01 Basics: Measures of central tendency

Peter Baumgartner

2018-05-11

Contents


```
## Loading required package: tidyverse
```

```
## -- Attaching packages ---------------------------------------------------------- tidyverse 1.2.1 --
```

```
## ✓ ggplot2 2.2.1     ✓ purrr   0.2.4
## ✓ tibble  1.4.2     ✓ dplyr   0.7.4
## ✓ tidyr   0.8.0     ✓ stringr 1.3.1
## ✓ readr   1.1.1     ✓ forcats 0.3.0
```

```
## -- Conflicts ------------------------------------------------------------- tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

1 How to get a small data set into R?

Number of friends of 11 Facebook user: 108, 103, 252, 121, 93, 57, 40, 53, 22, 116, 98.

1.1 Assign values to a vector

```
> # number of facebook friends = nff
> nff <- c(108, 103, 252, 121, 93, 57, 40, 53, 22, 116, 98)
> nff <- sort(nff)
> nff
```

```
 [1]  22  40  53  57  93  98 103 108 116 121 252
```

1.2 Read data values from keyboard

After running the following code you have to set your curso into the console and provide the data. There are 2 possibilities: * Enter the data manually and separate each entry with ENTER * Copy a string of data (e.g. from a PDF table), where each data is spearated by a blank

In both cases: Terminate the input with an extra ENTER

```
> nff_scan <- as.vector(scan(file = ""))
> nff <- sort(nff_scan)
> nff_scan
```

2 Mode

There is no mode-function in the base R module. See https://bit.ly/R-mode. But there are many possibilites to programm this function.

Before I will demonstrate this, I need to add another number into the data set in order to get the frequencey of one number higher than the others.

```
> nff_mode <- c(nff, 53)
```

Variante 1: Simple but goot!

```
> Mode <- function(x) {
+   ux <- unique(x)
+   ux[which.max(tabulate(match(x, ux)))]
+ }
> Mode(nff_mode)
```

```
[1] 53
```

Version 2: with NA

```
> Mode <- function(x, na.rm = FALSE) {
+     if (na.rm) {
+         x = x[!is.na(x)]
+     }
+     ux <- unique(x)
+     ux[which.max(tabulate(match(x, ux)))]
+ }
> Mode(nff_mode)
```

```
[1] 53
```

```
> library(modeest)
> mlv(nff_mode, method = "mfv")
```

```
Mode (most likely value): 53
Bickel's modal skewness: 0.5
Call: mlv.default(x = nff_mode, method = "mfv")
```

3 Median