Assignment 2

Introduction to Bayesian Data Analysis 2025

Ana Sofia Acevedo (03805441)

Preamble

- **Points**: Assignment 2 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 11, 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: 2025-06-09 15:05:10 CEST

```
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
```

```
library(ggplot2)
library(data.tree)
```

Task Set 1

For tasks 1.1-1.2, suppose there are 3 companies, Company A to C. Company A has a customer satisfaction rate of .60, Company B of .80, and Company C of .83. Further suppose that you receive 10 customer reviews (8 positive, 2 negative) for the same company, but you don't know for which company. Assume that Company A is three times more likely to obtain reviews than Company B and C.

Task 1.1

Which company maximizes the likelihood of the 10 reviews and which company has the highest posterior probability that it was rated?

Answer

```
#Assuming the satisfaction rate of 0.6 means that 60% of reviews are positive
data <- data.frame(company = c("A","B","C"), sr =c(.6, .8, .83),
                   prior = c(3/5,1/5,1/5))

# This function takes a data frame that has at least 3 columns, for the
# candidate (name first column and rate second column) and the prior
# (third column), a number of trials N and a number of success s.
```

```
updating <- function(data, N, s){
  candidates <- data[, 2]
  prior <- data[, 3]

  lh <- dbinom(s, N, prob = candidates)
  posterior <- prior*lh
  posterior_norm <- posterior / sum(posterior)

  data$likelihood <- lh
  data$posterior <- posterior_norm
  data[ , -(1:2)] <- round(data[ , -(1:2)], 4)

  return(data)
}

data <- updating(data, 10, 8)
data
```

```
  company   sr prior likelihood posterior
1       A 0.60   0.6     0.1209    0.3788
2       B 0.80   0.2     0.3020    0.3153
3       C 0.83   0.2     0.2929    0.3058
```

Task 1.2

Suppose you receive 10 more reviews (3 positive and 7 negative). Show that the posterior probability that Company B received the reviews decreases by ≈ 30 percentage points, when considering all 20 rather than only the first 10 reviews. To obtain the updated posterior probability, only compute the likelihood of the 10 most recent reviews.

Answer

```
# Yesterday's posterior is today's prior (this considers all the info from the
# 20 reviews), ansd computing the likelihood of ony the 10 most recent reviews
data_1 <- data[,1:3]
post_1 <- data$posterior
data_1$prior <- data$posterior
data_1 <- updating(data_1, 10, 3)
post_2 <- data_1$posterior
```

```r
# Posterior for company B before and after this update
post_1[2]
```

```
[1] 0.3153
```

```r
post_2[2]
```

```
[1] 0.0151
```

```r
# Difference between posteriors
post_1[2]-post_2[2]
```

```
[1] 0.3002
```

Task Set 2

For tasks 2.1 and 2.2, suppose there are Factory A, Factory B and Factory C, producing the same product. The head company receives 50% of its shipments from A and 25% each from B and C. Even though the machines, processes, and standards are virtually identical, the factories differ in their defect rates. Shipments from Factory A entail defective products 10% of the time, shipments from Factory B 15% of the time, and shipments from Factory C 20% .

Task 2.1

You receive a shipment from one of the factories, and upon inspection, you find that the shipment contains defective products. Compute the probability that the next shipment from this company will also contain defective products.

Answer

```
data <- data.frame(factory = c("A","B","C"), dr =c(.1, .15, .2),
                   prior = c(.5,.25,.25))
# Observation is that one defect occurred, so the belief is updated
data_1 <- updating(data, 1, 1)
data_1
```

```
  factory   dr prior likelihood posterior
1       A 0.10  0.50       0.10    0.3636
2       B 0.15  0.25       0.15    0.2727
3       C 0.20  0.25       0.20    0.3636
```

```
# Does A, or B, or C have a defect in the next shipment? Is calculated as an
# expected value
sum(data_1$dr*data_1$posterior)
```

```
[1] 0.149985
```

Task 2.2

Suppose the R&D department came up with a Machine Learning algorithm that (imperfectly) identifies the factory based on the shipped products. But the classification algorithm is imperfect. This is the information you have about the algorithm:

- Correct identification Factory A: 90% (False identification of B/C: 5% / 5%)
- Correct identification Factory B: 95% (False identification of A/C: 4% / 1%)
- Correct identification Factory C: 85% (False identification of A/B: 6% / 9%)

When applying the algorithm to the shipped products, the test is positive for Factory A. Including the defect data from 2.1, compute the posterior probability that your shipment is from Company A.

Answer

```
# By drawing a garden of forking data, and setting the prior probabilities as
# the posterior probabilities calculated in the last step
tree <- Node$new("Sample")

a <- tree$AddChild("A"); a$prob <- data_1[1,"posterior"]
b <- tree$AddChild("B"); b$prob <- data_1[2,"posterior"]
c <- tree$AddChild("C"); c$prob <- data_1[2,"posterior"]
a1 <- a$AddChild("A"); a1$prob <- 0.9
a2 <- a$AddChild("B"); a2$prob <- 0.05
a3 <- a$AddChild("C"); a3$prob <- 0.05
b1 <- b$AddChild("A"); b1$prob <- 0.04
b2 <- b$AddChild("B"); b2$prob <- 0.95
b3 <- b$AddChild("C"); b3$prob <- 0.01
c1 <- c$AddChild("A"); c1$prob <- 0.06
c2 <- c$AddChild("B"); c2$prob <- 0.09
c3 <- c$AddChild("C"); c3$prob <- 0.85

print(tree, "prob")
```

```
   levelName   prob
1  Sample        NA
2   ¦--A      0.3636
3   ¦   ¦--A 0.9000
4   ¦   ¦--B 0.0500
5   ¦   °--C 0.0500
6   ¦--B      0.2727
7   ¦   ¦--A 0.0400
8   ¦   ¦--B 0.9500
9   ¦   °--C 0.0100
10  °--C      0.2727
11      ¦--A 0.0600
12      ¦--B 0.0900
```

```
13      °--C 0.8500
```

To calculate the posterior using Bayes' the formula is

$$P(A_{True} \mid ML = A) = \frac{P(ML = A \mid A_{True}) \cdot P(A_{True})}{P(ML = A)}$$

Where $P(A_{True})$ is the probability that it actually comes from A and $P(ML = A)$ is that the machine learning algorithm predict it comes from factory A. Hence,

```r
# Since we include the defect, that is we have seen a defect and updated the
#beliefs, the prior is the posterior of the last point, and the likelihood is
# the probability that the ML algorithm gives A given that A is true (that if
# A really made the product the ML shows A), which is 0.9
num <- 0.9*data_1[1,"posterior"]
den <- data_1[1,"posterior"]*0.9+data_1[2,"posterior"]*0.04+data_1[3,"posterior"]*0.06
num/den
```

```
[1] 0.9090909
```

Task Set 3

For Task 3.1 and 3.2, suppose, one last time, you want to estimate the proportions of land on the earth's surface.

Task 3.1

Specify a prior distribution and store 10,000 random samples from it in a vector `sample`. Plot the prior distribution and briefly explain your choice of the prior.

Answer

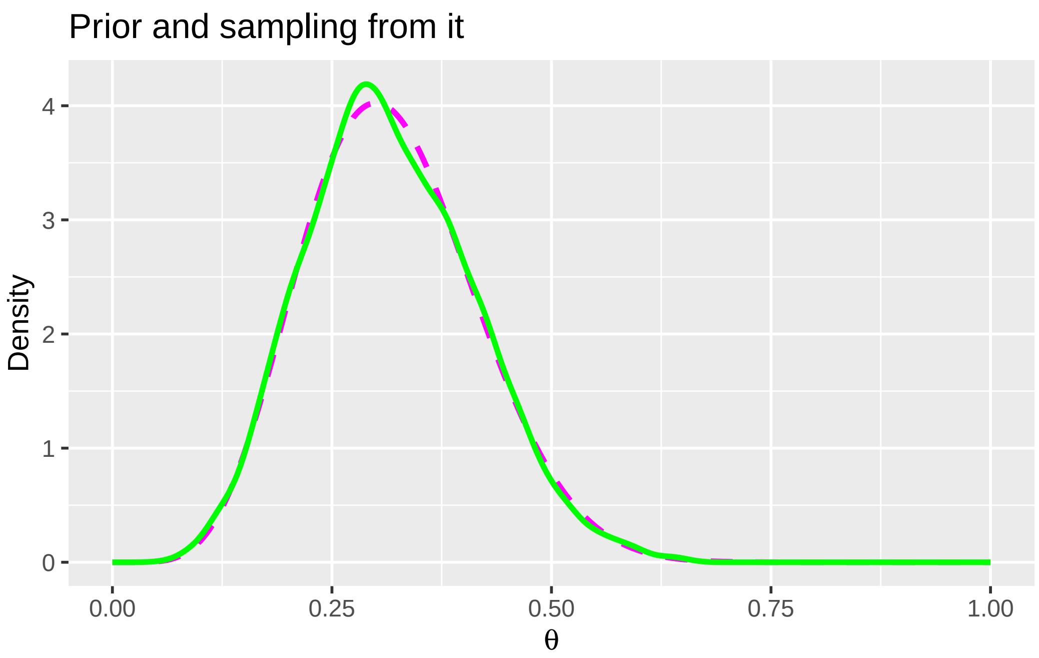
I chose a beta distribution with a mean close to 0.3 because the "general knowledge" is the Earth's surface is covered 70% water, but I considered values lower than 0.3 were more likely because of the melting of ice caps which could increase the percentage of water that covers the surface.

```r
a <- 7
b <- 15

theta <- seq(0,1, length.out = 1e4)
d <- dbeta(theta, shape1 = a, shape2 = b)
summary <- data.frame(theta, d)

# sample from prior
no <- 1e4
sample <- data.frame(smp = rbeta(no, a, b))

ggplot(summary) +
  geom_line(linewidth = 1, linetype = "dashed",  color = "#ff02ff",
            aes(x = theta, y = d)) +
  geom_density(data = sample, aes(x = smp), color = "green", linewidth = 1) +
  labs(x = expression(theta),
       y = "Density", title = "Prior and sampling from it")
```



Task 3.2

Run the following code chunk that uses your object `sample` to obtain prior probabilities for the possible proportions of land 0, 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9, 1 that approximate your prior distribution.

```
prop <- seq(0, 1, length.out = 12)
priors <- vector("numeric", length(prop))
for (i in seq_along(prop)){
priors[i] <- round( sum(sample >= prop[i] & sample < prop[i+1]) / 1e4 , 2)
}
poss <- tibble(prop_L = seq(0, 1, .1), prior = priors[1:11])
poss
```

```
# A tibble: 11 x 2
   prop_L prior
    <dbl> <dbl>
 1    0    0
 2    0.1  0.07
 3    0.2  0.27
 4    0.3  0.35
 5    0.4  0.22
 6    0.5  0.07
 7    0.6  0.01
 8    0.7  0
 9    0.8  0
10    0.9  0
11    1    0
```

Use these priors to compute the posterior probability after observing 26 times land in 100 globe tosses. Take 1,000 samples from the posterior distribution and with each sample, predict the outcome of 100 globe tosses. Plot the posterior predictions in a histogram.

Answer

I also plotted some generated data with the prior just so see how it would compare with the data generated after updating with the new observations.

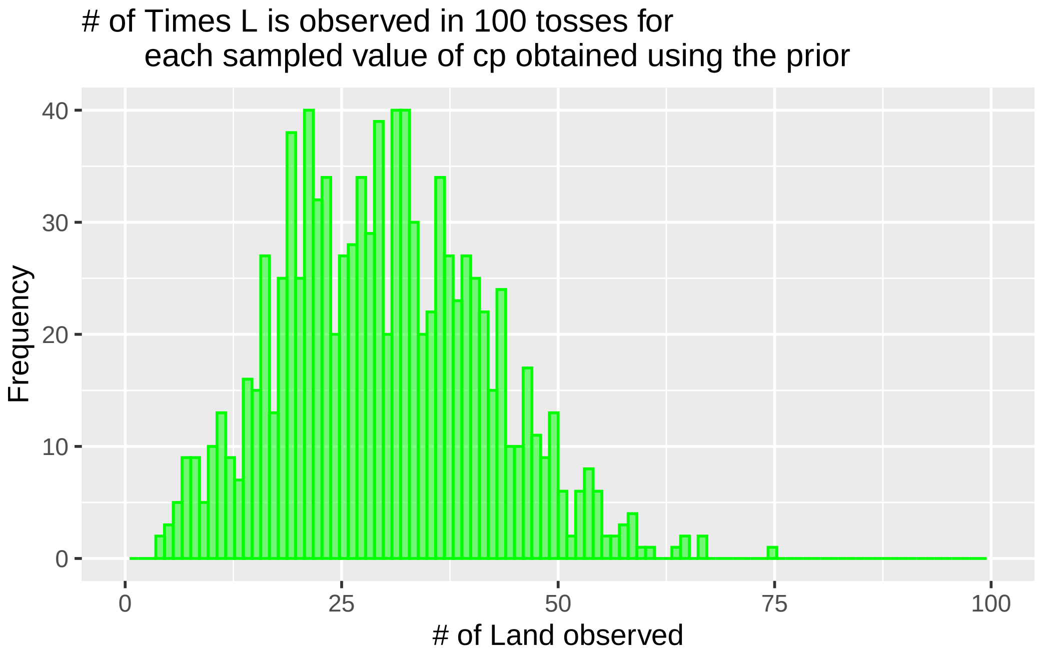
```r
# Sample from prior
N <- 1e3
sample <- sample(poss$prop_L, size=N, replace=T, prob=poss$prior)
sample <- data.frame(smp=sample)

# Predict using the prior
preds <- data.frame(L =vector("numeric", N))

for (i in seq_along(sample$smp)){
  preds[i, "L"] <- rbinom(n = 1, size = 100, prob = sample[i, "smp"])
}

preds %>% ggplot(aes(x=L)) +
  geom_histogram(fill = "green", color = "green",
                 alpha = .5, bins = 100) +
  scale_x_continuous(limits = c(0,100)) +
  labs(x = "# of Land observed",
       y = "Frequency", title = "# of Times L is observed in 100 tosses for
       each sampled value of cp obtained using the prior") +
  theme(plot.title = element_text(size=12))
```

```
Warning: Removed 2 rows containing missing values or values outside the scale range
(`geom_bar()`).
```



Now updating and using the posterior

```
# Updating with the new observations
lh <- dbinom(26, 100, prob = poss$prop_L)
posterior <- poss$prior*lh
posterior_norm <- posterior / sum(posterior)

poss$likelihood <- lh
poss$posterior <- posterior_norm
poss[ , -(1:2)] <- round(poss[ , -(1:2)], 4)
poss
```

```
# A tibble: 11 x 4
   prop_L prior likelihood posterior
    <dbl> <dbl>      <dbl>     <dbl>
 1    0    0          0         0
 2    0.1  0.07       0         0
 3    0.2  0.27       0.0316    0.282
 4    0.3  0.35       0.0613    0.709
 5    0.4  0.22       0.0012    0.0088
 6    0.5  0.07       0         0
 7    0.6  0.01       0         0
 8    0.7  0          0         0
 9    0.8  0         0          0
10    0.9  0         0          0
11    1    0         0          0
```

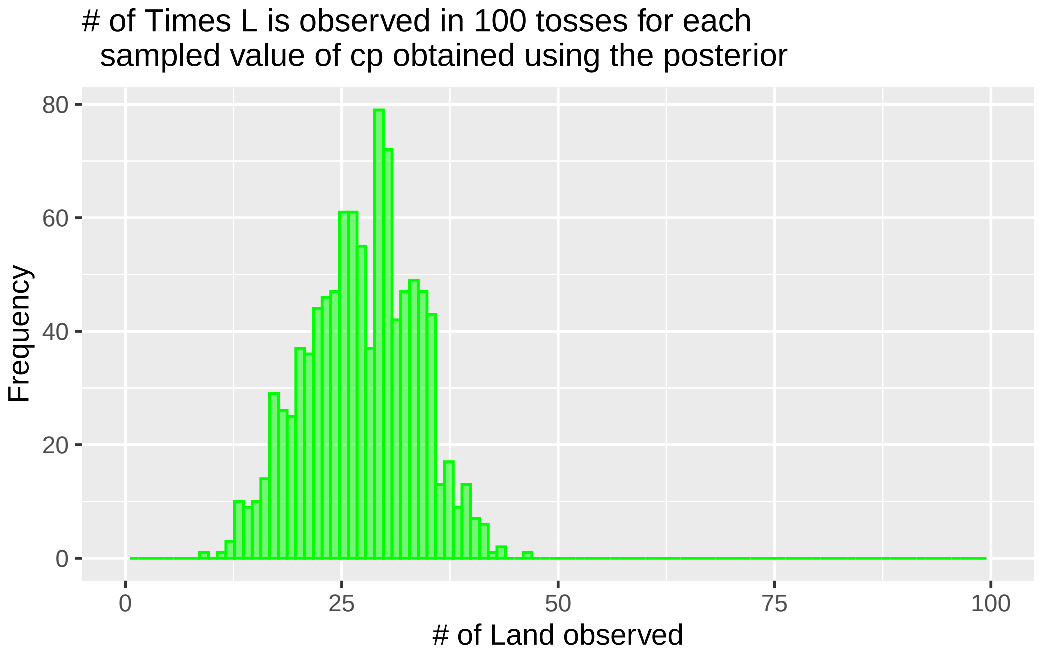
```
# Sample from posterior
N <- 1e3
post_sample <- sample(poss$prop_L, size=N, replace=T, prob=poss$posterior)
post_sample <- data.frame(smp=post_sample)

# Predict using the posterior
preds <- data.frame(L =vector("numeric", N))

for (i in seq_along(post_sample$smp)){
  preds[i, "L"] <- rbinom(n = 1, size = 100, prob = post_sample[i, "smp"])
}

preds %>% ggplot(aes(x=L)) +
  geom_histogram(fill = "green", color = "green",
                 alpha = .5, bins = 100) +
  scale_x_continuous(limits = c(0,100)) + labs(x = "# of Land observed",
  y = "Frequency", title = "# of Times L is observed in 100 tosses for each
  sampled value of cp obtained using the posterior") +
  theme(plot.title = element_text(size=12))
```

```
Warning: Removed 2 rows containing missing values or values outside the scale range
(`geom_bar()`).
```


# of Times L is observed in 100 tosses for each
sampled value of cp obtained using the posterior
Frequency
80
60
40
20
0
0
25
50
75
100
# of Land observed