Combinations

Zheng Pan

7/18/2021

Contents

Introduction **2**

code **3**

prerequisites . . . 3

load required libraries . . . 3

Input all possible choice of A, B and X . . . 3

Create a function named od() . . . 3

Create a function named conver_ele . . . 3

Construction of A4BX6 . . . 4

Construction of A4 . . . 4

Same for X . . . 5

Aggregate A, B and X . . . 6

Introduction

This is a tutorial on how do we merge vectors into a matrix, whose element is the combination of elements of vectors.

The library ‘dplyr’ in library ‘tidyverse’ provides functions that can easily get the job done. But we will not mention it here(maybe it will be mentioned in later updates).

In this tutorial we are just ganna use several ‘for’ iterations.

code

prerequisites

load required libraries

```r
library('tidyverse')
library('readr')
# This library entitles us to read .csv more properly
# by using read_csv() or read_csv2().
# Users can also use R's basic file input function: 'read.csv()'
```

If you get an error loading it, you should probably install or update some of required packages using :

```r
install.packages("NAME OF PACKAGE")
```

Input all possible choice of A, B and X

```r
A <- c('Li', 'Na', 'K', 'Rb', 'Cs')
B <- c('Ge', 'Sn', 'Pb')
X <- c('F', 'Cl', 'Br', 'I')
```

Create a function named od()

Function od() ranks the position of a vector ascendingly according to the position of the element predefined above.

```r
od <- function(f_s){
  str_c(sort((as.numeric(strsplit(f_s, ',')[[1]]))), collapse = ',')
}
```

For example:

```r
od("3,2,1,1")
```

```
## [1] "1,1,2,3"
```

The number indicates the position of elements in A, B or X.
That is to say, “3,2,1,1” means “K,Na,Li,Li” in A.

Create a function named conver_ele

Create a function through the name of element if converted from numbers od() produced.
It takes 2 parameter: f_s and f_pos.
f_s is the string of numbers that shall be produced by od().
f_pos is the position of element, if should be ‘A’ or ‘X’.

```
conver_ele <- function(f_s,f_pos){
  if(f_pos == "A"){
    str_c(A[as.numeric(strsplit(f_s, ',')[[1]])], collapse = ',')
  }else if(f_pos == "X"){
    str_c(X[as.numeric(strsplit(f_s, ',')[[1]])], collapse = ',')
  }else{
    NULL
  }
}
```

For example:

```
conver_ele(f_s = "1,1,2,3", f_pos = "A")
```

```
## [1] "Li,Li,Na,K"
```

```
conver_ele(f_s = "1,1,2,3", f_pos = "X")
```

```
## [1] "F,F,Cl,Br"
```

Construction of A4BX6

With such a long prelude, finally we can do what we want:

Construction of A4

The main idea is that first we use a 4-dimensional array to save the information of A, in which each dimension stands for ‘A1’,‘A2’,‘A3’ or ‘A4’ respectively.

Then we collapse it into one-dimensional vector.

```
A_arr <- array(rep(0, 5^4), c(5,5,5,5))
for (s1 in 1:5) {
  for (s2 in 1:5) {
    for (s3 in 1:5) {
      for (s4 in 1:5) {
            A_arr[s1, s2, s3, s4] <- od(paste(s1, s2, s3, s4, sep = ','))
      }
    }
  }
}
A_vec <- as.vector(A_arr)
```

First 10 elements of A_vec is :

```
data.frame(A = A_vec[1:10])
```

```
##          A
## 1  1,1,1,1
## 2  1,1,1,2
## 3  1,1,1,3
## 4  1,1,1,4
## 5  1,1,1,5
## 6  1,1,1,2
## 7  1,1,2,2
## 8  1,1,2,3
## 9  1,1,2,4
## 10 1,1,2,5
```

We can see that the second and the sixth element is same, we must remove duplicated elements.
That's why we rank them above.
Now we remove duplicated elements using the code below, which shows the meaning literally, through which we can feel the charm of R.

```
A_vec <- A_vec[!duplicated(A_vec)]
```

Now we convert numbers to names unsing conver_ele().

```
A_vec <- sapply(1:length(A_vec), function(q){
  conver_ele(A_vec[q], 'A')
})
```

The first 20 combinations of A are:

```
data.frame(A = A_vec[1:20])
```

```
##               A
## 1  Li,Li,Li,Li
## 2  Li,Li,Li,Na
## 3   Li,Li,Li,K
## 4  Li,Li,Li,Rb
## 5  Li,Li,Li,Cs
## 6  Li,Li,Na,Na
## 7   Li,Li,Na,K
## 8  Li,Li,Na,Rb
## 9  Li,Li,Na,Cs
## 10   Li,Li,K,K
## 11  Li,Li,K,Rb
## 12  Li,Li,K,Cs
## 13 Li,Li,Rb,Rb
## 14 Li,Li,Rb,Cs
## 15 Li,Li,Cs,Cs
## 16 Li,Na,Na,Na
## 17  Li,Na,Na,K
## 18 Li,Na,Na,Rb
## 19 Li,Na,Na,Cs
## 20   Li,Na,K,K
```

Same for X

```
X_arr <- array(rep(0, 4^6), c(4,4,4,4,4,4))
for (s1 in 1:4) {
  for (s2 in 1:4) {
    for (s3 in 1:4) {
      for (s4 in 1:4) {
        for (s5 in 1:4) {
          for (s6 in 1:4) {
            X_arr[s1, s2, s3, s4, s5, s6] <- od(paste(s1, s2, s3, s4, s5, s6, sep = ','))
          }
        }
      }
    }
  }
}
X_vec <- as.vector(X_arr)
X_vec <- X_vec[!duplicated(X_vec)]
X_vec <- sapply(1:length(X_vec), function(q){
  conver_ele(X_vec[q], 'X')
})
```

The first 20 combinations of X are:

```
data.frame(X = X_vec[1:20])
```

```
##                   X
## 1     F,F,F,F,F,F
## 2    F,F,F,F,F,Cl
## 3    F,F,F,F,F,Br
## 4     F,F,F,F,F,I
## 5   F,F,F,F,Cl,Cl
## 6   F,F,F,F,Cl,Br
## 7    F,F,F,F,Cl,I
## 8   F,F,F,F,Br,Br
## 9    F,F,F,F,Br,I
## 10    F,F,F,F,I,I
## 11 F,F,F,Cl,Cl,Cl
## 12 F,F,F,Cl,Cl,Br
## 13  F,F,F,Cl,Cl,I
## 14 F,F,F,Cl,Br,Br
## 15  F,F,F,Cl,Br,I
## 16   F,F,F,Cl,I,I
## 17 F,F,F,Br,Br,Br
## 18  F,F,F,Br,Br,I
## 19   F,F,F,Br,I,I
## 20    F,F,F,I,I,I
```

Aggregate A, B and X

```
A4BX6 <- array(0, c(length(A_vec), length(B), length(X_vec)))
for (a in 1:length(A_vec)) {
  for (b in 1:length(B)) {
    for (x in 1:length(X_vec)) {
      A4BX6[a, b, x] <-  str_c(A_vec[a], B[b], X_vec[x], sep = ',')
    }
  }
}
A4BX6 <- as.vector(A4BX6)
```

The first 20 combinations of A4BX6 are:

```
data.frame(A4BX6 = A4BX6[1:20])
```

```
##                         A4BX6
## 1  Li,Li,Li,Li,Ge,F,F,F,F,F,F
## 2  Li,Li,Li,Na,Ge,F,F,F,F,F,F
## 3   Li,Li,Li,K,Ge,F,F,F,F,F,F
## 4  Li,Li,Li,Rb,Ge,F,F,F,F,F,F
## 5  Li,Li,Li,Cs,Ge,F,F,F,F,F,F
## 6  Li,Li,Na,Na,Ge,F,F,F,F,F,F
## 7   Li,Li,Na,K,Ge,F,F,F,F,F,F
## 8  Li,Li,Na,Rb,Ge,F,F,F,F,F,F
## 9  Li,Li,Na,Cs,Ge,F,F,F,F,F,F
## 10   Li,Li,K,K,Ge,F,F,F,F,F,F
## 11  Li,Li,K,Rb,Ge,F,F,F,F,F,F
## 12  Li,Li,K,Cs,Ge,F,F,F,F,F,F
## 13 Li,Li,Rb,Rb,Ge,F,F,F,F,F,F
## 14 Li,Li,Rb,Cs,Ge,F,F,F,F,F,F
## 15 Li,Li,Cs,Cs,Ge,F,F,F,F,F,F
## 16 Li,Na,Na,Na,Ge,F,F,F,F,F,F
## 17  Li,Na,Na,K,Ge,F,F,F,F,F,F
## 18 Li,Na,Na,Rb,Ge,F,F,F,F,F,F
## 19 Li,Na,Na,Cs,Ge,F,F,F,F,F,F
## 20   Li,Na,K,K,Ge,F,F,F,F,F,F
```

The total number of combinations of A4BX6 is :17640.