Lecture 5 Module 2: Sample Counts and Proportions

Exercises

Exercises

Exercise 2.1: Sample sum and proportion

Here's the vector `second.logical.vector`:

```
second.logical.vector <-
    c(TRUE, FALSE, TRUE, TRUE, FALSE)
```

Using the `sum()` function, determine the total number of elements of `second.logical.vector` that have the value `TRUE`.

Using the `mean()` function, determine the proportion of elements of `second.logical.vector` that have the value `TRUE`.

Solution

Exercise 2.2: Sample sum and proportion

Consider this numeric vector:

```
second.numeric.vector <-
    c( -8.4, 7.3, 2.5, 0.1, -4.4, 6.3, 1.9 )
```

How many elements of this vector are greater than or equal to 4.2?

What proportion of elements of this vector are strictly less than 1.8?

Solution

Solutions to the Exercises

Exercise 2.1: Sample sum and proportion

Here's the vector `second.logical.vector`:

```
second.logical.vector <-
    c(TRUE, FALSE, TRUE, TRUE, FALSE)
```

Using the `sum()` function, determine the number of elements of `second.logical.vector` that have the value `TRUE`.

Using the `mean()` function, determine the proportion of elements of `second.logical.vector` that have the value `TRUE`.

Solution

To count the number of elements of the vector that have the value `TRUE`, we use the `sum()` function:

```
sum( second.logical.vector )
```

```
## [1] 3
```

To determine the proportion of elements of the vector that have the value `TRUE`, we use the `mean()` function:

```
mean( second.logical.vector )
```

```
## [1] 0.6
```

Exercise 2.2: Sample sum and proportion

Consider this numeric vector:

```
second.numeric.vector <-
    c( -8.4, 7.3, 2.5, 0.1, -4.4, 6.3, 1.9 )
```

How many elements of this vector are greater than or equal to 4.2?

What proportion of elements of this vector are strictly less than 1.8?

Solution

How many elements of this vector are greater than or equal to 4.2?

```
sum( second.numeric.vector >= 4.2 )
```

```
## [1] 2
```

What proportion of elements of this vector are strictly less than 1.8?

```
mean( second.numeric.vector < 1.8 )
```

```
## [1] 0.4285714
```