Introduction to R: practical 2
Dr Colin 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.

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 command:

```r
library("nclRcourses")
```

Question 1

Run the following R code

```r
x1 = GetNumericVector()
```

1. What is length of x1?

```r
length(x1)
```

```
## [1] 52655
```

2. What is the 55th element of x1?

```r
x1[55]
```

```
## [1] 38.3
```

3. What is the final element of x1?

```r
x1[length(x1)]
```

```
## [1] -3.3
```

```r
tail(x1, 1)
```

```
## [1] -3.3
```

4. What is the mean value of x1?

```r
mean(x1)
```

```
## [1] -1.334
```

5. What is the smallest value of x1?

```r
min(x1)
```

```
## [1] -87.3
```

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

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

```
## [1] 13048
```

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

```
## [1] 13048
```

7. How many values are greater than $\bar{x}_1 + 2sd(x_1)$, 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.

```r
m = mean(x1)
s = sd(x1)
sum(x1 > (m + 2 * s))
```

```
## [1] 1254
```

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

```r
sort(x1)[50]
```

```
## [1] -63.7
```

Question 2

Run the following R code

```r
y = GetDataFrame()
```

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

1. How many rows does y have?

```r
dim(y)[1]
```

```
## [1] 14937
```

2. How many columns does y have?

```r
dim(y)[2]
```

```
## [1] 5
```

3. What are the different cell types in this data set?

```r
unique(y$type)
```

```
## [1] mutant wild
## Levels: mutant wild
```

4. How many measurements have been made on `mutant` cells?

```r
sum(y$type == "mutant")
```

```
## [1] 7468
```

5. How many of probes have expression levels less than 5?

```r
sum(y$value < 5)
```

```
## [1] 456
```

6. How many measurements have been made at time point 0?

```r
sum(y$tps == 0)
```

```
## [1] 2985
```

7. How many mutant probes were measured at time point 0?

```r
sum(y$tps == 0 & y$type == "mutant")
```

```
## [1] 1493
```

Question 3

Run the following R code

```r
x2 = GetCharacterVector()
```

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

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

    ```r
    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).

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

    ##    W
    ## 2108
    ```

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

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

    ## [1] 2074
    ```

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