CGS698C, Assignment 02

Himanshu Yadav

Part 1: A simple binomial model

Suppose you do an experiment and collect data y. You assume the following:

(a) **The likelihood assumption**

The data in your experiment is generated by a probability mass function $f(x;\theta)$, such that $f(x;\theta) = \frac{10!}{x!(10-x)!}\theta^x(1-\theta)^{10-x}$

You can express the probability mass function as a likelihood function. Suppose the observed data is y, you can write the likelihood function as:

$$\mathcal{L}(\theta|y) = \frac{10!}{y!(10-y)!}\theta^y(1-\theta)^{10-y} \tag{1}$$

The above likelihood function is a function of θ when y is fixed.

(b) **The prior assumption**

The parameter θ of associated with the generative process has a real number value somewhere between 0 and 1, and each value between 0 and 1 is equally likely. This assumption can be expressed by a probability density function, say $f(\theta)$. Since $f(\theta)$ represent our prior assumption/belief about θ, we can call it the **prior density** or the prior distribution represented by $p(\theta)$. You assume the following about the prior density of θ

$$p(\theta) = \begin{Bmatrix} 1 & \text{when } 0 \leq \theta \leq 1 \\ 0 & \text{when } \theta < 0 \text{ or } \theta > 1 \end{Bmatrix} \tag{2}$$

We want to know what is the probability density of a particular value of θ that it would generate the data y under the assumptions (a) and (b). This probability density assigned to a particular value of θ given the data y and the assumptions is called the **posterior density**.

We want to infer the approximate function that assigns the posterior densities to each value of θ. This posterior density assigner function is called the **posterior distribution** of θ and is represented by $p(\theta|y)$.

The posterior distribution of θ is given by the Bayes' rule:

$$p(\theta|y) = \frac{\mathcal{L}(\theta|y)p(\theta)}{\int \mathcal{L}(\theta|y)p(\theta)\,d\theta} \tag{3}$$

You already know the likelihood function $L(\theta|y)$ (see Equation 1) and the prior density function $p(\theta)$ (see Equation 2).

Suppose you are given that

- The data: $y = 7$

- The marginal likelihood: $\int \mathcal{L}(\theta|y)p(\theta)\,d\theta = \frac{1}{11}$

You can use the above information and the Bayes' rule (Equation 3) to calculate the posterior density $p(\theta|y)$ for each value of θ.

1.1 Estimate the posterior density for the following values of θ.

(a) $\theta = 0.75$

(b) $\theta = 0.25$

(c) $\theta = 1$

1.2 Graph the posterior distribution of θ.

(Hint: Create a vector containing a lot of equidistant values of θ between say 0 and 1; calculate posterior density for each value in the vector; plot a graph with values of θ on the x-axis and associated posterior densities $p(\theta|y)$ on the y-axis.)

1.3 What value of θ has the maximum posterior density?

1.4 Compare the graphs of the likelihood function, the prior distribution, and the posterior distribution.

Part 2: A Gaussian model of reading

Suppose you do a reading experiment and collect reading times data y. The data y contains n independent and identically distributed datapoints $y_1, y_2, ..., y_n$. You make the following assumptions about the underlying generative process:

(a) **The likelihood assumption**

Each datapoint y_i comes from a normal distribution with mean μ and standard deviation σ:

$y_i \sim Normal(\mu, \sigma)$

The joint likelihood of the data $y_1, y_2, ..., y_n$ can be given by:

$\mathcal{L}(\mu, \sigma|y) = \frac{1}{(\sigma\sqrt{2\pi})^n} e^{-\frac{1}{2\sigma^2}\sum_{i=1}^{n}(y_i-\mu)^2}$

(b) **The prior assumptions**

Based on the previous research on the topic, you make the following assumptions about the standard deviation σ and the mean reading times μ.

$\sigma = 50$

$\mu \sim Normal(250, 25)$

The product of the likelihood and the prior density gives you **unnormalized posterior density**. The unnormalized posterior density of μ is given by

$$p'(\mu|\sigma, y) = \mathcal{L}(\mu, \sigma|y)p(\mu) \tag{4}$$

Suppose the observed reading times data y consists of 8 observations: 300, 270, 390, 450, 500, 290, 680, 450.

Answer the following:

2.1 Given the above data y and the likelihood and prior assumptions, calculate the unnormalized posterior density for the following values of μ.

- $\mu = 300$
- $\mu = 900$
- $\mu = 50$

2.2 Graph the **unnormalized** posterior distribution of μ.

2.3 Compare the graphs of the prior and the (unnormalized) posterior distribution of μ.

Part 3: The Bayesian learning

Model $\mathcal{M}1$ is a model for predicting the number of road accidents in single day in Berlin.

The likelihood of the model is given by:

$$\mathcal{L}(\lambda|k) = \frac{\lambda^k e^{-\lambda}}{k!}$$

where k is the number of road accidents, and λ is the parameter to encode the average rate of accidents in Berlin.

And, the prior on the parameter λ is given by:

$$\lambda \sim Gamma(40, 2)$$

We can derive the posterior distribution of λ analytically

$$\lambda \sim Gamma(40 + k, 3)$$

the distribution $Gamma(40 + k, 3)$ is the posterior distribution of λ after seeing the data k. Here, k is the number of road accidents in a day.

Suppose we collect data from 4 consecutive days. The data (number of road accidents) from 4 days is shown below:

25, 20, 23, 27

The model generates predictions for day 1 using the $Gamma(40, 2)$ prior.

Based on data from day 1 and the prior $Gamma(40, 2)$, the posterior disribution of λ is estimated. This posterior estimated from day 1 data becomes the prior for predicting about day 2.

The same process is repeated such that we obtain a new posterior of λ after seeing each day's data.

Finally, we obtain a posterior after fitting the model on day 4 data. This final posterior becomes the prior to predict the number of accidents on day 5.

3.1 What will be the prior on λ to generate predictions for day 5?

3.2 How many road accidents are predicted to happen on day 5?

Part 4: Model building in the Bayesian framework

4.1 The research problem

In a visual word recognition experiment, a participant has to recognize whether a string shown on the screen is a meaningful word (e.g., "book") or a non-word (e.g., "bktr"). The participant is asked to answer "yes" if the shown string is a meaningful word, and "no" if it is a meaningless non-word. Suppose a participant is shown n words and n non-words on the screen one by one and you record the recognition time for each word/non-word.

Say, T_w is the vector of word recognition times, and T_{nw} is the vector of non-word recognition times.

You ask the following question:

Does it take longer to recognize the non-words compared to the words?

Technically,

Is the mean recognition time for the non-words larger than the mean recognition time for the words?

4.2 Hypotheses

Null hypothesis: The mean recognition time for the words is equal to the mean recognition time for the non-words.

Lexical-access hypothesis: The mean recognition time for the words is longer than the mean recognition time for the non-words.

4.3 Models

A model is a set of statistical assumptions about the underlying generative process. A model should be able to generate data corresponding to a particular value(s) of the parameter(s) and it should be able to generate predictions based on the prior assumptions about the parameters.

We will express our models corresponding to each hypohesis using our assumptions about the likelihood and the priors.

Null hypothesis model

T_w is the vector of word recognition times; T_{nw} is the vector of non-word recognition times.

Likelihood:

$$T_w \sim Normal(\mu, \sigma)$$

$$T_{nw} \sim Normal(\mu + \delta, \sigma)$$

Priors:

$$\mu \sim Normal(300, 50)$$

$$\sigma = 60$$

$$\delta = 0$$

Lexical-access model
Likelihood:

$$T_w \sim Normal(\mu, \sigma)$$

$$T_{nw} \sim Normal(\mu + \delta, \sigma)$$

Priors:

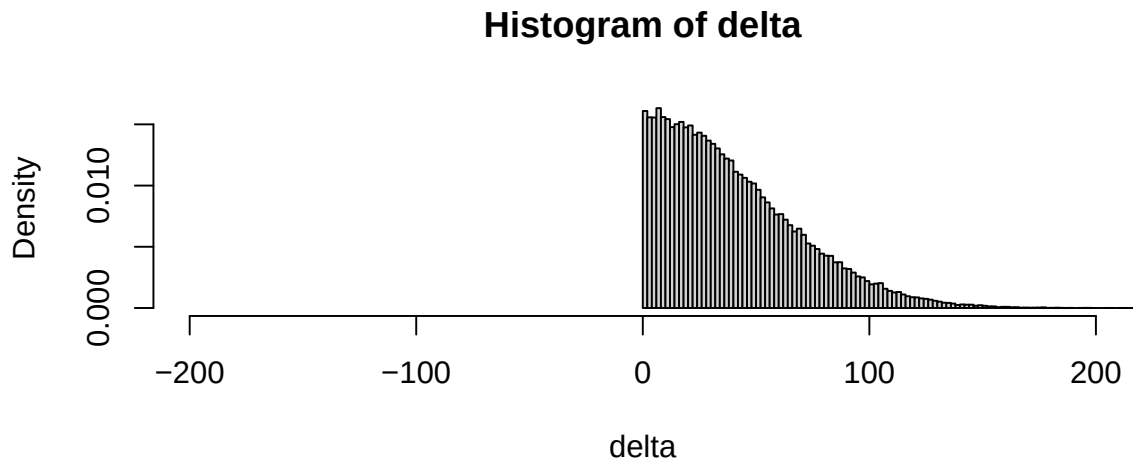
$$\mu \sim Normal(300, 50)$$

$$\sigma = 60$$

$$\delta \sim Normal_+(0, 50)$$

where $Normal_+()$ represent a truncated normal distribution such that δ would always be larger than 0.

```
library(truncnorm)
# You can generate from a truncated normal distribution using rtruncnorm
delta <- rtruncnorm(100000,a=0,b=Inf,mean=0,sd=50)
hist(delta,xlim = c(-200,200),probability = T,breaks = 100)
```




```
# You can calculate the probability density of obtaining a value x
# from the truncated normal distribution.
x <- 20
density_x <- dtruncnorm(x,a=0,b=Inf,mean=0,sd=50)
```

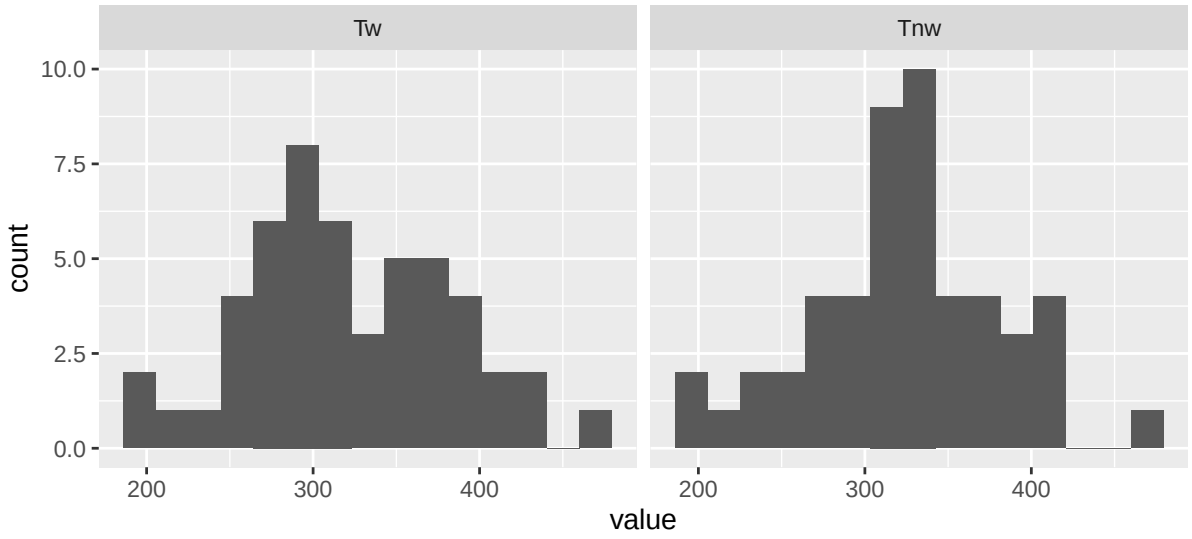
4.4 Data

The file *recognition.csv* on github contains the recognition times data for the words and non-words represented by the columns *Tw* and *Tnw*. Use the following code to load the data. (Alternatively, you can directly download the csv file and read it using `read.table`.)

```
dat <- read.table(
  "https://raw.githubusercontent.com/yadavhimanshu059/CGS698C/main/notes/Module-2/recognition.csv",
  sep=",",header = T)[,-1]
head(dat)
```

```
##         Tw      Tnw
## 1 285.0780 296.8060
## 2 267.5184 280.1157
## 3 289.9203 310.4417
## 4 399.0674 324.8276
## 5 359.9884 373.8152
## 6 403.3993 269.8220
```

```
## No id variables; using all as measure variables
```



4.5 Exercises

The unnormalized posterior density of μ for the Null hypothesis model is given by:

$$p'(\mu, \delta | T_w, T_{nw}) = \mathcal{L}(\mu, \delta, \sigma | T_w) \mathcal{L}(\mu, \delta, \sigma | T_{nw}) p(\mu) p(\delta)$$

You can calculate the term $\mathcal{L}(\mu, \delta | T_w) \mathcal{L}(\mu, \delta | T_{nw}$ for any value of μ and δ using the code $prod(dnorm(T_w, mean = \mu, sd = \sigma)) * prod(dnorm(T_{nw}, mean = \mu + \delta, sd = \sigma))$

and, $p(\mu)$ can be calculated using $dnorm(\mu, 300, 50)$, and similarly, $p(\delta)$ can be calculated using $dtruncnorm(\delta, a = 0, b = Inf, mean = 0, sd = 50)$ (if δ has a prior.)

4.5.1 Graph the unnormalized posterior distribution of μ for the Null hypothesis model.

4.5.2 Generate the prior predictions from the lexical-access model.

(Hint: Draw a vector of values for μ from its prior distribution $\mathcal{N}(300, 50)$; Draw a vector of values for δ from its prior $\mathcal{N}_+(0, 50)$. Plug each set of values of μ and δ in the generative process $\mathcal{N}(\mu + \delta, \sigma = 60)$ to generate non-word recognition times and plug them in $\mathcal{N}(\mu, \sigma = 60)$ to generate the word recognition times. Plot the recognition times as a histogram.)

4.5.3 Compare the prior predictions of the null hypothesis model and the lexical access model.

4.5.4 Compare the prior predictions of each model against the observed data T_w and T_{nw}. Which model seems more consistent with the data?

4.5.5 Graph the unnormalized posterior distribution of δ for the lexical-access model.