Assignment 3

Introduction to Bayesian Data Analysis 2025

Ana Sofia Acevedo (03805441)

Preamble

- **Points**: Assignment 3 comprises of 6 tasks, 2 points each (12 in total). 2 points are obtained for complete and correct answers. 1 point is obtained for a proper approach or if only part of the task is solved.
- **Submission**: Hand in the assignment as a PDF `Markdown` report. The report should show the results, the code that produced the results, and additional text or comment. The report should appear clean and be uploaded on Moodle until Wednesday, June 25, 9:45 am.
- **Collaboration**: Reports can be handed in as team work (max. 2 people). When working in teams, declare this on page 2. However, each collaborator needs to hand in a report via Moodle, stating their name, student number (p. 1), and their machine specification (p. 2).
- **Permitted and Prohibited**: You may use materials from this class (e.g., slides, code on GitHub) as well as online forums such as Stack Overflow to write your code. However, you are not allowed to post questions from the assignment online or prompt them (including paraphrases) to LLMs/chatbots. All use of LLMs/chatbots is generally not allowed. Solutions may not be shared with other students from the class (except 1 potential collaborator).

Authorship Information

1. Declaration of Collaboration

☐ Yes (Collaborator name)
☒ No

2. Declaration of Authorship

☒ I certify that this assignment represents my own work. I have not used any unauthorized or unacknowledged aids as stated in the preamble, including free or commercial systems or services offered on the internet or text generating systems embedded into software. I did not copy code from someone else nor did I share my code with someone else.

3. System Information

☒ I confirm that I generated the submitted PDF report myself using R version 4.5.0 (2025-04-11 ucrt) and `Quarto/RMarkdown`.

Machine stamp: x86_64-w64-mingw32/x64

Timestamp: 2025-06-20 13:26:23 CEST

```r
library(here)
```

```
here() starts at C:/Users/57314/BayesIntro25_forked
```

```r
library(tidyverse)
```

```
-- Attaching core tidyverse packages ------------------------ tidyverse 2.0.0 --
v dplyr     1.1.4     v readr     2.1.5
v forcats   1.0.0     v stringr   1.5.1
v ggplot2   3.5.2     v tibble    3.2.1
v lubridate 1.9.4     v tidyr     1.3.1
v purrr     1.0.4
```

```
-- Conflicts ------------------------------------------ tidyverse_conflicts() --
x dplyr::filter() masks stats::filter()
x dplyr::lag()    masks stats::lag()
i Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to becom
```

```r
library(rethinking)
```

```
Loading required package: cmdstanr
This is cmdstanr version 0.9.0
- CmdStanR documentation and vignettes: mc-stan.org/cmdstanr
- CmdStan path: C:/Users/57314/.cmdstan/cmdstan-2.36.0
- CmdStan version: 2.36.0
Loading required package: posterior
This is posterior version 1.6.1

Attaching package: 'posterior'

The following objects are masked from 'package:stats':

    mad, sd, var

The following objects are masked from 'package:base':

    %in%, match

Loading required package: parallel
```

```
rethinking (Version 2.42)

Attaching package: 'rethinking'

The following object is masked from 'package:purrr':

    map

The following object is masked from 'package:stats':

    rstudent
```

```r
library(ggplot2)
```

Load the data set `shaq` to solve the tasks below. If the `Markdown` document and the data set are stored in different folders (e.g., "BayesIntro/assignments/assignment_3.md" and "BayesIntro/data/shaq.csv" you can use the package `here` to load the data.

```r
shaq <- read.csv(here("data", "shaq.csv"))
```

Task Set 1

For Tasks 1.1 and 1.2, create a training data set `shaq_training` that contains all the data from the `Season` 1 to 5.

```r
shaq_training <- shaq[shaq$Season %in% c(1:5), ]
```

Task 1.1

Use the training data and estimate a simple regression model where you predict points (PTS) from field goal attempts (FGA). Specify the regression model such that the intercept represents the expected number of points, given an average number of FGA. Provide a table that summarizes the posterior distribution.

Answer

In this case, the model would be something like

$$PTS_i \sim N(\mu_i, \sigma)\mu_i = \alpha + \beta * (FGA_i - \overline{FGA})$$

For this, the priors that need to be set correspond to α, β, and σ.

```
#Calculate the mean of FGA for mean centering
FGA_bar <-round(mean(shaq_training$FGA),0)

# Establish model with priors
m_1 <- alist(

    PTS ~ dnorm(mu, sigma), # likelihood
    mu <- a + b * (FGA-FGA_bar),

    # priors
    a ~ dgamma(2,0.08),
    b ~ dunif(0,3),
    sigma ~ dunif(0,10)

  )

m_1_fit <- quap( m_1 , data=shaq_training )
precis(m_1_fit)
```

```
           mean         sd      5.5%     94.5%
a     27.241326 0.26775120 26.813408 27.669244
b      1.173304 0.05395665  1.087071  1.259537
sigma  4.977555 0.18921829  4.675147  5.279962
```

Task 1.2

Estimate a multiple regression model, where you add free throw attempts (FTA) as a second predictor. Again, the intercept should represent the expected number of points, given an average number of FGA and FTA. Provide a table that summarizes the posterior distribution.

Answer

In this case, the model can be modified to

$$PTS_i \sim N(\mu_i, \sigma) \mu_i = \alpha + \beta_1 * (FGA_i - \overline{FGA}) + \beta_2 * (FTA_i - \overline{FTA})$$

```
#Calculate the mean of FTA for mean centering
FTA_bar <-round(mean(shaq_training$FTA),0)
```

```
# Establish model with priors
m_2 <- alist(

    PTS ~ dnorm(mu, sigma), # likelihood
    mu <- a + b_1 * (FGA-FGA_bar) + b_2 * (FTA-FTA_bar),

    # priors
    a ~ dgamma(2,0.08),
    b_1 ~ dunif(0,3),
    b_2 ~ dunif(0,3),
    sigma ~ dunif(0,10)

  )

m_2_fit <- quap( m_2 , data=shaq_training )
precis(m_2_fit)
```

```
            mean         sd      5.5%      94.5%
a     27.2997782 0.23346133 26.926662 27.6728945
b_1    1.0495805 0.04849824  0.972071  1.1270901
b_2    0.6114512 0.05846933  0.518006  0.7048965
sigma  4.3388354 0.16493780  4.075233  4.6024378
```

Task Set 2

We know want to look how well the trained models from task set 1 predict out of sample. For the following tasks 2.1 and 2.2, create a test data set `shaq_test` that contains all the data from the `Season` 6 to 10.

```
shaq_test <- shaq[shaq$Season %in% c(6:10), ]
```

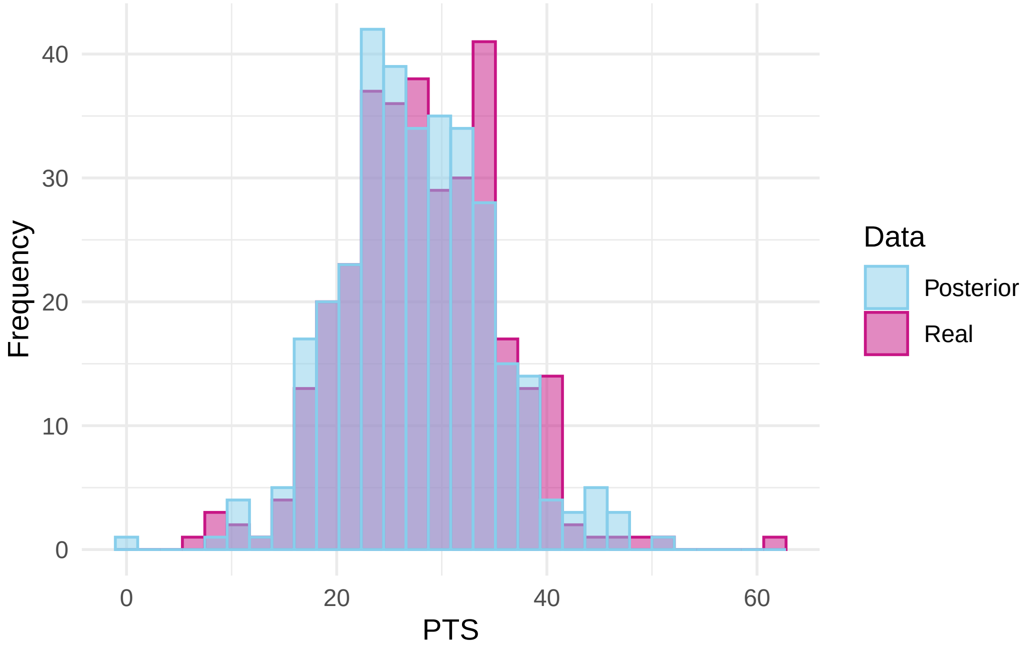
Task 2.1

Use posterior samples from the simple regression model that you estimated in Task 1.1 to predict the points in the Games from `Season` 6 to 10, using the FGA data from the games in these seasons. Create a plot that shows the predicted point distribution along the actual point distribution from Season `Season` 6 to 10.

Answer

```
#Draw posterior samples from the first model (a, b, sigma)
post_pred_1 <- extract.samples(m_1_fit, n=nrow(shaq_test))
#Get FGA from the test data
data <- data.frame(post_pred_1, FGA=shaq_test$FGA)
#Predict using posterior samples and FGA from test set
m_1_pred <- data %>% mutate(PTS = a + b*(FGA-FGA_bar)
                            + rnorm(nrow(shaq_test),0,sigma))

m_1_pred  %>% ggplot(aes(x = PTS)) +
  geom_histogram(data = shaq_test, aes(x = PTS, fill = "Real"),
                 alpha = 0.5, color = "#C71585", bins = 30) +
  geom_histogram(aes(fill = "Posterior"),
                 alpha = 0.5, color = "#87CEEB", bins = 30) +
  scale_fill_manual(values = c("Real" = "#C71585", "Posterior" = "#87CEEB")) +
  labs(x = "PTS",
       y = "Frequency",
       fill = "Data") +
  theme_minimal()
```



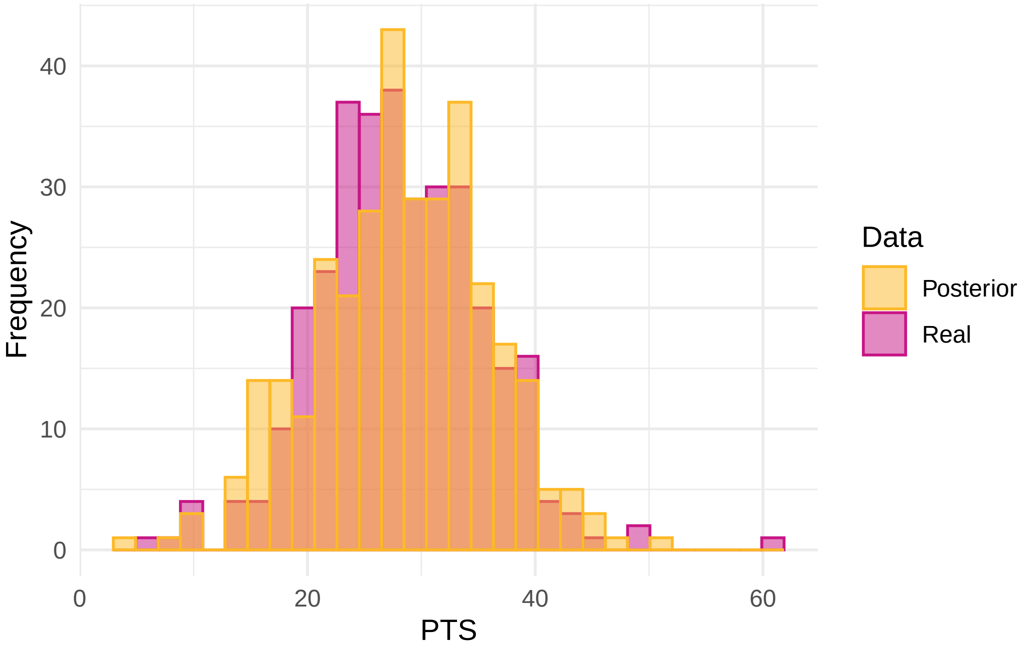
Task 2.2

Use posterior samples from the multiple regression model that you estimated in Task 1.2 to predict the points in the Games from `Season` 6 to 10, using the FGA and FTA data from the games in these seasons. Create a plot that shows the predicted point distribution along the actual point distribution from Season `Season` 6 to 10.

Answer

```
#Draw posterior samples from the second model (a, b_1, b_2, sigma)
post_pred_2 <- extract.samples(m_2_fit, n=nrow(shaq_test))
#Get FGA from the test data
data <- data.frame(post_pred_2, FGA=shaq_test$FGA, FTA=shaq_test$FTA)
#Predict using posterior samples and FGA from test set
m_2_pred <- data %>% mutate(PTS = a + b_1 * (FGA-FGA_bar) + b_2 * (FTA-FTA_bar)
                            + rnorm(nrow(shaq_test),0,sigma))

m_2_pred  %>% ggplot(aes(x = PTS)) +
  geom_histogram(data = shaq_test, aes(x = PTS, fill = "Real"),
                 alpha = 0.5, color = "#C71585", bins = 30) +
  geom_histogram(aes(fill = "Posterior"),
                 alpha = 0.5, color = "#FDB927", bins = 30) +
  scale_fill_manual(values = c("Real" = "#C71585", "Posterior" = "#FDB927")) +
  labs(x = "PTS",
       y = "Frequency",
       fill = "Data") +
  theme_minimal()
```


Frequency
40
30
20
10
0
0
20
40
60
PTS
Data
Posterior
Real

Task Set 3

Task 3.1

Write a function `error()` that takes the predicted points $\hat{y}$ and the observed points y to compute the sum of squared errors:

$$\sum_{i}^{n}(\hat{y}_i - y_i)^2$$

Compute the squared errors for the simple regression model and the multiple regression model. Which model makes better predictions for the test data?

Answer

```
error <- function(pred, real){
  sum((pred - real)^2)
}

error(m_1_pred$PTS, shaq_test$PTS)
```

```
[1] 17019.84
```

```
error(m_2_pred$PTS, shaq_test$PTS)
```

```
[1] 12564.31
```

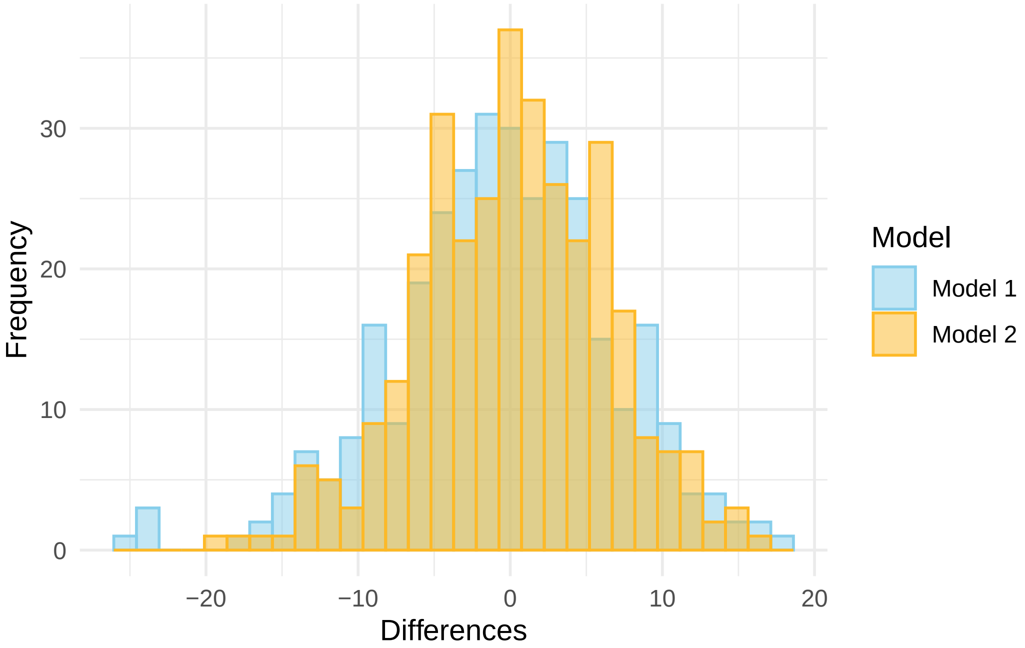
Task 3.2

For both models, compute the (non-squared) differences between each prediction and observation. Create a plot that shows the distributions of differences for both models.

Answer

```r
dif <- data.frame(
  dif_m1 = m_1_pred$PTS - shaq_test$PTS,
  dif_m2 = m_2_pred$PTS - shaq_test$PTS
)

ggplot() +
  geom_histogram(data = dif, aes(x = dif_m1, fill = "Model 1"),
                 alpha = 0.5, color = "#87CEEB", bins = 30) +
  geom_histogram(data = dif, aes(x = dif_m2, fill = "Model 2"),
                 alpha = 0.5, color = "#FDB927", bins = 30) +
  scale_fill_manual(values = c("Model 1" = "#87CEEB", "Model 2" = "#FDB927")) +
  labs(x = "Differences",
       y = "Frequency",
       fill = "Model") +
  theme_minimal()
```


Frequency
Differences
Model
Model 1
Model 2