Introduction to R Software

Introduction to Statistical Functions
:::
Central Tendency and Variation

Shalabh
Department of Mathematics and Statistics
Indian Institute of Technology Kanpur

Descriptive statistics:

First hand tools which gives first hand information.

- **Central tendency of data (Mean, median, mode, geometric mean, harmoninc mean etc.)**

- **Variation in data (variance, standard deviation, standard error, mean deviation etc.)**

Central tendency of the data

Gives an idea about the mean value of the data

The data is clustered around what value?

Data: $x_1, x_2, \ldots, x_n$

x : **Data vector**

Arithmetic mean (mean) $\bar{x} = \frac{1}{n}\sum_{i=1}^{n} x_i$

`mean(x)`

Central tendency of the data

Geometric mean

$$\overline{x}_{GM} = \left(\prod_{i=1}^{n} x_i \right)^{\frac{1}{n}}$$

`prod(x)^(1/length(x))`

(`length(x)` is equal to the number of elements in x)

Harmonic mean

$$\overline{x}_{HM} = \frac{n}{\frac{1}{n} \sum_{i=1}^{n} \frac{1}{x_i}}$$

`1/mean(1/x)`

Central tendency of the data

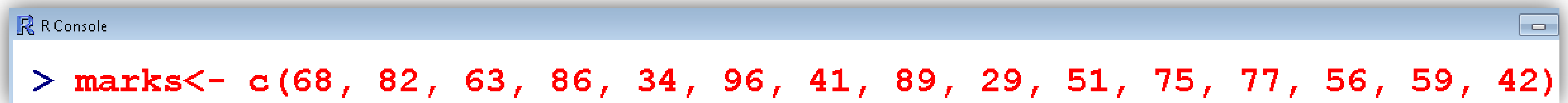
Median:

Value such that the number of observation above it is equal to the number of observation below it.

```
median(x)
```

Example

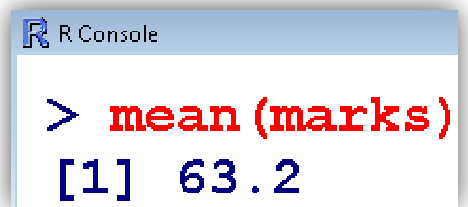
```
> marks<- c(68, 82, 63, 86, 34, 96, 41, 89,
29, 51, 75, 77, 56, 59, 42)
```



```
> marks<- c(68, 82, 63, 86, 34, 96, 41, 89, 29, 51, 75, 77, 56, 59, 42)
```

Arithmetic mean:

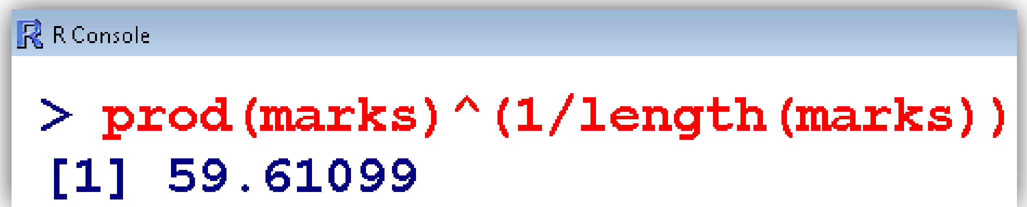
```
> mean(marks)
[1] 63.2
```



```
> mean(marks)
[1] 63.2
```

Geometric mean:

```
>  prod(marks)^(1/length(marks))
[1] 59.61099
```

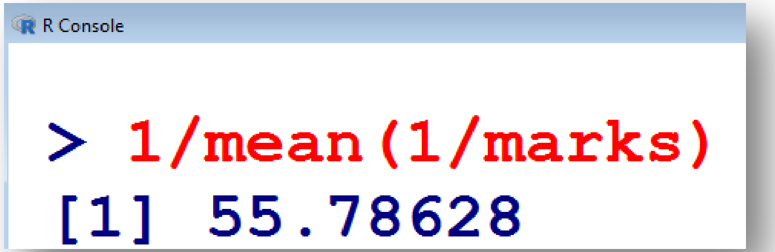


```
> prod(marks)^(1/length(marks))
[1] 59.61099
```

Example

Harmonic mean:

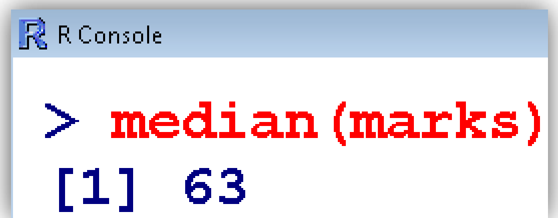
```
> 1/mean(1/marks)
[1] 55.78628
```



```
> 1/mean(1/marks)
[1] 55.78628
```

Median:

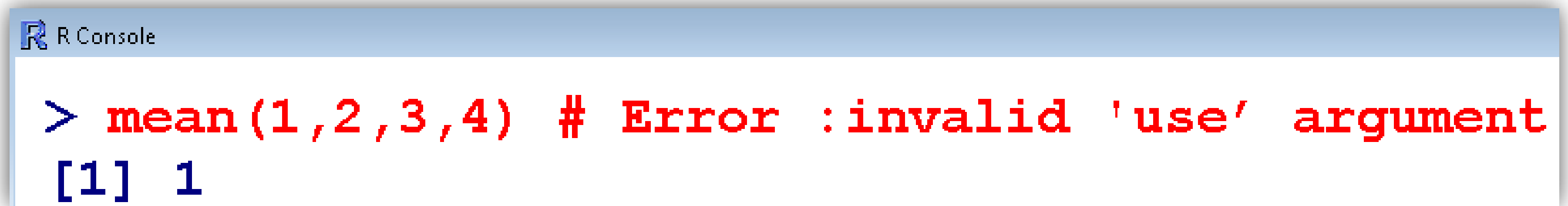
```
> median(marks)
[1] 63
```



```
> median(marks)
[1] 63
```

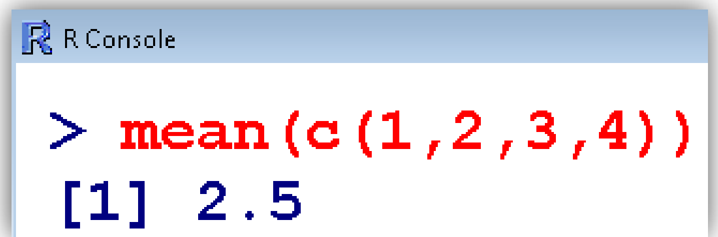
Doesn’t do what you would expect:

```
> mean(1,2,3,4) # Error :invalid 'use' argument
 [1] 1
```



```
> mean(1,2,3,4) # Error :invalid 'use' argument
 [1] 1
```

```
> mean(c(1,2,3,4))
 [1] 2.5
```



```
> mean(c(1,2,3,4))
 [1] 2.5
```

Variability

Spread and scatterdness of data around any point, preferebly the mean value.

Data set 1: 360, 370, 380

mean = (360 + 370 + 380)/3 = 370

Data set 2: 10, 100, 1000

mean = (10 + 100 + 1000)/3 = 370

How to differentiate between the two data sets?

Variability

Variance
$$\mathrm{var}(x) = \frac{1}{n}\sum_{i=1}^{n}(x_i - \bar{x})^2$$

x: data vector

```
var(x)
```

Positive square root of variance : <u>standard deviation</u>

```
sqrt(var(x))
```

Variability

Variance

Another variant,

$$\operatorname{var}(x) = \frac{1}{n-1}\sum_{i=1}^{n}(x_i - \bar{x})^2$$

If we want divisor to be n, then use

```
((n – 1)/n)*var(x)
```

where `n = length(x)`

Variability

Range:

maximum($x_1, x_2,\ldots, x_n$) − minimum($x_1, x_2,\ldots, x_n$)

```
max(x) – min(x)
```

Interquartile range:

Third quartile ($x_1, x_2,\ldots, x_n$) − First quartile ($x_1, x_2,\ldots, x_n$)

```
IQR(x)
```

Variability

Quartile deviation:

[Third quartile ($x_1, x_2, \ldots, x_n$) – First quartile ($x_1, x_2, \ldots, x_n$)]/2
= Interquartile range/2

`IQR(x)/2`

Mean deviation:

$$MD(x) = \frac{1}{n}\sum_{i=1}^{n} | x_i - \overline{x} |$$

`sum(abs(x-mean(x)))/length(x)`

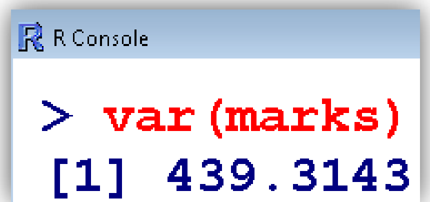
Example

`x`: data vector

```
> marks <- c(68, 82, 63, 86, 34, 96, 41, 89,
29, 51, 75, 77, 56, 59, 42)
```

Variance:

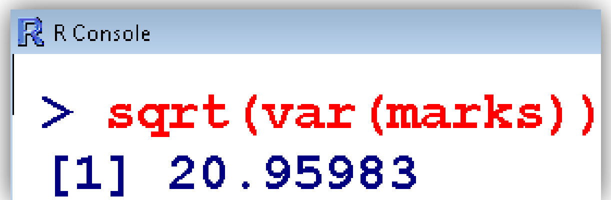
```
> var(marks)
[1] 439.3143
```



```
> var(marks)
[1] 439.3143
```

Standard deviation:

```
> sqrt(var(marks))
[1] 20.95983
```

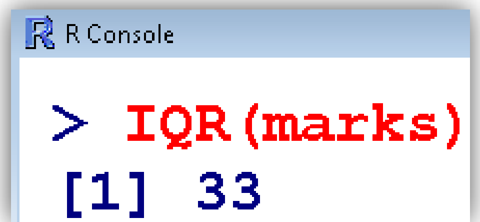


```
> sqrt(var(marks))
[1] 20.95983
```

Example

Interquartile Range:

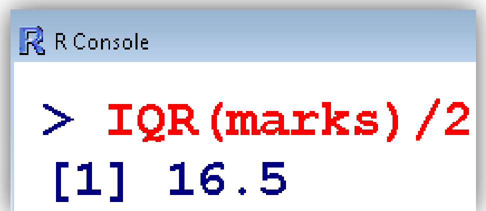
```
> IQR(marks)
[1] 33
```



```
> IQR(marks)
[1] 33
```

Quartile deviation :

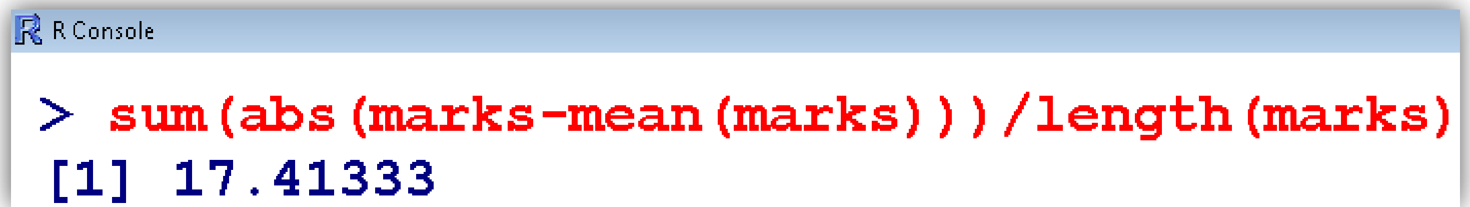
```
> IQR(marks)/2
[1] 16.5
```



```
> IQR(marks)/2
[1] 16.5
```

Mean deviation:

```
> sum(abs(marks-mean(marks)))/length(marks)
[1] 17.41333
```



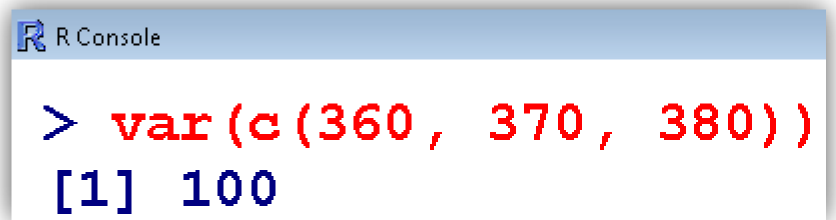
```
> sum(abs(marks-mean(marks)))/length(marks)
[1] 17.41333
```

Example

Data set 1: 360, 370, 380

mean = (360 + 370 + 380)/3 = 370

```
> var(c(360, 370, 380))
[1] 100
```

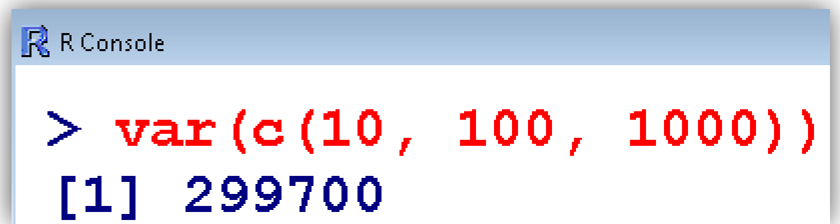


```
> var(c(360, 370, 380))
[1] 100
```

Data set 2: 10, 100, 1000

mean = (10 + 100 + 1000)/3 = 370 Same as of Data set 1

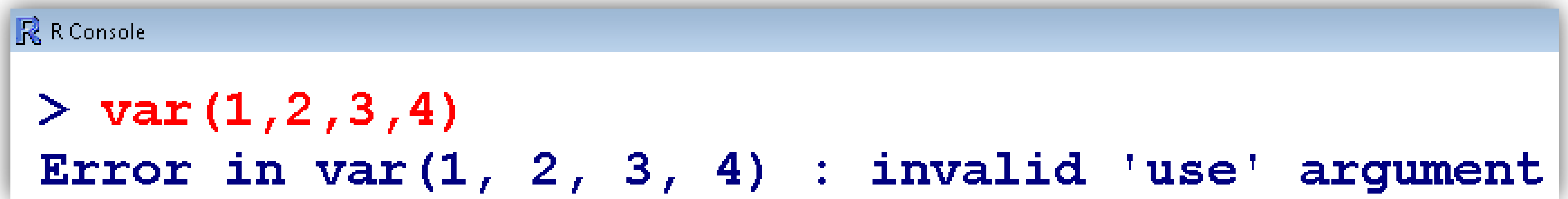
```
> var(c(10, 100, 1000))
[1] 299700
```



```
> var(c(10, 100, 1000))
[1] 299700
```

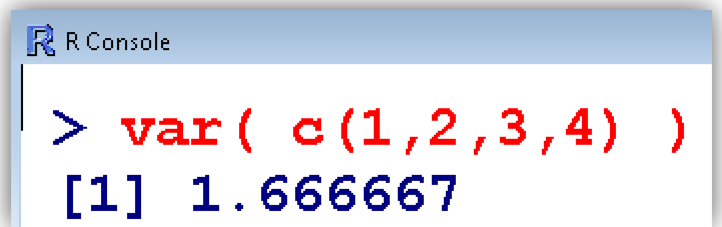
Doesn’t do what we would expect:

```
> var(1,2,3,4)
Error in var(1, 2, 3, 4) : invalid 'use’ argument
```



```
> var(1,2,3,4)
Error in var(1, 2, 3, 4) : invalid 'use' argument
```

```
> var( c(1,2,3,4) )
[1] 1.666667
```



```
> var( c(1,2,3,4) )
[1] 1.666667
```