UNIVERSITATEA DIN BUCUREȘTI
FACULTATEA DE MATEMATICĂ ȘI INFORMATICĂ

Simulare vaccinare Covid-19

Sistem de tip coadă cu două servere legate în serie

DOCUMENTAȚIE

Cozma Laura-Elena
Manolache Andrei
Matei Daniel

2021

1. Descrierea problemei

Proiectul implementează un sistem de tip coadă cu două servere legate în serie, ce are ca scop simularea venirii persoanelor la un centru de vaccinare Covid-19, fără a necesita o programare în prealabil.

Guvernul a observat ca prea mulți oameni iși fac programare la vaccin, dar până la urmă nu se mai prezintă. Astfel se gândește dacă nu ar fi mai util ca vaccinarea pe baza de programare să fie înlocuită cu vaccinarea la liber. Așadar, vom simula o astfel de situație pentru a observa dacă noua variantă este mai eficientă decât cea abordată în prezent. Întrucât persoanele care își administrează a doua doză trebuie să se prezinte într-o zi stabilită în funcție de administrarea primei doze, considerăm că centrul de vaccinare este destinat persoanelor care vor să își administreze prima doză a vaccinului, urmând ca a doua doză să fie administrată la un alt centru a cărui activitate nu va fi analizată în proiect.

„Clienții" vor fi reprezentați de oamenii ce vor veni la vaccinare pentru a-și administra prima doză, iar „serverele" vor fi reprezentate de un doctor și o asistentă medicală: doctorul se va ocupa de completarea fișelor necesare, iar asistenta de administrarea vaccinului. Când o persoană va sosi, dacă doctorul este liber, se va începe completarea fișei, în caz contrar, aceasta va intra în coada de așteptare. De asemenea, după completarea fișei, persoana va repeta procesul – se va vaccina dacă nu stă nimeni la coadă, altfel va intra în coadă. Atât completarea fișei cât și timpul pentru administrarea vaccinului diferă de la persoană la persoană.

Programul de funcționare va fi 8:00 – 20:00. Presupunem că oamenii nu vor mai sta la coadă dacă aceasta depășește 20 de persoane. De asemenea, o persoană va avea asociată o limită a răbdării, astfel că în cazul în care va aștepta la coadă mai mult de 60 de minute va pleca. Ne așteptăm ca dimineața între 9:00 și 11:00 și seara între 16:00 și 18:00 când persoanele vor ieși de la serviciu să fie perioadele de vârf. Considerăm că de-a lungul programului, atât doctorul cât și asistenta medicală nu vor lua pauze.

De asemenea, am limitat numărul de doze utilizate într-o zi la 100 și vom contoriza numărul de clienți pierduți din cauza lipsei vaccinului în ziua respectivă. În plus, dozele nefolosite într-o zi vor fi considerate pierdute, astfel că vom calcula o pierdere medie cauzată de un număr prea mic de persoane care s-au prezentat la vaccinare într-o zi.

2. Datele problemei

Sunt reprezentate de:

- funcția de intensitate $\lambda(t)$corespunzătoare unui proces Poisson neomogen care **simulează sosirea pacienților**

$$\lambda(t) = \begin{cases} -t^2 - 2t + 25, & t \in [0,3) \\ 14, & t \in [3,5] \\ -0{,}05t^2 - 0{,}2t + 12, & t \in (5,9) \\ \quad 11, & t \geq 9 \end{cases}$$

- variabila aleatoare PoissonY_1 de parametru $\lambda = 6$ ce descrie **timpul de servire** de la **primul server**

$$Y_1 \sim Pois(6)$$

$$f_2(x) = \begin{cases} \frac{2}{61} x\, e^{\frac{-x^2}{61}}, & x > 0 \\ 0, & x \leq 0 \end{cases}$$

- densitatea de probabilitate $f_2(x)$ce descrie **timpul de servire** de la **al doilea server**.
- **coadă de așteptare** cu valoarea implicită de **20 de persoane**. Dacă o persoană ajunge la server și în fața lui se află mai mult de 20 de persoane, atunci persoana respectivă pleacă
- **timpul maxim de așteptare** a unei persoane este de **60 de minute**. Dacă o persoană stă la coadă de mai mult de 60 de minute, atunci persoana respectivă va pleca

3. Rezolvarea problemelor teoretice

Pentru a determina valori pentru variabila aleatoare $\mathbf{Y_1}$ folosim metoda inversă. Mai exact

1. Generăm U variabilă aleatoare uniformă pe intervalul [0,1]

$$Y_1 = \begin{cases} y_1, & U < p_1 \\ y_2, & p1 \leq U < p_1 + p_2 \\ y_3, & p1 + p2 \leq U < p_1 + p_2 + p_3 \\ \dots & \\ y_j, & \sum_{i=1}^{j-1} p_i \leq U < \sum_{i=1}^{j} p_i \end{cases}$$

cu

$$y_i = \frac{e^{-\lambda} \cdot \lambda^i}{i!}$$

În partea de **implementare** este explicată metoda de obținere a valorii.

Pentru a determina valori pentru variabila aleatoare $\mathbf{Y_2}$ am folosit metoda respingerii. Mai exact, avem densitatea de probabilitate

$$f_2(x) = \begin{cases} \frac{2}{61} x\, e^{\frac{-x^2}{61}}, & x > 0 \\ 0, & x \leq 0 \end{cases}$$

ce ia valori pe intervalul (0, inf). Pentru a putea folosi metoda respingerii, am considerat G ~ Exp(2/61) variabilă aleatoare cu densitatea de probabilitate $\frac{2}{61} e^{-\frac{2}{61}x}$ pentru x > 0 (cele două funcții au acelasi suport).

Construim funcția

$$h(x) = \frac{f_2(x)}{g(x)}, x > 0$$

pentru care aflăm o constanta c > 0 astfel încât

$$h(x) \leq c \; \forall \, x > 0$$

$$x \cdot e^{\frac{2}{61}x - \frac{x^2}{61}} > 0$$

Derivăm h(x) și obținem

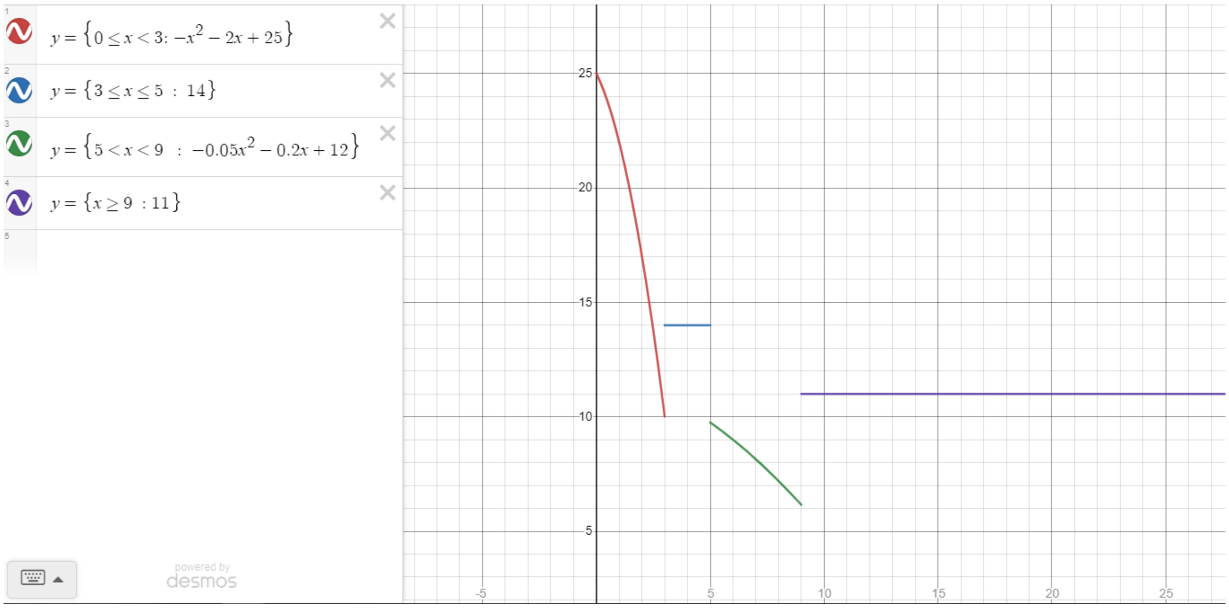
$$h'(x) = \frac{1}{61} \cdot e^{\frac{2}{61}x - \frac{x^2}{61}} \cdot (-2x^2 - 2x + 61)$$

Calculăm punctele de extrem ale funcției de gradul II, obținem soluția pozitivă x = $\frac{\sqrt{123}}{2}$ și reprezintă punctul de maxim al derivatei. Calcumăm

$$h\left(\frac{\sqrt{123}}{2}\right) = 4.04863$$

Deci putem considera **c=5.**

Pentru **funcția de intensitate** a procesului Poisson cu care ajung clienții, este necesară găsirea unei constante astfel încât $\lambda(t) \leq \lambda \;\; \forall t > 0$. Pentru acest lucru, ne folosim de **Desmos** unde desenăm funcția și obținem valoarea **c = 26**.



4. Implementare

S-a considerat că sosirea clienților s-a făcut după un proces Poisson neomogen cu funcția de intensitate λ(t) care este definită astfel:

```
LAMBDA <- function(t) {
  if (0 <= t & t < 3 * 60) {
    c <- -t ** 2 - 2 * t + 25
  } else if (3 * 60 <= t & t <= 5 * 60) {
    c <- 14
  } else if (5 * 60 < t & t < 9 * 60) {
    c <- -0.05 * (t ** 2) - 0.2 * t + 12
  } else {
    c <- 11
  }
  return (c)
}
```

Pentru a afla prima sosire a unei persoane după momentul de timp s corespunzător unui proces Poisson neomogen cu funcția de intensitate λ și maximul mLambda, am considerat momentul de timp **s** și am generat două variabile uniforme. Dacă valoarea aleatoare uniformă este mai mică decât raportul dintre funcția de intensitate în momentul respectiv și valoarea maximă a funcției intensitate, ne oprim.

```r
GET_TS <- function(s, lambda, mLambda) {
  ts <- t <- s

  while (TRUE) {
    # Simuleaza 2 valori uniforme in intervalul [0,1]
    U1 <- runif(1)
    U2 <- runif(1)

    # Noul moment de timp t
    t <- t - 1 / mLambda * log(U1)

    # Daca valoarea aleatoare uniforma este mai mica decat raportul dintre functia de intensitate
    # in momentul respectiv si valoarea maxima a functiei intensitate, atunci ne oprim
    if (U2 <= lambda(t) / mLambda) {
      ts <- t
      break
    }
  }
  ts
}
```

Am folosit o funcție care simulează o variabilă aleatoare exponențială de parametru lambda. Generează o variabilă aleatoare uniformă folosind metoda inversă:

$$X = F^{-1}(U) \Leftrightarrow u = F(x)$$

Știm că

$$F(x) = \begin{cases} 0, & \text{dacă } x < 0 \\ 1 - e^{-\lambda x}, & \text{dacă } x \geq 0. \end{cases}$$

Atunci

$$u = 1 - e^{-\lambda x} \Leftrightarrow -\lambda x = \ln(1-u)$$

$$\Leftrightarrow X = -\frac{1}{\lambda} \cdot \ln(1-u) \Leftrightarrow = -\frac{1}{\lambda} \cdot \ln(u)$$

```r
SIMULATE_EXP <- function(lambda) {
  # Genereaza o variabila aleatoare uniforma pe intervalul [0,1]
  u <- runif(1)
  # Construieste variabila exponentiala de parametru lambda
  x <- -1 / lambda * log(u)
  return (x)
}
```

Pentru a simula variabila aleatoare Y1, am folosit metoda inversă după cum am amintit mai sus.

Știm că $p_i = \frac{e^{-\lambda} \cdot \lambda^i}{i!}$ deci $p_{i+1} = \frac{e^{-\lambda} \cdot \lambda^{i+1}}{(i+1)!} = \frac{e^{-\lambda} \cdot \lambda^i \cdot \lambda}{i \cdot (i+1)} = p_i \cdot \frac{\lambda}{i+1}$

Astfel, ne folosim de relația de recurență pentru a obține o valoare dată de repartiție utilizând metoda inversă. Generăm o variabilă aleatoare uniformă pe [0, 1], calculăm probabilitatea în punctul i = 0 și inițializăm valoarea funcției de repartiție. Cât timp nu obțin o valoare suficient de bună, actualizez probabilitatea și repartiția în punctul următor și avansez.

```
GET_Y1 <- function(lambda = 6) {
  #  Se foloseste de relatia de recurenta p_j+1 = p_j * lambda / (j + 1)
  # Contor
  i <- 0
  # Probabilitatea curenta
  p <-  exp(1) ^ (-lambda)
  # Valoarea functiei de repartitie in punctul curent
  F <- p
  u <- runif(1, 0, 1)

  while (u >= F) {
    # Updatam noile valori ale probabilitatilor si functiei de repartitie
    p <- p * lambda / (i + 1)
    F <- F + p
    i <- i + 1
  }
  return (i)
}
```

Pentru a simula variabila aleatoare Y2, am folosit metoda respingerii după cum am amintit mai sus.

```
GET_Y2 <- function(x) {
  # Am ales g(y) = e^x, x > 0
  # Am calculat maximul functiei h(x) = f(x)/g(x),iar acesta este c = 5
  c <- 5
  while (T) {
    # Simuleaza o variabila aleatoare exponentiala de parametru lambda = 1
    y <- SIMULATE_EXP(1)
    # Simuleaza o variabila aleatoare uniforma cu valori in [0,1]
    u <- runif(1)
    if (u <= 2 / (61 * c) * y * exp(y - y ^ 2 / 61)) {
      return(y)
    }
  }
}
```

Funcția principală care simulează un astfel de sistem este **RUN_SIMULATION.**

Definim variabilele contor

```
# Momentul de timp in care un client ajunge in sistem
A1 <- c()
# Momentul de timp in care un client ajunge la serverul 2
A2 <- c()
# momentul de timp in care un client paraseste sistemul
D <- c()

# timpul maxim de asteptare dupa care un pacient pleaca
timp_asteptare_max <- 60

# valoarea maxima a functiei de intensitate
max_value <- 26
```

După care repetăm procesul cât timp centrul nu este închis și numărul de doze nu este consumat.

```
while (t < t_max && nr_doze > 0) {
```

Se disting 3 cazuri:

1. Sosește un client nou

   ```
   #Soseste un client nou
   if (ta == min(ta, t1, t2)) {
   ```

   În acest caz, adăugăm noul client în vectorii de output, incrementăm numărul de clienți din coadă și dacă serverul 1 este gol, atunci clientul nu așteaptă, ci intră la server. Verificăm și dacă este prea mare coada de așteptare. În cazul ăsta, clientul pleacă imediat

2. Serverul 1 se eliberează înainte de sosirea unui client nou

   ```
   else if (t1 < ta &&
            t1 <= t2) {
   ```

   În acest caz, verificăm dacă primul client din coadă nu așteaptă de prea mult timp. Dacă da, este scos din coadă.
   Altfel este pus în coada de așteptare de la al doilea server (verificându-se și dacă nu este prea mare coada de așteptare)

3. Serverul 2 se eliberează înainte de a sosi un client nou și înainte de finalizarea activității din serverul 1.

   ```
   else if (t2 < ta && t2 < t1)  {
   ```

   Clientul părăsește sistemul, marcăm acest lucru, vine următorul client la serverul 2

Rezultatele obținute în urma simulării

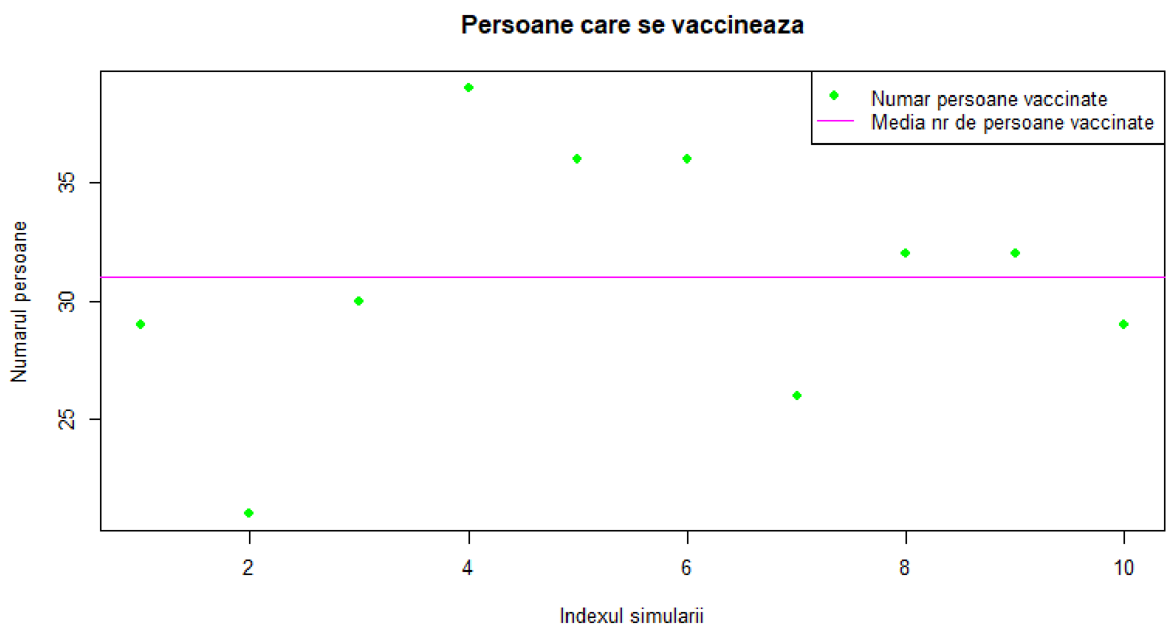
- Timpul minim, maxim și mediu, petrecut de un o persoană la vaccinare

S-a observat că o persoană care vine la centrul de vaccinare va sta în medie 25 de minute la coadă pentru a completa fișa de triaj, respectiv 5 minute pentru a se vaccina. Timpul maxim de așteptare pentru completarea fișei nu va depăși 60 de minute, întrucât persoana va atinge limita răbdării și va pleca, iar valoarea maximă a acesteia va fi aproximativ de 55 de minute, în timp ce timpul minim va fi de 5 minute. În cazul vaccinării propriu-zise, o persoană așteaptă în medie 5 minute, cu o limită maximă de 12 minute și una minimă de 2 minute.

	Timp minim	**Timp maxim**	**Timp mediu**
Completare fișă	5 min	55 min	25 min
Vaccinare	2 min	12 min	5 min

- Numărul mediu de persoane vaccinate într-un interval de timp

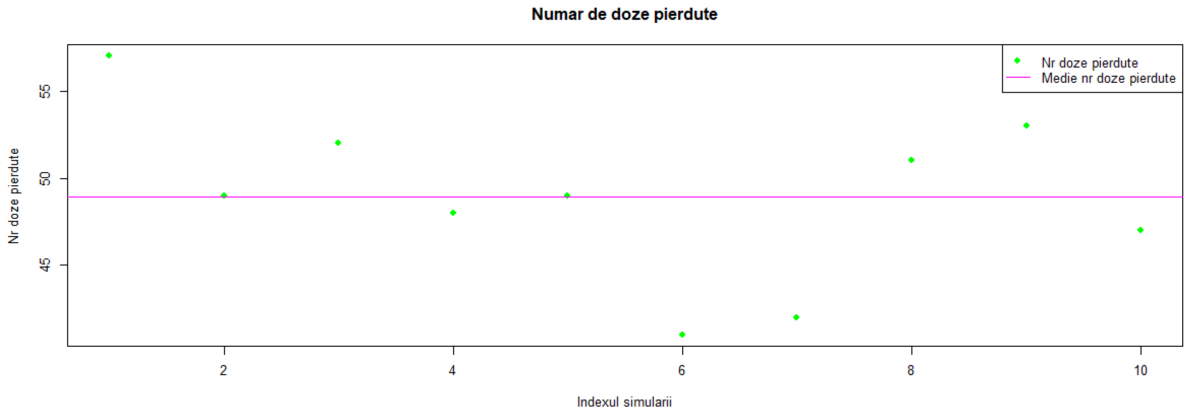
De-a lungul simulărilor, s-a observat o frecvență mai ridicată a persoanelor între orele 8:00 - 12:00. Astfel că multe persoane nu vor reuși să se vaccineze din cauza limitării cozii la maxim 20 de persoane. Un număr mediu de persoane vaccinate este de 54 de persoane, studiul făcându-se pe 10 simulări. În următorul grafic poate fi vizualizat numărul de persoane vaccinate corespunzătoare fiecarei simulări, împreună cu media persoanelor vaccinate în intervalul orar 11:00-18:00.



- Utilizarea dozelor de vaccin

În cadrul simulării, am considerat că centrul va fi dotat cu 100 de doze de vaccin. Dat fiind numărul scăzut de persoane care se vaccinează în decursul unei zile, nu rămâne nicio persoană nevaccinată din cauza numărului insuficient al dozelor.

De asemenea, am contorizat pierderea medie a dozelor cauzată de un număr prea mic de persoane care s-au prezentat la vaccinare într-o zi. Următorul grafic va ilustra numărul de doze pierdute în fiecare simulare, împreună cu numărul mediu de doze pierdute.



- Momentul de timp la care o persoană părăsește centrul de vaccinare

Am presupus că o persoană va părăsi centrul de vaccinare în două cazuri: coada deja existentă atinge limita maximă de 20 de persoane sau aceasta așteaptă de mai mult de o ora. Astfel, momentul de timp la care prima persoană care va pleca din unul dintre cele două motive menționate anterior va fi în medie de 250 minute după deschiderea centrului de vaccinare la ora 8:00. În medie 0-1 persoane vor părăsi pe zi centrul de vaccinare.

- Determinarea numărului de doze suplimentare administrate în cazul schimbării ipotezei

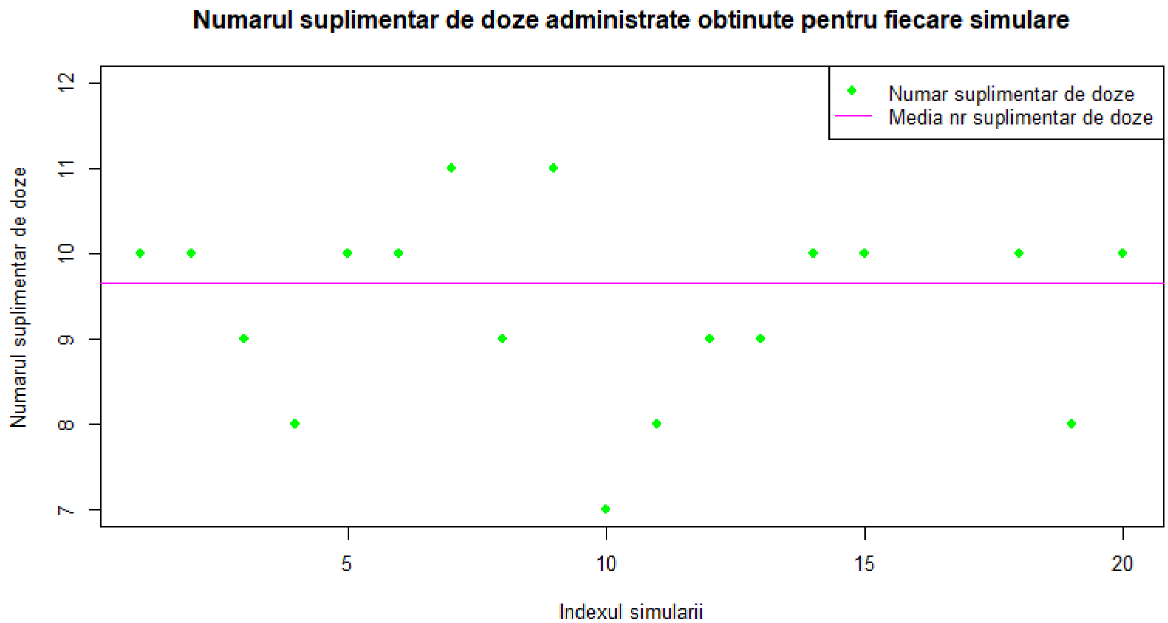
Au fost considerate trei cazuri care ar modifica câștigul mediu suplimentar:

1. Începerea programului cu o oră mai devreme

   Pentru începerea programului cu o oră mai devreme s-a înregistrat un câștig mediu de o doză administrată, frecvența persoanelor care vin atunci

2. Prelungirea cu o oră a programului curent

   În cazul prelungirii programului curent cu o ora câștigul este cel mai mare, de aproximativ 10 doze noi administrate.



3. Mărirea dimensiunii maxime a cozii de așteptare

    În cazul în care dimensiunea maximă a cozii de așteptare se dublează, câștigul maxim va fi de 7 doze administrate, cel mediu fiind de 5.

6. Concluzii

Centrele de vaccinare fără programare reprezintă o modalitate eficientă de administrare a vaccinului împotriva Covid-19 în cazul în care persoanele nu s-ar prezenta într-un număr foarte mare între anumite ore. În acest caz nefavorabil, mulți oameni ar pleca din cauza cozilor prea lungi sau a lipsei răbdării, conducând la un număr mic de persoane vaccinate și o pierdere mare a dozelor ce nu au fost administrate.