Practical 2 solutions

Dr Colin S. Gillespie

Often when we commence an analysis we want to partition the data in different ways. For example, selecting data points greater than a particular value. In this practical we will investigate how this is done using R's logical operators.

If you have your own data set, try the questions below, then load your own data.

Question difficulty

Some of the questions below are straight forward, others are bit more tricky.

- The last sub-questions of Q1 and Q2 are hard.

Before starting the practical, run the following commands:

```
library(nclRintroduction)
```

Question 1

Run the following R code

```
x1 = GetNumericVector()
```

1. What is length of `x1`?

```
length(x1)

## [1] 52655
```

2. What is the 55^{th} element of `x1`?

```
x1[55]

## [1] 38.3
```

3. What is the final element of `x1`?

```
x1[length(x1)]

## [1] -3.3
```

4. What is the mean value of `x1`?

```
mean(x1)

## [1] -1.334
```

5. What is the smallest value of x1?

```
min(x1)

## [1] -87.3
```

6. How many values are greater than the first quartile but less than the median?

```
q1 = quantile(x1)[2]
med = median(x1)
length(x1[x1 > q1 & x1 < med])

## [1] 13048

## Or
sum(x1 > q1 & x1 < med)

## [1] 13048
```

7. How many values are greater than the $\bar{x}_1 + 2sd(x_1)$, where sd is the sample standard deviation?

$\bar{x}_1$ is the mean value and $sd(x_1)$ is the standard deviation of x_1.

```
length(x1[x1 > (mean(x1) + 2 * sd(x1))])

## [1] 1254

## Or
sum(x1 > (mean(x1) + 2 * sd(x1)))

## [1] 1254
```

8. **Tricky:** What is the 50^{th} smallest value in x1?

```
sort(x1)[50]

## [1] -63.7
```

Question 2

Run the following R code

```
y = GetDataFrame()
```

The data frame y is a subset of the movie data we use in the lectures.

1. How many rows does y have?

```
dim(y)[1]

## [1] 4784
```

```
## Or
nrow(y)

## [1] 4784
```

2. How many columns does y have?

```
dim(y)[2]

## [1] 24

## Or
ncol(y)

## [1] 24
```

3. How many movies are there where budget is known, i.e. where `Budget` $!= -1$?

```
nrow(y[y$Budget != -1, ])

## [1] 1763

## Or
sum(y$Budget != -1)

## [1] 1763
```

4. How many movies are there where the rating is less than 2.5 or greater than 7.5 (not including 2.5 and 7.5)? How would you include the values 2.5 and 7.5 in your condition?

```
nrow(y[y$Rating < 2.5 | y$Rating > 7.5, ])

## [1] 406

##
sum(y$Rating < 2.5 | y$Rating > 7.5)

## [1] 406
```

5. How many movies are there where the length is greater than 120 and have a rating of more than 7.5 - (not including 120 and 7.5)?

```
nrow(y[y$Length > 120 & y$Rating > 7.5, ])

## [1] 111
```

```
## Or
sum(y$Length > 120 & y$Rating > 7.5)

## [1] 111
```

6. How many movies were made in 1980 and have a rating above 5.0?

```
nrow(y[y$Year == 1980 & y$Rating > 5, ])

## [1] 4

## Or
sum(y$Year == 1980 & y$Rating > 5)

## [1] 4
```

7. How many movies are classified as Action?

```
sum(y$Action)

## [1] 913
```

8. How many movies were classified as both Action and Animation?

```
sum(y$Action == 1 & y$Animation == 1)

## [1] 19
```

Question 3

Run the following R code

```
x2 = GetCharacterVector()
```

In the following questions, the function `table` is quite useful, especially when combined with `sum`.

1. How many times does "A" appear in `x2`?

```
length(x2[x2 == "A"])

## [1] 2095

## Or
sum(x2 == "A")

## [1] 2095
```

2. Which letter appears the most? If more than one letter appears, just give the first letter (if the letters were sorted in alphabetical order).

```
sort(table(x2), decreasing = TRUE)[1]

##    W
## 2108
```

3. **Very tricky:** How many pairs of letters are there in `x2`.[1]

[1] For example, in `AABCCC` we would have 3 pairs: `AA`, `CC` and `CC`.

```
l = length(x2)
x2_a = x2[1:(l - 1)]
x2_b = x2[2:l]
sum(x2_a == x2_b)

## [1] 2074
```