Advanced Statistical Modelling - Exercise 2.5

Filip Chmielowski

2024-10-22

Package

```
# Install and load the package
#install.packages('R2jags')
library(R2jags)
```

```
## Loading required package: rjags
```

```
## Loading required package: coda
```

```
## Linked to JAGS 4.3.1
```

```
## Loaded modules: basemod,bugs
```

```
##
## Attaching package: 'R2jags'
```

```
## The following object is masked from 'package:coda':
##
##     traceplot
```

Data preparation

```
# Clearing the workspace
rm(list=ls())

# List of the boys' and girls' grades
boys <- c(9.6, 7.0, 5.0, 8.0, 8.4, 6.4)
girls <- c(6.1, 9.1, 8.8, 5.7, 8.9, 6.1, 6.5)

# Combine the data into a single data frame, where 1 is for boys, 0 is for girls
sex <- c(rep(1, length(boys)), rep(0, length(girls)))
grades <- c(boys, girls)
summary(grades)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   5.000   6.100   7.000   7.354   8.800   9.600
```

```r
data <- list(
  y = grades,          # Grades vector
  sex = sex,           # Sex vector
  N = length(grades)   # Number of students
)
```

The amount of the given grades (the size of the dataset) is very small so it is expected that there will be kind of big uncertainty in the results.

a) Write the bayesian model and justify the chosen prior distribution.

```r
# JAGS model specification with truncated (0-10) normal distribution for grades
# beta0: Expected value of the grade
# beta1: How much expected value of the grade changes for boys compared to girls
model.bug <- "
  model
  {
    for (i in 1:N)
    {
      y[i] ~ dnorm(mu[i], tau) T(0, 10)
      mu[i] <- beta0 +
               beta1 * sex[i]
    }

    # Priors
    beta0 ~ dnorm(7, 0.1) T(0, 10)  # Normal prior, mean 7, variance 10, precision 0.1
    beta1 ~ dnorm(0, 1)             # Normal prior, mean 0, variance 1, precision 1
    sigma ~ dunif(0, 1.5)           # Uniform prior, mean 0, variance 0.67, precision 1.5
    tau <- 1 / (sigma * sigma)      # Precision = 1 / variance
  }
"
```

The model uses a Bayesian linear regression approach with a normal likelihood. This model was chosen because it captures continuous data (like grades) in an effective way. The prior distributions were chosen to be weakly informative, allowing the data to influence posterior estimates.

The model estimates 'beta0' as the average grade for girls, and 'beta1' as the difference in the grades between boys and girls. The positive or negative value of 'beta1' will indicate whether boys, on average, score higher or lower grades than girls. The credibility interval around 'beta1' also provides insights into whether this difference is statistically significant or not.

The prior distribution chosen for 'beta0' circulates around the value of 7, reflecting a general expectation of the central average for the grades. I chose the normal distribution with the mean equal to 7 because I think that 7 is the expected grade of the students (not 5, which is in the center of the range 0-10, because usually most of the students pass the course). Meanwhile, 'beta1' has a mean prior of 0, reflecting no assumed differences in the grades between boys and girls.

The 'sigma' parameter's uniform prior of the interval (0, 1.5) indicates a bit conservative approach to the expected variability in the grades. That's because I assumed no large deviations in this expected variability.

b) Update the model and draw the posterior distribution for every parameter.

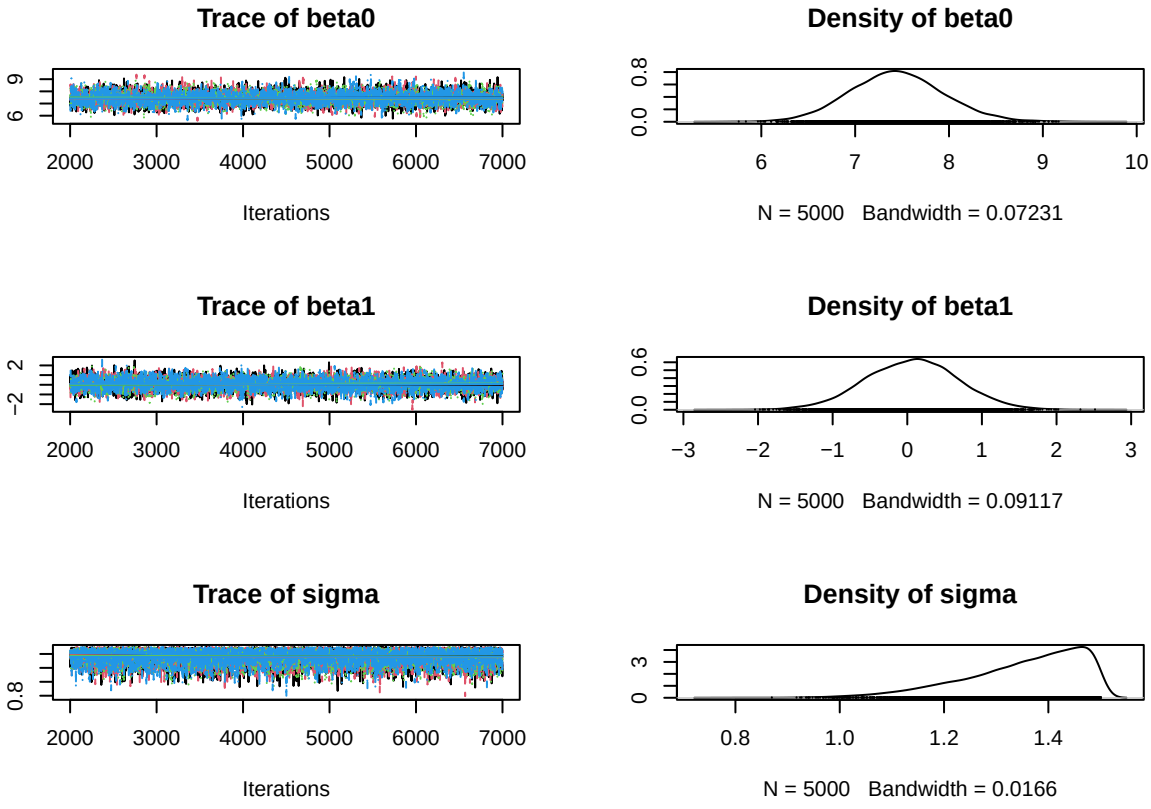
```
# Parameters of the simulations
Iter <- 5000
Burn <- 1000
Chain <- 4

# Compile the model using 4 chains
model <- jags.model(textConnection(model.bug), data = data, n.chains = Chain, n.adapt = Burn)
```

```
## Compiling model graph
##    Resolving undeclared variables
##    Allocating nodes
## Graph information:
##    Observed stochastic nodes: 13
##    Unobserved stochastic nodes: 3
##    Total graph size: 42
##
## Initializing model
```

```
# Update the model with burn-in phase of 1000 iterations
update(model, Burn)

# Sample from the posterior distribution
samples <- coda.samples(model, variable.names = c("beta0", "beta1", "sigma"), n.iter = Iter)
# I don't have to do it manually with mean() and quantile() functions,
# because the 'coda' library does it for me automatically

# Print a summary of the posterior distribution for beta1, beta2 and sigma
summary(samples)
```

```
##
## Iterations = 2001:7000
## Thinning interval = 1
## Number of chains = 4
## Sample size per chain = 5000
##
## 1. Empirical mean and standard deviation for each variable,
##    plus standard error of the mean:
##
##          Mean     SD  Naive SE Time-series SE
## beta0 7.44558 0.5015 0.0035463       0.006477
## beta1 0.06136 0.6234 0.0044081       0.007746
## sigma 1.35221 0.1135 0.0008028       0.001363
##
## 2. Quantiles for each variable:
##
##         2.5%     25%     50%    75% 97.5%
## beta0  6.487  7.1068 7.43471 7.7693 8.467
## beta1 -1.169 -0.3639 0.07159 0.4809 1.274
## sigma  1.083  1.2847 1.37648 1.4437 1.495
```

```
# It is good to know the values of the means of each of the parameters

# Plot posterior distributions for beta1, beta2 and sigma
plot(samples)
```



The posterior distributions for 'beta0', 'beta1', and 'sigma' are derived from the observed data. The posterior for 'beta0' (the mean grade for girls) and 'beta1' (the difference in the grades between boys and girls) show how the model estimates these parameters. On the plots it is possible to see that the most likely values of 'beta0' is about 7.2, of 'beta1' is about 0 and of 'sigma' is about 1.5.

c) Do you think there are differences between the grades got by the boys and the grades got by the girls?

```
# Convert the posterior samples into a matrix
samples_matrix <- as.matrix(samples)

# Extract the samples for beta1 (the difference between boys' and girls' grades)
beta1_samples <- samples_matrix[, "beta1"]

# Calculate the 95% credible interval for beta1 (the range: 2.5% - 97.5%)
credible_interval_beta1 <- quantile(beta1_samples, probs = c(0.025, 0.975))

# Print the credible interval of 95% for beta1
print("Credible interval of 95% for beta1 (difference between boys and girls): ")
```

```
## [1] "Credible interval of 95% for beta1 (difference between boys and girls): "
```

```
print(credible_interval_beta1)
```

```
##      2.5%     97.5%
## -1.169374  1.274302
```

```
# Check if the calculated credible interval of 95% for beta1 includes 0
if (credible_interval_beta1[1] > 0 | credible_interval_beta1[2] < 0)
{
  print("There is A significant difference between boys' and girls' grades.")
} else
{
  print("There is NO significant difference between boys' and girls' grades.")
}
```

```
## [1] "There is NO significant difference between boys' and girls' grades."
```

By interpreting the posterior distribution of 'beta1', we can assess if there is a significant difference between boys' and girls' grades (or not). If the 90% credible interval of 'beta1' does not contain 0, it suggests that there is a significant difference between these grades. Otherwise, there is no significant difference between them.

The final result is that there is NO significant difference between boys' and girls' grades, because the value of the first quantile of 2.5% is lower than 0 (about -1.15) and the value of the second quantile of 97.5% is bigger than 0 (about 1.30), so neither of the two conditions are met.

d) A girl could not take the exam, because she stayed trapped in an elevator. Calculate a 90% credibility interval for this student's final grade.

```
# Convert the posterior samples into a matrix
samples_matrix <- as.matrix(samples)

# Extract the posterior samples for beta0 and sigma
beta0_samples <- samples_matrix[, "beta0"]
sigma_samples <- samples_matrix[, "sigma"]

# Generate predicted grades for the new girl (sex = 0) using truncated normal bounds
predicted_grades <- rnorm(length(beta0_samples), mean = beta0_samples, sd = sigma_samples)

# Apply the truncation: any values < 0 set to 0, any values > 10 set to 10
predicted_grades <- pmin(pmax(predicted_grades, 0), 10)

# Calculate the credible interval for 90% of the predicted grade (the range: 5% - 95%)
credible_interval <- quantile(predicted_grades, probs = c(0.05, 0.95))

# Print the credible interval for 90% of the new girl's final grade
print("Credible interval of 90% for the missing girl's final grade: ")
```

```
## [1] "Credible interval of 90% for the missing girl's final grade: "
```

```
round(quantile(credible_interval, c(0.05, 0.95)), 3)
```

```
##    5%   95%
## 5.303 9.572
```

To predict the grade of a girl who missed the exam, the model can calculate the posterior predictive distribution based on the observed grades of other girls. The 90% credibility interval around this prediction gives a range where her grade would be likely to fall.

The 90% credibility interval of the missing girl's final grade is likely to fall in the range between about 5.31 and about 9.59 - these are the values of the quantiles of 5% and 95% respectively.