Control Structures

Gerardo Ferrara

Master in Economics and Complexity, Collegio Carlo Alberto

Fall 2013

Introduction

Control structures in R allow you to control the flow of execution of the program, depending on runtime conditions. Common structures are:

- **if, else**: testing a condition.

- **for**: execute a loop a fixed number of times.

- **while**: execute a loop while a condition is true.

- **repeat**: execute an infinite loop. They should generally be avoided, even if they are theoretically correct.

- **break**: break the execution of a loop.

Control Structures: if

$$if(< condition >)\{$$
$$\#do\ something$$
$$\}$$
$$else\{$$
$$\#do\ something\ different$$
$$\}$$
$$if(< condition1 >)\{$$
$$\#do\ something$$
$$\}$$
$$else\ if(< condition2 >)\{$$
$$\#do\ something\ different$$
$$\}$$
$$else\{$$
$$\#do\ something\ different$$
$$\}$$

Control Structures: for

For loops take an interator variable and assign it successive values from a sequence or vector. They are most commonly used for iterating over the elements of an object (list, vector, etc.).

```
for(j in 1 : 10){
                 #do something
                 }
```

Loops can also be nested:

```
> m <- matrix(1 : 6, 2, 3)
for(i in seq_len(nrow(m))){
 for(j in seq_len(ncol(m))){
                           print(x[i, j])
                           }
                           }
```

Be careful with nesting though. Nesting beyond 2 or 3 levels is often very difficult to read/understand.

Control Structures: while

While loops begin by testing a condition. If it is true, then they execute the loop body. Once the loop body is executed, the condition is tested again, and so forth.

> $count <- 0$
while($count < 0$){
                        #do something
                        $count <- count + 1$
                        }

Sometimes there will be more than one condition in the test. Conditions are always evaluated from left to right.

> $count <- 4$
while($count >= 5$ && $count <= 9$){
                                        #do something
                                        $count <- count + 1$
                                        }

repeat initiates an infinite loop; these are not commonly used in statistical applications but they do have their uses. The only way to exit a **repeat** loop is to call **break**.

```
repeat{
        #do something
        if (condition)
        break
        }
```

We can re-write a function using a **repeat** loop instead of a for **loop**.