HY390.51 Project Report

Michalis Karagiannakis - csd4355

Michalis Saridakis - csd4528

Περίληψη του αποτελέσματος(ερώτημα 9ο): Μετά από προσεκτική ανάγνωση και επεξεργασία των δοσμένων δεδομένων, με την πλήρη εκτέλεση του προγράμματος, καταλήγουμε (βλέπε διαγράμματα ερώτημα 8ο) ότι υπάρχει μειωμένη εμφάνιση κρουσμάτων Sars-CoV2 κατά τις πρώτες 50 μέρες του χρόνου καθώς και από την ημέρα 150 και έπειτα μέχρι το τέλος του χρόνου. Στα διαστήματα αυτά παρατηρείται ότι η χαμηλότερη εμφάνιση είναι από την ημέρα 190 και μέχρι τέλος του χρόνου. Αντίθετα, η περίοδος με τα περισσότερα κρούσματα είναι τις ημέρες 70-150. Με τα ύψιστα κρούσματα να εμφανίζονται την περίοδο ημερών 100-110. Με βάση αυτά τα αποτελέσματα παρατηρούμε ότι στις αρχές και στα τέλη του χρόνου όπου ο καιρός είναι συνήθως ψυχρός και βροχερός εμφανίζονται λιγότερα κρούσματα, πιθανόν λόγω χαμηλής κινητικότητας, ενώ την περίοδο που ο καιρός ζεσταίνει δηλαδή αρχές του μηνός Μαρτίου (περίπου) ανεβαίνει απότομα και ο αριθμός των κρουσμάτων.

Ιδέα του προγράμματος: Το πρόγραμμα μας ξεκινάει διαβάζοντας τα raw δεδομένα από το διαδίκτυο, τα αποθηκεύει σε κατάλληλες δομές και μετά τα χωρίζει με τρόπο τέτοιο ώστε να διευκολύνεται η σύγκριση και ο υπολογισμός. Έπειτα θα δούμε τις συναρτήσεις που είναι υπεύθυνες για τον υπολογισμό των k, w και D αντίστοιχα, ακολουθούμενες από τις κλήσεις τους. Στην συνέχεια έχουμε οπτικοποίηση των raw δεδομένων των συναρτήσεων και έπειτα την κανονικοποίηση αυτών. Προς το τέλος, έχουμε την συνάρτηση, και την κλήση της, για τον υπολογισμό των Ευκλείδειων αποστάσεων. Τέλος, έχουμε την δημιουργία και εκτύπωση των απαραίτητων Διαγραμμάτων και Ιστογραμμάτων για την οπτικοποίηση και ανάλυση των τελικών αποτελεσμάτων.

Εκτέλεση του Προγράμματος: Η συνολική εκτέλεση του προγράμματος χρειάζεται περίπου 2 λεπτά χρόνο, και επιτυγχάνεται τρέχοντας το πρόγραμμα μαζεμένο καθώς είναι χωρισμένο με κατάλληλο τρόπο για σειριακή εκτέλεση.

Ερώτημα 1ο:
Με σκοπό τον υπολογισμό των k, w και D αντίστοιχα, και με βάση τους δοσμένους από την εκφώνηση τύπους γράψαμε, δοκιμάσαμε και εκτελέσαμε τις παρακάτω συναρτήσεις:

```
# Function Υπολογισμου k
calculate_k <- function(input_vector){
  diff_count = vector("numeric")

  for(i in 1:50) {
    for(j in i:50) {
      if(i == j) {
        next;
```

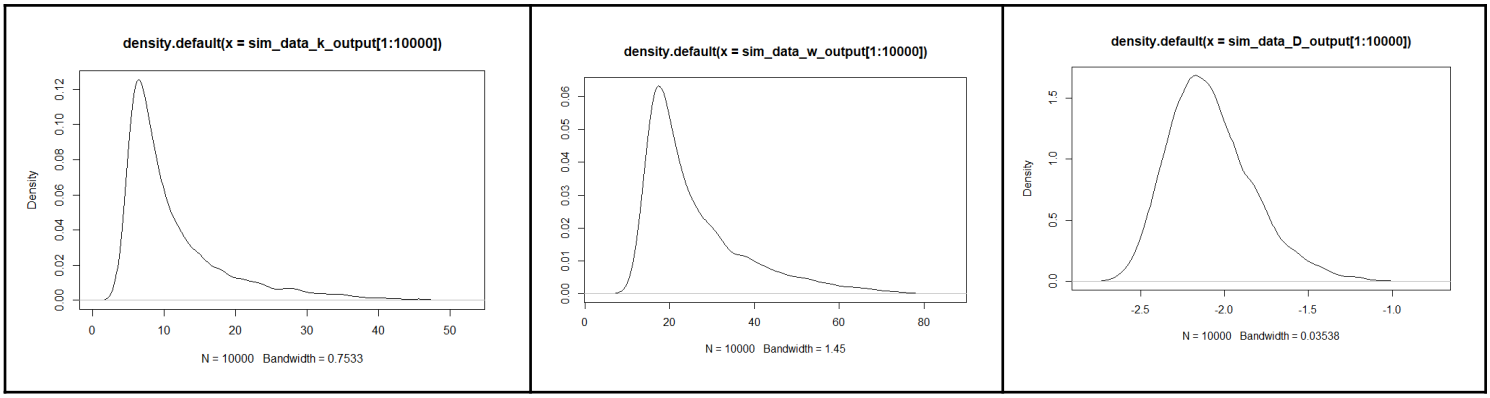
```
      }
      num_differences <- sum(strsplit(input_vector[[i]], "")[[1]] !=
strsplit(input_vector[[j]], "")[[1]])

      # Output the number of different digits
      diff_count <- append(diff_count,num_differences)
    }
  }
  #Υπολογισμος k
  k = sum(diff_count/length(diff_count))
  k

}

#function υπολογισμου w
calculate_w <- function(s){
  a1 = sum(1/(1:49))

  w = s/a1
  w
}

#Function υπολογισμου D
calculate_D <- function(k,w,s){
  n=50
  a1 = sum(1/(1:49))
  a2 = sum(1/((1:49)**2))
  a2

  b2 = (2*(n**2+n+3))/(9*n*(n-1))
  b2

  b1 = (n+1)/(3*(n-1))
  b1

  c2 = b2 - (n+2)/(a1*n) + a2/(a1**2)
  c2

  c1 = b1 - 1/a1
  c1

  e2 = c2/((a1**2)+a2)
  e2

  e1 = c1/a1
  e1

  D = (k-w) / sqrt((e1*s) + (e2*s*(s-1)))
  D
}
```

Ερώτημα 2ο:
Για τον υπολογισμό των ζητούμενων δεδομένων έχουμε εκτελέσει τις παρακάτω κλήσεις συναρτήσεων μας.

```
#Κληση Υπολογισμου κ για τα observed data

k <- calculate_k(observed_data)
w <- calculate_w(length(observed_data_splitted[[1]]))
D <- calculate_D(k,w,length(observed_data_splitted[[1]]))

#Κληση Υπολογισμου κ για τα simulated data

begin=1
final=50
sim_data_k_output = vector("numeric")
sim_data_w_output = vector("numeric")
sim_data_D_output = vector("numeric")

for (i in 1:10000) {

  sim_data_k_output <-
append(sim_data_k_output,calculate_k(simulated_data[begin:final]))
  sim_data_w_output <-
append(sim_data_w_output,calculate_w(length(simulated_data_splitted[[begin]])))
  sim_data_D_output <-
append(sim_data_D_output,calculate_D(sim_data_k_output[i],sim_data_w_output[i],l
ength(simulated_data_splitted[[begin]])))

  begin=final+1
  final= final + 50
}
```

Στην συνέχεια, με τα παραγόμενα από τις συναρτήσεις δεδομένα, και τον παρακάτω κώδικα, παράγουμε τα παρακάτω διαγράμματα:

```
#Δημιουργια plot απο τα output των συναρτησεων για τα simulated data

plot(density(sim_data_k_output[1:10000]))
plot(density(sim_data_w_output[1:10000]))
plot(density(sim_data_D_output[1:10000]))
```



Ερώτημα 3ο:

Για την κανονικοποίηση και τον υπολογισμό μέσων όρων και διασπορών , γράψαμε και εκτελέσαμε τον παρακάτω κώδικα:

```
#Κανονικοποίηση

#Για το w
mean_w = mean(sim_data_w_output)
var_w = var(sim_data_w_output)
normalized_w = (sim_data_w_output - mean_w)/var_w

#Για το w0
normalized_w0 = (w - mean_w)/var_w

#Για το k
mean_k = mean(sim_data_k_output)
var_k = var(sim_data_k_output)
normalized_k = (sim_data_k_output - mean_k)/var_k

#Για το κ0
normalized_k0 = (k - mean_k)/var_k

#Για το D
mean_D = mean(sim_data_D_output)
var_D = var(sim_data_D_output)
normalized_D = (sim_data_D_output - mean_D)/var_D

#Για το D0
normalized_D0 = (D - mean_D)/var_D
```

Ερώτημα 4ο:
Για τον υπολογισμό των ευκλείδειων αποστάσεων μεταξύ των observed και των simulated datasets, και με βάση τον δοσμένο από την εκφώνηση τύπο, καταλήξαμε στην εξής συνάρτηση:

```
#Υπολογισμός ευκλείδιων αποστάσεων

calc_euclidian_d <- function(){
  d = sqrt((normalized_D0 - normalized_D)**2 + (normalized_w0 - normalized_w)**2
+ (normalized_k0 - normalized_k)**2)
}
#Κληση της συναρτηση και αποθηκευση τον αποτελεσματων σε array
euclidian_d = calc_euclidian_d()

temp_array <-matrix(nrow=10000,ncol = 2)
temp_array[1]<-euclidian_d
```

Ερώτημα 5ο:
Βρήκαμε τις 500 μικρότερες αποστάσεις και κρατήσαμε τους δείκτες τους με τον εξής τρόπο:

```
#Εύρεση των 500 μικρότερων αποστάσεων και κρατάμε τα indexes τους σε ένα νέο
vector
smallest_distances_indexes = order(euclidian_d,decreasing = FALSE)[1:500]

smallest_distances_indexes
```

Ερώτημα 6ο:
Εξάγαμε τα δεδομένα από τα data_parameters με τις παρακάτω εντολές:

```
# Ευρεση τον αντιστοιχων τιμων με βαση τα indexes απο το data_parameters
data_corrensponding_values =
as.double(data_parameters[smallest_distances_indexes[1:500]])

data_corrensponding_values
```

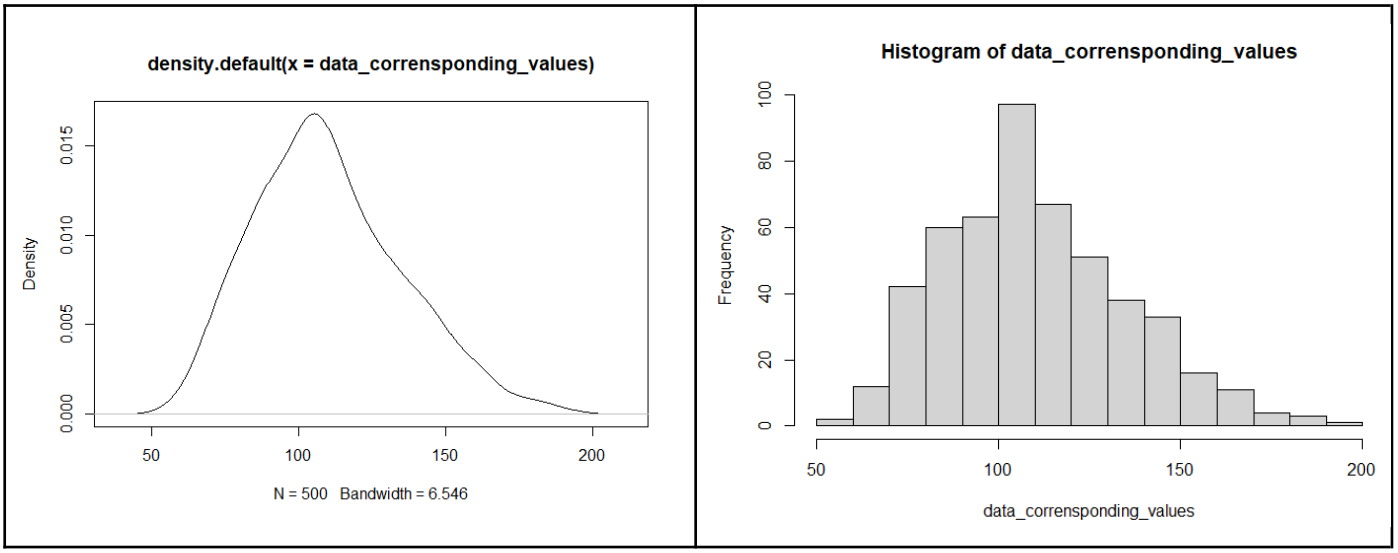
Ερώτημα 7ο:
Υπολογίσαμε τον μέσο όρο των 500 τιμών από το 6ο ερώτημα ως εξής:

```
#Ευρεση του μεσου ορου των data_corrensponding_values
mean_cor_values = mean(data_corrensponding_values)
mean_cor_values
```

Ερώτημα 8ο:

Παράξαμε τα παρακάτω Διαγράμματα και Ιστογράμματα για τις 500 τιμές που υπολογίσαμε στο 7ο ερώτημα, με τον κώδικα:

```
#Κατασκευη ιστογραμματος για τα data_corrensponding_values
hist(data_corrensponding_values)
#κατασκευη 2ου ιστογραμματος για ευκολοτερη διακριση των διαστημματων με σκοπο την αναφορα
hist(data_corrensponding_values,breaks = 4,xlim = c(1,365))

#Κατασκευη διαγραμματος πυκνοτητας για τα data_corrensponding_values
plot(density(data_corrensponding_values))
```



Καθώς και αυτό το πιο συμπιεσμένο Ιστόγραμμα για να μας βοηθήσει στην εξαγωγή συμπερασμάτων.