Mini-Project 4: Bayesian Data Analysis

Tanner Bessette

2025-04-02

Statement of Integrity

I have followed all rules for collaboration for this project, and I have not used generative AI on this project. Tanner Bessette

Question of interest and overview

The question of interest that I am answering for this project is: What is the probability that Nadal wins a point on his own serve, against his primary rival, Djokovic, on a clay court at the French Open?

An appropriate distribution for this data is a Binomial distribution, because Nadal either won a point or did not win a point on each of his serves. The conjugate prior (that we used in class) for Binomial data is the Beta distribution.

I will be answering this question using Bayesian analyses with three priors: a non-informative prior, an informative prior based on a clay court match the two played in the previous year, and an informative prior based on a sports announcer.

Justifying decisions made to make the two informative priors

Non-informative prior: Beta(1,1)

Informative prior based on a clay-court match the two played in the previous year. In that match, Nadal won 46 out of 66 points on his own serve. The standard error of this estimate is 0.05657:

Want $\mu = 46/66 = 0.69697$, so $46/66 = \alpha/(\alpha + \beta)$

–> $46/66 * (\alpha + \beta) = \alpha$

–> $46/66 * \beta = 20/66 * \alpha$

and want $\sigma = 0.05657$.

```
library(tibble)
library(tidyverse)

target_mean <- 0.69697

alphas <- seq(0.1, 100, length.out = 1000)
betas <- (20/66) * alphas / (46/66)

param_df <- tibble(alphas, betas)
param_df <- param_df |> mutate(vars = (alphas * betas) /
                   ((alphas + betas)^2 * (alphas + betas + 1)))

target_var <- 0.05657^2

param_df <- param_df |> mutate(dist_to_target = abs(vars - target_var))
param_df
```

```
# A tibble: 1,000 x 4
   alphas  betas   vars dist_to_target
    <dbl>  <dbl>  <dbl>          <dbl>
 1    0.1 0.0435 0.185          0.182
 2    0.2 0.0870 0.164          0.161
 3    0.3 0.130  0.148          0.144
 4    0.4 0.174  0.134          0.131
 5    0.5 0.217  0.123          0.120
 6    0.6 0.261  0.113          0.110
 7    0.7 0.304  0.105          0.102
 8    0.8 0.348  0.0983         0.0951
 9    0.9 0.391  0.0922         0.0890
10    1   0.435  0.0867         0.0835
# i 990 more rows
```

```
param_df |> filter(dist_to_target == min(dist_to_target))
```

```
# A tibble: 1 x 4
  alphas betas    vars dist_to_target
   <dbl> <dbl>   <dbl>          <dbl>
1   45.3  19.7 0.00320   0.0000000904
```

So, for the informative prior based on a clay-court match the two played in the previous year, we will use: Beta(45.3, 19.7).

Informative prior based on a sports announcer, who claims that they think Nadal wins about 75% of the points on his serve against Djokovic. They are also "almost sure" that Nadal wins no less than 70% of his points on serve against Djokovic:

Want $\mu = 0.75$, so $\alpha/(\alpha + \beta) = 0.75$

–> $0.75 * (\alpha + \beta) = \alpha$

–> $0.75 * \beta = 0.25 * \alpha$

(We will treat "almost sure" as a 0.02 chance)

```r
## trying to get a mean of 0.75 and a probability
## that lambda is less than 0.70 equal to 0.02
alphas <- seq(0.01, 500, length.out = 2000)
betas <- alphas * 0.25 / 0.75

target_prob <- 0.02
prob_less_point_70 <- pbeta(0.70, alphas, betas)

tibble(alphas, betas, prob_less_point_70) |>
  mutate(close_to_target = abs(prob_less_point_70 - target_prob)) |>
  filter(close_to_target == min(close_to_target))
```

```
# A tibble: 1 x 4
  alphas betas prob_less_point_70 close_to_target
   <dbl> <dbl>              <dbl>           <dbl>
1   252.  84.0             0.0200       0.0000155
```

So, for the informative prior with Nadal winning 75% of the points on his serves and "almost sure" that Nadal wins no less than 70% of his serves we will use: Beta(252, 84).

What did I assume when I made these priors?

For both of the informative priors, we are assuming independence of each serve, and that there is no effect of player momentum, fans, or weather on whether or not Nadal wins his serve.

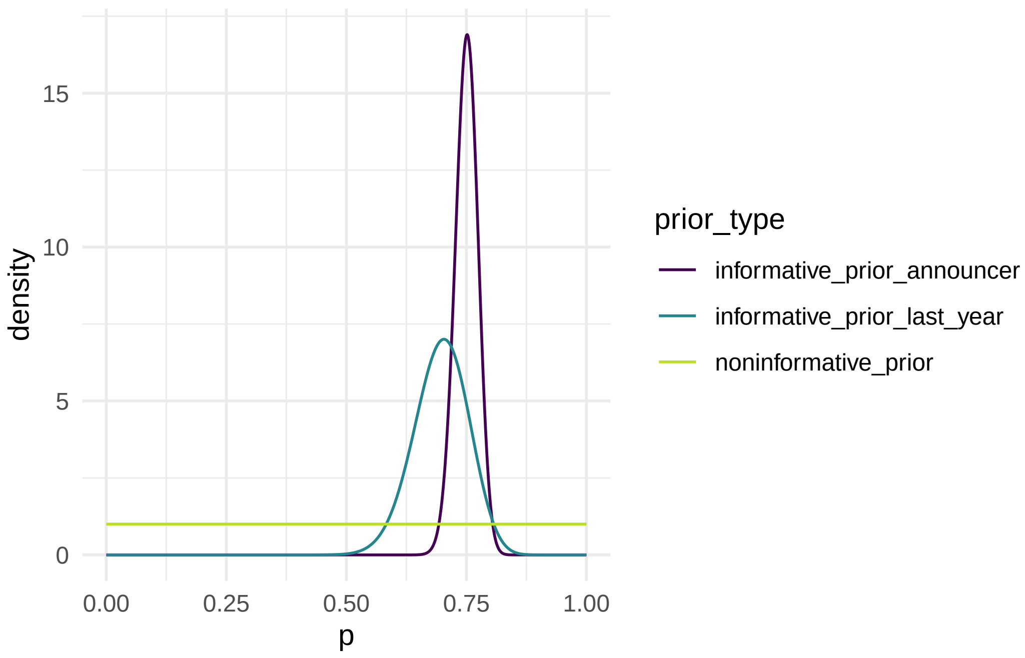
For the informative prior based on the previous year's match, we are assuming that this game last year is indicative of the players' abilities in the French Open this year.

For the sports announcer informative prior, we are assuming that it is okay to use 0.98 as "almost sure", and we are assuming that the sports announcer is well-educated on tennis.

Graph of three priors

Code for graphing priors:

```r
library(tidyverse)
ps <- seq(0, 1, length.out = 1000)

noninformative_alpha <- 1
noninformative_beta <- 1

informative_alpha_last_year <- 45.3
informative_beta_last_year <- 19.7

informative_alpha_announcer <- 252
informative_beta_announcer <- 84

noninformative_prior <- dbeta(ps, noninformative_alpha,
                              noninformative_beta)

informative_prior_last_year <- dbeta(ps, informative_alpha_last_year,
                           informative_beta_last_year)

informative_prior_announcer <- dbeta(ps, informative_alpha_announcer,
                           informative_beta_announcer)

prior_plot <- tibble(ps, noninformative_prior,
  informative_prior_last_year, informative_prior_announcer) |>
  pivot_longer(2:4, names_to = "prior_type", values_to = "density")

ggplot(data = prior_plot, aes(x = ps, y = density,
                              colour = prior_type)) +
  geom_line() +
  scale_colour_viridis_d(end = 0.9) +
  theme_minimal() +
  labs(x = "p")
```



Graph of three posteriors (and work to obtain them)

Now, we want to use the 2020 French Open data to update our prior for the probability that Nadal wins a point on serve. In that tournament, the two players played in the final. In that final, Nadal served 84 points and won 56 of those points.

Work for non-informative posterior:

Beta(1,1) –> Beta(1 + 56, 1 + 84 - 56) –> Beta(57, 29) is the posterior distribution.

Posterior mean: 57 / (57 + 29) = 57/86 = 0.6628

90% credible interval for p:

```
c(qbeta(0.025, 57, 29), qbeta(0.975, 57, 29))
```

```
[1] 0.5601601 0.7582486
```

According to our model, there is a 90% chance that the proportion of serves that Nadal would win is between 0.5602 and 0.7582.

Work for informative posterior based on last year's match:

Beta(45.3, 19.7) –> Beta(45.3 + 56, 19.7 + 84 - 56) –> Beta(101.3, 47.7) is the posterior distribution.

Posterior mean: 101.3 / (101.3 + 47.7) = 0.6798658

90% credible interval for p:

```
c(qbeta(0.025, 101.3, 47.7), qbeta(0.975, 101.3, 47.7))
```

```
[1] 0.6030531 0.7521048
```

According to our model, there is a 90% chance that the proportion of serves that Nadal would win is between 0.6031 and 0.7521.

Work for informative posterior based on announcer:

Beta(252, 84) –> Beta(252 + 56, 84 + 84 - 56) –> Beta(308, 112) is the posterior distribution.

Posterior mean: 308 / (308 + 112) = 0.7333

90% credible interval for p:

```
c(qbeta(0.025, 308, 112), qbeta(0.975, 308, 112))
```

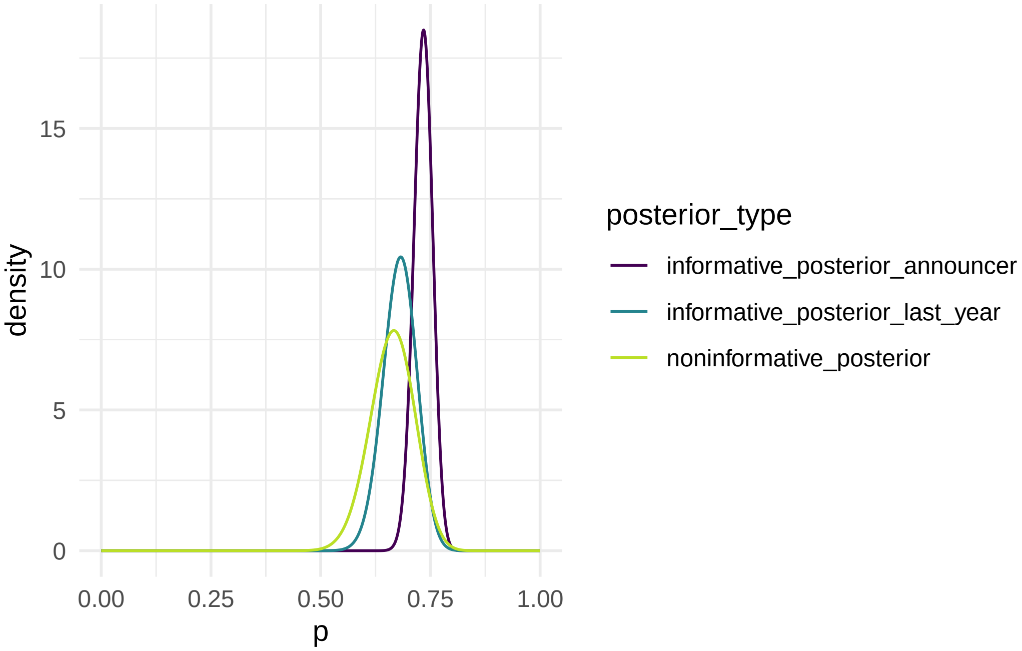
```
[1] 0.6900703 0.7744917
```

According to our model, there is a 90% chance that the proportion of serves that Nadal would win is between 0.6901 and 0.7745.

Code for graphing posteriors:

```
library(tidyverse)
ps <- seq(0, 1, length.out = 1000)

noninformative_alpha <- 57
noninformative_beta <- 29

informative_alpha_last_year <- 101.3
informative_beta_last_year <- 47.7

informative_alpha_announcer <- 308
informative_beta_announcer <- 112
```

```
noninformative_posterior <- dbeta(ps, noninformative_alpha,
                                  noninformative_beta)

informative_posterior_last_year <- dbeta(ps, informative_alpha_last_year,
                               informative_beta_last_year)

informative_posterior_announcer <- dbeta(ps, informative_alpha_announcer,
                               informative_beta_announcer)

prior_plot <- tibble(ps, noninformative_posterior,
  informative_posterior_last_year, informative_posterior_announcer) |>
  pivot_longer(2:4, names_to = "posterior_type", values_to = "density")

ggplot(data = prior_plot, aes(x = ps, y = density,
                              colour = posterior_type)) +
  geom_line() +
  scale_colour_viridis_d(end = 0.9) +
  theme_minimal() +
  labs(x = "p")
```



Comparison of the three posteriors

They should be a bit different from each other: why?

They are different from each other because the prior distributions are all different. So, some posteriors are updated with greater or smaller changes based on the data that is observed. The priors all had different alpha and beta parameters, so the variances and means of the posteriors are all different.

If you had to choose one to use, which one would you choose here and why?

If I had to choose one posterior, I would choose the one that had the prior of the match data from the previous year. I would choose this because it is based on real data, and had a very similar prior mean to posterior mean. Its prior mean (0.69697) was also extremely similar to the observed data from the French Open's match (0.66667).

The variance of each posterior should be different. Why do you think one posterior has a lower variance than the other two?

One posterior has a lower variance than the other two: the announcer's informative posterior. This is because the prior had specific requirements that needed to be met, which meant the alpha and beta parameters of the prior were extremely large - 252 and 84, so there is a higher level of confidence.

Brief conclusion of what I found in this mini-project

From the three posteriors, it appears that Nadal would win between 65% and 75% of his serves against Djokovic. The prior that had much higher alpha and beta parameters (announcer's) has much lower variance in the posterior than the non-informative, or the one based on last year's match data. The stronger the prior beliefs are, the less the posterior will be updated by real data.

I think one takeaway is also that it is more reliable to trust real data, or to use a non-informative prior, rather than being overly confident in the precision of somebody's subjective guess.