Fundamental data type in R

Mike Yang

10 November 2018

```
library(tidyverse)
```

Summary

> Although R has object-oriented features, methods are not part of the class definitions. Methods in R look just like normal functions, such as summary().

> Data type in R

- **vector**
  - roughly, there are two different kinds of vectors: atomic vectors and lists (generic vectors).
  - a single number (e.g., 3) is treated as a vector with only one element.
- **list**
  - Virtually every object you'll run across in R is really just a list.
  - Most objects that you'll deal with have the type "list".
  - If you want to actually "extract" an element from a list, you need to use the "[[]]" style of indexing.
  - If you have named elements of a list, instead of using the "[[]]" style of indexing mentioned previously, you can access them by appending $name instead.
- **matrix**
- **data frame**
  - data frames are lists of data representing different kinds of information taken at the same times, places, etc.
  - When you read in data from an external source, it will typically be in a data frame.

Create data type in R

- The easiest way to directly create a multi-element vector is using the **c()** (concatenate) function.
- You can create lists directly using the **list()** function.
- Matrices can be created directly using **matrix()**.

creating a list

```
df <- list(a = c(1,2), b = c(2,4), c = "mikey")
summary(df)
```

File failed to load: /extensions/MathZoom.js

```
##   Length Class  Mode     
## a 2      -none- numeric  
## b 2      -none- numeric  
## c 1      -none- character
```

access list element

```
df[1]
```

```
## $a
## [1] 1 2
```

```
class(df[1])
```

```
## [1] "list"
```

```
df[[1]]
```

```
## [1] 1 2
```

```
class(df[[1]])
```

```
## [1] "numeric"
```

```
df$a
```

```
## [1] 1 2
```

```
class(df$a)
```

```
## [1] "numeric"
```

```
# convert list to tibble
df_table <- as.tibble(df)
```

create a matrix

```
m <- matrix(c(1,2,3,4), nrow = 2, ncol = 2)
```

return first row and first column of the matrix

```
m[1,1]
```

File failed to load: /extensions/MathZoom.js

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

return column of the matrix

```
m[,2]
```

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

return row of the matrix

```
m[2,]
```

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

File failed to load: /extensions/MathZoom.js