Assignment-5 Q&A Bilge Salman (03796071), Salim Kaplan (0378856) GitHub link: https://github.com/salimkaplan/BayesIntro24-Assignments/tree/main

```r
# load packages here
library(rethinking)
library(tidyverse)
library(tidyr)

knitr::opts_chunk$set(tidy = TRUE, tidy.opts = list(width.cutoff = 50))
```

Load the data set RiskyChoice.csv to solve the Task Set 1. Use the read_csv2() function instead of read_csv().

```r
# Load the data set
data <- read_csv2("RiskyChoice.csv")
```

Task Set 1

1.1

Create a reduced data table with only one row per subject that shows the number of solved choices problems (nChoice) and the number of correct choices (nCorrect) for each subject along with the other variables. Remove the subjects with missing values. Print the data of the first 10 subjects.

```r
# 1.1
# Summarize data
reduced_data <- data %>%
  group_by(Subject) %>%
  summarise(
    nChoice = sum(CorrectChoice, na.rm = TRUE) + sum(RiskyChoice, na.rm = TRUE),
    nCorrect = sum(CorrectChoice, na.rm = TRUE),
    Numeracy = first(Numeracy),
    AgeGroup = first(AgeGroup),
    Gender = first(Gender)
  ) %>%
  na.omit()

print(head(reduced_data, 10))
```

```
# A tibble: 10 x 6
   Subject nChoice nCorrect Numeracy AgeGroup Gender
     <dbl>   <dbl>    <dbl>    <dbl> <chr>    <chr>
 1       1      90       61        9 younger  female
 2       2     119       66        7 younger  female
 3       3     128       80       10 younger  male
 4       4     112       71       10 younger  female
 5       5     110       58        9 younger  male
 6       6     113       69        9 younger  female
 7       7     128       75       10 younger  male
 8       8      96       66       10 younger  female
 9       9     104       62        9 younger  female
10      10     112       63        7 younger  female
```

1.2

Run a Bayesian regression model that predicts nCorrect from Numeracy using fixed intercepts and fixed slopes. Standardize the predictor before running the model and compute the WAIC of the model.

```r
# 1.2
reduced_data$Numeracy_s <- scale(reduced_data$Numeracy)

data_list <- list(
  nCorrect = reduced_data$nCorrect,
```

```
  Numeracy = reduced_data$Numeracy_s
)
```

```
# Bayesian regression model
model_fixed <- ulam(
  alist(
    nCorrect ~ dnorm(mu, sigma),
    mu <- a + b * Numeracy,
    a ~ dnorm(0, 1),
    b ~ dnorm(0, 1),
    sigma ~ dexp(1)
  ),
  data = data_list,
  chains = 4,
  cores = 4,
  log_lik = TRUE)
```

```
# WAIC
waic_fixed <- WAIC(model_fixed)
print(waic_fixed)
```

```
      WAIC      lppd   penalty  std_err
1 1344.172 -671.9007 0.1852271 4.509108
```

1.3

Run a Bayesian regression model that predicts nCorrect from Numeracy using random intercepts and fixed slopes. Standardize the predictor before running the model and compute the WAIC of the model.

```
# 1.3
# list for ulam with subject indices
data_list_random <- list(
  nCorrect = reduced_data$nCorrect,
  Numeracy = reduced_data$Numeracy_s,
  Subject = as.integer(as.factor(reduced_data$Subject)))
```

```
# Bayesian regression model
model_random <- ulam(
  alist(
```

```
    nCorrect ~ dnorm(mu, sigma),
    mu <- a[Subject] + b * Numeracy,
    a[Subject] ~ dnorm(0, 1),
    b ~ dnorm(0, 1),
    sigma ~ dexp(1)),
  data = data_list_random,
  chains = 4,
  cores = 4,
  log_lik = TRUE)
```

```
# WAIC
waic_random <- WAIC(model_random)
print(waic_random)
```

```
      WAIC      lppd   penalty  std_err
1 1354.188 -676.8693 0.2246044 4.408366
```

```
compare_models <- compare(model_fixed, model_random)
print(compare_models)
```

```
                 WAIC       SE    dWAIC       dSE     pWAIC      weight
model_fixed  1344.172 4.509108  0.00000        NA 0.1852271 0.993359966
model_random 1354.188 4.408366 10.01595 0.1032475 0.2246044 0.006640034
```

Task Set 2

2.1

Create a data table that entails 10,000 posterior samples (rows) for each subject-specific (columns) intercept. Convert the sampled values into probabilities and print the first 10 samples of the first 10 subjects.

```
# posterior samples for subject-specific intercepts
posterior_samples <- extract.samples(model_random)

subject_intercepts <- posterior_samples$a

# converting the sampled values into probabilities using the logistic function
subject_probabilities <- 1 / (1 + exp(-subject_intercepts))
```

```r
posterior_table <- as.data.frame(subject_probabilities)

print(head(posterior_table[, 1:10], 10))
```

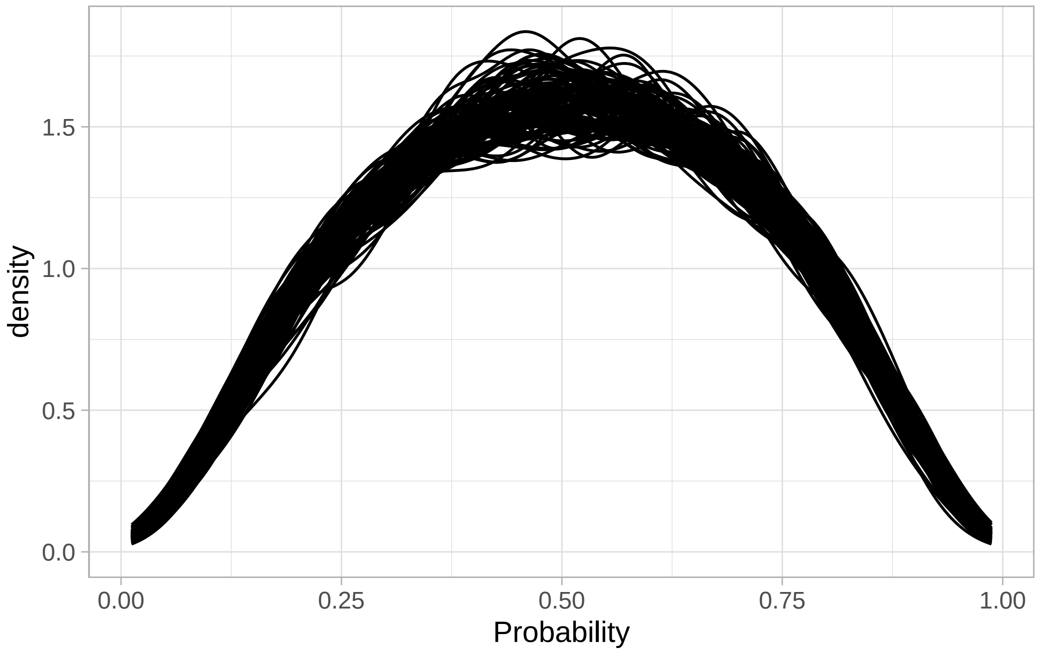
```
          V1         V2        V3        V4         V5        V6        V7
1  0.2696085 0.61871556 0.6410955 0.4637811 0.78132592 0.1610873 0.4646618
2  0.5094834 0.23740854 0.6781483 0.4910517 0.50130954 0.6113254 0.3199582
3  0.3340802 0.75669786 0.2592174 0.3119950 0.50995643 0.4335904 0.7511902
4  0.1956496 0.81462695 0.4565707 0.2190023 0.86344468 0.3101832 0.3173192
5  0.4688422 0.73969232 0.6570510 0.6626084 0.55356615 0.8313619 0.4662696
6  0.6779614 0.05831673 0.3818081 0.3606023 0.10515454 0.2060830 0.4747894
7  0.3397544 0.94380809 0.6147035 0.5888412 0.92279408 0.6939130 0.4590224
8  0.4513618 0.16550159 0.3723966 0.3751098 0.09888458 0.3376226 0.2808492
9  0.6134748 0.15640153 0.2212822 0.4947427 0.25111019 0.3471705 0.3824292
10 0.2583677 0.67378623 0.6079938 0.7777254 0.32621047 0.2968574 0.7213581
          V8        V9       V10
1  0.5322405 0.6603078 0.4737659
2  0.6114590 0.7196546 0.7475183
3  0.1893744 0.4344261 0.3329024
4  0.2996355 0.5988808 0.1197468
5  0.8500738 0.7809806 0.5931641
6  0.3435876 0.1906781 0.4315507
7  0.6292804 0.7982657 0.5815869
8  0.5660544 0.4465881 0.4737123
9  0.5118659 0.4630259 0.3149232
10 0.2542941 0.3681248 0.4702479
```

2.2

Use the posterior samples to plot the posterior distribution of all subject-specific intercepts to show the variability in the performance among subjects. Use the converted values (probabilities).

```r
# pivot data to a longer format
posterior_long <- pivot_longer(posterior_table, cols = everything(), names_to = "Subject", va

# posterior density plot
ggplot(posterior_long, aes(x = Probability, group = Subject)) +
  geom_density(alpha = 0.5) +
  theme_light()
```



2.3

Consider the following posterior summaries and trace plots. Which model was estimated and what might be the cause of the convergence problems?

- For the estimated model, the parameters mu_a and tau_a are representing the mean and standard deviation of the intercepts.
- The same applies for the parameters mu_b and tau_b for the slopes, suggesting a multilevel model with random intercepts and slopes.
- tau_b is showing signs of convergence issues, which may indicate that the model is not well-specified or that the data.
- Having a low value ess_bulk of 43.68 and an rhat of 1.11 which should have been close to 1 may indicate potential convergence issues.