Documentatie proiect PS

Grupa: 243

Membri:

* Gheorghe Bogdan-Alexandru
* Andrei Cristian-David
* Sincari Sebastian-George
* Cosor Mihail

Se consideră o activitate care presupune parcurgerea secvențială a n etape. Timpul necesar finalizării etapei i de către o persoană A este o variabilă aleatoare ( ) T Exp i i : λ . După finalizarea etapei i , A va trece ȋn etapa i +1 cu probabilitatea αi sau va opri lucrul cu probabilitatea 1−αi . Fie T timpul total petrecut de persoana A ȋn realizarea activității respective.

set.seed(123)

n <- 100 #Nr etape

lambda <- runif(n, 0.5, 2) #Rata exp pt fiecare etapa

alpha <- runif(n-1, 0.8, 1) #Prob trecerii la urmatoarea etapa

alpha<-c(1,alpha)#prob ca o persoana sa participe la prima etapa este 100%

nrSimulari <- 1000000 #Nr simulari

simulator <- function() {

Ti <- rexp(n, rate = lambda) # Timpul pentru fiecare etapă

stop\_point <- which(runif(n - 1) > alpha[-1])[1] #Determinam momentul opririi

#Timpul total va fi cel pana la oprire/finalul vectorului daca nu se opreste

if (!is.na(stop\_point)) {

return(sum(Ti[1:stop\_point]))

} else {

return(sum(Ti))

}

}

# Simulam 10^6

valoriT <- replicate(nrSimulari, simulator())

1) Construiți un algoritm ȋn R care simulează 6 10 valori pentru v.a. T și ȋn baza acestora aproximați E T( ). Reprezentați grafic ȋntr-o manieră adecvată valorile obținute pentru T . Ce puteți spune despre repartiția lui T ?

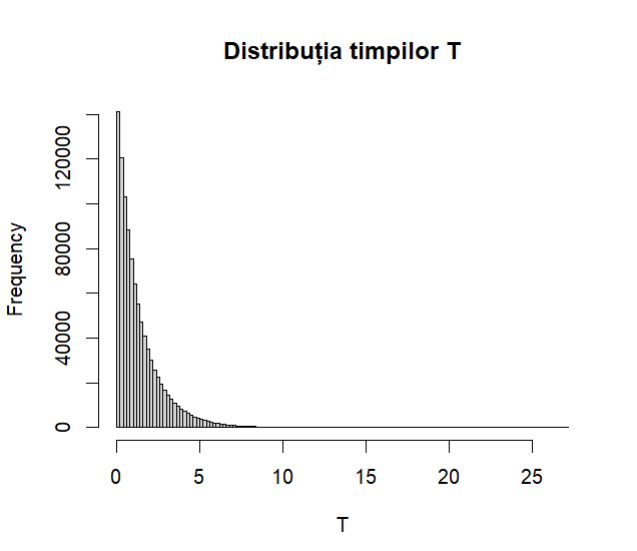
approx\_E\_T <- mean(valoriT)

hist(valoriT, breaks = 100, main = "Distribuția timpilor T", xlab = "T")

Explicatii:

Pentru calculul aproximativ al mediei am facut pur si simplu media valorilor rezultate in urma simularii

Afisand in histograma valorile obtinute in valoriT rezulta distrbutia lui T. Se poate observa ca reprezentarea grafica a timpilor T urmeaza o distributie exponentiala



Rezultat:

approx\_E\_T = 9.51113170988666

2) Calculați valoarea exactă a lui E [T] și comparați cu valoarea obținută prin simulare.

exact\_E\_T <- sum(1 / lambda \* cumprod( alpha ))

Explicatii:

Valoarea exacta a lui E[T] se calculeaza dupa urmatoarea formula:

E[T]=E[T1​]+α1​E[T2​]+α1​α2​E[T3​]+…+ α1α2...αn-1E[Tn]

Am folosit functia cumprod care ne returneaza un vector cu produsul comulativ. Alpha are 99 de valori deoarece exista n-1 pasi pentru trecerea de la o etapa la cealalta. Am atasat acestui vector la inceput valoarea 1 deoarece probabilitatea ca persoana sa participe la prima etapa este 100%. Apoi am inmultit produsul cumulativ la 1/λ deoarece X~Exp(λ) => E[X] = 1/λ si in final am facut suma tuturor mediilor pe fiecare etapa rezultand E[T].

In urma analizarii rezultatului exact observam ca este foarte apropiat de rezultatul aproximativ obtinut prin simulare

Rezultat:

exact\_E\_T = 9.51299920302413

3) Ȋn baza simulărilor de la 1) aproximați probabilitatea ca persoana A să finalizeze activitatea.

prob\_finalizare <- mean(valoriT >= sum(1 / lambda))

Explicatii:

Am calculat probabilitatea ca o persoana sa finalizeze fiecare etapa luand ca exemplu in simularea noastra un numar de 100 de etape si o probabilitate de a trece de la o etapa la alta >=80% pentru un rezultat mai consistent.

In scrierea codului am calculat valoarea teoretica a timpului total pe care o persoana trebuie sa il petreaca in cele 100 de etape pentru a spune ca a finalizat. Astfel am calculat media unui vector de valori booleene care valoarea true corespunde rezultatului unei simulari>= valoarea timpului adica persoanele care au finalizat respectiv false pentru persoanele care nu au reusit sa finalizeze. Rezultatul este mic datorita numarului de etape, acesta ar creste o data cu micsorarea numarului de etape/ ar scadea o data cu cresterea numarului de etape. Totodata, rezultatul va scadea odata cu micsorarea marginii inferioare a valorilor probabilitatilor din alpha si invers.

Rezultat:

prob\_finalizare = 0.00004

4) Ȋn baza simulărilor de la 1) aproximați probabilitatea ca persoana A să finalizeze activitatea ȋntr-un timp mai mic sau egal cu σ .

sigma <-94

prob\_sigma<- mean(valoriT <= sigma & valoriT >= sum(1 / lambda))

Explicatii:

Ne-am folosit de modalitatea de rezolvare a exercitiului anterior, astfel am folosit un vector de valori booleene care verifica 2 conditii <= sigma si >= valoarea timpului total, apoi am facut media si ne-a rezultat valoarea probabilitatii ca o persoana sa finalizeze intr-un timp mai mic decat sigma. Este normal ca valoarea acestei probabilitati sa fie mai mica sau egala cu valoarea probabilitatii de a finaliza.

Rezultat:

prob\_sigma = 0.000007

5) Ȋn baza simulărilor de la 1) determinați timpul minim și respectiv timpul maxim ȋn care persoana A finalizează activitatea și reprezentați grafic timpii de finalizare a activității din fiecare simulare. Ce puteți spune despre repartiția acestor timpi de finalizare a activității?

timpMin <- min(valoriT[valoriT >= sum(1 / lambda)] )

timpMax <- max(valoriT[valoriT >= sum(1 / lambda)] )

cat("Timpul minim:", timpMin, "Timpul maxim:", timpMax, "\n")

hist(valoriT[valoriT >= sum(1 / lambda)], breaks=50, main="Distribuția timpurilor de finalizare (valide)", xlab="Timpul de finalizare (T)", col="lightblue", border="black")

Explicatii:

Am considerat variabilele timpMin si timpMax pentru valorile minime si maxime a timpilor persoanelor care au finalizat evenimentul. am folosit functia min pentru minim si ca parametru am dat lista de timpi simulati, filtrata de valorile mai mari decat media timpului total. Am folosit dinnou pentru reprezentarea grafica o histograma deoarece ne ajuta la identificarea repartitiei. 2 observatii sunt faptul ca numarul de persoane care au reusit sa finalizeze evenimentul este foarte mic in comparatie cu numarul de simulari si ca se vede ca repartitia este uniforma deoarece pentru alegerea valorilor random am folosit runif.

A graph of a number of columns

Description automatically generated with medium confidence

Rezultat:

timpMax = 108.54443085275

timpMin = 92.0171680316328

6) Ȋn baza simulărilor de la 1) aproximați probabilitatea ca persoana A să se oprească din lucru ȋnainte de etapa k , unde 1<k ≤ n . Reprezentați grafic probabilitățile obținute ȋntr-o manieră corespunzătoare. Ce puteți spune despre repartiția probabilităților obținute?

k <- 5

PStopK <- mean(sapply(valoriT, function(x) {

if(x<=sum(1/lambda[1:k-1]))

return (TRUE)

else

return(FALSE)

}))

Explicatii:

Am creat o variabila numita PstopK care primeste probabilitatea ca persoana A sa se opreasca din lucru inainte de etapa k . Pentru aceasta probabilitate am aplicat ca parametru unei functii fiecare valoare simulata. Functia returneaza TRUE daca timpul este mai mic decat media timpului total pana la pasul k-1(adica poate face maxim k-1 pasi), si FALSE altfel. Variabila primeste media valorilor din lista booleana.

Rezultat:

PstopK = 0.284293 pentru k=5

PstopK = 0.51482 pentru k=10

PstopK = 0.991332 pentru k=50

calculate\_PStopBeforeK <- function(valoriT, lambda, alpha, n) {

# Calculam timpii cumulativi pentru fiecare etapa

cumulative\_times <- cumsum(1 / lambda)

# Calculam probabilitatile pentru fiecare k

PStopBeforeK <- sapply(2:n, function(k) {

mean(valoriT <= cumulative\_times[k - 1]) # Probabilitatea de oprire inainte de etapa k

})

return(PStopBeforeK)

}

PStopBeforeK <- calculate\_PStopBeforeK(valoriT, lambda, alpha, n)

barplot(PStopBeforeK, names.arg = 2:n, col = "lightblue",

main = "Probabilitățile de oprire înainte de etapa k",

xlab = "Etapa k", ylab = "Probabilitatea de oprire",

xlim = c(1, n), ylim = c(0, 1))

A graph of a bar graph

Description automatically generated

Observatii:

Se poate observa ca o persoana cu cat ajunge mai departe in etape cu atat are sanse mai mari sa finalizeze evenimentul ,iar in etapele initiale sansele sunt foarte mici. Aceste observatii sunt date datorita pantei abrupte in faza initiala ,respectiv panta lina de la final.