

R Basics

Peter Solymos

Point count data analysis workshop, AOS 2019, Anchorage AK, 24 June 2019

This short document is intended to help you brush up your R skills. If you feel that these R basics are not very familiar, I suggest to take a look at some introductory R books, such as this preprint version of Norman Matloff's *The Art of R Programming* book: <http://heather.cs.ucdavis.edu/~matloff/132/NSPpart.pdf>, check out Chapters 1–6.

R is a great calculator

```
1 + 2
```

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

Assign a value and print an object

```
(x = 2) # shorthand for print
```

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

```
print(x)
```

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

```
x == 2 # logical operator, not assignment
```

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

```
y <- x + 0.5
```

```
y # another way to print
```

```
## [1] 2.5
```

Logical operators

```
x == y # equal
```

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

```
x != y # not equal
```

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

```
x < y # smaller than
```

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

```
x >= y # greater than or equal
```

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

Vectors and sequences

```
x <- c(1, 2, 3)
x
```

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

```
1:3
```

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

```
seq(1, 3, by = 1)
```

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

```
rep(1, 5)
```

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

```
rep(1:2, 5)
```

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

```
rep(1:2, each = 5)
```

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

Vector operations, recycling

```
x + 0.5
```

```
## [1] 1.5 2.5 3.5
```

```
x * c(10, 11, 12, 13)
```

```
## Warning in x * c(10, 11, 12, 13): longer object length is not a multiple of
## shorter object length
```

```
## [1] 10 22 36 13
```

Indexing vectors, ordering

```
x[1]
```

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

```
x[c(1, 1, 1)] # a way of repeating values
```

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

```
x[1:2]
```

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

```
x[x != 2]
```

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

```
x[x == 2]
```

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

```
x[x > 1 & x < 3]
```

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

```
order(x, decreasing=TRUE)
```

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

```
x[order(x, decreasing=TRUE)]
```

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

```
rev(x) # reverse
```

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

Character vectors, NA values, and sorting

```
z <- c("b", "a", "c", NA)
```

```
z[z == "a"]
```

```
## [1] "a" NA
```

```
z[!is.na(z) & z == "a"]
```

```
## [1] "a"
```

```
z[is.na(z) | z == "a"]
```

```
## [1] "a" NA
```

```
is.na(z)
```

```
## [1] FALSE FALSE FALSE TRUE
```

```
which(is.na(z))
```

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

```
sort(z)
```

```
## [1] "a" "b" "c"
```

```
sort(z, na.last=TRUE)
```

```
## [1] "a" "b" "c" NA
```

Special values

```
as.numeric(c("1", "a")) # NA: not available (missing or invalid)
```

```
## Warning: NAs introduced by coercion
```

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

```
0/0 # NaN: not a number
```

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

```
1/0 # Inf
```

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

```
-1/0 # -Inf
```

```
## [1] -Inf
```

Matrices and arrays

```
(m <- matrix(1:12, 4, 3))
```

```
##      [,1] [,2] [,3]
## [1,]    1    5    9
## [2,]    2    6   10
## [3,]    3    7   11
## [4,]    4    8   12
```

```
matrix(1:12, 4, 3, byrow=TRUE)
```

```
##      [,1] [,2] [,3]
## [1,]    1    2    3
## [2,]    4    5    6
## [3,]    7    8    9
## [4,]   10   11   12
```

```
array(1:12, c(2, 2, 3))
```

```
## , , 1
##
##      [,1] [,2]
## [1,]    1    3
## [2,]    2    4
##
## , , 2
##
##      [,1] [,2]
## [1,]    5    7
## [2,]    6    8
##
## , , 3
##
##      [,1] [,2]
## [1,]    9   11
```

```
## [2,] 10 12
```

Attribues

```
dim(m)
```

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

```
dim(m) <- NULL  
m
```

```
## [1] 1 2 3 4 5 6 7 8 9 10 11 12
```

```
dim(m) <- c(4, 3)  
m
```

```
##      [,1] [,2] [,3]  
## [1,] 1    5    9  
## [2,] 2    6   10  
## [3,] 3    7   11  
## [4,] 4    8   12
```

```
dimnames(m) <- list(letters[1:4], LETTERS[1:3])  
m
```

```
##   A B C  
## a 1 5 9  
## b 2 6 10  
## c 3 7 11  
## d 4 8 12
```

```
attributes(m)
```

```
## $dim  
## [1] 4 3  
##  
## $dimnames  
## $dimnames[[1]]  
## [1] "a" "b" "c" "d"  
##  
## $dimnames[[2]]  
## [1] "A" "B" "C"
```

Matrix indices

```
m[1:2,]
```

```
##   A B C  
## a 1 5 9  
## b 2 6 10
```

```

m[1,2]

## [1] 5
m[,2]

## a b c d
## 5 6 7 8
m[,2,drop=FALSE]

##      B
## a 5
## b 6
## c 7
## d 8
m[2]

## [1] 2
m[rownames(m) == "c",]

##      A B C
##      3 7 11
m[rownames(m) != "c",]

##      A B C
## a 1 5 9
## b 2 6 10
## d 4 8 12
m[rownames(m) %in% c("a", "c", "e"),]

##      A B C
## a 1 5 9
## c 3 7 11
m[!(rownames(m) %in% c("a", "c", "e")),]

##      A B C
## b 2 6 10
## d 4 8 12

```

Lists and indexing

```

l <- list(m = m, x = x, z = z)
l

## $m
##      A B C
## a 1 5 9

```

```
## b 2 6 10
## c 3 7 11
## d 4 8 12
##
## $x
## [1] 1 2 3
##
## $z
## [1] "b" "a" "c" NA
```

```
l$ddd <- sqrt(l$x)
l[2:3]
```

```
## $x
## [1] 1 2 3
##
## $z
## [1] "b" "a" "c" NA
```

```
l[["ddd"]]
```

```
## [1] 1.000000 1.414214 1.732051
```

Data frames

```
d <- data.frame(x = x, sqrt_x = sqrt(x))
d
```

```
##   x    sqrt_x
## 1 1 1.000000
## 2 2 1.414214
## 3 3 1.732051
```

Structure

```
str(x)
```

```
##   num [1:3] 1 2 3
```

```
str(z)
```

```
##   chr [1:4] "b" "a" "c" NA
```

```
str(m)
```

```
##   int [1:4, 1:3] 1 2 3 4 5 6 7 8 9 10 ...
##   - attr(*, "dimnames")=List of 2
##   ..$ : chr [1:4] "a" "b" "c" "d"
##   ..$ : chr [1:3] "A" "B" "C"
```

```
str(l)
```

```
## List of 4
## $ m : int [1:4, 1:3] 1 2 3 4 5 6 7 8 9 10 ...
## ..- attr(*, "dimnames")=List of 2
## .. ..$ : chr [1:4] "a" "b" "c" "d"
## .. ..$ : chr [1:3] "A" "B" "C"
## $ x : num [1:3] 1 2 3
## $ z : chr [1:4] "b" "a" "c" NA
## $ ddd: num [1:3] 1 1.41 1.73
```

```
str(d)
```

```
## 'data.frame': 3 obs. of 2 variables:
## $ x : num 1 2 3
## $ sqrt_x: num 1 1.41 1.73
```

```
str(as.data.frame(m))
```

```
## 'data.frame': 4 obs. of 3 variables:
## $ A: int 1 2 3 4
## $ B: int 5 6 7 8
## $ C: int 9 10 11 12
```

```
str(as.list(d))
```

```
## List of 2
## $ x : num [1:3] 1 2 3
## $ sqrt_x: num [1:3] 1 1.41 1.73
```

Summaries

```
summary(x)
```

```
##      Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##      1.0    1.5    2.0    2.0    2.5    3.0
```

```
summary(z)
```

```
##      Length      Class      Mode
##      4 character character
```

```
summary(m)
```

```
##           A           B           C
## Min.      :1.00   Min.      :5.00   Min.      : 9.00
## 1st Qu.:1.75   1st Qu.:5.75   1st Qu.: 9.75
## Median :2.50   Median :6.50   Median :10.50
## Mean    :2.50   Mean    :6.50   Mean    :10.50
## 3rd Qu.:3.25   3rd Qu.:7.25   3rd Qu.:11.25
## Max.    :4.00   Max.    :8.00   Max.    :12.00
```

```
summary(l)
```

```
##      Length Class  Mode
```



```
## m    12    -none- numeric
## x     3    -none- numeric
## z     4    -none- character
## ddd   3    -none- numeric
```

```
summary(d)
```

```
##           x           sqrt_x
##  Min.      :1.0    Min.      :1.000
## 1st Qu.:1.5    1st Qu.:1.207
##  Median :2.0    Median :1.414
##   Mean  :2.0    Mean   :1.382
## 3rd Qu.:2.5    3rd Qu.:1.573
##   Max.  :3.0    Max.    :1.732
```

Key takehome

A *matrix* is a vector with `dim` attribute, elements are in same mode.

A *data frame* is a list where length of elements match and elements can be in different mode.