

Kaitlyn Valentinetti

September 21, 2022

Eco 634: Michael Nelson

Lab 2: R Fundamentals 2

1. `vec_2 <- vec_1 == 3`
2. It's a bad idea to determine which elements in `vec_1` have a value of 3 by visual inspection because the vector is extremely long, and it would take hours to sort through of all the values visually. Secondly, there is a high probability that measurement error will happen, and a value would be missed, so the total number would be incorrect.
3. I didn't get the same count of "3" each time because R is generating a sample of ten random values that are between 0 and 12. The sample function used randomly chooses values, so each entry will generate a unique vector; therefore, there will be a different count of "3" each time.
4. Using a logical test is a safe way to select entries with a value of 3 because it is a binary code that filters elements into two categories: true or false. When 3 is categorized as true and all other values are false, it simplifies the data categorically and makes it easy to select entries with values of 3 and discard everything else.
5. It's a bad practice to perform logical "by hand" subsetting because it is not easily reproducible, especially with larger data sets. Manually subsetting is time-consuming and risks human error, and if collaborators are interested in replicating your test, it will take them even longer to reproduce what you did successfully. However, logical testing in R is highly efficient because once the code is run, it can be copied and pasted by collaborators to utilize even quicker than the code was generated. In this sense, the code produces a reusable formula that is quick, easy, and accessible, regardless of the data set size.
6.

```
for (i in 1:10)
{
  print(paste0("This is a loop iteration: ", i))
}
```
7.

```
n = 5
for(i in 1:n)
```

```
{  
  print(paste0("This is a loop iteration: ", i))  
}
```

8. n= 17

```
vec_1 <- sample(10, n, replace = TRUE)  
for(i in 1:n)  
{  
  print(paste0("The element of vec_1 at index ", i, " is: ", vec_1[[i]]))  
}
```

9. create_and_print_vec = function(n, min = 1, max = 10)

```
{  
  vec_3 <- sample(min:max, n, replace= TRUE)  
  for(i in 1:n)  
  {  
    print(paste0("The element at index ", i, " is ", vec_3[[i]]))  
  }}  
create_and_print_vec(5)
```