

Onderzoekstechnieken

TijsMartens

6 april 2019

aan de slag

algemeen

alle huidige objecten weergeven

```
objects()
```

```
## character(0)
```

objecten verwijderen

```
rm (lijst, meanY)
```

```
## Warning in rm(lijst, meanY): object 'lijst' not found
```

```
## Warning in rm(lijst, meanY): object 'meanY' not found
```

```
objects()
```

```
## character(0)
```

toekenning van een nieuw object

```
nieuweVar <- "dit is een test"  
nieuweVar
```

```
## [1] "dit is een test"
```

toekenning van een lijst

- let op: R is niet null based, het eerste element heeft dus index 1 en niet 0

```
lijst <- c(10, 15, 20, 35)  
lijst
```

```
## [1] 10 15 20 35
```

```
lijst[1]
```

```
## [1] 10
```

csv bestanden

een csv file inlezen

```
library(readr)
```

```
## Warning: package 'readr' was built under R version 3.5.3
```

```
puntenlijst <- read_csv("C:/Users/tijsm/Google Drive/HoGent 2018-2019/2e semester/Onderzoekstechnieken/
```

```
## Parsed with column specification:
```

```
## cols(
```

```
##   Groep = col_character(),
```

```
##   Score = col_double()
## )
```

```
puntenlijst
```

```
## # A tibble: 203 x 2
##   Groep Score
##   <chr> <dbl>
## 1 A      12.5
## 2 A       7.5
## 3 A      NA
## 4 A      21
## 5 A      10
## 6 A     20.5
## 7 A      14
## 8 A       4
## 9 A       7
## 10 A     NA
## # ... with 193 more rows
```

werken met een lijst

de kolomnamen vragen

```
names(puntenlijst)
```

```
## [1] "Groep" "Score"
```

kolommen

```
attributes(puntenlijst)
```

```
## $names
## [1] "Groep" "Score"
##
## $class
## [1] "spec_tbl_df" "tbl_df"      "tbl"        "data.frame"
##
## $row.names
##  [1]  1  2  3  4  5  6  7  8  9 10 11 12 13 14 15 16 17
## [18] 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34
## [35] 35 36 37 38 39 40 41 42 43 44 45 46 47 48 49 50 51
## [52] 52 53 54 55 56 57 58 59 60 61 62 63 64 65 66 67 68
## [69] 69 70 71 72 73 74 75 76 77 78 79 80 81 82 83 84 85
## [86] 86 87 88 89 90 91 92 93 94 95 96 97 98 99 100 101 102
## [103] 103 104 105 106 107 108 109 110 111 112 113 114 115 116 117 118 119
## [120] 120 121 122 123 124 125 126 127 128 129 130 131 132 133 134 135 136
## [137] 137 138 139 140 141 142 143 144 145 146 147 148 149 150 151 152 153
## [154] 154 155 156 157 158 159 160 161 162 163 164 165 166 167 168 169 170
## [171] 171 172 173 174 175 176 177 178 179 180 181 182 183 184 185 186 187
## [188] 188 189 190 191 192 193 194 195 196 197 198 199 200 201 202 203
##
## $spec
## cols(
##   Groep = col_character(),
##   Score = col_double()
## )
```

data types

numbers

```
a <- 3
a
```

```
## [1] 3
```

```
b <- sqrt(a*a+3)
b
```

```
## [1] 3.464102
```

een lijst van nummers maken

- een lijst van 10 cijfers

```
c <- numeric(10)
c
```

```
## [1] 0 0 0 0 0 0 0 0 0 0
```

type controlleren

```
typeof(a)
```

```
## [1] "double"
```

strings

```
a <- "dit is een sting"
a
```

```
## [1] "dit is een sting"
```

```
b <- c("dit", "is", "een", "lijst", "van", "strings")
b
```

```
## [1] "dit"      "is"       "een"      "lijst"    "van"      "strings"
```

```
b[2]
```

```
## [1] "is"
```

factors

als een nominale variabele moet vervormd worden naar een integer waarde maken we gebruik van het commando “factor”

- voorbeeld van een nominale variabele is vb. “veel”, “goed”, “tevreden”, ...

```
a <- c(1,2,3,4)
b <- c(2,4,6,8)
levels <- factor("A","B","A","B")
```

data frames

verschillende vectoren van verschillende types nemen en deze allemaal in dezelfde variabele(lijt) opslaan

```
bubba <- data.frame(first=a, second=b, f=levels)
bubba
```

```
##   first second    f
## 1     1      2 <NA>
## 2     2      4 <NA>
## 3     3      6 <NA>
## 4     4      8 <NA>
```

```
bubba$first
```

```
## [1] 1 2 3 4
```

```
bubba$f
```

```
## [1] <NA> <NA> <NA> <NA>
## Levels: A
```

tables

```
a <- factor(c("A","G","A","D","A","A","G","D","F","V"))
results <- table(a)
results
```

```
## a
## A D F G V
## 4 2 1 2 1
```

```
summary(results)
```

```
## Number of cases in table: 10
## Number of factors: 1
```

matrix

aanmaken van een matrix

```
A = matrix(
  c(2,4,3,1,5,7), #data
  nrow = 2,      #aantal rijen
  ncol = 3,      #aantal kolommen
  byrow = TRUE   #vullen per rij (niet per kolom)
)
A
```

```
##      [,1] [,2] [,3]
## [1,]    2    4    3
## [2,]    1    5    7
```

selecteren van elementen binnnen een matrix

```
A[2, 3] #element op rij 2 kolom 3
```

```
## [1] 7
```

```
A[2, ] # volledig tweede rij
```

```
## [1] 1 5 7
```

```
A[,c(1,3)] #eerste en derde kolom
```

```
##      [,1] [,2]
## [1,]    2    3
```

```
## [2,]    1    7
```

Oefeningen

oefening 1.2.

opgave

Bekijk de dataset mtcars. Geef de waarde terug voor de eerste rij, tweede kolom. Geef ook het aantal rijen, het aantal kolommen. Geef ook een preview van het volledige data frame. Geef enkel de kolom terug met de definities van de cylinders. Om een data frame te bekomen met de twee kolommen mpg en hp, pakken we de kolomnamen in een indexvector in met single square bracket operator. Probeer ook eens op te zoeken hoe je een rijrecord van de ingebouwde data set mtcars bepaalt.

oplossing

Geef de waarde terug voor de eerste en de tweede rij

```
mtcars[1]
```

```
##                mpg
## Mazda RX4      21.0
## Mazda RX4 Wag  21.0
## Datsun 710      22.8
## Hornet 4 Drive  21.4
## Hornet Sportabout 18.7
## Valiant        18.1
## Duster 360     14.3
## Merc 240D      24.4
## Merc 230       22.8
## Merc 280       19.2
## Merc 280C      17.8
## Merc 450SE     16.4
## Merc 450SL     17.3
## Merc 450SLC    15.2
## Cadillac Fleetwood 10.4
## Lincoln Continental 10.4
## Chrysler Imperial 14.7
## Fiat 128       32.4
## Honda Civic    30.4
## Toyota Corolla 33.9
## Toyota Corona  21.5
## Dodge Challenger 15.5
## AMC Javelin    15.2
## Camaro Z28     13.3
## Pontiac Firebird 19.2
## Fiat X1-9      27.3
## Porsche 914-2  26.0
## Lotus Europa   30.4
## Ford Pantera L 15.8
## Ferrari Dino   19.7
## Maserati Bora   15.0
## Volvo 142E     21.4
```

```
mtcars[2]
```

```
##                                cyl
## Mazda RX4                     6
## Mazda RX4 Wag                 6
## Datsun 710                    4
## Hornet 4 Drive                6
## Hornet Sportabout             8
## Valiant                      6
## Duster 360                   8
## Merc 240D                    4
## Merc 230                     4
## Merc 280                     6
## Merc 280C                    6
## Merc 450SE                   8
## Merc 450SL                   8
## Merc 450SLC                  8
## Cadillac Fleetwood           8
## Lincoln Continental           8
## Chrysler Imperial            8
## Fiat 128                     4
## Honda Civic                  4
## Toyota Corolla               4
## Toyota Corona                4
## Dodge Challenger             8
## AMC Javelin                  8
## Camaro Z28                   8
## Pontiac Firebird             8
## Fiat X1-9                    4
## Porsche 914-2                4
## Lotus Europa                 4
## Ford Pantera L               8
## Ferrari Dino                 6
## Maserati Bora                8
## Volvo 142E                   4
```

geef het aantal rijen en kolommen

```
nrow(mtcars)
```

```
## [1] 32
```

```
ncol(mtcars)
```

```
## [1] 11
```

preview van dataframe

```
mtcars
```

```
##      mpg  cyl  disp  hp drat   wt  qsec vs am gear carb
## Mazda RX4      21.0   6 160.0 110 3.90 2.620 16.46 0  1   4    4
## Mazda RX4 Wag  21.0   6 160.0 110 3.90 2.875 17.02 0  1   4    4
## Datsun 710     22.8   4 108.0  93 3.85 2.320 18.61 1  1   4    1
## Hornet 4 Drive  21.4   6 258.0 110 3.08 3.215 19.44 1  0   3    1
## Hornet Sportabout 18.7   8 360.0 175 3.15 3.440 17.02 0  0   3    2
## Valiant        18.1   6 225.0 105 2.76 3.460 20.22 1  0   3    1
## Duster 360     14.3   8 360.0 245 3.21 3.570 15.84 0  0   3    4
## Merc 240D      24.4   4 146.7  62 3.69 3.190 20.00 1  0   4    2
## Merc 230       22.8   4 140.8  95 3.92 3.150 22.90 1  0   4    2
```

## Merc 280	19.2	6	167.6	123	3.92	3.440	18.30	1	0	4	4
## Merc 280C	17.8	6	167.6	123	3.92	3.440	18.90	1	0	4	4
## Merc 450SE	16.4	8	275.8	180	3.07	4.070	17.40	0	0	3	3
## Merc 450SL	17.3	8	275.8	180	3.07	3.730	17.60	0	0	3	3
## Merc 450SLC	15.2	8	275.8	180	3.07	3.780	18.00	0	0	3	3
## Cadillac Fleetwood	10.4	8	472.0	205	2.93	5.250	17.98	0	0	3	4
## Lincoln Continental	10.4	8	460.0	215	3.00	5.424	17.82	0	0	3	4
## Chrysler Imperial	14.7	8	440.0	230	3.23	5.345	17.42	0	0	3	4
## Fiat 128	32.4	4	78.7	66	4.08	2.200	19.47	1	1	4	1
## Honda Civic	30.4	4	75.7	52	4.93	1.615	18.52	1	1	4	2
## Toyota Corolla	33.9	4	71.1	65	4.22	1.835	19.90	1	1	4	1
## Toyota Corona	21.5	4	120.1	97	3.70	2.465	20.01	1	0	3	1
## Dodge Challenger	15.5	8	318.0	150	2.76	3.520	16.87	0	0	3	2
## AMC Javelin	15.2	8	304.0	150	3.15	3.435	17.30	0	0	3	2
## Camaro Z28	13.3	8	350.0	245	3.73	3.840	15.41	0	0	3	4
## Pontiac Firebird	19.2	8	400.0	175	3.08	3.845	17.05	0	0	3	2
## Fiat X1-9	27.3	4	79.0	66	4.08	1.935	18.90	1	1	4	1
## Porsche 914-2	26.0	4	120.3	91	4.43	2.140	16.70	0	1	5	2
## Lotus Europa	30.4	4	95.1	113	3.77	1.513	16.90	1	1	5	2
## Ford Pantera L	15.8	8	351.0	264	4.22	3.170	14.50	0	1	5	4
## Ferrari Dino	19.7	6	145.0	175	3.62	2.770	15.50	0	1	5	6
## Maserati Bora	15.0	8	301.0	335	3.54	3.570	14.60	0	1	5	8
## Volvo 142E	21.4	4	121.0	109	4.11	2.780	18.60	1	1	4	2

preview van slechts 2 kolommen

```
kolommen <- c(1,4)
```

```
mtcars[kolommen]
```

##	mpg	hp
## Mazda RX4	21.0	110
## Mazda RX4 Wag	21.0	110
## Datsun 710	22.8	93
## Hornet 4 Drive	21.4	110
## Hornet Sportabout	18.7	175
## Valiant	18.1	105
## Duster 360	14.3	245
## Merc 240D	24.4	62
## Merc 230	22.8	95
## Merc 280	19.2	123
## Merc 280C	17.8	123
## Merc 450SE	16.4	180
## Merc 450SL	17.3	180
## Merc 450SLC	15.2	180
## Cadillac Fleetwood	10.4	205
## Lincoln Continental	10.4	215
## Chrysler Imperial	14.7	230
## Fiat 128	32.4	66
## Honda Civic	30.4	52
## Toyota Corolla	33.9	65
## Toyota Corona	21.5	97
## Dodge Challenger	15.5	150
## AMC Javelin	15.2	150
## Camaro Z28	13.3	245

```
## Pontiac Firebird      19.2 175
## Fiat X1-9             27.3  66
## Porsche 914-2        26.0  91
## Lotus Europa         30.4 113
## Ford Pantera L       15.8 264
## Ferrari Dino         19.7 175
## Maserati Bora        15.0 335
## Volvo 142E           21.4 109
```

oefening 1.3.

opgave

Maak zelf een willekeurige datafile aan in Excel en probeer deze in te lezen in R. Zijn er nog dataformaten die ondersteund worden door R?

oplossing

skip