

Home Study 1: To be completed by Monday 26th October

MAS2905: Probability and Bayesian Methods with R

Newcastle University
`lee.fawcett@ncl.ac.uk`

Semester 1, 2020/21

What you should have done so far...

- Attended the first synchronous online lecture, 12-1pm Wednesday 21st October
 - Course overview
 - R preliminaries: installation, data types and vectors
- Worked through **Example 2.1**: You should now be ready to check your answers!
- If you've not done the above, you must do so before completing the activities in this video

Example 2.1: Solutions

Answers for x

(a) What type of data is in the vector x?

```
is.double(x)  
[1] TRUE
```

(b) How many values are in the vector x?

```
length(x)  
[1] 9764
```

(c) What is the smallest value in the vector x?

```
min(x)  
[1] -3.808057
```

(d) What is the value of x in position 3500?

```
x[3500]  
[1] -0.33692
```

(e) What is the 97th smallest value in x?

```
sort(x)[97]  
[1] -2.398929
```

(f) Find $\sum_{i=5612}^{8664} x_i$.

```
sum(x[5612:8664])  
[1] -54.8225
```

Answers for y

(a) What type of data is in the vector y?

```
is.logical(y)  
[1] TRUE
```

(b) What is the first value of y?

```
y[1]  
[1] FALSE
```

(c) What is the last value of y?

```
y[length(y)]  
[1] FALSE
```

(d) How many times does TRUE occur?

```
length(y[y==TRUE])  
[1] 4969
```

(e) When is the 472nd occurrence of TRUE?

```
which(y==TRUE)[472]  
[1] 953
```

(f) How many occurrences of TRUE are there between positions 504 and 593 (inclusive)?

```
sum(y[504:593])  
[1] 49
```

2.4 Dataframes

A dataframe is a special kind of object in R.

We use dataframes for storing and managing datasets that have a rectangular structure.

Typically the rows correspond to **cases** and the columns **variables**.

The crucial difference between a dataframe and a matrix is that all values in a matrix must be of the same type, whereas in a dataframe we can have responses for cases on variables of different types.

2.4 Dataframes

For example, we might have survey results on 10 individuals, who have each been asked 5 questions.

- This would give a dataframe with 10 rows and 5 columns
- We call this a “**10 by 5 dataframe**” (generally, we refer to $r \times c$ dataframes)
- The variable types need not all be the same:
 - The first column could be a vector of doubles recording the individuals' height in metres
 - The second column could be a vector of logicals taking the value TRUE if they like Marmite
 - The third column could be a vector of characters taking the name of their favourite book, etc.

2.4.1 Creating and querying your own dataframe

You can construct your own dataframe using R's `data.frame` command. For example:

```
1 height_m = c(1.81, 1.75, 1.58)
2 weight_kg = c(NA, NA, 70.3)
3 marmite = c(TRUE, FALSE, FALSE)
4 book = c("Midnight in Chernobyl", "To Kill a
           Mockingbird", "Life of Pi")
5
6 (small_survey = data.frame(height_m, weight_kg, marmite
                             , book))
7
```

	height_m	weight_kg	marmite	book
1	1.81	NA	TRUE	Midnight in Chernobyl
2	1.75	NA	FALSE	To Kill a Mockingbird
3	1.58	70.3	FALSE	Life of Pi

2.4.1 Creating and querying your own dataframe

If you had a large dataframe and wanted to quickly know its size, you could use the `dim` command. For example:

```
1 dim(small.survey)
2 [1] 3 4
```

2.4.1 Creating and querying your own dataframe

We can extract elements from a dataframe using square brackets (just as we demonstrated in Section 2.3.2 for vector extraction).

If the columns have labels, we can also use the \$ command.

```
1 small.survey[2, 4]      # Extract row 2, column 4
2 [1] To Kill a Mockingbird
3
4 small.survey[, 4]      # Extract all rows from column 4
5 [1] Midnight in Chernobyl    To Kill a Mockingbird
6     Life of Pi
7
8 small.survey$book      # Does exactly the same as line 4!
9 [1] Midnight in Chernobyl    To Kill a Mockingbird
10     Life of Pi
11
12 small.survey[3, ]      # Extract all columns from row 3
13     height_m weight_kg marmite      book
14 3         1.58      70.3    FALSE Life of Pi
```

2.4.1 Creating and querying your own dataframe

Though used less, the `subset` command can be applied to extract information from selected columns only. For example:

```
1 (small.survey2 = subset(small.survey, select = 3:4))  
2   marmite                book  
3 1    TRUE Midnight in Chernobyl  
4 2   FALSE To Kill a Mockingbird  
5 3   FALSE           Life of Pi
```

And what if you wanted the height and favourite book of respondents who didn't like Marmite?

```
1 subset(small.survey, marmite == FALSE, select=c(1,4))  
2   height_m                book  
3 2     1.75 To Kill a Mockingbird  
4 3     1.58           Life of Pi
```

2.4.1 Creating and querying your own dataframe

Though used less, the `subset` command can be applied to extract information from selected columns only. For example:

```
1 (small.survey2 = subset(small.survey, select = 3:4))
2   marmite                book
3 1    TRUE Midnight in Chernobyl
4 2   FALSE To Kill a Mockingbird
5 3   FALSE      Life of Pi
```

And what if you wanted the height and favourite book of respondents who didn't like Marmite?

```
1 subset(small.survey, marmite == FALSE, select=c(1,4))
2   height_m                book
3 2    1.75 To Kill a Mockingbird
4 3    1.58      Life of Pi
```

2.4.2 The movies dataframe

The Movies dataset is available from within the `ncldata` package.

Recall that you can access this in the following way:

```
1 install.packages("ncldata", repos="http://R-Forge.R-  
   Project.org", type="source")  
2 library(ncldata)  
3 data(movies)
```

We will now explore the movies dataset in RStudio.

2.4.2 The movies dataframe

The Movies dataset is available from within the `ncldata` package.

Recall that you can access this in the following way:

```
1 install.packages("ncldata", repos="http://R-Forge.R-  
    Project.org", type="source")  
2 library(ncldata)  
3 data(movies)
```

We will now explore the movies dataset in RStudio.

2.4.2 The movies dataframe

The Movies dataset is available from within the `ncldata` package.

Recall that you can access this in the following way:

```
1 install.packages("ncldata", repos="http://R-Forge.R-  
    Project.org", type="source")  
2 library(ncldata)  
3 data(movies)
```

We will now explore the movies dataset in RStudio.

Example 2.2



Follow the steps at the start of Section 2.4.2 to load the `ncldata` package and to read in the `movies` dataframe. Then answer the following questions.

- (a) How many people submitted votes for the films *“10 Things I Hate About You”*, *“Midnight in the Garden of Good and Evil”* and *“Zoolander”*?
- (b) How many films are classified as “Action”? How many are classified as “Romance”? How many “Short” films are there in the database?
- (c) How many films were made between the Years 1970 and 2000 (inclusive)?
- (d) How many movies are longer than two hours in duration?
- (e) How many movies made since the year 2005 (not including 2005) are longer than three hours in duration?

Example 2.3



Load and explore the `music` dataset from the `ncldata` package by typing the following lines of code:¹

```
1 data(music)
2 # What's this dataset all about? Let's find out
3 ?music
```

¹This will only work if you have correctly installed the `ncldata` package and read the package into your working RStudio space using the `library` command; see example 2.2 and Section 2.4.2)

Example 2.3

- (a) How many tracks do we have data for, and on how many variables?
- (b) How many tracks do we have from *The Beatles* in this dataframe? What about *Lady GaGa*?
- (c) How many tracks have a tempo of at least 130 beats per minute?
- (d) Of the tracks with a tempo of at least 130 beats per minute, how many have been released since the year 2000 (including the year 2000)?
- (e) What is the most popular key for tracks in this database?

Home Study 1: To be completed by Monday 26th October

- That's this Home Study activity completed. Well done!
- Before the end of **Monday 26th October**, you should:
 - Complete **Examples 2.2** and **2.3**
 - Watch the short video "**Home Study 2**", which will reveal answers to Examples 2.2 and 2.3 before covering Section 2.5 of the course notes; work through **Example 2.4**
- You must complete these activities by the dates shown, to help prepare you for next week's activities.

Home Study 1: To be completed by Monday 26th October

- That's this Home Study activity completed. Well done!
- Before the end of **Monday 26th October**, you should:
 - Complete **Examples 2.2** and **2.3**
 - Watch the short video “**Home Study 2**”, which will reveal answers to Examples 2.2 and 2.3 before covering Section 2.5 of the course notes; work through **Example 2.4**
- You must complete these activities by the dates shown, to help prepare you for next week's activities.