

Homework #3

Chapters 2.1, 2.2, 2.3 and 2.4

Brynjólfur Gauti Guðrúnar Jónsson

```
```{r}
#| output: hide

library(tidyr)
library(dplyr, warn.conflicts = FALSE)
library(ggplot2)
library(patchwork)
library(gt)
library(invgamma)
theme_set(theme_classic())
```
```

Chapter 2.4

Exercise 6

An assembly line relies on accurate measurements from an image-recognition algorithm at the first stage of the process. It is known that the algorithm is unbiased, so assume that measurements follow a normal distribution with mean zero,

$$Y_i | \sigma^2 \stackrel{\text{iid}}{\sim} \text{Normal}(0, \sigma^2).$$

Some errors are permissible, but if σ exceeds the threshold c then the algorithm must be replaced.

You make $n = 20$ measurements and observe

$$\sum_{i=1}^n Y_i = -2 \quad \text{and} \quad \sum_{i=1}^n Y_i^2 = 15,$$

and conduct a Bayesian analysis with $\text{InvGamma}(a, b)$ prior. compute the posterior probability that $\sigma > c$ for:

a. $c = 1$ and $a = b = 0.1$

Solution:

Our likelihood is

$$\begin{aligned} f(\mathbf{Y}|\sigma^2) &\propto \prod_{i=1}^n \frac{1}{\sigma} \exp \left[-\frac{(Y_i - \mu)^2}{2\sigma^2} \right] \\ &\propto (\sigma^2)^{-n/2} \exp \left(-\frac{\text{SSE}}{2\sigma^2} \right). \end{aligned}$$

In our case, we can write out SSE as

$$\begin{aligned} \text{SSE} &= \sum_{i=1}^n (Y_i - \mu)^2 \\ &= \sum_{i=1}^n Y_i^2 - 2Y_i\mu + \mu^2 \\ &= \sum_{i=1}^n Y_i^2, \quad \mu = 0. \end{aligned}$$

Our inverse gamma prior is

$$\pi(\sigma^2) \propto (\sigma^2)^{-(a+1)} \exp \left(-\frac{b}{\sigma^2} \right).$$

Combining our likelihood and prior we get the posterior

$$\begin{aligned} p(\sigma^2|\mathbf{Y}) &\propto f(\mathbf{Y}|\sigma^2)\pi(\sigma^2) \\ &\propto (\sigma^2)^{-(A+1)} \exp \left(-\frac{B}{\sigma^2} \right), \end{aligned}$$

where $A = n/2 + a$ and $B = \text{SSE}/2 + b$, and we therefore see that

$$\sigma^2 | \mathbf{Y} \sim \text{InvGamma}(A, B).$$

By using the right values for n , SSE , a , b , and c we can easily use R's inbuilt `pgamma` function to answer the question. To use the `pgamma` function, we just need to remember that if

$$X \sim \text{Gamma}(\alpha, \beta),$$

then

$$\frac{1}{X} \sim \text{InvGamma}(\alpha, \beta).$$

Thus, since σ^2 is assumed to follow an inverse gamma distribution we can answer the question by calculating

$$P\left(\frac{1}{\sigma^2} < \frac{1}{c^2}\right).$$

```

```{r}
n <- 20
SSE <- 15
c <- 1
a <- 0.1
b <- 0.1

A <- n/2 + a
B <- SSE/2 + b

pgamma(1/c^2, shape = A, rate = B)
```

```

```
[1] 0.2249838
```

We could also use the `pinvgamma` function from the `invgamma` package

```
1 - pinvgamma(c^2, shape = A, rate = B)
```

```
[1] 0.2249838
```

```

plot_solution <- function(A, B, c) {
  p1 <- ggplot() +
    stat_function(
      geom = "area",
      fun = dinvgamma,
      xlim = c(0, c),
      args = list(shape = A, rate = B),
      alpha = 0.4
    ) +
    stat_function(
      geom = "area",
      fun = dinvgamma,
      xlim = c(c, qinvgamma(0.999, shape = A, rate = B)),
      args = list(shape = A, rate = B)
    ) +
    annotate(
      geom = "text",
      x = 1.1 * c + 0.1,
      y = dgamma(1/c, shape = A, rate = B) + 0.05,
      label = expression(sigma>c)
    ) +
    coord_cartesian(expand = FALSE) +
    labs(
      subtitle = "Inverse gamma",
      x = expression(sigma)
    )

  p2 <- ggplot() +
    stat_function(
      geom = "area",
      fun = dgamma,
      xlim = c(0, 1/c),
      args = list(shape = A, rate = B)
    ) +
    stat_function(
      geom = "area",
      fun = dgamma,
      xlim = c(1/c, qgamma(0.999, shape = A, rate = B)),
      args = list(shape = A, rate = B),
      alpha = 0.4
    ) +

```

```

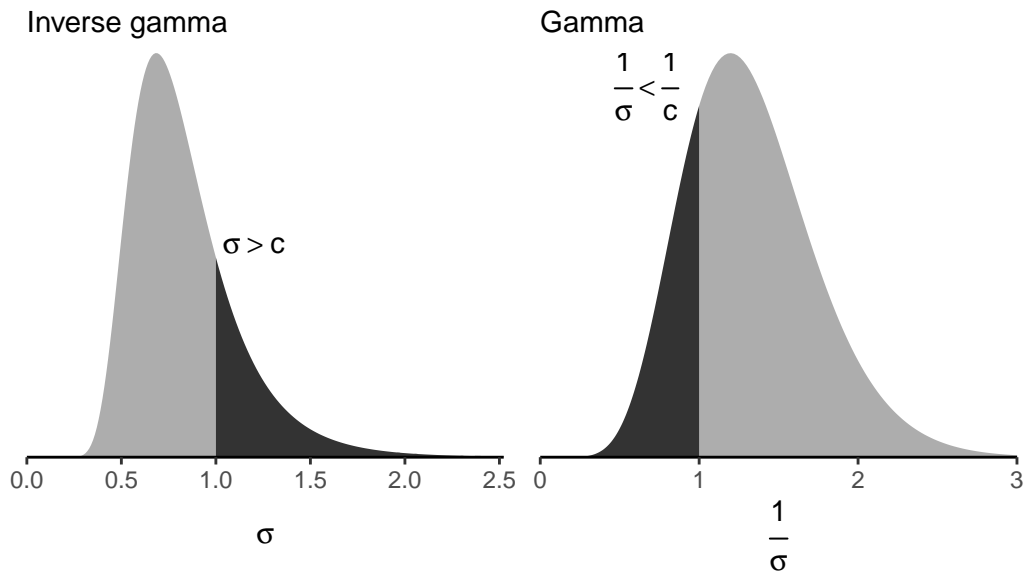
    annotate(
      geom = "text",
      x = 1/ (c * 1.3) - 0.1,
      y = dgamma(1/c, shape = A, rate = B) + 0.05,
      label = expression(frac(1,sigma)<frac(1,c))
    ) +
    coord_cartesian(expand = FALSE) +
    labs(
      subtitle = "Gamma",
      x = expression(frac(1,sigma))
    )

p1 + p2 +
  plot_annotation(
    title = "Comparing solutions using the Gamma or Inverse Gamma"
  ) &
  theme(
    axis.text.y = element_blank(),
    axis.ticks.y = element_blank(),
    axis.title.y = element_blank(),
    axis.line.y = element_blank()
  )
}

plot_solution(A, B, c)

```

Comparing solutions using the Gamma or Inverse Gamma



b. $c = 1$ and $a = b = 1.0$

Solution:

```

```{r}
n <- 20
SSE <- 15
c <- 1
a <- 1
b <- 1

A <- n/2 + a
B <- SSE/2 + b

pgamma(1/c, shape = A, rate = B)
```

```

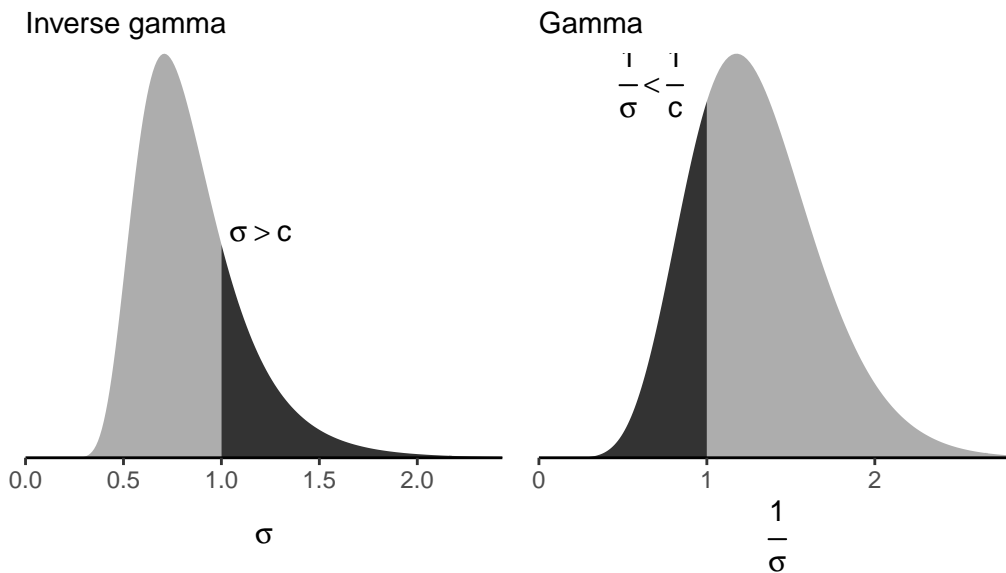
[1] 0.236638

```
1 - pinvgamma(c, shape = A, rate = B)
```

```
[1] 0.236638
```

```
plot_solution(A, B, c)
```

Comparing solutions using the Gamma or Inverse Gamma



c. $c = 2$ and $a = b = 0.1$

Solution:

```
```\{r\}  
n <- 20
SSE <- 15
c <- 2
a <- 0.1
b <- 0.1
```

```

A <- n/2 + a
B <- SSE/2 + b

pgamma(1/c, shape = A, rate = B)
```

```

[1] 0.005214567

```

1 - pinvgamma(c, shape = A, rate = B)

```

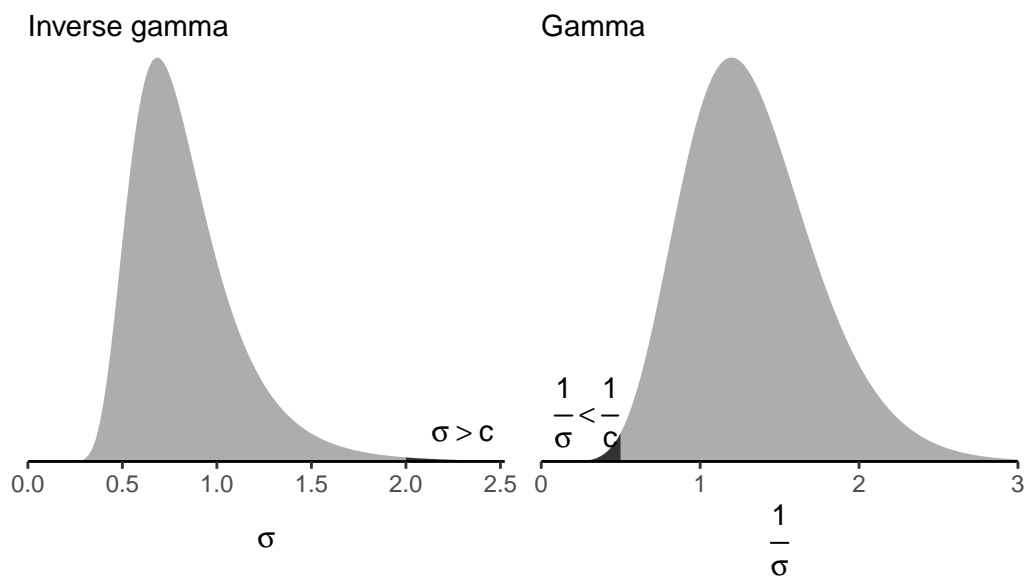
[1] 0.005214567

```

plot_solution(A, B, c)

```

Comparing solutions using the Gamma or Inverse Gamma



d. $c = 2$ and $a = b = 1.0$

Solution:


```

 $\sigma$ 
n <- 20
SSE <- 15
c <- 2
a <- 1
b <- 1

A <- n/2 + a
B <- SSE/2 + b

pgamma(1/c, shape = A, rate = B)

```

[1] 0.004434094

```

1 - pinvgamma(c, shape = A, rate = B)

```

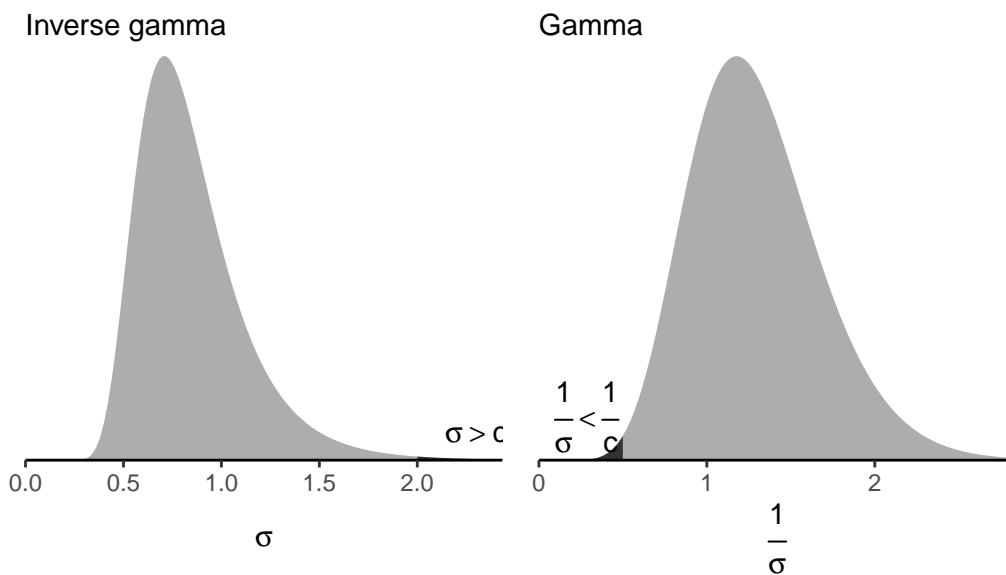
[1] 0.004434094

```

plot_solution(A, B, c)

```

Comparing solutions using the Gamma or Inverse Gamma



Exercise 16

Say $Y|\lambda \sim \text{Gamma}(1, \lambda)$.

- a. Derive and plot the Jeffreys' prior for λ
-

Solution: Jeffreys' prior is

$$\pi(\theta) \propto \sqrt{I(\theta)},$$

where $I(\theta)$ is the expected Fisher information, defined as

$$I(\theta) = -E \left(\frac{d^2 \log f(Y|\theta)}{d\theta^2} \right).$$

The PDF of the Gamma distribution can be written

$$f(x|a, b) = \frac{b^a}{\Gamma(a)} x^{a-1} \exp(-xb).$$

In our case, $a = 1$ and $b = \lambda$

$$\begin{aligned} f(Y|\lambda) &= \frac{\lambda^1}{\Gamma(1)} Y^0 \exp(-\lambda Y) \\ &= \lambda \exp(-\lambda Y). \end{aligned}$$

Taking the log we get

$$\begin{aligned} l(Y|\theta) &= \log f(Y|\lambda) \\ &= \log \lambda - \lambda Y \\ \frac{dl}{d\lambda} &= \frac{1}{\lambda} - Y \\ \frac{dl^2}{d\lambda^2} &= -\frac{1}{\lambda^2} \\ &= -I(\lambda). \end{aligned}$$

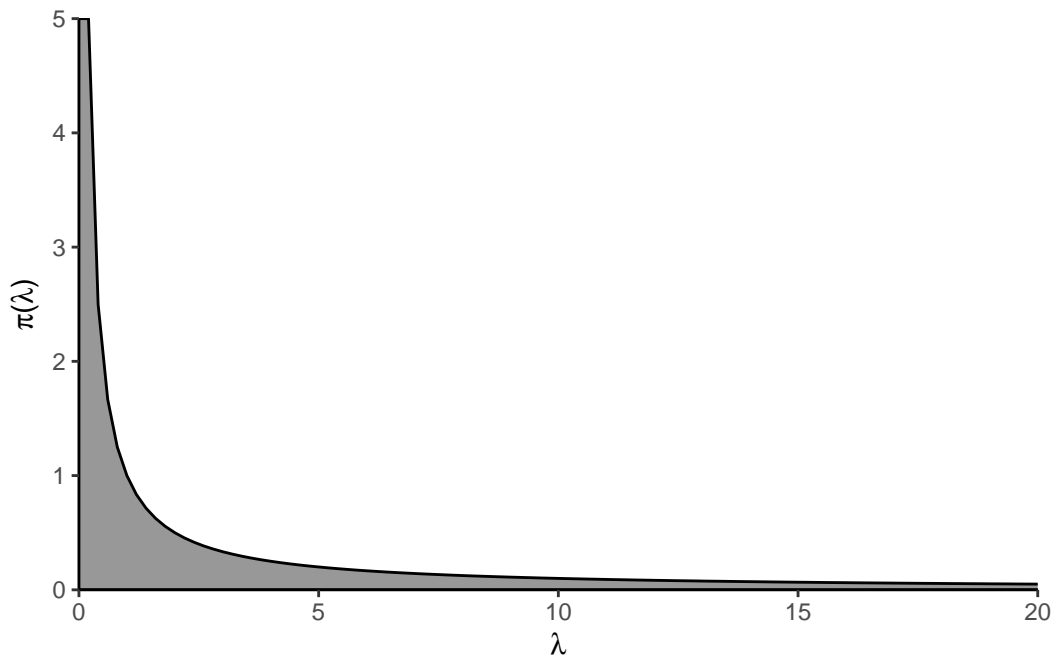
We thus see that

$$\begin{aligned}
 \pi(\theta) &= \sqrt{I(\theta)} \\
 &= \sqrt{\frac{1}{\lambda^2}} \\
 &= \frac{1}{\lambda}
 \end{aligned}$$

```

ggplot() +
  stat_function(
    geom = "area",
    fun = function(x) 1 / x,
    xlim = c(0, 20),
    col = "black",
    alpha = 0.5
  ) +
  coord_cartesian(expand = FALSE) +
  labs(
    x = expression(lambda),
    y = expression(paste(pi, "(", lambda, ")"))
  )

```



b. Is this prior proper?

Solution: This is not a proper prior since it does not integrate to one.

c. Derive the posterior and give conditions on Y to ensure it is proper.

Solution:

Our posterior is

$$\begin{aligned} p(\lambda|Y) &= \frac{f(\mathbf{y}|\lambda)\pi(\lambda)}{\int_0^\infty f(\mathbf{y}|\lambda)\pi(\lambda)d\lambda} \\ &= \frac{\lambda \exp(-\lambda Y) \cdot \frac{1}{\lambda}}{\int_0^\infty \lambda \exp(-\lambda Y) \cdot \frac{1}{\lambda} d\lambda} \\ &= \frac{\exp(-\lambda Y)}{\int_0^\infty \exp(-\lambda Y) d\lambda} \end{aligned}$$

The indefinite integral of $\exp(-\lambda Y)$ is

$$\int \exp(-\lambda Y) = -\frac{1}{Y} \exp(-\lambda Y).$$

If $Y = 0$, then our posterior distribution will not integrate to one and thus will not be proper. We therefore require that $Y > 0$.

Exercise 18

The data in the table below are the result of a survey of commuters in 10 counties likely to be affected by a proposed addition of a high occupancy vehicle (HOV) lane.

```
d <- tribble(
  ~County, ~Approve, ~Disapprove,
  1, 12, 50,
  2, 90, 150,
  3, 80, 63,
  4, 5, 10,
  5, 63, 63,
  6, 15, 8,
  7, 67, 56,
  8, 22, 19,
  9, 56, 63,
  10, 33, 19
)

d |>
  gt()
```

| County | Approve | Disapprove |
|--------|---------|------------|
| 1 | 12 | 50 |
| 2 | 90 | 150 |
| 3 | 80 | 63 |
| 4 | 5 | 10 |
| 5 | 63 | 63 |
| 6 | 15 | 8 |
| 7 | 67 | 56 |
| 8 | 22 | 19 |
| 9 | 56 | 63 |
| 10 | 33 | 19 |

- a. Analyze the data in each county separately using the Jeffreys' prior distribution and report the posterior 95% credible set for each county.

Solution: We assume the data follow a binomial distribution

$$f(Y|p) = \binom{N}{Y} p^Y (1-p)^{N-Y}.$$

To derive Jeffreys' prior we first find the expected Fisher information

$$\begin{aligned}
l(Y|p) &= \log f(Y|p) \\
&= \log \binom{N}{Y} + Y \log p + (N - Y) \log(1 - p) \\
\frac{dl}{dp} &= \frac{Y}{p} + \frac{N - Y}{1 - p} \\
\frac{dl^2}{dp^2} &= -\frac{Y}{p^2} - \frac{N - Y}{(1 - p)^2}.
\end{aligned}$$

The expected value of Y is Np and the expected value of $N - Y$ is $N - Np$.

$$\begin{aligned}
I(p) &= \frac{Np}{p^2} + \frac{N - Np}{(1 - p)^2} \\
&= \frac{N}{p} + \frac{N}{1 - p} \\
&= \frac{N}{p(1 - p)}.
\end{aligned}$$

Jeffreys' prior is therefore

$$\pi(p) \propto \sqrt{\frac{n}{p(1 - p)}} \propto p^{1/2-1}(1 - p)^{1/2-1},$$

which is the kernel of a Beta(0.5, 0.5) PDF. We can thus see that Jeffreys' prior for a binomial proportion is a Beta(0.5, 0.5) distribution, and the posterior will be a Beta(0.5 + Y , 0.5 + ($N - Y$)) distribution.

```

```{r}
jeffreys_results <- d |>
 mutate(
 A = Approve + 0.5,
 B = Disapprove + 0.5,
 Lower = qbeta(0.025, A, B),
 Upper = qbeta(0.975, A, B)
)

jeffreys_results |>
 gt() |>
 cols_hide(
 columns = c(Approve, Disapprove, A, B)
) |>
 tab_spanner(

```

```

 label = "95% Credible Interval",
 columns = c(Lower, Upper)
) |>
 fmt_percent(
 columns = c(Lower, Upper)
)
...

```

County	95% Credible Interval	
	Lower	Upper
1	11.04%	30.45%
2	31.55%	43.74%
3	47.76%	63.89%
4	14.03%	58.42%
5	41.35%	58.65%
6	44.89%	81.98%
7	45.65%	63.08%
8	38.58%	68.24%
9	38.25%	56.01%
10	49.93%	75.54%

- b. Let  $\hat{p}_i$  be the sample proportion of commuters in county  $i$  that approve of the HIV lane (e.g.  $\hat{p}_1 = 12/(12 + 50) = 0.194$ ). Select  $a$  and  $b$  so that the mean and variance of the  $\text{Beta}(a, b)$  distribution match the mean and variance of the sample proportions  $\hat{p}_1, \dots, \hat{p}_{10}$ .

**Solution:**

```

d |>
 mutate(
 p = Approve / (Approve + Disapprove)
) |>
 summarise(
 mean = mean(p),
 var = var(p)
) |>

```

```
gt() |>
cols_label(
 mean = "Mean",
 var = "Variance"
)
```

Mean	Variance
0.4800001	0.02025887

So we want the following to be approximately true

$$\frac{\alpha}{\beta} = 0.48 \quad \frac{\alpha\beta}{(\alpha + \beta)^2(\alpha + \beta + 1)} = 0.02$$

We can solve this simply by creating a target function and using R's built-in `optim` function

```
```{r}
my_fun <- function(pars) {

  alpha <- pars[1]
  beta <- pars[2]

  # Calculate difference between current mean and desired mean
  diff_mean <- alpha / beta - 0.48

  # Calculate difference between current variance and desired variance
  diff_var <- (alpha * beta) / ((alpha + beta)^2 * (alpha + beta + 1)) - 0.02

  # Return the squared differences of both
  diff_mean^2 + diff_var^2
}

results <- optim(c(1, 1), fn = my_fun)

alpha <- results$par[1]
beta <- results$par[2]

alpha
beta
```
```



```
[1] 3.236171
[1] 6.742171
```

- 
- c. Conduct an empirical Bayesian analysis by computing the 95% posterior credible sets that results from analyzing each county separately using the  $\text{Beta}(a, b)$  prior you computed in *b*.
- 

**Solution:**

```
```{r}
emp_bayes_results <- d |>
  mutate(
    A = Approve + alpha,
    B = Disapprove + beta,
    Lower = qbeta(0.025, A, B),
    Upper = qbeta(0.975, A, B)
  )

emp_bayes_results |>
  gt() |>
  cols_hide(
    columns = c(Approve, Disapprove, A, B)
  ) |>
  tab_spanner(
    label = "95% Credible Interval",
    columns = c(Lower, Upper)
  ) |>
  fmt_percent(
    columns = c(Lower, Upper)
  )
```
```

---

| County | 95% Credible Interval |        |
|--------|-----------------------|--------|
|        | Lower                 | Upper  |
| 1      | 12.60%                | 31.25% |
| 2      | 31.42%                | 43.37% |

---

|    |        |        |
|----|--------|--------|
| 3  | 46.50% | 62.21% |
| 4  | 16.38% | 52.15% |
| 5  | 40.37% | 57.09% |
| 6  | 38.39% | 71.59% |
| 7  | 44.34% | 61.22% |
| 8  | 35.99% | 63.06% |
| 9  | 37.43% | 54.54% |
| 10 | 46.08% | 70.33% |

---

- 
- d. How do the results from *a.* and *c.* differ? What are the advantages and disadvantages of these two analyses?
- 

**Solution:** The Empirical Bayes (EB) prior pulls each individual proportion slightly towards the overall mean. Since each county gets the same EB prior the effect is that counties with fewer votes get pulled more towards the overall mean, thus stabilizing the inferences in smaller counties by “*borrowing*” information from other counties.

The downside to using the EB prior is that the inference for each county is slightly biased towards the overall mean.

This is an example of the bias/variance trade-off in statistics. We often want to reduce the variance in our statistical inference at the cost of increasing the bias in our inferences.

```
```{r}
#| code-fold: true

emp_bayes_results |>
  mutate(
    type = "Empirical Bayes"
  ) |>
  bind_rows(
    jeffreys_results |>
      mutate(
        type = "Jeffreys"
      )
  ) |>
  ggplot(aes(x = County, ymin = Lower, ymax = Upper, col = type)) +
  geom_hline(
```

```

    yintercept = 0.48,
    lty = 2,
    linewidth = 0.5
) +
geom_linerange(
  linewidth = 1,
  position = position_dodge(width = 0.3)
) +
scale_x_continuous(
  breaks = 1:10
) +
scale_y_continuous(
  limits = c(0, 1),
  labels = scales::label_percent(),
  expand = expansion()
) +
annotate(
  geom = "text",
  x = 1.5,
  y = 0.51,
  label = "Overall Mean"
) +
theme(legend.position = c(0.15, 0.9)) +
labs(
  x = "County",
  y = expression(hat(p)),
  col = NULL,
  title = "Comparing the Jeffreys and Empirical Bayes intervals",
  subtitle = "Empirical Bayes pulls the intervalls slightly towards the overall mean"
)
...

```

Comparing the Jeffreys and Empirical Bayes intervals

Empirical Bayes pulls the intervals slightly towards the overall mean

