Tutorial_23Sep2024

Yin Chen Wan

2024-09-21

Tutorial

Some super basic recap

Vector

Accessing the vector content: vector[position number]:

```
words <- c("one", "two", "three", "four", "five")
words
```

```
## [1] "one"   "two"   "three" "four"  "five"
```

```
#Accessing each element
words[1]
```

```
## [1] "one"
```

```
words[2]
```

```
## [1] "two"
```

```
words[3]
```

```
## [1] "three"
```

```
words[4]
```

```
## [1] "four"
```

```
words[5]
```

```
## [1] "five"
```

```
#Accessing more than 1 element
words[1:3]
```

```
## [1] "one"   "two"   "three"
```

```
words[3:5]
```

```
## [1] "three" "four"  "five"
```

```
#Putting stuff in vector (e.g. what if I want a vector with "one", "two", "three", "five"?)
#words[1:3, 5] #error #exceeds the dimension
words[1:3]
```

```
## [1] "one"   "two"   "three"
```

```
words[5]
```

```
## [1] "five"
```

```r
c(words[1:3], words[5]) #try running the functions inside individually
```

```
## [1] "one"   "two"   "three" "five"
```

```r
one_two_three_five <- c(words[1:3], words[5])
one_two_three_five_four <- c(one_two_three_five, words[4])
```

Data Frame

Since "gene_condition_source_id.txt" file is so big, I will use a small section of it as an example:

```r
#Read the "gene_condition_source_id.txt" file
dataFrame = read.table("gene_condition_source_id.txt", header=TRUE, sep="\t", fill=T, comment.char = ""

#Look at the top few rows
head(dataFrame)
```

```
##   X.GeneID AssociatedGenes RelatedGenes ConceptID
## 1   144568           A2ML1               C1833692
## 2    53947          A4GALT               C3549485
## 3     8086            AAAS               C0271742
## 4    79719           AAGAB               CN031225
## 5       16           AARS1               C4225361
## 6       16           AARS1               C5562057
##                                     DiseaseName    SourceName      SourceID
## 1                Otitis media, susceptibility to         MONDO MONDO:0008162
## 2                      Blood group, P1PK system NCBI curation
## 3       Glucocorticoid deficiency with achalasia NCBI curation
## 4     Palmoplantar keratoderma, punctate type 1A         MONDO MONDO:0007858
## 5 Developmental and epileptic encephalopathy, 29         MONDO MONDO:0014593
## 6       Trichothiodystrophy 8, nonphotosensitive         MONDO MONDO:0030517
##   DiseaseMIM LastUpdated
## 1     166760 Feb 16 2016
## 2     111400 Mar 10 2022
## 3     231550 Feb 16 2016
## 4     148600 May 21 2021
## 5     616339 Jan 21 2022
## 6     619691 Apr 19 2022
```

```r
#save the top few rows as an example we will run
dataFrame_head <- head(dataFrame)
```

Accessing the data frame content: dataframe[row number, column number]: *Tip is to keep checking the table itself*

```r
#Accessing each cell/element
dataFrame_head[1,1] #row 1, column 1
```

```
## [1] 144568
```

```r
#going through each row of column 1
dataFrame_head[1,1]
```

```
## [1] 144568
```

```r
dataFrame_head[2,1]
```

```
## [1] 53947
```

```
dataFrame_head[3,1]
```

```
## [1] 8086
```

```
dataFrame_head[4,1]
```

```
## [1] 79719
```

```
dataFrame_head[5,1]
```

```
## [1] 16
```

```
dataFrame_head[6,1]
```

```
## [1] 16
```

```
#now play with column 2, 3...

#going through each column of row 1
dataFrame_head[1,1]
```

```
## [1] 144568
```

```
dataFrame_head[1,2]
```

```
## [1] "A2ML1"
```

```
dataFrame_head[1,3]
```

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

```
dataFrame_head[1,4]
```

```
## [1] "C1833692"
```

```
dataFrame_head[1,5]
```

```
## [1] "Otitis media, susceptibility to"
```

```
dataFrame_head[1,6]
```

```
## [1] "MONDO"
```

```
dataFrame_head[1,7]
```

```
## [1] "MONDO:0008162"
```

```
dataFrame_head[1,8]
```

```
## [1] 166760
```

```
dataFrame_head[1,9]
```

```
## [1] "Feb 16 2016"
```

```
dataFrame_head[1,10] #Oops, there are only 9 columns!
```

```
## NULL
```

```
#now play with column 2, 3...
```

Accessing data frame entire row/column:

```
#Accessing row
dataFrame_head[1, ]
```

```
##   X.GeneID AssociatedGenes RelatedGenes ConceptID
## 1   144568           A2ML1               C1833692
##                       DiseaseName SourceName      SourceID DiseaseMIM
## 1 Otitis media, susceptibility to      MONDO MONDO:0008162     166760
##   LastUpdated
## 1 Feb 16 2016
```

```
dataFrame_head[2, ]
```

```
##   X.GeneID AssociatedGenes RelatedGenes ConceptID              DiseaseName
## 2    53947          A4GALT               C3549485 Blood group, P1PK system
##      SourceName SourceID DiseaseMIM LastUpdated
## 2 NCBI curation              111400 Mar 10 2022
```

```
dataFrame_head[3, ]
```

```
##   X.GeneID AssociatedGenes RelatedGenes ConceptID
## 3     8086            AAAS               C0271742
##                               DiseaseName    SourceName SourceID DiseaseMIM
## 3 Glucocorticoid deficiency with achalasia NCBI curation              231550
##   LastUpdated
## 3 Feb 16 2016
```

```
#Accessing column
dataFrame_head[, 9]
```

```
## [1] "Feb 16 2016" "Mar 10 2022" "Feb 16 2016" "May 21 2021" "Jan 21 2022"
## [6] "Apr 19 2022"
```

```
dataFrame_head[, 2]
```

```
## [1] "A2ML1"  "A4GALT" "AAAS"   "AAGAB"  "AARS1"  "AARS1"
```

```
dataFrame_head[, 5]
```

```
## [1] "Otitis media, susceptibility to"
## [2] "Blood group, P1PK system"
## [3] "Glucocorticoid deficiency with achalasia"
## [4] "Palmoplantar keratoderma, punctate type 1A"
## [5] "Developmental and epileptic encephalopathy, 29"
## [6] "Trichothiodystrophy 8, nonphotosensitive"
```

```
#Accessing column by column name 'S'
dataFrame_head$AssociatedGenes
```

```
## [1] "A2ML1"  "A4GALT" "AAAS"   "AAGAB"  "AARS1"  "AARS1"
```

```
dataFrame_head$SourceName
```

```
## [1] "MONDO"         "NCBI curation" "NCBI curation" "MONDO"
## [5] "MONDO"         "MONDO"
```

Comparison operators

```
<           less than
<=          less than or equal to
>           greater than
>=          greater than or equal to
==          exactly equal to
!=          not equal to
```

```
!x          not x
x | y       x OR y
x & y       x AND y e.g. if((x != 3) & (x > 0))
```

Comparison operators examples. Vectors:

```
words <- c("one", "two", "three", "four", "five")
words == ("four")
```

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

```
words[words == ("four")]
```

```
## [1] "four"
```

```
numbers <- runif(30, min = 0, max = 50)
numbers > 19
```

```
##  [1]  TRUE  TRUE FALSE  TRUE FALSE FALSE FALSE  TRUE  TRUE  TRUE  TRUE FALSE
## [13]  TRUE  TRUE  TRUE FALSE  TRUE FALSE  TRUE  TRUE  TRUE  TRUE FALSE  TRUE
## [25]  TRUE FALSE FALSE  TRUE FALSE FALSE
```

```
numbers[numbers > 19]
```

```
##  [1] 31.71032 34.09286 35.28114 29.63248 29.82017 26.55791 24.90940 28.49029
##  [9] 38.29023 41.29100 25.93658 40.11405 47.57433 37.59773 19.07986 44.42583
## [17] 42.03688 21.06002
```

Data frames:

```
dataFrame_head$AssociatedGenes == "AAGAB"
```

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

```
dataFrame_head$AssociatedGenes[dataFrame_head$AssociatedGenes == "AAGAB"]
```

```
## [1] "AAGAB"
```

```
dataFrame_head$AssociatedGenes[dataFrame_head$AssociatedGenes == "TP53"]
```

```
## character(0)
```

Data frame example: How many diseases associated with AARS1?

```
dataFrame_head$AssociatedGenes == "AARS1"
```

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

```
dataFrame_head$AssociatedGenes[dataFrame_head$AssociatedGenes == "AARS1"]
```

```
## [1] "AARS1" "AARS1"
```

```
dataFrame_head$DiseaseName[dataFrame_head$AssociatedGenes == "AARS1"]
```

```
## [1] "Developmental and epileptic encephalopathy, 29"
## [2] "Trichothiodystrophy 8, nonphotosensitive"
```

If else statements

if (a condition is met) { do something }

Example 1 using only "if":

```r
pH <- 10

if (pH < 7) {
  print("acidic")
}

if (pH == 7) {
  print("neutral")
}

if (pH > 7) {
  print("basic")
}
```

```
## [1] "basic"
```

Example 2 using "if" and "else:

```r
pH <- 8

if (pH < 7) {
  print("acidic")
} else if (pH == 7) {
  print("neutral")
} else {
  print("basic")
}
```

```
## [1] "basic"
```

For loops

for (element in (a series of something)) { do something }

You need to know how many iterations beforehand

Super break down of a for loop using an example:

```r
words <- c("one", "two", "three", "four", "five")

i=1
words[i]
```

```
## [1] "one"
```

```r
i=2
words[i]
```

```
## [1] "two"
```

```r
i=3
words[i]
```

```
## [1] "three"
```

```r
i=4
words[i]
```

```
## [1] "four"
```

```
i=5
words[i]
```

```
## [1] "five"
```

```
for (i in 1:length(words)) {
  print(i)
  print(words[i])
}
```

```
## [1] 1
## [1] "one"
## [1] 2
## [1] "two"
## [1] 3
## [1] "three"
## [1] 4
## [1] "four"
## [1] 5
## [1] "five"
```

For loop example 1 (vector): Add one to every element in a vector

```
numbers <- 100:120

#First make sure you are looping fine
for (each_number in 1:length(numbers)) {
  print(each_number)
}
```

```
## [1] 1
## [1] 2
## [1] 3
## [1] 4
## [1] 5
## [1] 6
## [1] 7
## [1] 8
## [1] 9
## [1] 10
## [1] 11
## [1] 12
## [1] 13
## [1] 14
## [1] 15
## [1] 16
## [1] 17
## [1] 18
## [1] 19
## [1] 20
## [1] 21
```

```
for (i in 1:length(numbers)) {
  print(i)
}
```

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

```
## [1] 2
## [1] 3
## [1] 4
## [1] 5
## [1] 6
## [1] 7
## [1] 8
## [1] 9
## [1] 10
## [1] 11
## [1] 12
## [1] 13
## [1] 14
## [1] 15
## [1] 16
## [1] 17
## [1] 18
## [1] 19
## [1] 20
## [1] 21
```

```
for (i in 1:length(numbers)) {
  print(numbers[i])
}
```

```
## [1] 100
## [1] 101
## [1] 102
## [1] 103
## [1] 104
## [1] 105
## [1] 106
## [1] 107
## [1] 108
## [1] 109
## [1] 110
## [1] 111
## [1] 112
## [1] 113
## [1] 114
## [1] 115
## [1] 116
## [1] 117
## [1] 118
## [1] 119
## [1] 120
```

```
#Now 'do something'
for (i in 1:length(numbers)) {
  print(numbers[i]+1)
}
```

```
## [1] 101
## [1] 102
## [1] 103
## [1] 104
```

```
## [1] 105
## [1] 106
## [1] 107
## [1] 108
## [1] 109
## [1] 110
## [1] 111
## [1] 112
## [1] 113
## [1] 114
## [1] 115
## [1] 116
## [1] 117
## [1] 118
## [1] 119
## [1] 120
## [1] 121
```

```
for (i in 1:length(numbers)) {
  added1 <- (numbers[i]) +1
  print(added1)
}
```

```
## [1] 101
## [1] 102
## [1] 103
## [1] 104
## [1] 105
## [1] 106
## [1] 107
## [1] 108
## [1] 109
## [1] 110
## [1] 111
## [1] 112
## [1] 113
## [1] 114
## [1] 115
## [1] 116
## [1] 117
## [1] 118
## [1] 119
## [1] 120
## [1] 121
```

```
#try something else! E.g. add 2 to every element etc.
for (i in 1:length(numbers)) {
  print(numbers[i]+2)
}
```

```
## [1] 102
## [1] 103
## [1] 104
## [1] 105
## [1] 106
## [1] 107
```

```
## [1] 108
## [1] 109
## [1] 110
## [1] 111
## [1] 112
## [1] 113
## [1] 114
## [1] 115
## [1] 116
## [1] 117
## [1] 118
## [1] 119
## [1] 120
## [1] 121
## [1] 122
```

```
for (i in 1:length(numbers)) {
  added2 <- (numbers[i]) +2
  print(added2)
}
```

```
## [1] 102
## [1] 103
## [1] 104
## [1] 105
## [1] 106
## [1] 107
## [1] 108
## [1] 109
## [1] 110
## [1] 111
## [1] 112
## [1] 113
## [1] 114
## [1] 115
## [1] 116
## [1] 117
## [1] 118
## [1] 119
## [1] 120
## [1] 121
## [1] 122
```

```
#Easily check your answers with vectorisation
numbers+1
```

```
##  [1] 101 102 103 104 105 106 107 108 109 110 111 112 113 114 115 116 117 118 119
## [20] 120 121
```

For loop example 2 (looping through the rows of a data frame): What are the associated diseases in the data frame?

```
#First make sure you are looping fine
for (i in 1:nrow(dataFrame_head)) {
  print(i)
}
```

```
## [1] 1
## [1] 2
## [1] 3
## [1] 4
## [1] 5
## [1] 6
```

```
#Looping through the rows of the associated diseases column
for (i in 1:nrow(dataFrame_head)) {
  print(dataFrame_head[i, 5])
}
```

```
## [1] "Otitis media, susceptibility to"
## [1] "Blood group, P1PK system"
## [1] "Glucocorticoid deficiency with achalasia"
## [1] "Palmoplantar keratoderma, punctate type 1A"
## [1] "Developmental and epileptic encephalopathy, 29"
## [1] "Trichothiodystrophy 8, nonphotosensitive"
```

```
for (i in 1:nrow(dataFrame_head)) {
  print(dataFrame_head$DiseaseName[i])
}
```

```
## [1] "Otitis media, susceptibility to"
## [1] "Blood group, P1PK system"
## [1] "Glucocorticoid deficiency with achalasia"
## [1] "Palmoplantar keratoderma, punctate type 1A"
## [1] "Developmental and epileptic encephalopathy, 29"
## [1] "Trichothiodystrophy 8, nonphotosensitive"
```

```
#Easily check answer
dataFrame_head$DiseaseName
```

```
## [1] "Otitis media, susceptibility to"
## [2] "Blood group, P1PK system"
## [3] "Glucocorticoid deficiency with achalasia"
## [4] "Palmoplantar keratoderma, punctate type 1A"
## [5] "Developmental and epileptic encephalopathy, 29"
## [6] "Trichothiodystrophy 8, nonphotosensitive"
```

While loops

counter = 1 for (a condition) { do something

counter= counter+1 }

+1 so that it can move forward

Super break down of a loop using an example:

```
words <- c("one", "two", "three", "four", "five")

i=1
words[i]
```

```
## [1] "one"
```

```
i=2
words[i]
```

```
## [1] "two"
```

```
i=3
words[i]
```

```
## [1] "three"
```

```
i=4
words[i]
```

```
## [1] "four"
```

```
i=5
words[i]
```

```
## [1] "five"
```

```
i=1
while (i<6) {
  print(i)

  i=i+1 #remove and see what happens if you don't add to the counter?
}
```

```
## [1] 1
## [1] 2
## [1] 3
## [1] 4
## [1] 5
```

```
i=1
while (i<6) {
  print(i)
  print(words[i])

  i=i+1
}
```

```
## [1] 1
## [1] "one"
## [1] 2
## [1] "two"
## [1] 3
## [1] "three"
## [1] 4
## [1] "four"
## [1] 5
## [1] "five"
```

```
#what if you set more than length of vector?
#vector only has 5 elements, so if it's more than 5 then it will be error
```

While loop example 1 (vector): Add one to every element in vector For loop example 1 (vector): Add one to every element in a vector

```
numbers <- 100:120
```

```r
#First make sure you are looping fine
i=1
while (i<21) {
  print(i)

  i=i+1
}
```

```
## [1] 1
## [1] 2
## [1] 3
## [1] 4
## [1] 5
## [1] 6
## [1] 7
## [1] 8
## [1] 9
## [1] 10
## [1] 11
## [1] 12
## [1] 13
## [1] 14
## [1] 15
## [1] 16
## [1] 17
## [1] 18
## [1] 19
## [1] 20
```

```r
i=1
while (i<21) {
  print(numbers[i])

  i=i+1
}
```

```
## [1] 100
## [1] 101
## [1] 102
## [1] 103
## [1] 104
## [1] 105
## [1] 106
## [1] 107
## [1] 108
## [1] 109
## [1] 110
## [1] 111
## [1] 112
## [1] 113
## [1] 114
## [1] 115
## [1] 116
## [1] 117
## [1] 118
```

```
## [1] 119
```

```
#Now 'do something'
for (i in 1:length(numbers)) {
  print(numbers[i]+1)
}
```

```
## [1] 101
## [1] 102
## [1] 103
## [1] 104
## [1] 105
## [1] 106
## [1] 107
## [1] 108
## [1] 109
## [1] 110
## [1] 111
## [1] 112
## [1] 113
## [1] 114
## [1] 115
## [1] 116
## [1] 117
## [1] 118
## [1] 119
## [1] 120
## [1] 121
```

```
#try something else! E.g. add 2 to every element etc.

#Easily check your answers with vectorisation
numbers+1
```

```
##  [1] 101 102 103 104 105 106 107 108 109 110 111 112 113 114 115 116 117 118 119
## [20] 120 121
```

While loop example 2 (how to loop through more than 1 at a time): Here are the temperatures of 28 days (1 month/ 4 weeks). What is the average temperature per week? # kindly look like the assignement

```
monthly_temp <- runif(28, min=0, max = 40)
monthly_temp
```

```
##  [1] 33.499612 17.614426 38.392068  1.200974 28.567060 15.696759  6.342629
##  [8] 35.412466 26.619012 19.325693 20.022859 37.565015  0.130359  7.251136
## [15] 28.920464  8.627942 25.596648 27.514464  3.681061 19.790150  2.696620
## [22] 23.149022 32.183270 28.353371 30.408070  9.480723 28.452995 20.935466
```

Instead of looping through every single number, you loop through every seven days

```
#want to get every 7 days in 1 iteration

#first week
i=1
first_week <- monthly_temp[i: (i+7)]
first_week
```

```
## [1] 33.499612 17.614426 38.392068  1.200974 28.567060 15.696759  6.342629
```

```
## [8] 35.412466
```

```
length(first_week)
```

```
## [1] 8
```

```
first_week <- monthly_temp[i: (i+6)]
length(first_week)
```

```
## [1] 7
```

```
mean(first_week)
```

```
## [1] 20.18765
```

```
#2nd week
i=8
second_week <- monthly_temp[i: (i+6)]
length(second_week)
```

```
## [1] 7
```

```
mean(second_week)
```

```
## [1] 20.90379
```

```
#3rd week
i=15
third_week <- monthly_temp[i: (i+6)]
length(third_week)
```

```
## [1] 7
```

```
mean(third_week)
```

```
## [1] 16.68962
```

```
#what code is changing and what code is the same?

#First make sure you are looping fine
i=1
while (i<29) {
  print(i)

  i=i+7 #the difference between every iteration is 7 days
}
```

```
## [1] 1
## [1] 8
## [1] 15
## [1] 22
```

```
#now that the loop is working, enter what your task
i=1
while (i<29) {
  print(i)
  week <- monthly_temp[i: (i+6)]
  print(week)
  print(length(week))
  print(mean(week))
```

```
  i=i+7 #the difference between every iteration is 7 days
}
```

```
## [1] 1
## [1] 33.499612 17.614426 38.392068  1.200974 28.567060 15.696759  6.342629
## [1] 7
## [1] 20.18765
## [1] 8
## [1] 35.412466 26.619012 19.325693 20.022859 37.565015  0.130359  7.251136
## [1] 7
## [1] 20.90379
## [1] 15
## [1] 28.920464  8.627942 25.596648 27.514464  3.681061 19.790150  2.696620
## [1] 7
## [1] 16.68962
## [1] 22
## [1] 23.149022 32.183270 28.353371 30.408070  9.480723 28.452995 20.935466
## [1] 7
## [1] 24.70899
```

What if I want to put all averages into a vector? vector <- c(vector, things_to_put_in)

```
avg_temp <- c() #create an empty vector to put the average temperature in

#1st week
i=1
first_week <- monthly_temp[i: (i+6)]
avg_temp <- c(avg_temp, mean(first_week))
avg_temp
```

```
## [1] 20.18765
```

```
#2nd week
i=8
second_week <- monthly_temp[i: (i+6)]
avg_temp <- c(avg_temp, mean(second_week))
avg_temp
```

```
## [1] 20.18765 20.90379
```

```
#3rd week
i=15
third_week <- monthly_temp[i: (i+6)]
avg_temp <- c(avg_temp, mean(third_week))
avg_temp
```

```
## [1] 20.18765 20.90379 16.68962
```

```
#what code is changing and what code is the same?

avg_temp <- c()
i=1
while (i<29) {
  week <- monthly_temp[i: (i+6)]
  avg_temp <- c(avg_temp, mean(week))

  i=i+7 #the difference between every iteration is 7 days
```

```
}
avg_temp
```

```
## [1] 20.18765 20.90379 16.68962 24.70899
```

Putting if else staments into a loop

for (element in (a series of something)) { if (a condition is met) { do something } else { do something } }

Make sure the if else statement is working first! if else statement in a while loop example: Here are the temperatures of 28 days (1 month/ 4 weeks). How is the weather for each week? Assuming temperatures less than 10 are cold, temperatures between 10-20 is warm, and temperatures above 20 is hot. *live coding example*

```
# Example of monthly temperatures (just dummy data for 28 days)
monthly_temp <- c(15, 12, 10, 13, 17, 19, 14, 22, 23, 25, 20, 21, 18, 19,
                  16, 14, 13, 12, 10, 9, 8, 7, 6, 5, 4, 3, 2, 1)

# 1st week
i = 1
first_week <- monthly_temp[i:(i+6)]
weather <- mean(first_week)

if (weather < 10) {
  print("cold")
} else if (weather > 20) {
  print("hot")
} else {
  print("warm")
}
```

```
## [1] "warm"
```

```
# 2nd week
i = 8
second_week <- monthly_temp[i:(i+6)]  # Adjust this to cover days 8 to 14
weather <- mean(second_week)

if (weather < 10) {
  print("cold")
} else if (weather > 20) {
  print("hot")
} else {
  print("warm")
}
```

```
## [1] "hot"
```

```
# 3rd week
i = 15
third_week <- monthly_temp[i:(i+6)]  # Adjust this to cover days 15 to 21
weather <- mean(third_week)

if (weather < 10) {
  print("cold")
} else if (weather > 20) {
  print("hot")
} else {
```

```
  print("warm")
}
```

```
## [1] "warm"
```

```
# 4th week
i = 22
fourth_week <- monthly_temp[i:(i+6)]  # Adjust this to cover days 22 to 28
weather <- mean(fourth_week)

if (weather < 10) {
  print("cold")
} else if (weather > 20) {
  print("hot")
} else {
  print("warm")
}
```

```
## [1] "cold"
```

```
#we have a partial code for this (see R chunk above)
#live coding example
```

TIPS

1. Write out the code for the first few iterations Check that your task is working. Identify what parts are the same and what parts are changing.

2. Check your looping is okay.

for (i in 1:vector) { print(i) } *Make sure i is going in an order you expect*

3. Replace the content in loop with your if else statement

Make sure the statement is accepting input from the loop