Lecture 5 Module 3: Logical Indexing

Exercises

Exercises

Exercise 3.1: Logical indexing

Construct a logical indexing vector to select the elements "Cat", "Mouse", and "Hedgehog" from `character.vector`.

Solution

Exercise 3.2: Using a comparison operation

The `rivers` vector is a built-in vector consisting of the lengths of the longest 141 rivers in North America.

Select the lengths of the rivers that are strictly longer than 1000 miles.

Solution

Exercise 3.3: Brand-specific sum and mean

We have a small dataset of 7 transactions, and for each transaction we have the brand of cereal and the number of boxes that were purchased:

Transaction	Brand Name	Number of Boxes
1	SBZ	4
2	SBZ	6
3	KYM	3
4	SBZ	5
5	HKT	1
6	KYM	2
7	HKT	2

Let's put the brand and number of boxes sold into vectors:

```
brand.name.vector <-
    c( "SBZ", "SBZ", "KYM", "SBZ",
       "HKT", "KYM", "HKT" )

number.of.boxes.sold.vector <-
    c( 4, 6, 3, 5, 1, 2, 2 )
```

Using logical indexing, determine the total number of boxes of Sugar Bomz sold, and also the average number of boxes sold per transaction.

Solution

Solutions to the Exercises

Exercise 3.1: Logical indexing

Let's construct a vector of character strings:

```
character.vector <-
    c( "Red", "Cat", "Dog", "Pen", "Phone",
       "White", "Mouse", "Bird", "Piano", "Bob",
       "Anita", "Hedgehog" )
```

Construct a logical indexing vector to select the elements "Cat", "Mouse", and "Hedgehog" from `character.vector`.

Solution

For your reference, here's the `character.vector`:

```
character.vector
```

```
##  [1] "Red"      "Cat"      "Dog"      "Pen"      "Phone"    "White"
##  [7] "Mouse"    "Bird"     "Piano"    "Bob"      "Anita"    "Hedgehog"
```

Now we'll construct this logical vector:

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

Now we'll perform logical indexing:

```
character.vector[ logical.indexing.vector ]
```

```
## [1] "Cat"      "Mouse"    "Hedgehog"
```

Exercise 3.2: Using a comparison operation

The `rivers` vector is a built-in vector consisting of the lengths of the longest 141 rivers in North America.

Using logical indexing with a vectorized comparison to construct a vector with the values of the lengths of the rivers that are strictly longer than 1000 miles.

Solution

```
rivers[ rivers > 1000 ]
```

```
##  [1] 1459 1450 1243 2348 1171 3710 2315 2533 1306 1054 1270 1885 1100 1205 1038
## [16] 1770
```

Exercise 3.3: Brand-specific sum and mean

We have a small dataset of 7 transactions, and for each transaction we have the brand of cereal and the number of boxes that were purchased:

Transaction	Brand Name	Number of Boxes
1	SBZ	4
2	SBZ	6
3	KYM	3
4	SBZ	5
5	HKT	1
6	KYM	2
7	HKT	2

Let's put the brand and number of boxes sold into vectors:

```
brand.name.vector <-
    c( "SBZ", "SBZ", "KYM", "SBZ",
       "HKT", "KYM", "HKT" )

number.of.boxes.sold.vector <-
    c( 4, 6, 3, 5, 1, 2, 2 )
```

Using logical indexing, determine the total number of boxes of Sugar Bomz sold, and also the average number of boxes sold per transaction.

Solution

```
sbz.logical.indexing.vector <-
    brand.name.vector == "SBZ"

sbz.number.of.boxes.sold.vector <-
    number.of.boxes.sold.vector[
        sbz.logical.indexing.vector
    ]

sbz.total.boxes.sold <-
    sum( sbz.number.of.boxes.sold.vector )

cat(
    "Total number of boxes of Sugar Bomz sold:",
    formatC(
        sbz.total.boxes.sold,
        format = "f",
        digits = 2
    )
)
```

```
## Total number of boxes of Sugar Bomz sold: 15.00
```

Now for the mean number of boxes of Sugar Bomz sold per transaction:

```r
sbz.mean.boxes.sold <-
    mean( sbz.number.of.boxes.sold.vector )

cat(
    "Mean number of boxes of Sugar Bomz sold:",
    formatC(
        sbz.mean.boxes.sold,
        format = "f",
        digits = 2
    )
)
```

```
## Mean number of boxes of Sugar Bomz sold: 5.00
```