Homework #3

Dustin Leatherman

October 5, 2020

1

> The data set NBAclutchFT.csv has the overall free throw proportion and results of free throws taken in pressure situations, defined as "clutch", for ten National Basketball Association players (those that received the most votes for the Most Valuable Player award) for the 2019-2020 season. Since the overall proportion is computed using a large sample size, assume it is fixed and analyze the clutch data for each player separately using Bayesian methods

a

> Describe your model including the likelihood and prior

The reponse variable is the proportion of successful "clutch" shots. A Binomial Distribution with n = attempts and θ = clutch.makes / clutch.attempts is a reasonable Likelihood distribution. The Prior information known about a given player is their overall free throw percentage. Bounded between 0 and 1, this can be represented by a Beta Prior so we can take advantage of the Beta-Binomial conjugate prior and likelihood.

```
# 5 percentage points of error allowed. I don't know if this is actually a good number. I don't really
acceptable_error <- 0.05

# add
clutch.plus <-
  # estimate beta parameters for each player since they have varying Freethrow Percentages
  cbind(clutch, t(sapply(clutch$FT.Pct, function(x) beta.prior(x, acceptable_error)))) %>%
  mutate(
    clutch.pct = clutch.makes / clutch.attempts
  ) %>%
  mutate(
    prior.a = unlist(a),
    prior.b = unlist(b),
    post.a = clutch.makes + prior.a,
    post.b = clutch.attempts - clutch.makes + prior.b
  ) %>%
  select(-a, -b)
```

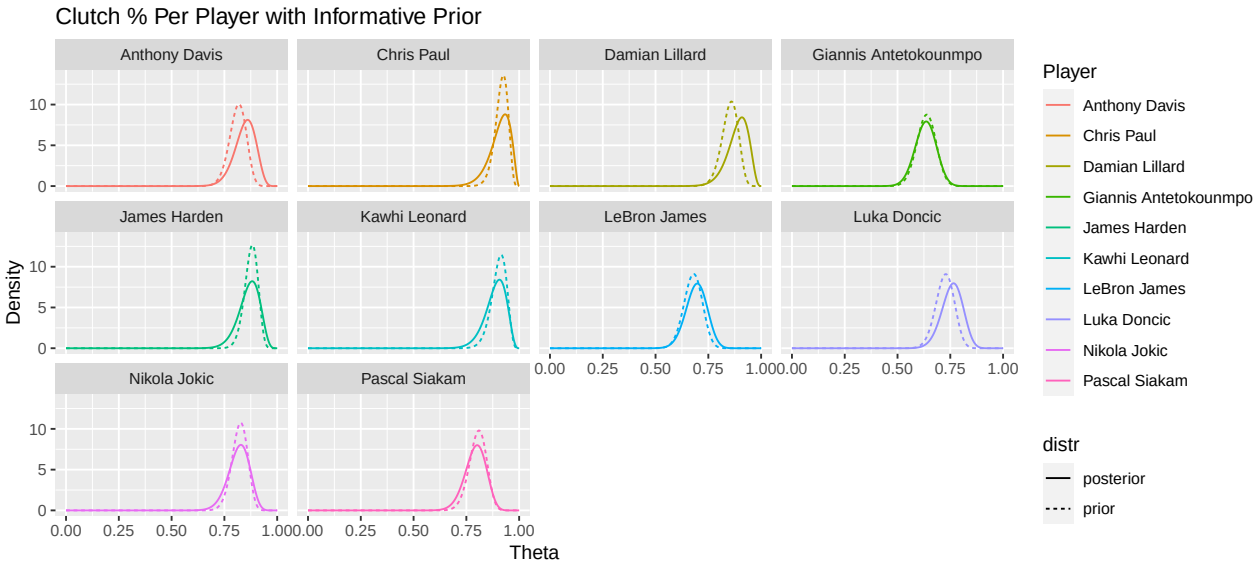
The parameters for the prior beta distribution can be estimated using the overall free throw percentage as the mean and an acceptable range of error as the standard deviation.

b

> Plot your posteriors of the clutch success probability

```
# base thetas to compute densities
theta <- seq(0, 1, 0.001)

# produce posterior density function for each row.
```

```r
post.density <-
  apply(clutch.plus, 1, function(x){
    dbeta(
        theta,
        shape1 = as.numeric(x["post.a"]),
        shape2 = as.numeric(x["post.b"])
    )
  })

# produce prior density for each row.
prior.density <-
  apply(clutch.plus, 1, function(x){
    dbeta(
        theta,
        shape1 = as.numeric(x["prior.a"]),
        shape2 = as.numeric(x["prior.b"])
    )
  })

# associate the player with the function and make it a dataframe
# Format density to a single column by player for nice graphing
density.formatted <-
  # add "player" column to densities. This assumes that the
  # density and player lists match order. As in player 1 is associated
  # with density 1, etc. Since this is doing matrix operations, this is a safe assumption
  inner_join(
    # since there are a bunch of unnamed columns, dynamic columns of the form
    # V1, V2, ..., V1000 are created for each theta.
    as.data.frame(t(post.density)) %>% mutate(Player = clutch.plus$Player),
    as.data.frame(t(prior.density)) %>% mutate(Player = clutch.plus$Player),
    clutch.plus,
    by = "Player",
    # add suffix so its easier to tell which var is which
    suffix = c(".prior", ".posterior")
  ) %>%
  # Get the density columns and turn them into rows.
  select(starts_with("V"), Player) %>%
  pivot_longer(starts_with("V"), names_to = "columnName", values_to = "density") %>%
  mutate(
    # had trouble with getting correct values for str_split
    # so needed to do a roundabout way to get the distribution
    distr_obj = unlist(str_split_fixed(columnName, pattern = "\\.", n = 2)),
    distr = distr_obj[,2],
    # remake theta. This is gross but dealing with this format has also been unpleasant
    theta = (as.numeric(str_remove(columnName, "V") %>% str_remove(".posterior") %>% str_remove(".prior
  )

ggplot(density.formatted, aes(x = theta, y = density, color = Player, linetype = distr)) +
  geom_line() +
  facet_wrap(~Player) +
  labs(x = "Theta", y = "Density") +
  ggtitle("Clutch % Per Player with Informative Prior")
```



c

Summarize the Posteriors in a table

```
credible.int <-
  apply(clutch.plus, 1, function(x){
    qbeta(
        c(0.025, 0.975),
        shape1 = as.numeric(x["post.a"]),
        shape2 = as.numeric(x["post.b"])
    )
  })

# join credible interval back onto our normal data
density.formatted.plus <-
  density.formatted %>%
  inner_join(
    as.data.frame(t(credible.int)) %>%
      mutate(Player = clutch.plus$Player) %>%
      rename(ucl = V2, lcl = V1),
    by = "Player"
  )


# build summary table for posterior distribution
density.formatted.plus %>%
  inner_join(clutch.plus, by = "Player") %>%
  select(Player, lcl, ucl, post.a, post.b) %>%
  distinct() %>%
  mutate(
    post.mean = post.a / (post.a + post.b),
    post.sd = sqrt(post.a*post.b/((post.a+post.b)^2*(post.a+post.b+1)))
  ) %>%
  select(Player, post.mean, post.sd, lcl, ucl) %>%
  kable(
```

```
    col.names = c("Player", "Mean", "Std. Dev", "95th Lower C.I.", "95th Upper C.I.")
  ) %>%
  kable_styling(bootstrap_options = "striped", latex_options = "hold_position")
```

Player	Mean	Std. Dev	95th Lower C.I.	95th Upper C.I.
Giannis Antetokounmpo	0.6360378	0.0452768	0.5451859	0.7222837
LeBron James	0.6773121	0.0435760	0.5891584	0.7595778
James Harden	0.8754058	0.0319706	0.8064273	0.9309732
Luka Doncic	0.7241618	0.0437470	0.6345626	0.8055476
Kawhi Leonard	0.9022345	0.0361757	0.8208301	0.9608727
Anthony Davis	0.8121037	0.0398449	0.7281585	0.8836357
Chris Paul	0.9148559	0.0304990	0.8463439	0.9645187
Damian Lillard	0.8508907	0.0389146	0.7671161	0.9186543
Nikola Jokic	0.8209884	0.0372698	0.7424650	0.8879259
Pascal Siakam	0.8029911	0.0408299	0.7172140	0.8765589

d

Test the Hypothesis that the clutch proportion is less than the overall proportion

```
# convert density rows back into columns so means can be compared.
# produces a DataFrame with 3 columns: name, posterior (list), prior (list)
hyp.means <-
  density.formatted %>%
    select(Player, distr, density) %>%
    pivot_wider(
      id_cols = Player,
      names_from = "distr",
      values_from = "density",
      values_fn = list
    )

# compute probability that posterior is greater than the prior
data.frame("clutch.prob" = apply(hyp.means, 1, function(x) mean(x$posterior > x$prior))) %>%
  kable(
    col.names = c("Probability"),
    caption = "Estimate of Posterior Probability that Clutch Percentages are greater than Overall Percen
  ) %>%
  kable_styling(bootstrap_options = "striped", latex_options = "hold_position")
```

e

Are the results sensitive to your prior?

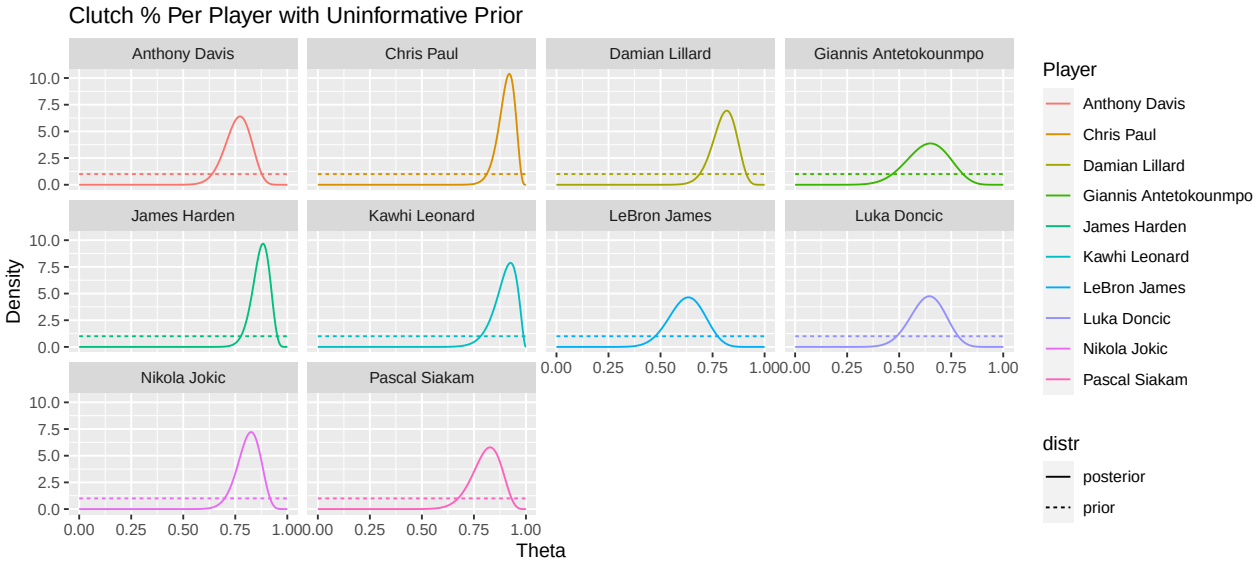
```
# redo our analysis with an uninformative prior.
# TODO: If this is useful, turn these into functions for reuse later.
clutch.plus.uninf <-
  clutch.plus %>%
  mutate(
    prior.a = 1,
    prior.b = 1,
```

Table 1: Estimate of Posterior Probability that Clutch Percentages are greater than Overall Percentages

Probability
0.8991009
0.8531469
0.9210789
0.7712288
0.9140859
0.8391608
0.9250749
0.8421578
0.9130869
0.8991009

```
    post.a = clutch.makes + prior.a,
    post.b = clutch.attempts - clutch.makes + prior.b
  )

post.density.uninf <-
  apply(clutch.plus.uninf, 1, function(x){
    dbeta(
        theta,
        shape1 = as.numeric(x["post.a"]),
        shape2 = as.numeric(x["post.b"])
    )
  })

prior.density.uninf <-
  apply(clutch.plus.uninf, 1, function(x){
    dbeta(
        theta,
        shape1 = as.numeric(x["prior.a"]),
        shape2 = as.numeric(x["prior.b"])
    )
  })

# associate the player with the function and make it a dataframe
# Format density to a single column by player for nice graphing
density.formatted <-
  inner_join(
    as.data.frame(t(prior.density.uninf)) %>% mutate(Player = clutch.plus.uninf$Player),
    as.data.frame(t(post.density.uninf)) %>% mutate(Player = clutch.plus.uninf$Player),
    clutch.plus.uninf,
    by = "Player",
    suffix = c(".prior", ".posterior")
  ) %>%
  select(starts_with("V"), Player) %>%
  pivot_longer(starts_with("V"), names_to = "columnName", values_to = "density") %>%
  # remake theta. This is gross but dealing with this format has also been unpleasant
  mutate(
    distr_obj = unlist(str_split_fixed(columnName, pattern = "\\.", n = 2)),
    distr = distr_obj[,2],
```

```
    theta = (as.numeric(str_remove(columnName, "V") %>% str_remove(".posterior") %>% str_remove(".prior
  )

ggplot(density.formatted, aes(x = theta, y = density, color = Player, linetype = distr)) +
  geom_line() +
  facet_wrap(~Player) +
  labs(x = "Theta", y = "Density") +
  ggtitle("Clutch % Per Player with Uninformative Prior")
```



Comparing the graphs in b with graphs using an uninformative prior, the results appear to be sensitive to the prior. The distributions are flatter, with smaller means, and generally smaller modes using the uninformative prior compared to the informative prior.

2

Say that $Y|\theta \sim Bin(n, \theta)$, and $Z|\theta \sim Bin(M, \theta)$ and that Y,Z are independent given θ. Identify a conjugate prior for θ and find the corresponding Posterior Distribution.

Uninformative Prior: $\theta \sim Beta(1, 1)$

Posteriors:

$$Y|\theta \sim Beta(Y + 1, n - Y + 1)$$

$$Z|\theta \sim Beta(Z + 1, M - Z + 1)$$

$$\theta|Y, Z \sim Beta(Y + Z + 1, n + M - Y - Z + 1)$$

Since $Y|\theta$ and $Z|\theta$ are independent, their product is a beta distribution.