

Lecture 15 - R Software

Ashutosh Rajput
Department of Mathematics

Rajdhani College
University of Delhi

CONSTRUCTING DATA OBJECTS

This section introduces you to constructing other objects, including lists, data frames, and matrix objects.

Making Lists

```
> mow; unmow; data3; data7; data8
[1] 12 15 17 11 15
[1] 8 9 7 9
[1] 6 7 8 7 6 3 8 9 10 7 6 9
[1] 23.0 17.0 12.5 11.0 17.0 12.0 14.5 9.0 11.0 9.0 12.5 14.5
[13] 17.0 8.0 21.0
[1] "Jan" "Feb" "Mar" "Apr" "May" "Jun" "Jul" "Aug" "Sep"
[10] "Oct" "Nov" "Dec"
```

To join several vector object together we use the `list()` command as

```
> grass.list = list(mow, unmow, data3, data7, data8)
> grass.list
[[1]]
[1] 12 15 17 11 15
```

```
[[4]]
[1] 23.0 17.0 12.5 11.0 17.0 12.0 14.5 9.0 11.0 9.0 12.5 14.5 17.0
[14] 8.0 21.0
```

```
[[5]]
[1] "Jan" "Feb" "Mar" "Apr" "May" "Jun" "Jul" "Aug" "Sep"
[10] "Oct" "Nov" "Dec"
```

Making Data Frames

A data frame is a collection of columns of data. The important thing is that all the columns must be of the same length. Any “short” vector will be padded out using NA. To make a data frame you use the `data.frame()` command:

```
my.frame = data.frame(item1, item2, item3)
```

Note: In the parentheses you give the names of the objects that you want to use as the columns, separated with commas.

```
> sample1 = c(5,6,9,12,8)
```

```
> sample2 = c(7,9,13,10,NA)
```

```
> sample1 ; sample2
```

```
[1] 5 6 9 12 8
```

```
[1] 7 9 13 10 NA
```

Therefore, the command gives

```
> my.frame = data.frame(sample1, sample2)
> my.frame
```

	sample1	sample2
1	5	7
2	6	9
3	9	13
4	12	10
5	8	NA

We can use a command called `rep()` to repeat items multiple times. The general form of the command is: `rep(item, times)`

```
> sample1 = c(5,6,9,12,8,4,7,11)
> sample2 = c(7,9,13,10, rep(NA, 2))
> sample1 ; sample2
[1] 5 6 9 12 8 4
[1] 7 9 13 10 NA NA
> my.frame = data.frame(sample1, sample2)
```

```
> my.frame
  sample1 sample2
1      5      7
2      6      9
3      9     13
4     12     10
5      8     NA
6      4     NA
```

Making Matrix Objects

If you have vectors of data that you want to form the columns of your matrix, you can combine them using the `cbind()` command:

```
> sample1 ; sample2
[1] 5 6 9 12
[1] 7 9 13 10
> cmat = cbind(sample1, sample2)
> cmat
```

```
  sample1 sample2
[1,]     5      7
[2,]     6      9
[3,]     9     13
[4,]    12     10
```

We can use the `rbind()` command in a similar fashion:

```
> rmat = rbind(sample1, sample2)
> rmat
```

	[,1]	[,2]	[,3]	[,4]	[,5]
sample1	5	6	9	12	8
sample2	7	9	13	10	NA

★ To add another column in an existing matrix, we write it as;

```
> sample3
[1] "a" "b" "c" "d" "e"
> mix.mat = cbind(sample1, sample2, sample3)
> mix.mat
```

	sample1	sample2	sample3
[1,]	"5"	"7"	"a"
[2,]	"6"	"9"	"b"
[3,]	"9"	"13"	"c"
[4,]	"12"	"10"	"d"
[5,]	"8"	NA	"e"

Remark: The NA always remains as it is (note that it does not appear in quotation marks).

Suppose, the matrix data be given as:

```
> mat = matrix(all.samples, ncol = 2)
> mat
      [,1] [,2]
[1,]    5    7
[2,]    6    9
[3,]    9   13
[4,]   12   10
[5,]    8   NA
```

Then, the `dimnames()` command can also be used as an instruction within the `matrix()` command like so:

```
> cnam = c('Sample1', 'Sample2')
> rnam = c('Site1', 'Site2', 'Site3', 'Site4', 'Site5')
> mat = matrix(all.samples, ncol = 2, dimnames = list(rnam, cnam))
> mat
      Sample1 Sample2
Site1      5      7
Site2      6      9
Site3      9     13
Site4     12     10
Site5      8     NA
```