UNIVERSITY OF CALGARY

R Wizardry, week 1 (continuation): Data Types

January 18th, 2017

R data Types

R has a wide variety of data types including:

- Scalars (a single number)
- Vectors (numerical, character, logical)
- Matrices
- Arrays
- Data frames
- Lists
- S4

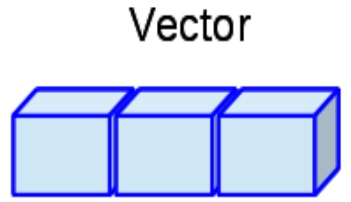
From “R in Action” by Robert I. Kabacoff

Scalars and Vectors

- Scalar:

  X <- 5 ; b <- 10

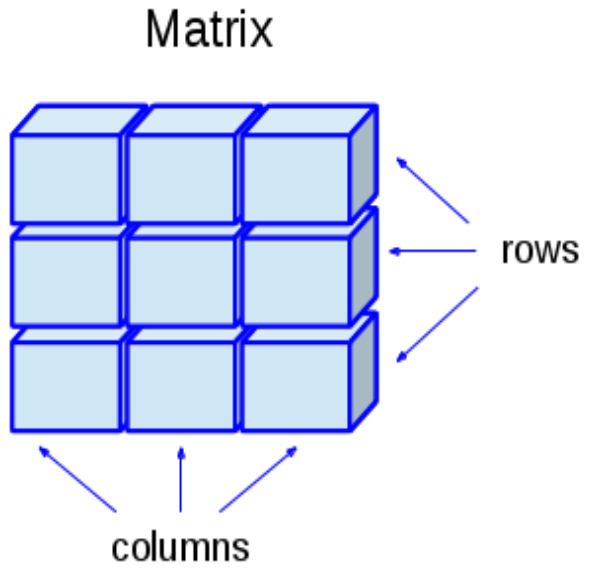
- Vectors (numerical, character, logical):




```
a <- c(1,2,5.3,6,-2,4) # numeric vector
b <- c("one","two","three") # character vector
c <- c(TRUE,TRUE,TRUE,FALSE,TRUE,FALSE) #logical
vector
```

Matrices

All columns in a matrix must have the same mode (numeric, character, etc.) and the same length.




```
#Generates 5 x 4 numeric matrix :
y <- matrix(1:20, nrow = 5, ncol = 4)

# Another example
cells <- c(1, 26, 24, 68)
rnames <- c("R1", "R2")
cnames <- c("C1", "C2")
mymatrix <- matrix(cells, nrow = 2, ncol = 2, byrow =
T, dimnames=list(rnames, cnames))
```

Arrays

An **n-dimensional array** is a set of stacked matrices of identical dimensions

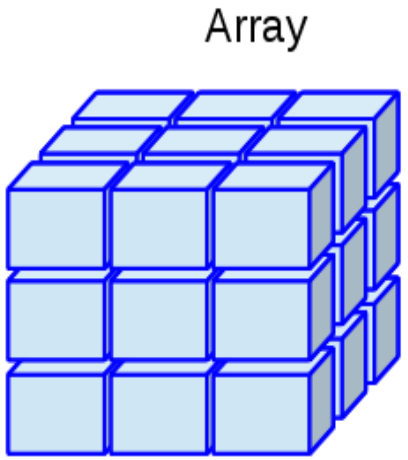
- a <- matrix(8, 2, 3) # Creates a 2 x 3 matrix populated with 8's.
- b <- matrix(9, 2, 3) # Creates a 2 x 3 matrix populated wi

```
> array(c(a, b), c(2, 3, 2)) # Creates a 2 x 3 x 2 array with the first
, , 1

     [,1] [,2] [,3]
[1,]    8    8    8
[2,]    8    8    8

, , 2

     [,1] [,2] [,3]
[1,]    9    9    9
[2,]    9    9    9
```



http://r.iq.harvard.edu/docs/zelig/3.4-8/Ways_to_create.html

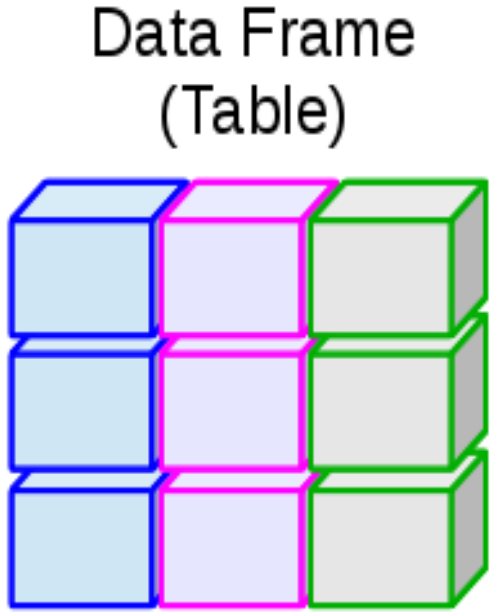
Data Frames

- A data frame is more general than a matrix, in that different columns can have different modes (numeric, character, factor, etc.).

```
d <- c(1,2,3,4)
e <- c("red", "white", "red", NA)
f <- c(TRUE,TRUE,TRUE,FALSE)

mydata <- data.frame(d,e,f)
```

```
>     d     e     f
      1   red  TRUE
      2 white  TRUE
      3   red  TRUE
      4  <NA> FALSE
```



Lists

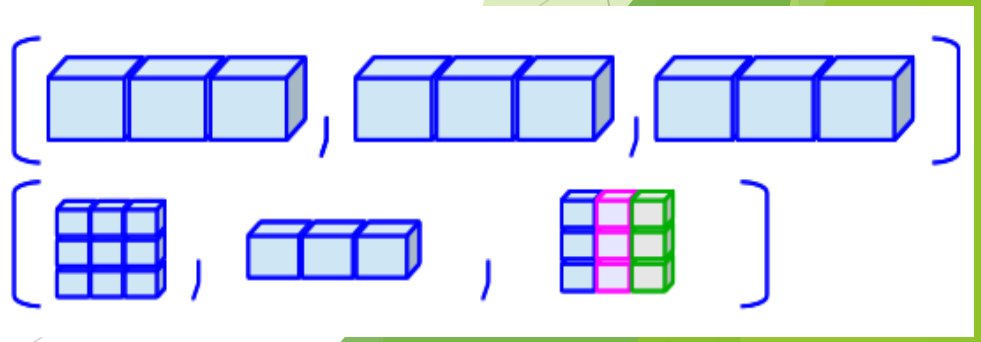
- An ordered collection of objects (components). A list allows you to gather a variety of (possibly unrelated) objects under one name.

Example of a list with 4 components -a string, a numeric vector, a matrix, and a scalar:

```
w <- list(name = "Fred", mynumbers = a, mymatrix = y, age = 5.3)
```

Example of a list containing two lists

```
v <- c(list1, list2)
```



S4 Objects

- Similar to a vector, except it has slots that can have different types of variables (character, numeric, etc.)
- Little more work to set up than other data types.

Example:

```
setClass("fieldsite",
slots=list(name="character",
size="numeric", species="integer"))

s <- new("fieldsite", name="lakeawesor
size=3.14, species=9L)
```

```
> s
An object of class "fieldsite"
Slot "name":
[1] "lakeawesome"

Slot "size":
[1] 3.14

Slot "species":
[1] 9
```

For more information and replicable examples go to:

- http://www.statmethods.net/input/datatypes.html

“R in Action” by Robert I. Kabacoff