Εργασία στο μάθημα Επιστήμη των Δεδομένων και Αναλυτική (Data Science and Analytics)

Τριανταφυλλίδου Μαρία

2023-06-26

Εισαγωγή

Η παρούσα εργασία αναλύει και παρουσιάζει δεδομένα χρησιμοποιώντας την γλώσσα R. Η γλώσσα προγραμματισμού R παρέχει ένα ισχυρό περιβάλλον για την ανάλυση και την οπτικοποίηση δεδομένων. Η εργασία ξεκινά με την ανάγνωση και την επεξεργασία ενός αρχείου CSV που περιέχει συλλογικά δεδομένα. Χρησιμοποιώντας τη γλώσσα R, εκτελούμε αναλύσεις και υπολογισμούς πάνω στα δεδομένα, προετοιμάζοντας τα για την παρουσίαση.

R Markdown

This is an R Markdown document. Markdown is a simple formatting syntax for authoring HTML, PDF, and MS Word documents. For more details on using R Markdown see http://rmarkdown.rstudio.com (http://rmarkdown.rstudio.com).

When you click the **Knit** button a document will be generated that includes both content as well as the output of any embedded R code chunks within the document. You can embed an R code chunk like this:

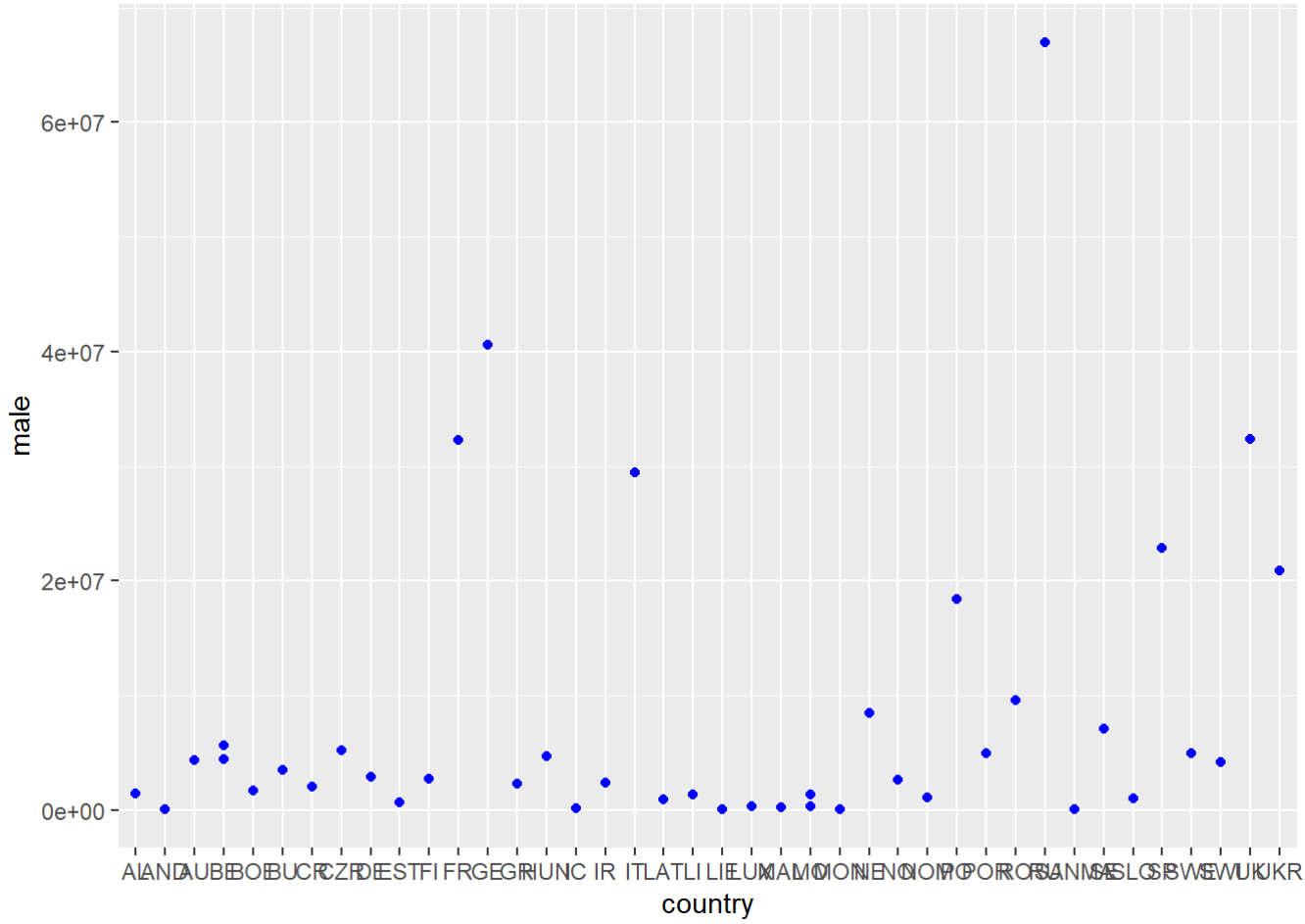
```
summary(countryTotal)
```

```
##    country               male          Population Female    physicians
##  Length:42          Min.   :   16453   Min.   :   17381   Min.   :    0
##  Class :character   1st Qu.: 1028710   1st Qu.: 1038178   1st Qu.:    0
##  Mode  :character   Median : 2776054   Median : 2835602   Median :30343
##                     Mean   : 8466682   Mean   : 8968383   Mean   :24472
##                     3rd Qu.: 6687986   3rd Qu.: 5674312   3rd Qu.:40019
##                     Max.   :66964302   Max.   :77378095   Max.   :62149
##    refugee
##  Length:42
##  Class :character
##  Mode  :character
##
##
##
```

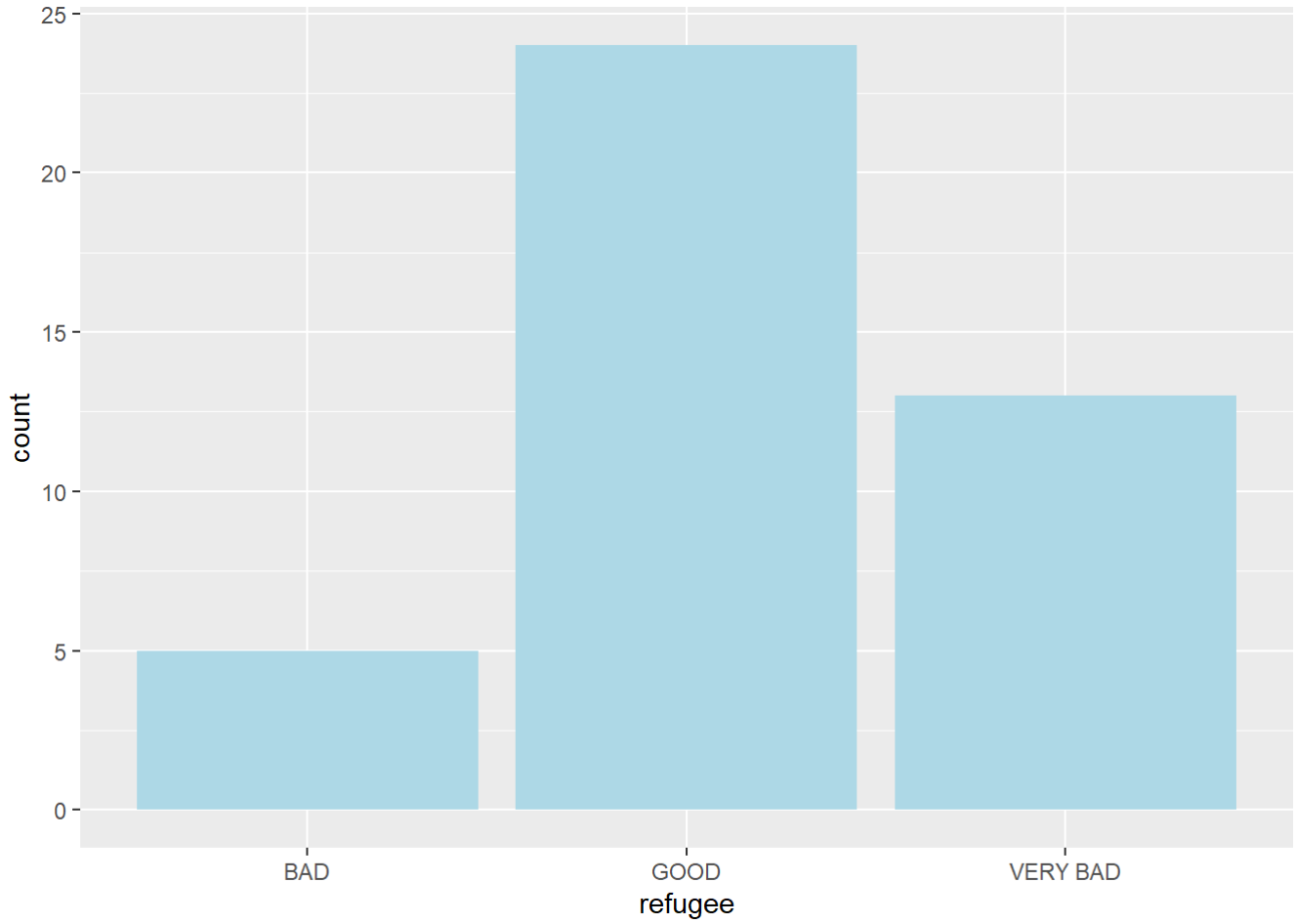
Σχεδιαγράμματα

Για να γίνει η παρουσίαση ακόμα πιο πειστική και εύκολη στην κατανόηση, συμπεριλαμβάνουμε γραφήματα και διαγράμματα που δημιουργούνται επίσης με τη χρήση της γλώσσας R. Τα γραφήματα αυτά βοηθούν στην οπτικοποίηση των δεδομένων και στην εύρυθμη παρουσίαση των αναλύσεων που πραγματοποιούνται.

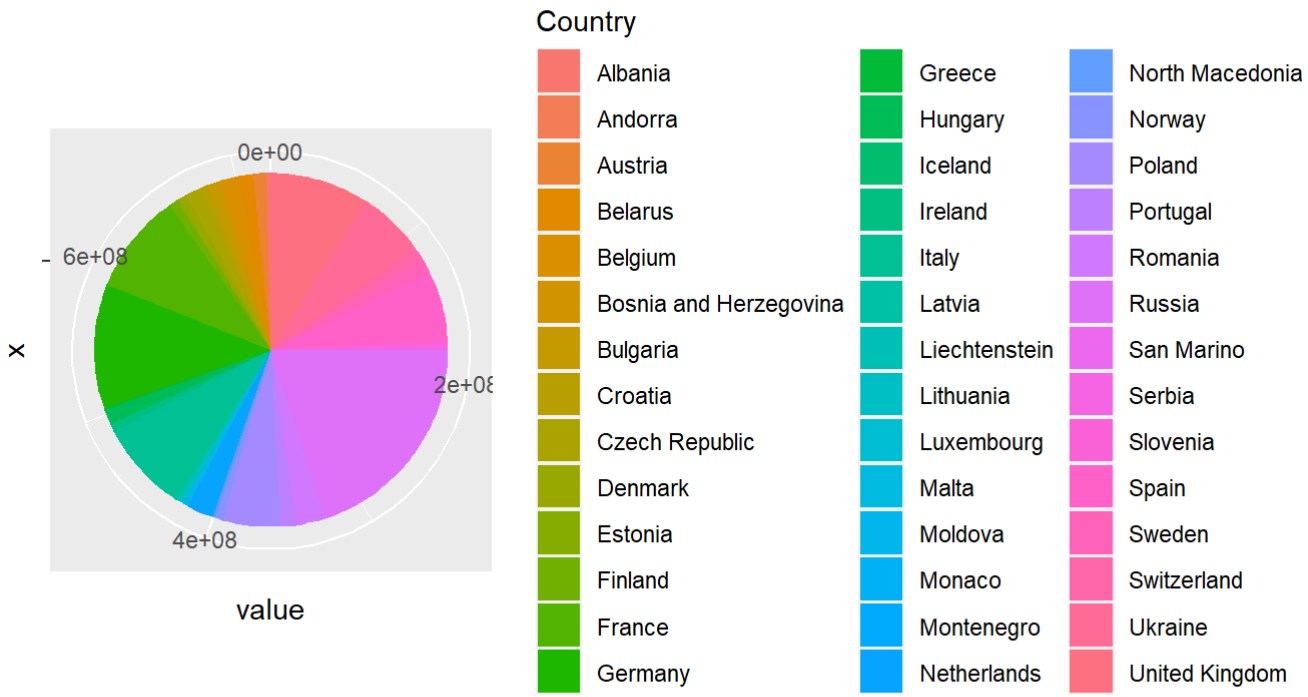
```
library(ggplot2)
plot <- ggplot(data = countryTotal) +
  geom_point(mapping = aes(x = country, y = male), color = "blue")
plot
```



```
library(ggplot2)
plot <- ggplot(data = countryTotal) +
  geom_bar(mapping = aes(x = refugee), fill = "lightblue")
plot
```




```
# Basic piechart
plot <- ggplot(data, aes(x="", y=value, fill=Country)) +
  geom_bar(stat="identity", width=1) +
  coord_polar("y", start=0)
plot
```



Inner Join με R

Τα Inner Join ανήκουν στις διαδικασίες συνένωσης (join) που μπορούν να γίνουν στη γλώσσα προγραμματισμού R, και χρησιμοποιούνται για να συνδυάσουν δεδομένα από διάφορες πηγές βάσης δεδομένων με βάση ένα κοινό πεδίο ή σύνολο πεδίων.

Συγκεκριμένα, το Inner Join επιστρέφει μόνο τις εγγραφές που έχουν κοινές τιμές στα πεδία που καθορίζονται για τη συνένωση. Αυτό σημαίνει ότι μόνο οι εγγραφές που έχουν αντίστοιχες τιμές στο κοινό πεδίο θα επιστραφούν στο αποτέλεσμα του Inner Join.

Ένα παράδειγμα inner join είναι το παρακάτω που εννώνει δυο πίνακες με την στήλη country

```
merged_data <- merge(total, countryTotal, by = "country")

#View of inner join only for two columns from six total columns
print(merged_data[c("country","PopulationTotal")])
```

```
##    country PopulationTotal
## 1       AL         2876101
## 2      AND           72540
## 3       AU         8736668
## 4       BE        11331422
## 5       BE        11331422
## 6       BE         9469379
## 7       BE         9469379
## 8      BOE         3480986
## 9       BU         7127822
## 10      CR         4174349
## 11     CZR        10566332
## 12      DE         5728010
## 13     EST         1315790
## 14      FI         5495303
## 15      FR        66724104
## 16      GE        82348669
## 17      GR         1236443
## 18     HUN         9814023
## 19      IC          335439
## 20      IR         4755335
## 21      IT        60627498
## 22     LAT         1959537
## 23      LI         2868231
## 24     LIE           37609
## 25     LUX          582014
## 26     MAL          455356
## 27      MO         2803186
## 28      MO         2803186
## 29      MO          622303
## 30      MO          622303
## 31     MON           37071
## 32      NE        17030314
## 33      NO         5234519
## 34     NOM         2072490
## 35      PO        37970087
## 36     POR        10325452
## 37      RO        19702267
## 38      RU       144342397
## 39   SANMA           33834
## 40      SE         3672802
## 41     SLO         2065042
## 42      SP        46484062
## 43     SWE         9923085
## 44     SWI         8373338
## 45      UK        65611593
## 46     UKR        45004673
```

Δομές επανάληψης

Οι for loops είναι ένας από τους βασικούς τρόπους επανάληψης κώδικα στη γλώσσα προγραμματισμού R. Οι for loops σας επιτρέπουν να εκτελέσετε ένα συγκεκριμένο τμήμα κώδικα επανειλημμένα για μια ορισμένη σειρά τιμών ή αντικειμένων.

Η σύνταξη μιας for loop στην R είναι η εξής:

```
for (i in 1:5) {
  # Κώδικας που θέλουμε να εκτελεστεί
}
```

Ας δούμε ένα απλό παράδειγμα. Ας υποθέσουμε πως θέλουμε να εκτυπώσουμε τυχαία δέκα αριθμούς απο το 1 εως το 100. Μπορούμε να χρησιμοποιήσουμε μια for loop για αυτό το σκοπό:

```
output <- for (i in 1:10) {
  random_number <- sample(1:100, 1)
  result <- paste0(i,"-> ", random_number)
  print(result)
}
```

```
## [1] "1-> 23"
## [1] "2-> 3"
## [1] "3-> 8"
## [1] "4-> 22"
## [1] "5-> 22"
## [1] "6-> 42"
## [1] "7-> 50"
## [1] "8-> 8"
## [1] "9-> 47"
## [1] "10-> 84"
```

Σε αυτό το παράδειγμα, η μεταβλητή “i” λαμβάνει τις τιμές από το 1 έως το 10, και κάθε φορά εκτελείται ο κώδικας μέσα στη for loop, ο οποίος εκτυπώνει την τρέχουσα τιμή της μεταβλητής “i”.

Function με R

Οι συναρτήσεις (functions) αποτελούν ένα βασικό στοιχείο της γλώσσας προγραμματισμού R. Μια συνάρτηση είναι ένα μπλοκ κώδικα που εκτελεί μια συγκεκριμένη λειτουργία και μπορεί να κληθεί (να εκτελεστεί) από άλλο τμήμα του κώδικα.

```
my_function <- function(arg1, arg2, ...) {
  # Κώδικας που εκτελεί τη λειτουργία της συνάρτησης
  # Επιστρέφει το αποτέλεσμα (αν χρειάζεται)
}
```

Σε αυτήν τη σύνταξη, το my_function είναι το όνομα που δίνουμε στη συνάρτηση, arg1, arg2, κ.λπ. είναι οι παράμετροι που μπορεί να δεχθεί η συνάρτηση, και ο κώδικας που ακολουθεί εκτελείται όταν κληθεί η συνάρτηση. Αν χρειάζεται, μπορούμε να επιστρέψουμε ένα αποτέλεσμα με την εντολή return().

Ας δούμε ένα παράδειγμα για να κατανοήσουμε καλύτερα. Ας υποθέσουμε ότι θέλουμε να ορίσουμε μια συνάρτηση που υπολογίζει τον παραγοντικό ενός αριθμού¨

```r
#Function that calculate factorial
factorial <- function(n) {
  result <- 1
  for (i in 1:n) {
    result <- result * i
  }
  return(result)
}

# Usage example
number <- 4
factorial_result <- factorial(number)
sprintf("Ο παραγοντικός του αριθμού: %s",  number)
```

```
## [1] "Ο παραγοντικός του αριθμού: 4"
```

```r
print(factorial_result)
```

```
## [1] 24
```