Descriptive Statistics With R Software

Calculations with R Software

:::

Built-in Commands and Missing Data Handling

Shalabh

Department of Mathematics and Statistics

Indian Institute of Technology Kanpur

Built in commands

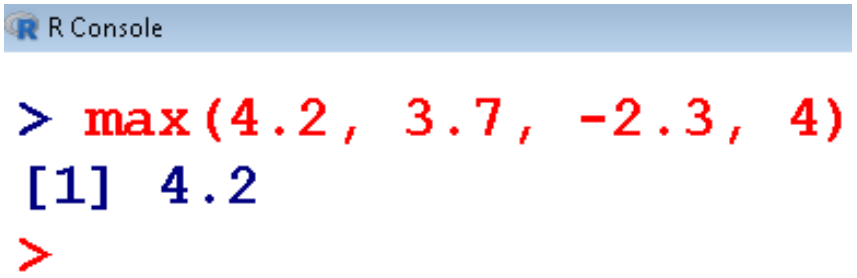
Some commands are readily available in R to compute

mathematical functions.

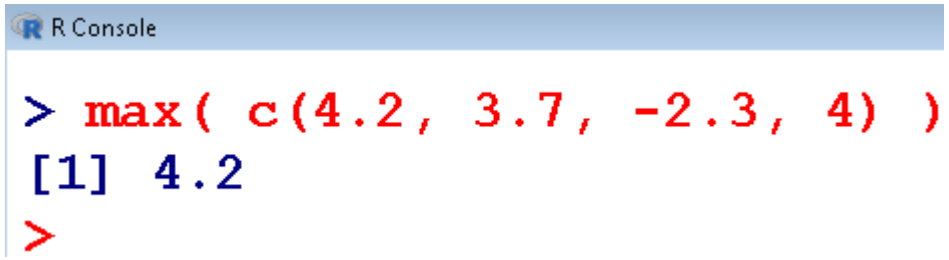
How to use them and utilize them in computing various quantities.

Maximum

```
> max(4.2, 3.7, -2.3, 4)
[1] 4.2
```

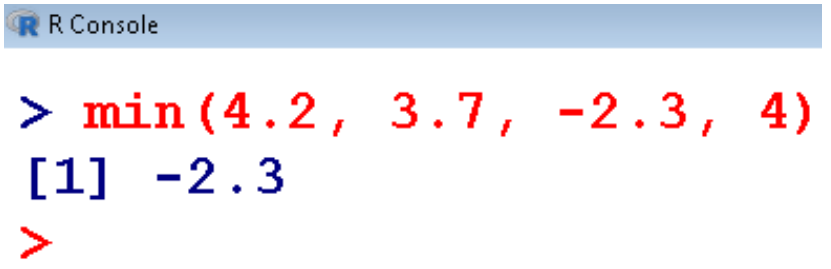



```
> max( c(4.2, 3.7, -2.3, 4) )
[1] 4.2
```

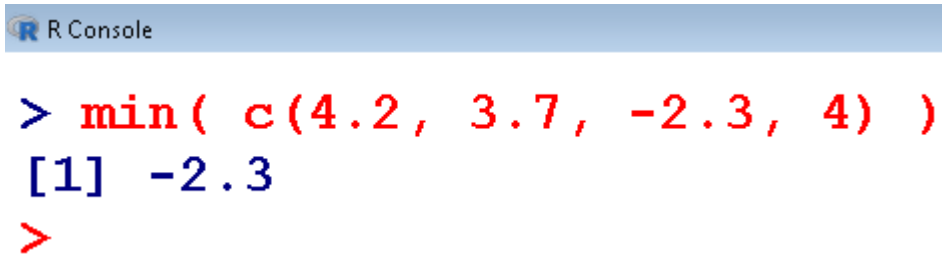


Minimum

```
> min(4.2, 3.7, -2.3, 4)
[1] -2.3
```




```
> min( c(4.2, 3.7, -2.3, 4) )
[1] -2.3
```



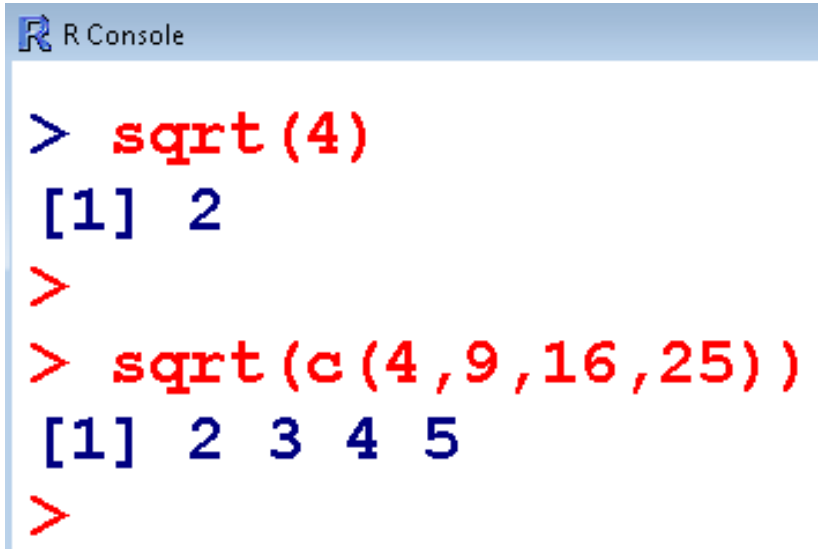
Overview Over Other Functions

`abs()`	Absolute value
`sqrt()`	Square root
`round(), floor(), ceiling()`	Rounding, up and down
`sum(), prod()`	Sum and product
`log(), log10(), log2()`	Logarithms
`exp()`	Exponential function
`sin(), cos(), tan(),` `asin(), acos(), atan()`	Trigonometric functions
`sinh(), cosh(), tanh(),` `asinh(), acosh(), atanh()`	Hyperbolic functions

Examples

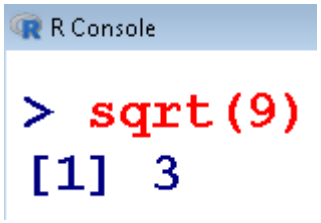
```
> sqrt(4)
[1] 2

> sqrt(c(4,9,16,25))
[1] 2 3 4 5
```

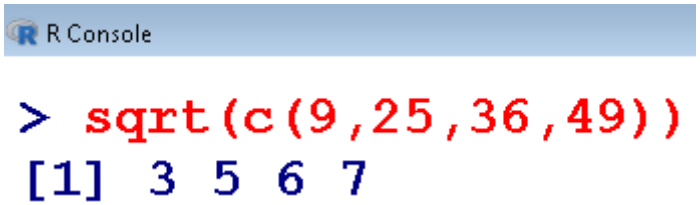


Examples

```
> sqrt(9)
[1] 3
```

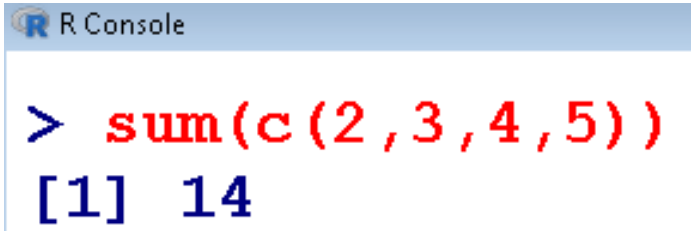



```
> sqrt(c(9,25,36,49))
[1] 3 5 6 7
```

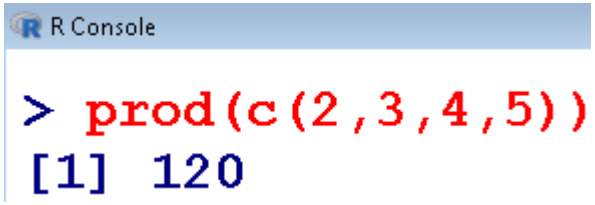


Examples

```
> sum(c(2,3,4,5))
[1] 14
```




```
> prod(c(2,3,4,5))
[1] 120
```



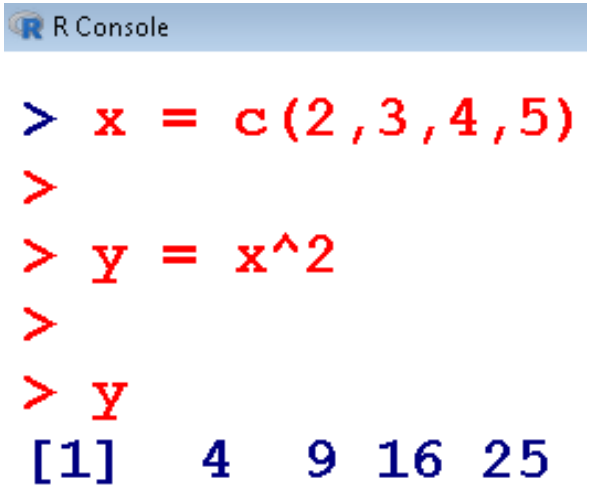
Assignments

An assignment can also be used to save values in variables:

```
> x = c(2,3,4,5)

> y = x^2

> y
[1] 4 9 16 25
```



Examples

To find sum of squares:

```
> x = c(2,3,4,5)
```

$$z = \sum_{i=1}^{n} x_i^2$$

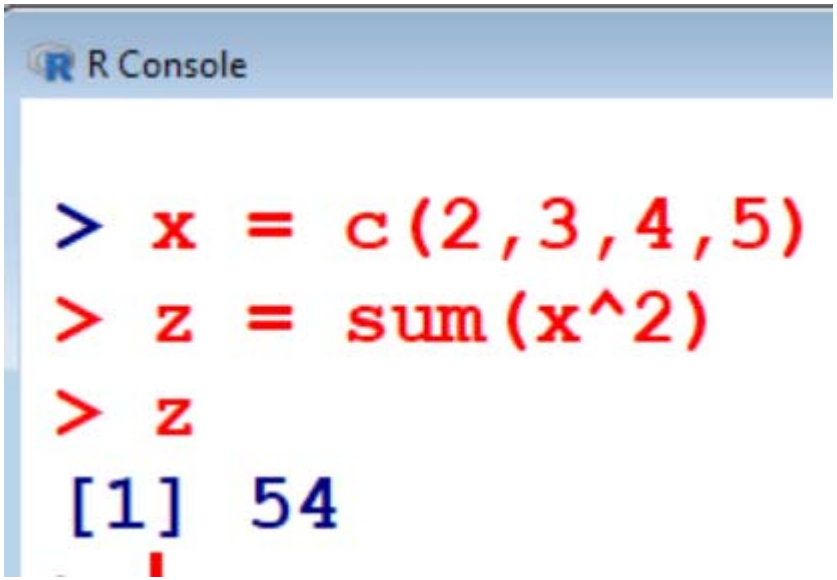
Examples

To find sum of squares:

```
> x = c(2,3,4,5)

> z = sum(x^2)

> z
[1] 54
```



Examples

To find sum of squares of deviation from mean

```
> x = c(2,3,4,5)
```

$$z = \sum_{i=1}^{n} (x_i - \overline{x})^2 = \sum_{i=1}^{n} x_i^2 - n\overline{x}^2$$

Examples

To find sum of squares of deviation from mean

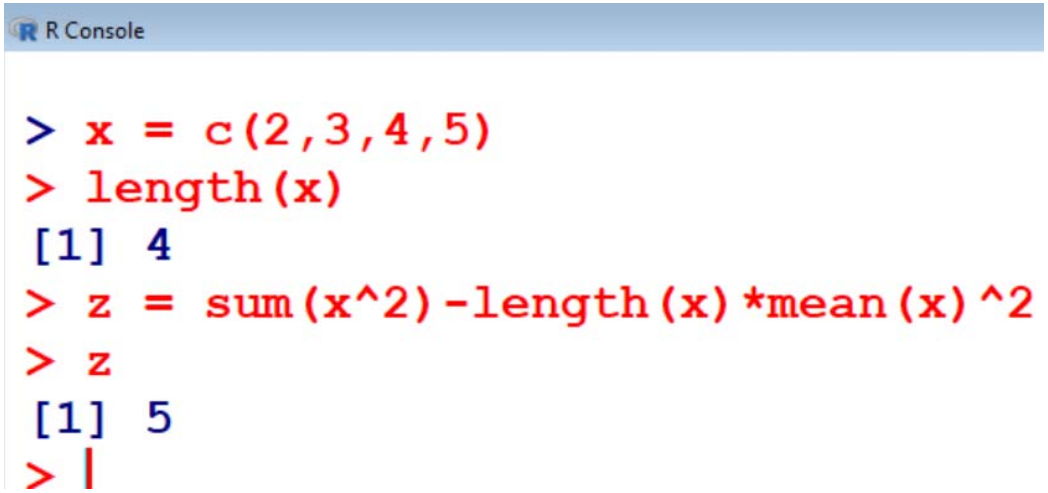
```
> x = c(2,3,4,5)
```

$$z = \sum_{i=1}^{n} (x_i - \overline{x})^2 = \sum_{i=1}^{n} x_i^2 - n\overline{x}^2$$

```
> z = sum(x^2)-length(x)*mean(x)^2

> length(x)
[1] 4

> z
[1] 5
```



Examples

To find sum of cross product:

```
> x1 = c(2,3,4,5)

> x2 = c(6,7,8,9)
```

$$2\times 6+3\times 7+4\times 8+5\times 9$$

```
> z = sum(x1*x2)

> z
[1] 110
```

Missing data

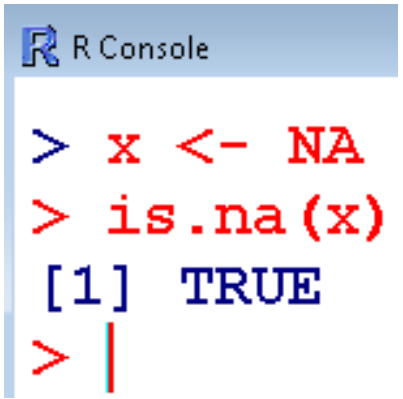
R represents missing observations through the data value **NA**

NA : Reserved word

Missing values can be detected in a data vector by using **is.na**

Create a data with NA.

```
> x <- NA      # assign NA to variable x
> is.na(x)     # is it missing?
 [1] TRUE
```



Missing data

TRUE and FALSE are logical operators that are used to compare expressions.

TRUE and FALSE are reserved words.

T can also be used in place of TRUE .

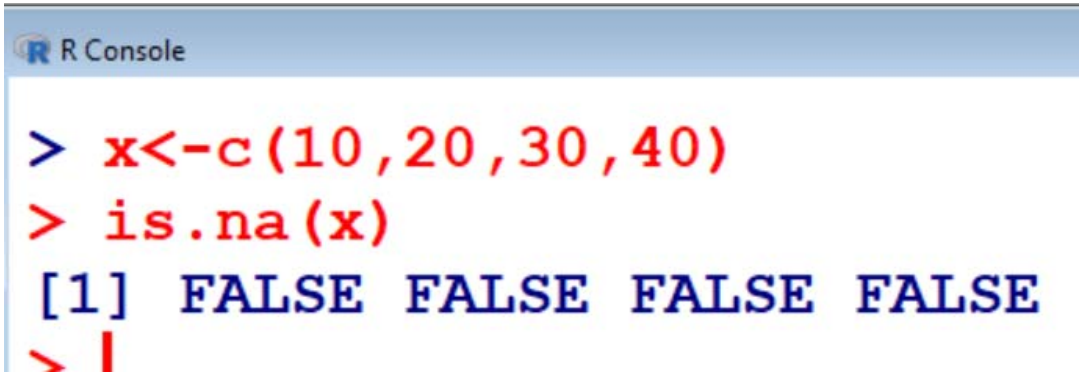
F can also be used in place of FALSE.

TRUE and FALSE are not the same as `true` and `false` respectively.

Missing data

How to know if any value is missing in a data vector?

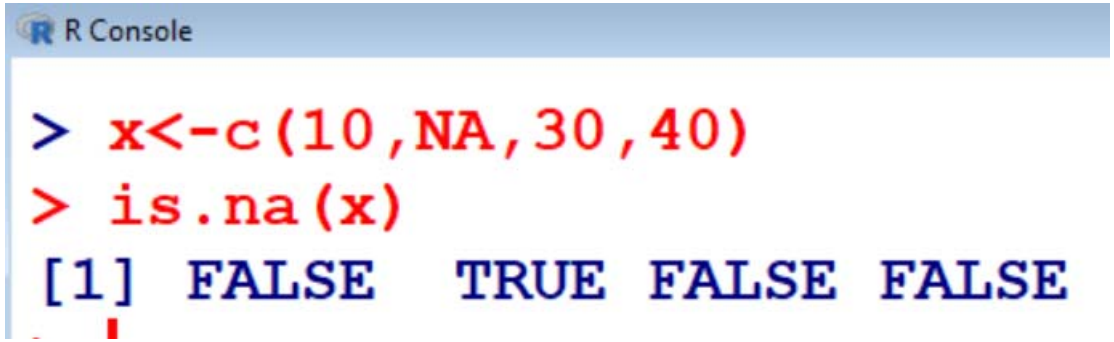
```
> x <- c(10,20,30,40)
> is.na(x)
 [1] FALSE FALSE FALSE FALSE
```



Missing data

How to know if any value is missing in a data vector?

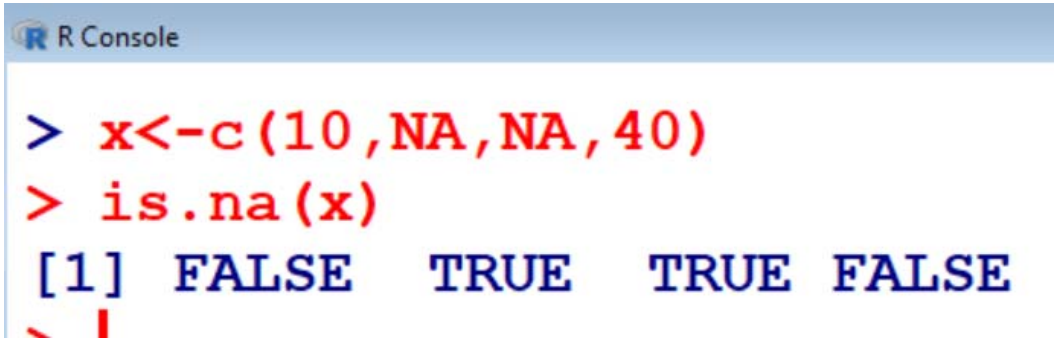
```
> x <- c(10,NA,30,40)
> is.na(x)
 [1] FALSE  TRUE FALSE FALSE
```



Missing data

How to know if any value is missing in a data vector?

```
> x <- c(10,NA,NA,40)
> is.na(x)
 [1] FALSE  TRUE TRUE FALSE
```



Example : How to work with missing data

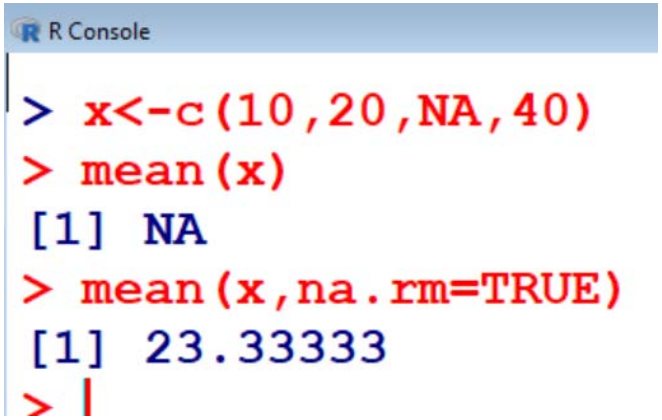
```
> x <- c(10,20,NA,40)
```
 # data vector

```
> mean(x)
 [1] NA
```

$$\frac{10+20+\mathrm{NA}+40}{4}$$

```
> mean(x, na.rm = TRUE)
 [1] 23.33333
```
 # NAs can be removed

$$\frac{10+20+40}{3}=23.33$$

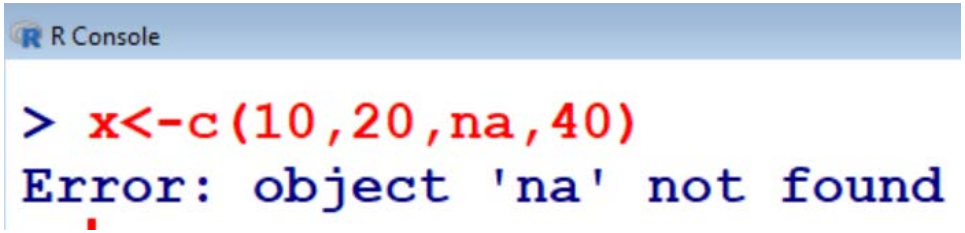


Example : How to work with missing data

The null object, called NULL, is returned by some functions and expressions.

Note that NA and NULL are not the same.

Note that NA and na are not the same.



NA is a placeholder for something that exists but is missing.

NULL stands for something that never existed at all.