Descriptive Statistics With R Software

Moments

::

Sheppard‘s Correction, Absolute Moments and Computation of Moments

Shalabh

Department of Mathematics and Statistics

Indian Institute of Technology Kanpur

Sheppard‘s Correction for Moments

We assume in grouped data that the frequencies are concentrated at the middle part of the class interval.

This asumption does not hold true in general, and “grouping error“ is introduced.

Sheppard‘s Correction for Moments

Such an effect can be corrected in calculating the moments by using the information on width of the class interval.

Let *c* be the width of the class interval.

Prof. W. F. Sheppard proved that if the frequency distribution is contineous and the frequency tapers off to zero in both directions, the “grouping effect“ can be corrected as follows:

Sheppard‘s Correction for Moments

Raw Moments

$$\mu'_{1(corr)} = \mu'_1$$

$$\mu'_{2(corr)} = \mu'_2 - \frac{c^2}{12}$$

$$\mu'_{3(corr)} = \mu'_3 - \frac{c^2}{4}\mu'_1$$

$$\mu'_{4(corr)} = \mu'_4 - \frac{c^2}{2}\mu'_2 + \frac{7}{240}c^4$$

Central Moments

$$\mu_{2(corr)} = \mu_2 - \frac{c^2}{12}$$

$$\mu_{3(corr)} = \mu_3$$

$$\mu_{4(corr)} = \mu_4 - \frac{c^2}{2}\mu_2 + \frac{7}{240}c^4$$

Absolute Moments

The r^{th} (sample) absolute moment based on observations $x_1, x_2, \ldots, x_n$ is defined as

- **For ungrouped (discrete) data**

$$|\mu|_r = \frac{1}{n}\sum_{i=1}^{n}|x_i - \overline{x}|^r$$

- **For grouped (continuous) data**

$$|\mu|_r = \frac{1}{n}\sum_{i=1}^{K} f_i\,|x_i - \overline{x}|^r$$

where $n = \sum_{i=1}^{K} f_i, \quad \overline{x} = \frac{1}{n}\sum_{i=1}^{K} f_i x_i$

Moments

R commands

Install package

```
install.packages("moments")

library(moments)
```

Sample moments are computed by the command

```
all.moments(x, order.max = 2, central = FALSE,
absolute = FALSE, na.rm = FALSE)
```

Usage

`x`   A numeric vector, matrix or data frame of data.

For matrices and data frames, each column is a random variable

Moments

R commands

`order.max` Maximum order of the moments to be computed with a default value of 2.

`central` Logical value, if TRUE, central moments are computed. Otherwise, raw moments are computed.

`absolute` Logical value, if TRUE, absolute moments are computed. Otherwise, standard moments are computed.

`na.rm` Logical value, if TRUE, remove NA values. Otherwise, keep NA values.

Moments

Example:

Following are the time taken (in seconds) by 20 participants in a race: 32, 35, 45, 83, 74, 55, 68, 38, 35, 55, 66, 65, 42, 68, 72, 84, 67, 36, 42, 58.

```
> time = c(32, 35, 45, 83, 74, 55, 68, 38, 35,
55, 66, 65, 42, 68, 72, 84, 67, 36, 42, 58)
```

```
> install.packages("moments")
```

```
> library(moments)
```

Moments

Example:

Raw moments: `order.max = 2`

```
> all.moments(time, order.max = 2)
[1]  1.0   56.0  3405.2
```

Raw moments: `order.max = 4`

```
> all.moments(time, order.max = 4)
[1]  1.0   56.0   3405.2  221096.0  15080073.2
```

Moments

Example:

Central moments: `order.max = 2`

```
> all.moments(time, order.max=2, central=TRUE)
[1] 1.0      0.0    269.2
```

Central moments: `order.max = 4`

```
> all.moments(time, order.max=4, central=TRUE)
[1] 1.0      0.0    269.2    254.4   123324.4
```

Moments

Example:

Absolute moments: `order.max = 2`

```
> all.moments(time, order.max=2, absolute=TRUE)

[1] 1.0   56.0   3405.2
```

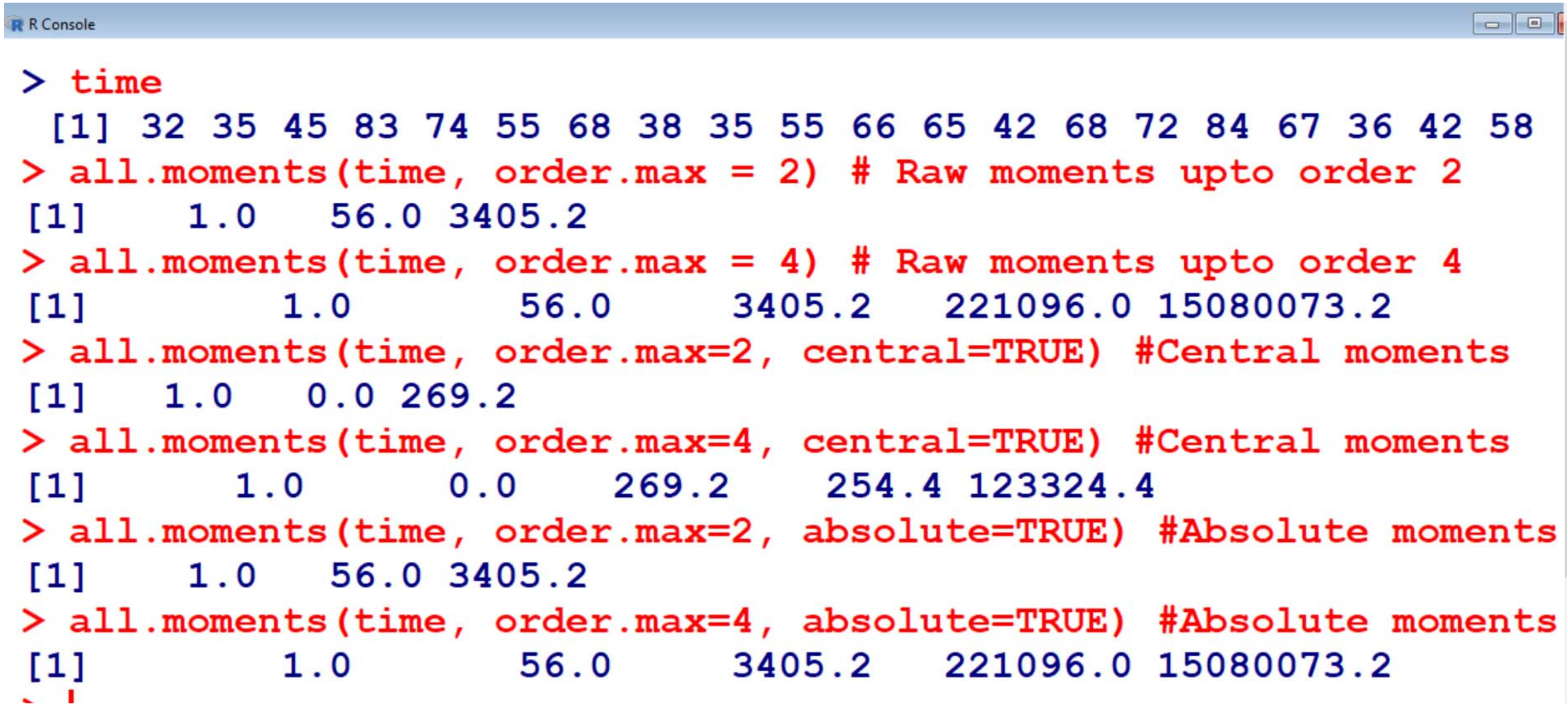
Absolute moments: `order.max = 4`

```
> all.moments(time, order.max=2, absolute=TRUE)

[1] 1.0   56.0   3405.2   221096.0   15080073.2
```

Moments

Example:



```
> time
 [1] 32 35 45 83 74 55 68 38 35 55 66 65 42 68 72 84 67 36 42 58
> all.moments(time, order.max = 2) # Raw moments upto order 2
[1]    1.0   56.0 3405.2
> all.moments(time, order.max = 4) # Raw moments upto order 4
[1]         1.0        56.0      3405.2    221096.0 15080073.2
> all.moments(time, order.max=2, central=TRUE) #Central moments
[1]   1.0   0.0 269.2
> all.moments(time, order.max=4, central=TRUE) #Central moments
[1]      1.0       0.0    269.2     254.4 123324.4
> all.moments(time, order.max=2, absolute=TRUE) #Absolute moments
[1]    1.0   56.0 3405.2
> all.moments(time, order.max=4, absolute=TRUE) #Absolute moments
[1]         1.0        56.0      3405.2    221096.0 15080073.2
```

Moments

Example: Handling missing values

Suppose two data points are missing in the earlier example where the time taken (in seconds) by 20 participants in a race. They are recorded as NA

NA, NA, 45, 83, 74, 55, 68, 38, 35, 55, 66, 65, 42, 68, 72, 84, 67, 36, 42, 58.

```
> time.na = c(NA, NA, 45, 83, 74, 55, 68, 38,
35, 55, 66, 65, 42, 68, 72, 84, 67, 36, 42, 58)
```

Moments

Example: **Handling missing values**

Raw moments: **First four moments**

```
> all.moments(time.na, order.max=4, na.rm=TRUE)
[1] 1.000   58.500  3658.611   241459.833
16614014.611
```

Central moments: **First four moments**

```
> all.moments(time.na, order.max=4,
central=TRUE, na.rm=TRUE)
[1] 1.0000  0.0000   236.3611   -223.1667
101119.6736
```

Moments

Example: **Handling missing values**

Absolute moments: **First four moments**

```
> all.moments(time.na, order.max=4,
absolute=TRUE, na.rm=TRUE)
[1] 1.000   58.500   3658.611   241459.833
16614014.611
```