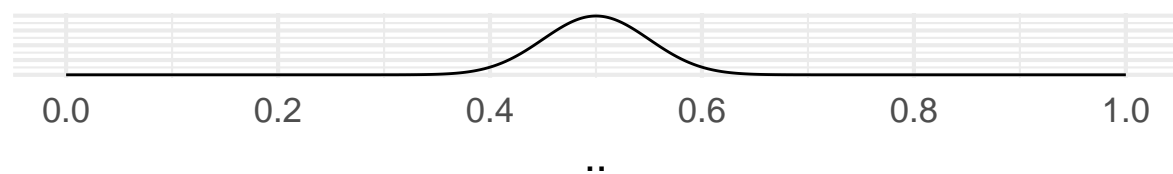
Likelihood for 1 heads in 100 flips



Likelihood for 99 heads in 100 flips



Likelihood for 50 heads in 100 flips



Comparing the plots for the extreme observations to the plot for the middling one we can see that there's a larger range of plausible values of the bias when the observation is somewhere near the middle of the range. When the observation is at the extreme there's only a very narrow range of values of the bias that could plausibly produce the observation. In a sense, we're less uncertain about the extremes and more uncertain in the middle. Quantifying this uncertainty means putting more prior mass at the extremes (where we're certain) and less in the middle (where more uncertain about the relation between parameters and observations).

There's also another hurdle to coming up with a prior that is “uninformative”, and this has to do with how we parametrise problems. Let's say that I'm interested in computing a posterior estimate about the probability π of a coin landing heads. I use a uniform prior so that for any value of θ_n , $p(\theta_1) = p(\theta_2)$. After

working out my posterior I give you the estimate. You, however, are not interested in the probability π . Instead, you're interested in the log odds ϕ , so you transform π into ϕ using $\phi = \log \frac{\pi}{1-\pi}$.

Now let's say another another statistician is analysing the same data, but they've decided to work in ϕ space from the get go. So to analyse the data, they first convert it into ϕ , construct a prior so that for any θ_n , $p(\theta_1) = p(\theta_2)$. They then take the data, and the prior, to compute the posterior.

It turns out that when this is done, the two estimates—the one converted from π to ϕ and the one done wholly in ϕ won't agree. That is because what is uniform for the one isn't uniform for the other. So which way of asking the question is the right one? Do you ask about π or ϕ ? The answer is that there is, and there shouldn't, be a right way to ask the question. If the parameters are equivalent (there's a one to one translation from one to the other) and to use a non-informative prior then your choice of parameterisation shouldn't inform your posterior. There is, however, a way to construct prior that will allow you to get the same answer for the two questions. This is by using **Jeffreys' rules**. The resulting prior is often called a **Jeffreys' prior**. For our coin flipping example, the prior constructed from the parameter space of the probability of show heads turns out to be $Beta(\frac{1}{2}, \frac{1}{2})$.

This property of Jeffreys’ priors is certainly useful. However, there’s no free lunch in statistics, and using Jeffreys’ priors involves a cost. The cost is that to construct a **Jeffreys’** prior you need to take into account the **universe of possible results**. That is, the method of constructing Jeffrey’s priors takes into account data that only might occur. As a consequence, the Jeffreys prior for our two coin flipping example—flipping for 10 flips and flipping until 2 tails—will be different. The Jeffreys prior for n flips is $Beta(\frac{1}{2}, \frac{1}{2})$. While the prior for the flipping until x tails is $Beta(1, \frac{1}{2})$.

```
## Warning in grid.Call(C_textBounds, as.graphicsAnnot(x$label), x$x, x$y, :  
## conversion failure on 'p()' in 'mbsToSbcs': dot substituted for <ce>  
  
## Warning in grid.Call(C_textBounds, as.graphicsAnnot(x$label), x$x, x$y, :  
## conversion failure on 'p()' in 'mbsToSbcs': dot substituted for <b8>  
  
## Warning in grid.Call(C_textBounds, as.graphicsAnnot(x$label), x$x, x$y, :  
## conversion failure on 'p()' in 'mbsToSbcs': dot substituted for <ce>  
  
## Warning in grid.Call(C_textBounds, as.graphicsAnnot(x$label), x$x, x$y, :  
## conversion failure on 'p()' in 'mbsToSbcs': dot substituted for <b8>  
  
## Warning in grid.Call(C_textBounds, as.graphicsAnnot(x$label), x$x, x$y, :  
## conversion failure on 'p()' in 'mbsToSbcs': dot substituted for <ce>  
  
## Warning in grid.Call(C_textBounds, as.graphicsAnnot(x$label), x$x, x$y, :  
## conversion failure on 'p()' in 'mbsToSbcs': dot substituted for <b8>  
  
## Warning in grid.Call(C_textBounds, as.graphicsAnnot(x$label), x$x, x$y, :  
## conversion failure on ' ' in 'mbsToSbcs': dot substituted for <ce>  
  
## Warning in grid.Call(C_textBounds, as.graphicsAnnot(x$label), x$x, x$y, :  
## conversion failure on ' ' in 'mbsToSbcs': dot substituted for <b8>  
  
## Warning in grid.Call(C_textBounds, as.graphicsAnnot(x$label), x$x, x$y, :  
## conversion failure on ' ' in 'mbsToSbcs': dot substituted for <ce>  
  
## Warning in grid.Call(C_textBounds, as.graphicsAnnot(x$label), x$x, x$y, :  
## conversion failure on ' ' in 'mbsToSbcs': dot substituted for <b8>  
  
## Warning in grid.Call(C_textBounds, as.graphicsAnnot(x$label), x$x, x$y, :  
## conversion failure on ' ' in 'mbsToSbcs': dot substituted for <ce>  
  
## Warning in grid.Call(C_textBounds, as.graphicsAnnot(x$label), x$x, x$y, :  
## conversion failure on 'p()' in 'mbsToSbcs': dot substituted for <ce>
```

[illegible]


```

## Warning in grid.Call.graphics(C_text, as.graphicsAnnot(x$label), x$x, x$y, :
## conversion failure on 'p()' in 'mbscsToSbcs': dot substituted for <ce>

## Warning in grid.Call.graphics(C_text, as.graphicsAnnot(x$label), x$x, x$y, :
## conversion failure on 'p()' in 'mbscsToSbcs': dot substituted for <b8>

## Warning in grid.Call(C_textBounds, as.graphicsAnnot(x$label), x$x, x$y, :
## conversion failure on ' ' in 'mbscsToSbcs': dot substituted for <ce>

## Warning in grid.Call(C_textBounds, as.graphicsAnnot(x$label), x$x, x$y, :
## conversion failure on ' ' in 'mbscsToSbcs': dot substituted for <b8>

## Warning in grid.Call(C_textBounds, as.graphicsAnnot(x$label), x$x, x$y, :
## conversion failure on ' ' in 'mbscsToSbcs': dot substituted for <ce>

## Warning in grid.Call(C_textBounds, as.graphicsAnnot(x$label), x$x, x$y, :
## conversion failure on ' ' in 'mbscsToSbcs': dot substituted for <b8>

## Warning in grid.Call.graphics(C_text, as.graphicsAnnot(x$label), x$x, x$y, :
## conversion failure on ' ' in 'mbscsToSbcs': dot substituted for <ce>

## Warning in grid.Call.graphics(C_text, as.graphicsAnnot(x$label), x$x, x$y, :
## conversion failure on ' ' in 'mbscsToSbcs': dot substituted for <b8>

## Warning in grid.Call(C_textBounds, as.graphicsAnnot(x$label), x$x, x$y, :
## conversion failure on 'p()' in 'mbscsToSbcs': dot substituted for <ce>

## Warning in grid.Call(C_textBounds, as.graphicsAnnot(x$label), x$x, x$y, :
## conversion failure on 'p()' in 'mbscsToSbcs': dot substituted for <b8>

## Warning in grid.Call(C_textBounds, as.graphicsAnnot(x$label), x$x, x$y, :
## conversion failure on 'p()' in 'mbscsToSbcs': dot substituted for <ce>

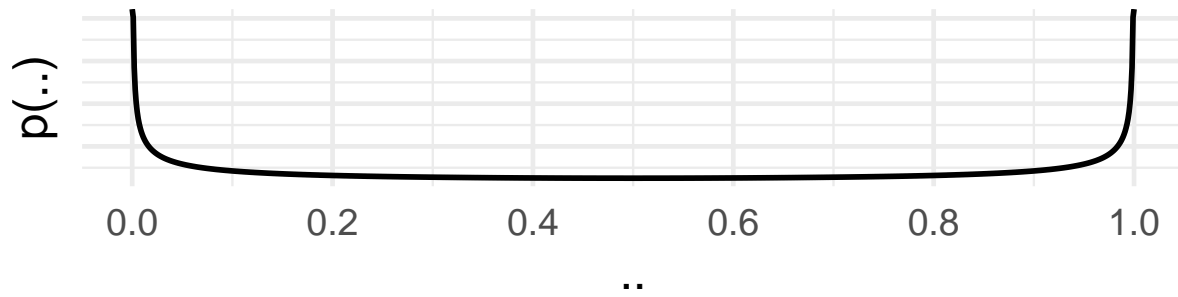
## Warning in grid.Call(C_textBounds, as.graphicsAnnot(x$label), x$x, x$y, :
## conversion failure on 'p()' in 'mbscsToSbcs': dot substituted for <b8>

## Warning in grid.Call.graphics(C_text, as.graphicsAnnot(x$label), x$x, x$y, :
## conversion failure on 'p()' in 'mbscsToSbcs': dot substituted for <ce>

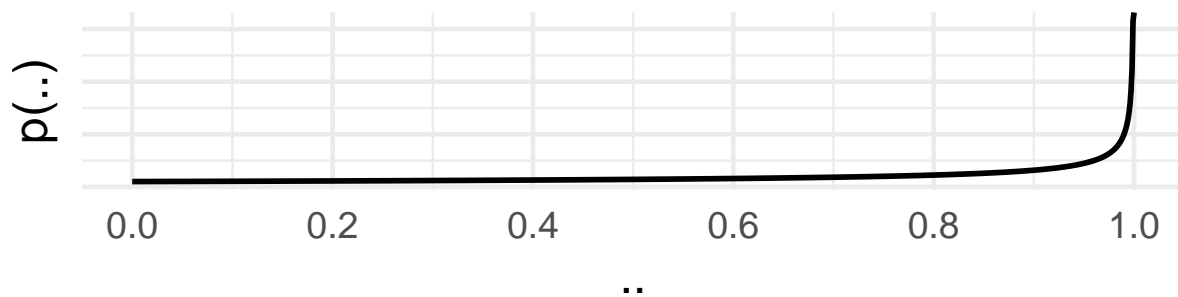
## Warning in grid.Call.graphics(C_text, as.graphicsAnnot(x$label), x$x, x$y, :
## conversion failure on 'p()' in 'mbscsToSbcs': dot substituted for <b8>

```

Jeffreys prior for binomial



Jeffreys prior for negative binomial



Priors that reflect our beliefs about the world.

Finally, you can choose priors that reflect your beliefs about the world. In the coin flipping example, this would involve picking a prior that reflects what you **believe** about fair and biased coins: Do biased coins show heads more often, tails more often, extreme values more often, and so on. When we're using **Bayes factors** in experimental psychology, this would mean picking priors that describe our scientific theories—that is, picking **priors** that makes predictions about data that are consistent with the predictions about data that our theories make.

Do this is hard, but we'll cover some strategies when we get to actually computing **Bayes factors** for some problems. But it's also important to remember that you don't need to restrict yourself to one model. You can have many models and you can see how changes in your model change your inferences. This is sometimes captured by the idea of *robustness regions*. However, it's also important to note that if different **reasonable** priors lead to different conclusions then there might not be **one answer** to your problem. That is, it important to recognise that uncertainty in the nature of your models can lead to uncertainty in your conclusions.