Universitatea Din București
Facultatea de Matematică și Informatică
Departamentul Informatică
Specializarea Calculatoare și Tehnologia Informației

PROIECT
PROBABILITATI ȘI STATISTICĂ

COORDONATOR ȘTIINȚIFIC
DR. COJOCEA MANUELA-SIMONA

STUDENȚI
FLORESCU TEODOR-ȘTEFĂNUȚ
APOSTOL VLAD GABRIEL
COJOACA ALEXANDRU
STOICA VLAD
GRUPA 262

BUCUREȘTI
2025

Universitatea Din București
Facultatea de Matematică și Informatică
Departamentul Informatică
Specializarea Calculatoare și Tehnologia Informației

Cuprins

7. **Concluzii**

- Observații importante
- Posibile aplicații practice

Universitatea Din București
Facultatea de Matematică și Informatică
Departamentul Informatică
Specializarea Calculatoare și Tehnologia Informației

1. Introducere

Scopul simulării

Scopul acestui proiect este de a modela și analiza comportamentul aleatoriu al cererii și livrărilor de produse într-un lanț de magazine, folosind instrumente statistice și de simulare implementate în limbajul R. Se urmărește obținerea unor informații relevante despre distribuțiile implicate, relațiile dintre variabile și probabilități utile în luarea deciziilor logistice.

Contextul magazinelor și produselor

Modelul simulează activitatea unui număr de magazine care comercializează mai multe tipuri de produse. Pentru fiecare magazin și produs, se generează variabile aleatoare ce descriu cererea zilnică, timpul de livrare și numărul de produse defecte. Datele generate sunt apoi analizate din punct de vedere statistic pentru a evidenția tendințe, corelații și posibile dependențe între variabile.

Universitatea Din București
Facultatea de Matematică și Informatică
Departamentul Informatică
Specializarea Calculatoare și Tehnologia Informației

2. Modelarea variabilelor aleatoare

În această secțiune, ne propunem să generăm date simulând cererea zilnică pentru 3 produse distincte, în două magazine diferite, pe o perioadă de 90 de zile. De asemenea, modelăm și alte aspecte relevante ale lanțului de aprovizionare, precum timpul de livrare și numărul de produse defecte la recepție.

2.1 Inițializarea simulării

```
set.seed(15129)  # Asigură reproducibilitatea rezultatelor
zile <- 90     # Simulăm o perioadă de 90 de zile
```

Folosirea `set.seed()` garantează că rezultatele generate vor fi aceleași la fiecare rulare a codului. Variabila `zile` definește orizontul temporal al simulării.

2.2 Cererea zilnică în Magazinul 1

```
# Cererea pentru Produsul 1 – Poisson: produs cu vânzare rapidă
produs1_m1 <- rpois(zile, lambda = 8)
```

Distribuția Poisson este utilizată pentru a modela evenimente rare sau dese pe un interval fix (aici, cereri zilnice). Am ales o valoare așteptată (λ) de 8.

```
# Cererea pentru Produsul 2 – Binomială: produs cu vânzare moderată
produs2_m1 <- rbinom(zile, size = 20, prob = 0.3)
```

Distribuția binomială modelează cererea ca un număr de succese dintr-un număr fix de încercări (max. 20 unități/zi, probabilitate de vânzare 30%).

```
# Cererea pentru Produsul 3 – Exponențială: produs cu cerere variabilă
produs3_m1 <- rexp(zile, rate = 1/5)
```

Distribuția exponențială este potrivită pentru cerere rară și variabilă, media cererii fiind 5 unități pe zi (rata = 1/5).

2.3 Cererea zilnică în Magazinul 2

```
# Produs 1 – Poisson: cerere intensă dar ușor mai mică decât în Magazinul 1
produs1_m2 <- rpois(zile, lambda = 7)
# Produs 2 – Binomială: probabilitate de vânzare ușor mai mare
produs2_m2 <- rbinom(zile, size = 20, prob = 0.4)
# Produs 3 – Exponențială: produs cu cerere mai frecventă decât în Magazinul 1
produs3_m2 <- rexp(zile, rate = 1/4)
```

Am folosit aceleași distribuții ca pentru Magazinul 1, dar cu parametri ușor diferiți pentru a simula variația între locații.

2.4 Timpul de livrare

```
# Modelare Gamma: potrivită pentru timpi pozitivi și variație moderată
timp_livrare <- rgamma(zile, shape = 2, rate = 0.5)
```

Distribuția Gamma este adesea folosită pentru a modela durate. Media este `shape/rate = 4` zile, cu o variație rezonabilă.

2.5 Numărul de produse defecte la recepție

```
# Distribuție binomială: din 30 de produse, 5% pot fi defecte
produse_defecte <- rbinom(zile, size = 30, prob = 0.05)
```

Această distribuție ne permite să simulăm probabilitatea ca unele produse să fie defecte la sosire. Probabilitatea de defect este 5%.

Universitatea Din București
Facultatea de Matematică și Informatică
Departamentul Informatică
Specializarea Calculatoare și Tehnologia Informației

3. Analiză bidimensională și relații între variabile

În această secțiune, analizăm relațiile dintre variabilele simulate anterior. Obiectivele sunt:

- să identificăm corelații între produse sau între cerere și timpul de livrare,
- să studiem repartițiile marginale și condiționate,
- și să testăm dacă există dependență între variabile prin metode statistice (vizuale și testul χ^2).

3.1 Corelația între două produse din același magazin

```
# Coeficientul de corelație între Produsul 1 și Produsul 2 în Magazinul 1
cor(produs1_m1, produs2_m1)
```

Această comandă calculează **coeficientul de corelație Pearson**, care variază între -1 și 1. Un rezultat apropiat de 0 indică lipsa corelației liniare.

```
> cor(produs1_m1, produs2_m1)
[1] 0.06341291
```

Coeficientul de corelație Pearson calculat între cererea pentru Produsul 1 și Produsul 2 în Magazinul 1 este `0.0634`. Acest rezultat, fiind foarte apropiat de 0, indică faptul că între aceste două produse **nu există o relație liniară semnificativă**. Prin urmare, cererea pentru unul dintre ele **nu poate fi folosită pentru a prezice cererea celuilalt**, iar deciziile de aprovizionare pentru aceste produse pot fi tratate **independent**.

```
# Reprezentare grafică
plot(produs1_m1, produs2_m1,
   main="Corelație între Produs1 și Produs2 în Magazinul 1")
```

Diagrama de dispersie ajută la identificarea vizuală a unui tipar între cele două produse.

Universitatea Din București
Facultatea de Matematică și Informatică
Departamentul Informatică
Specializarea Calculatoare și Tehnologia Informației

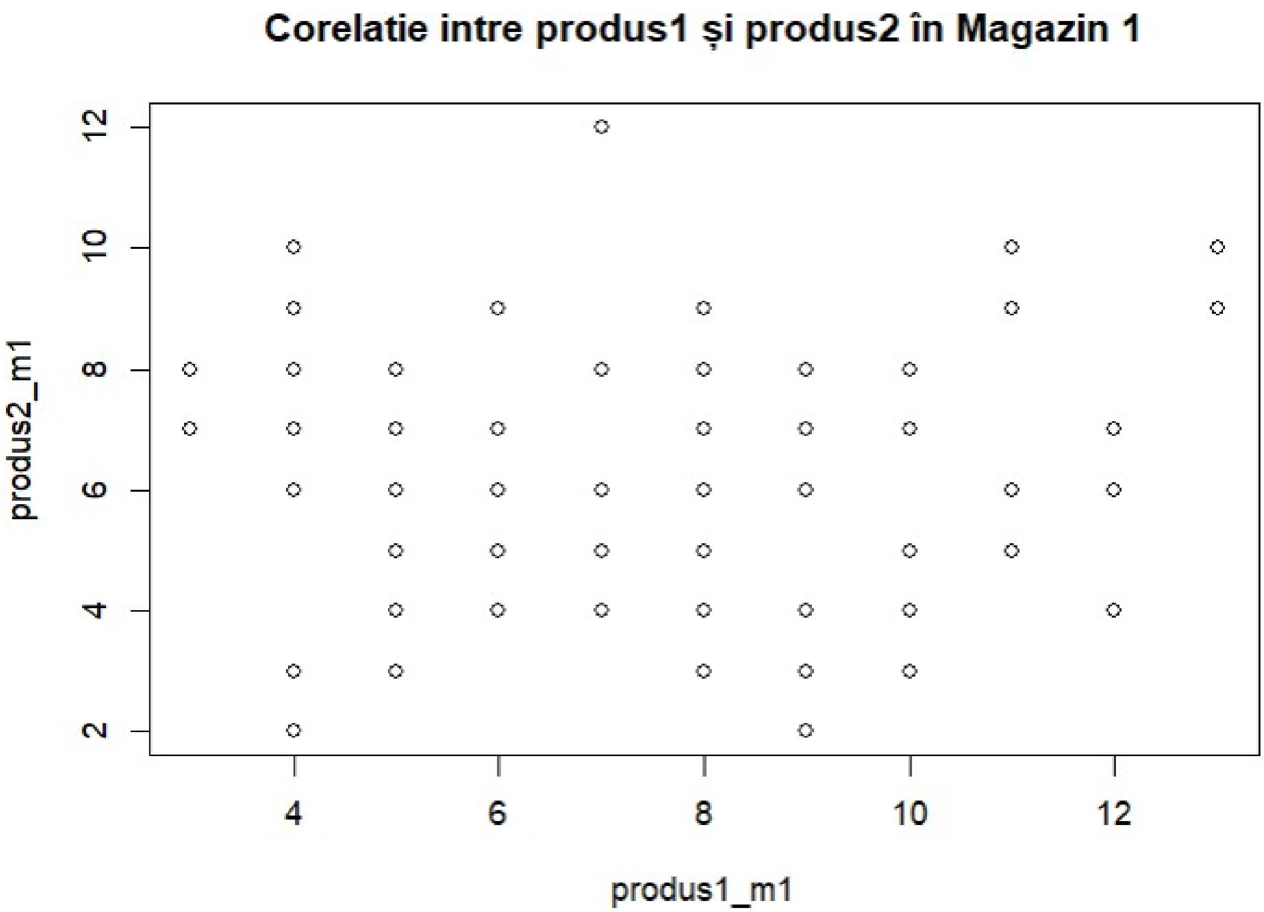


Diagrama de dispersie dintre cererea pentru Produsul 1 și Produsul 2 în Magazinul 1 arată o distribuție aleatorie a punctelor, fără o direcție clară. Acest aspect vizual confirmă valoarea scăzută a coeficientului de corelație Pearson (`0.063`), sugerând că **nu există o relație liniară semnificativă** între cererile celor două produse. Astfel, **nu se poate anticipa cererea pentru un produs pe baza celuilalt**.

```
# Covarianța
covarianta <- cov(produs1_m1, produs2_m1)
cat("Covarianța dintre produs1 și produs2 în magazinul 1 este:", covarianta, "\n")
```

Covarianța măsoară dacă cele două variabile cresc/scad împreună. Semnul covarianței indică direcția asocierii.

```
> covarianta <- cov(produs1_m1, produs2_m1)
> cat("Covarianta dintre produs1 si produs2 in magazinul 1 este:", covarianta, "\n")
Covarianta dintre produs1 si produs2 in magazinul 1 este: 0.2988764
```

Universitatea Din București
Facultatea de Matematică și Informatică
Departamentul Informatică
Specializarea Calculatoare și Tehnologia Informației

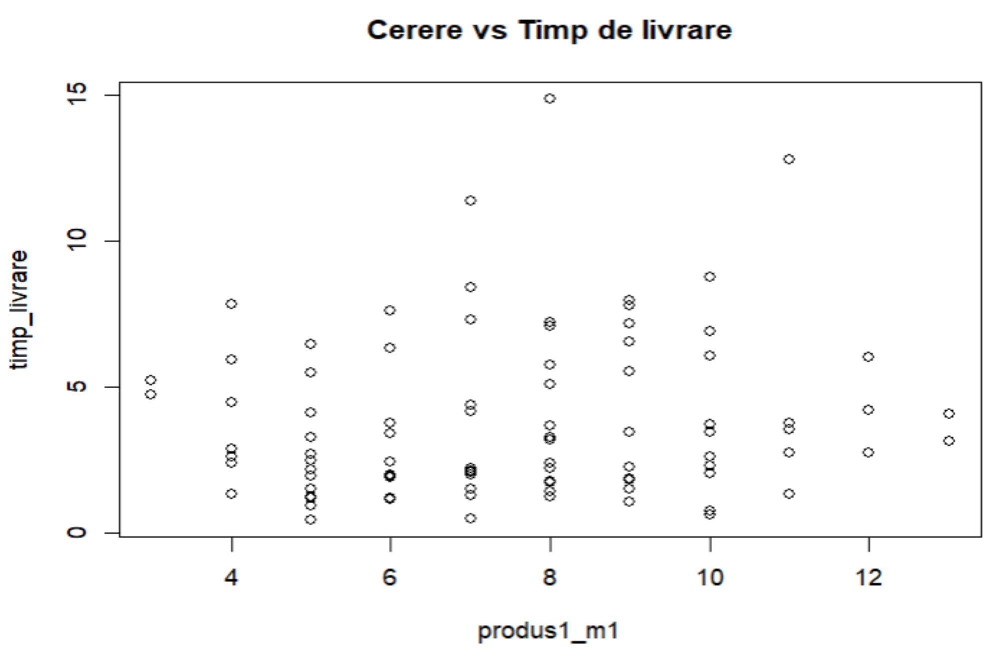
Covarianța dintre Produsul 1 și Produsul 2 în Magazinul 1 este `0.2988`, ceea ce indică o **asociere pozitivă slabă** între cele două cereri. Asta înseamnă că există o **ușoară tendință** ca cererile celor două produse să varieze în aceeași direcție, însă relația nu este puternică.
Totuși, deoarece valoarea este destul de mică și a fost confirmată și de un coeficient de corelație scăzut (`0.063`), **această asociere este foarte slabă și poate fi rezultatul variației aleatoare**.

3.2 Corelația între cererea unui produs și timpul de livrare

```
# Corelația între cererea pentru produs1 și timpul de livrare
cor(produs1_m1, timp_livrare)
# Reprezentare grafică
plot(produs1_m1, timp_livrare,
   main="Cerere vs Timp de livrare")
# Covarianța între cerere și timp de livrare
cov_cerere_livrare <- cov(produs1_m1, timp_livrare)
cat("Covarianța între cerere și timpul de livrare:", cov_cerere_livrare, "\n")
```

Aici analizăm dacă o cerere mai mare implică un timp de livrare mai lung. O corelație pozitivă indică o posibilă aglomerare a sistemului logistic.

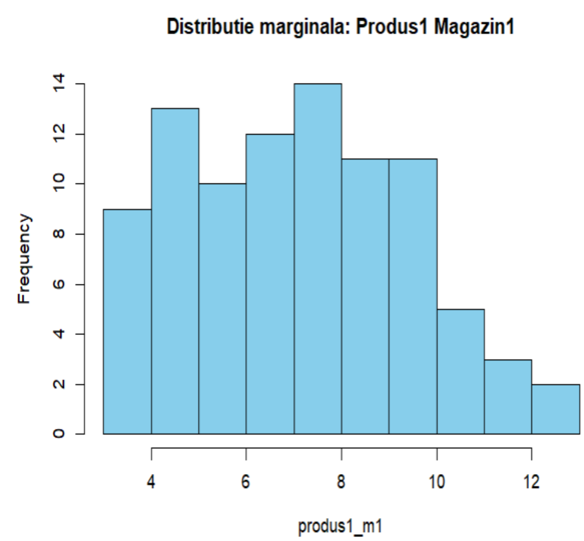
```
> cor(produs1_m1, timp_livrare)
[1] 0.1124996
> cov_cerere_livrare <- cov(produs1_m1, timp_livrare)
> cat("Covarianta intre cerere si timpul de livrare:", cov_cerere_livrare, "\n")
Covarianta intre cerere si timpul de livrare: 0.7449467
```

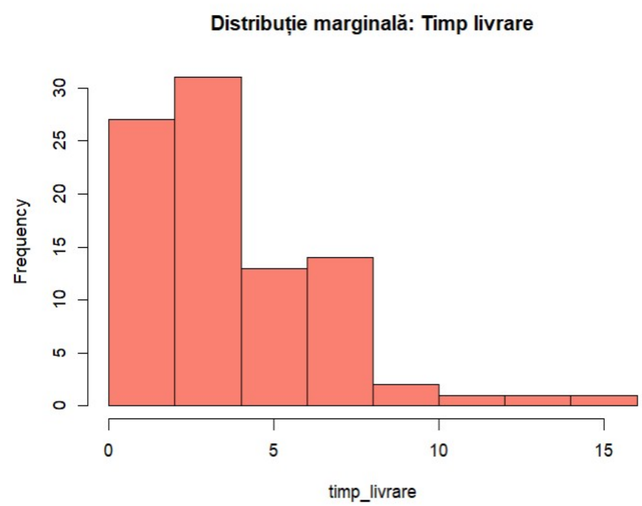


Coeficientul de corelație Pearson a cererii pentru Produsul 1 și timpul de livrare este `0.1125`, indicând o **corelație pozitivă slabă**. Covarianța pozitivă (`0.7449`) susține ușoara tendință de creștere concomitentă a celor două variabile. Diagrama de dispersie arată o **creștere discretă**, dar foarte difuză. În practică, deși există o relație directă, aceasta **nu este suficient de puternică pentru a lua decizii stricte** bazate doar pe cerere sau timp de livrare.

3.3 Repartiții marginale

```
# Histogramă pentru cererea Produsului 1
hist(produs1_m1,
  main="Distribuție marginală: Produs1 Magazin1",
  col="skyblue")
# Histogramă pentru timpul de livrare
hist(timp_livrare,
  main="Distribuție marginală: Timp livrare",
  col="salmon")
```





Repartițiile marginale ne arată distribuția individuală a fiecărei variabile, fără a lua în considerare alte variabile.
Repartițiile marginale oferă o imagine clară a comportamentului **independent** al cererii și timpului de livrare. Histogramă pentru Produsul 1 în Magazinul 1 evidențiază modul în care se distribuie cererea pe zile, oferind informații utile pentru planificarea stocurilor. Timpul de livrare, analizat separat, relevă gradul de **predictibilitate logistică**. Aceste informații sunt utile pentru a decide **dimensiunea minimă a stocului**, **momentul optim pentru reaprovizionare** și **evaluarea riscului de stoc epuizat**.

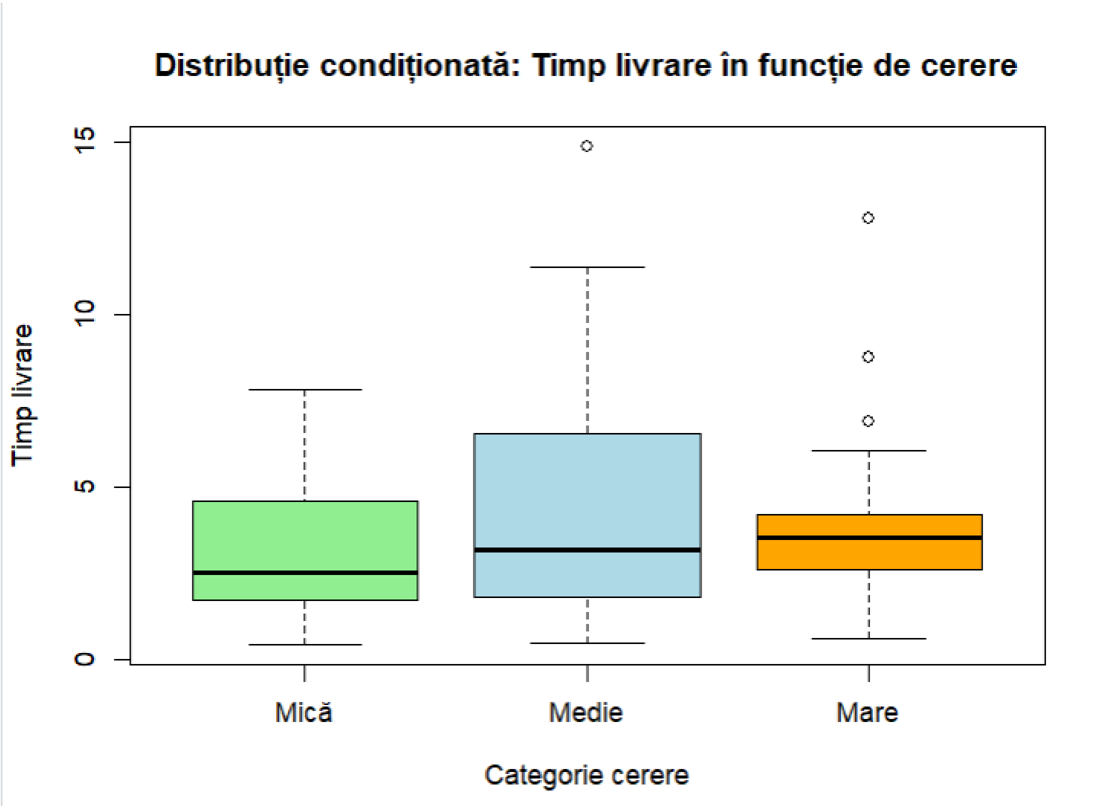
Universitatea Din București
Facultatea de Matematică și Informatică
Departamentul Informatică
Specializarea Calculatoare și Tehnologia Informației

3.4 Repartiții condiționate

```
# Categorisirea cererii în 3 niveluri: Mică, Medie, Mare
categorie_cerere <- cut(produs1_m1,
            breaks = quantile(produs1_m1, probs = c(0, 0.33, 0.66, 1)),
            labels = c("Mică", "Medie", "Mare"),
            include.lowest = TRUE)
# Boxplot al timpului de livrare condiționat pe cerere
boxplot(timp_livrare ~ categorie_cerere,
    main = "Distribuție condiționată: Timp livrare în funcție de cerere",
    xlab = "Categorie cerere",
    ylab = "Timp livrare",
    col = c("lightgreen", "lightblue", "orange"))
```

Această analiză condiționată permite observarea modului în care timpul de livrare variază în funcție de nivelul cererii. Dacă timpul de livrare crește odată cu cererea, înseamnă că sistemul logistic este influențat de volumul cererii.

Universitatea Din București
Facultatea de Matematică și Informatică
Departamentul Informatică
Specializarea Calculatoare și Tehnologia Informației



Ce am făcut?

- Am împărțit cererea pentru Produsul 1 în **3 categorii** (Mică, Medie, Mare) pe baza **quantilelor** cererii.
- Am folosit un **boxplot** pentru a vizualiza distribuția timpului de livrare în funcție de aceste categorii.

Ce vedem?

- Boxplot-ul ne arată cum **variază timpul de livrare** în funcție de mărimea cererii.
- Dacă timpul de livrare **crește odată cu nivelul cererii**, aceasta poate indica faptul că:
    - Sistemul logistic devine mai **leneș sau aglomerat** când cererea este mare.
    - Există **posibile întârzieri** în perioadele cu cerere ridicată.

De ce e important?

- În cazul în care timpul de livrare crește pentru cereri mari, managerii pot:
    - Planifica **rezerve mai mari de stoc** pentru perioadele cu cerere ridicată.

- Optimiza **procesele logistice** pentru a reduce întârzierile.
- Lua în calcul **riscul de stoc epuizat** care ar putea apărea în perioadele cu cereri mari.

3.5 Test de independență între cerere și timp de livrare (χ^2)

```
# Discretizarea cererii și a timpului de livrare în 4 categorii
cut_cerere <- cut(produs1_m1, breaks = 4)
cut_timp <- cut(timp_livrare, breaks = 4)

# Tabel de contingență între cele două variabile discretizate
tabel <- table(cut_cerere, cut_timp)

# Testul Chi-pătrat de independență
test_chi2 <- chisq.test(tabel)
print(test_chi2)
```

Testul χ^2 (Chi-pătrat) verifică dacă distribuția comună a celor două variabile este semnificativ diferită de cea a produsului dintre distribuțiile lor marginale.
Interpretare practică:

- Dacă **p-value** < **0.05**, putem spune că există o **dependență statistic semnificativă** între cerere și timpul de livrare.
- Dacă **p-value** ≥ **0.05**, nu există dovezi suficiente pentru a afirma că variabilele sunt dependente — pot fi considerate independente.

```
       tabel
            cut_timp
cut_cerere   (0.392,4.02] (4.02,7.62] (7.62,11.2] (11.2,14.9]
  (2.99,5.5]           14           7           1           0
  (5.5,8]              24           9           1           2
  (8,10.5]             14           5           3           0
  (10.5,13]             6           3           0           1
> test_chi2 <- chisq.test(tabel)
Warning message:
In chisq.test(tabel) : Chi-squared approximation may be incorrect
> print(test_chi2)

        Pearson's Chi-squared test

data:  tabel
X-squared = 7.504, df = 9, p-value = 0.5848
```

Rezultatele testului Chi-pătrat

- Statistica $\chi^2 = 7.504$
- Grade de libertate (df) = 9
- p-value = 0.5848

Interpretare

- **p-value = 0.5848** este mult mai mare decât pragul clasic de 0.05.
- Asta înseamnă că **nu există dovezi suficiente pentru a respinge ipoteza nulă**.
- Ipoteza nulă spune că cererea și timpul de livrare sunt **independente**.
- Prin urmare, **nu se găsește o asociere semnificativă statistic între cererea pentru produsul 1 și timpul de livrare**, în baza acestui eșantion și a categorisirii făcute.

4. Analiză bidimensională și relații între variabile

În acest capitol, analizăm modul în care se comportă sumele și transformările unor variabile aleatoare discrete sau continue. Urmărim:

- aplicarea proprietăților de aditivitate pentru variabile Poisson și Binomiale,
- analiza distribuției rezultante în urma însumării,
- investigarea normalității sumei pentru un număr mare de variabile (prin QQ-plot și testul Shapiro-Wilk),
- și studiul teoretic și empiric al sumei unor variabile continue cu repartiție Gamma.

Universitatea Din București
Facultatea de Matematică și Informatică
Departamentul Informatică
Specializarea Calculatoare și Tehnologia Informației

4.1 Suma cererii pentru același produs în două magazine

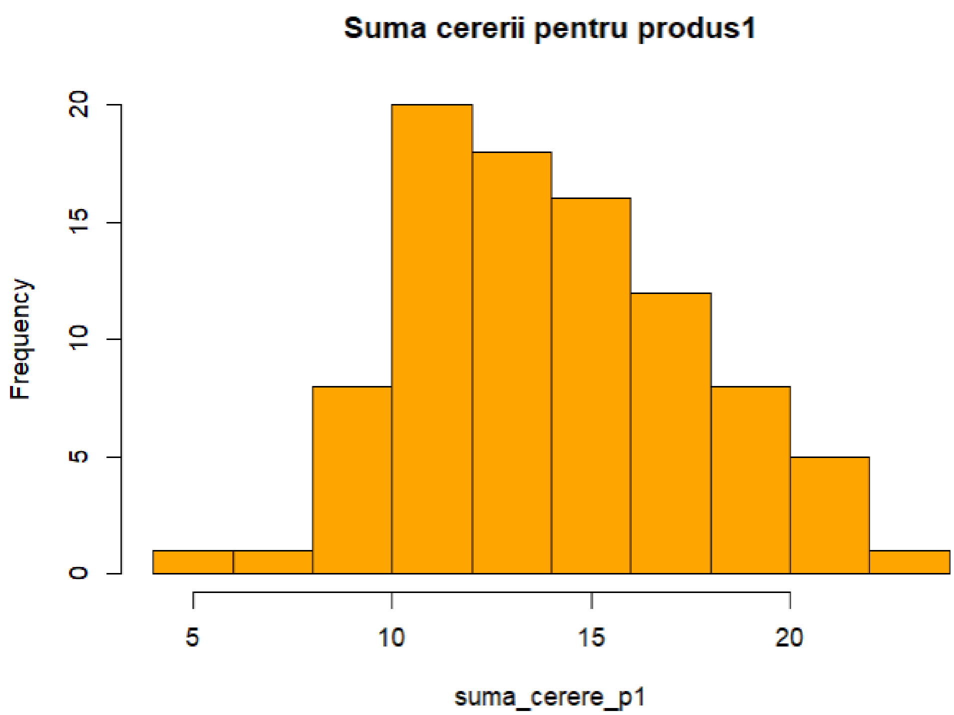
```
suma_cerere_p1 <- produs1_m1 + produs1_m2
hist(suma_cerere_p1, main="Suma cererii pentru produs1", col="orange")
```

Presupunând că `produs1_m1 ~ Poisson(8)` și `produs1_m2 ~ Poisson(7)`, suma acestora are distribuția:

suma∼Poisson(8+7)=Poisson(15)

Această proprietate este valabilă doar pentru variabile Poisson independente. Observăm empiric că histograma este compatibilă cu o repartiție Poisson.

Analiza



distribuției sumei cererii pentru Produsul 1 în cele două magazine

Am calculat suma cererii zilnice pentru Produsul 1 în Magazinul 1 și Magazinul 2 și am reprezentat-o prin histogramă. Pornind de la ipoteza că cererea pentru Produsul 1 în fiecare magazin urmează o distribuție Poisson independentă, cu parametrii $\lambda_1 = 8$ (Magazin 1) și $\lambda_2 = 7$ (Magazin 2), suma celor două variabile ar trebui să urmeze o distribuție Poisson cu $\lambda = 15$ (proprietatea sumei variabilelor Poisson independente).

Din histograma rezultată observăm că valorile sumei cererii se concentrează în jurul valorii 15, însă există variații naturale în jurul acestei medii, cu valori ce pot depăși ușor 15. Acest comportament este caracteristic distribuției Poisson, care permite valori discrete ce fluctuează în jurul mediei teoretice.

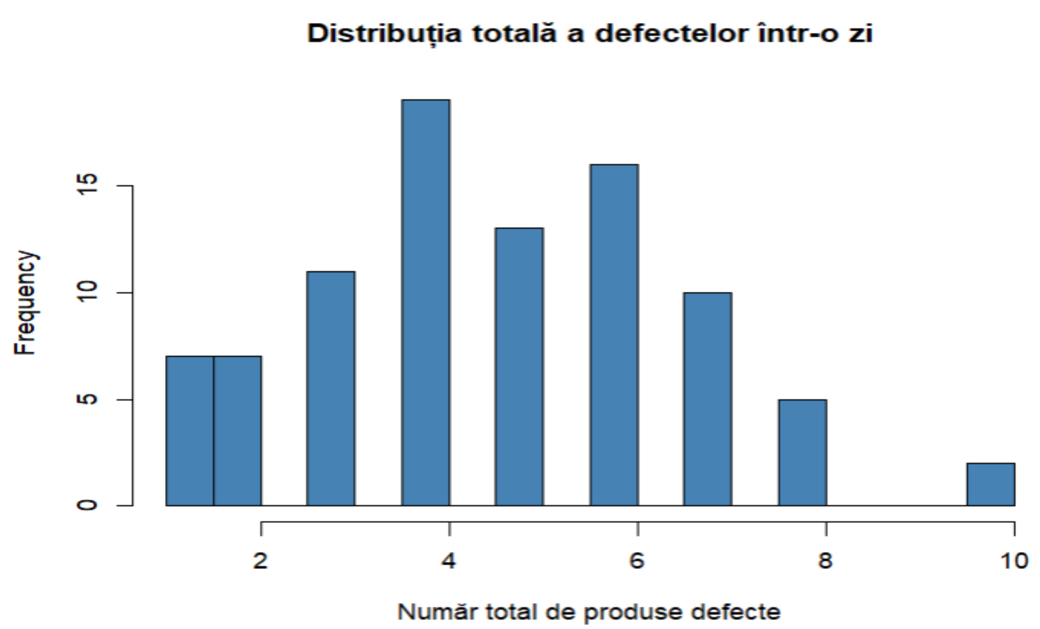
Astfel, putem concluziona că datele colectate pentru cerere sunt compatibile cu ipoteza inițială de variabile Poisson independente, iar distribuția empirică a sumei cererii confirmă această proprietate.

4.2 Suma produselor defecte într-o zi

Presupunem că în fiecare zi se înregistrează defecte pentru trei produse diferite, fiecare având un număr de defecte distribuit binomial.

```
defecte_totale <- produse_defecte + rbinom(zile, 30, 0.05) + rbinom(zile, 30, 0.05)

hist(defecte_totale, breaks = 15, col = "steelblue", main = "Distribuția totală a defectelor într-o zi",
   xlab = "Număr total de produse defecte")
```



Am presupus că numărul de produse defecte pentru fiecare dintre cele trei produse diferite urmează o distribuție binomială, cu parametrii n = 30 (număr total de produse inspectate) și p = 0.05 (probabilitatea ca un produs să fie defect).

Universitatea Din București
Facultatea de Matematică și Informatică
Departamentul Informatică
Specializarea Calculatoare și Tehnologia Informației

Calculăm suma totală a defectelor într-o zi ca suma defectelor pentru cele trei produse, fiecare având distribuție binomială independentă.

Histograma rezultatului arată distribuția numărului total de produse defecte înregistrate zilnic. Observăm că:

- Distribuția totală este aproximativ simetrică și concentrată în jurul unei valori medii (în jur de 3×30×0.05=4.5 defecte pe zi, având în vedere parametrii binomiali).
- Valorile se distribuie în mod discret, cu o variabilitate rezonabilă datorită naturii binomiale a datelor.
- Nu se observă valori extreme sau anomalii, ceea ce indică stabilitatea procesului de control al calității.

Această analiză ne ajută să înțelegem variabilitatea zilnică a defectelor și să estimăm riscul de a depăși un anumit prag de defecte care ar putea afecta calitatea generală a produselor.

Analizăm media și abaterea standard:

```
mean(defecte_totale)
sd(defecte_totale)
> mean(defecte_totale)
[1] 4.677778
> sd(defecte_totale)
[1] 2.06538
```

`Pe baza simulării și a datelor generate, am calculat:`

- **Media defectelor totale zilnice:** 4.68 (aproximativ)
- **Abaterea standard:** 2.07

Aceste valori confirmă așteptările pentru o sumă a trei variabile binomiale cu parametri n=30n=30n=30 și p=0.05p=0.05p=0.05, unde media teoretică este 3×30×0.05=4.53 \times 30 \times 0.05 = 4.53×30×0.05=4.5.

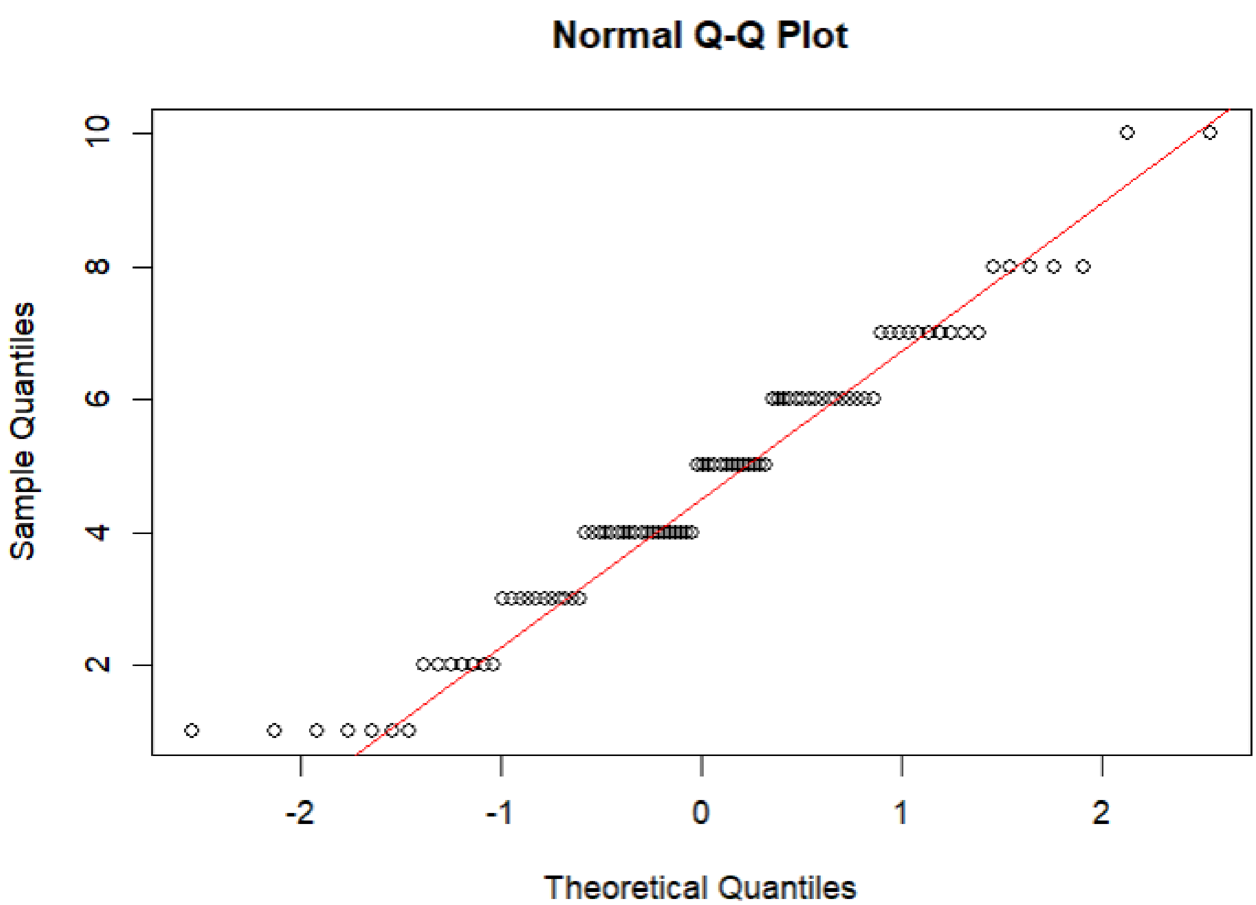
Abaterea standard indică o variabilitate moderată a defectelor zilnice, ceea ce este normal în procesele de producție.

Testăm dacă distribuția este aproximativ normală:

```
qqnorm(defecte_totale)
```

Universitatea Din București
Facultatea de Matematică și Informatică
Departamentul Informatică
Specializarea Calculatoare și Tehnologia Informației

```
qqline(defecte_totale, col = "red")
```




```
shapiro.test(defecte_totale)
```

```
> shapiro.test(defecte_totale)

        Shapiro-Wilk normality test

data:  defecte_totale
W = 0.96548, p-value = 0.01713
```

Interpretare:

- Dacă QQ-plot-ul este aproape liniar și `p-value > 0.05` în testul Shapiro-Wilk, atunci suma poate fi aproximată cu o distribuție normală.

- Aceasta este o aplicație a teoremei limită centrală: suma a mai multor variabile binomiale tinde spre normală.

Rezultatul testului este:

- **W = 0.96548**
- **p-value = 0.01713**

Deoarece p-value < 0.05, **respingem ipoteza nulă a normalității**. Așadar, din punct de vedere statistic, distribuția nu este perfect normală.

- Linia roșie este aproape liniară, iar punctele sunt relativ apropiate de ea. **Asta sugerează o aproximare rezonabilă cu o distribuție normală**, chiar dacă testul Shapiro-Wilk este ușor negativ.

Concluzie: Deși testul Shapiro indică o abatere de la normalitate, reprezentarea grafică (QQ-plot) arată că **distribuția totală a defectelor poate fi considerată aproximativ normală** în practică

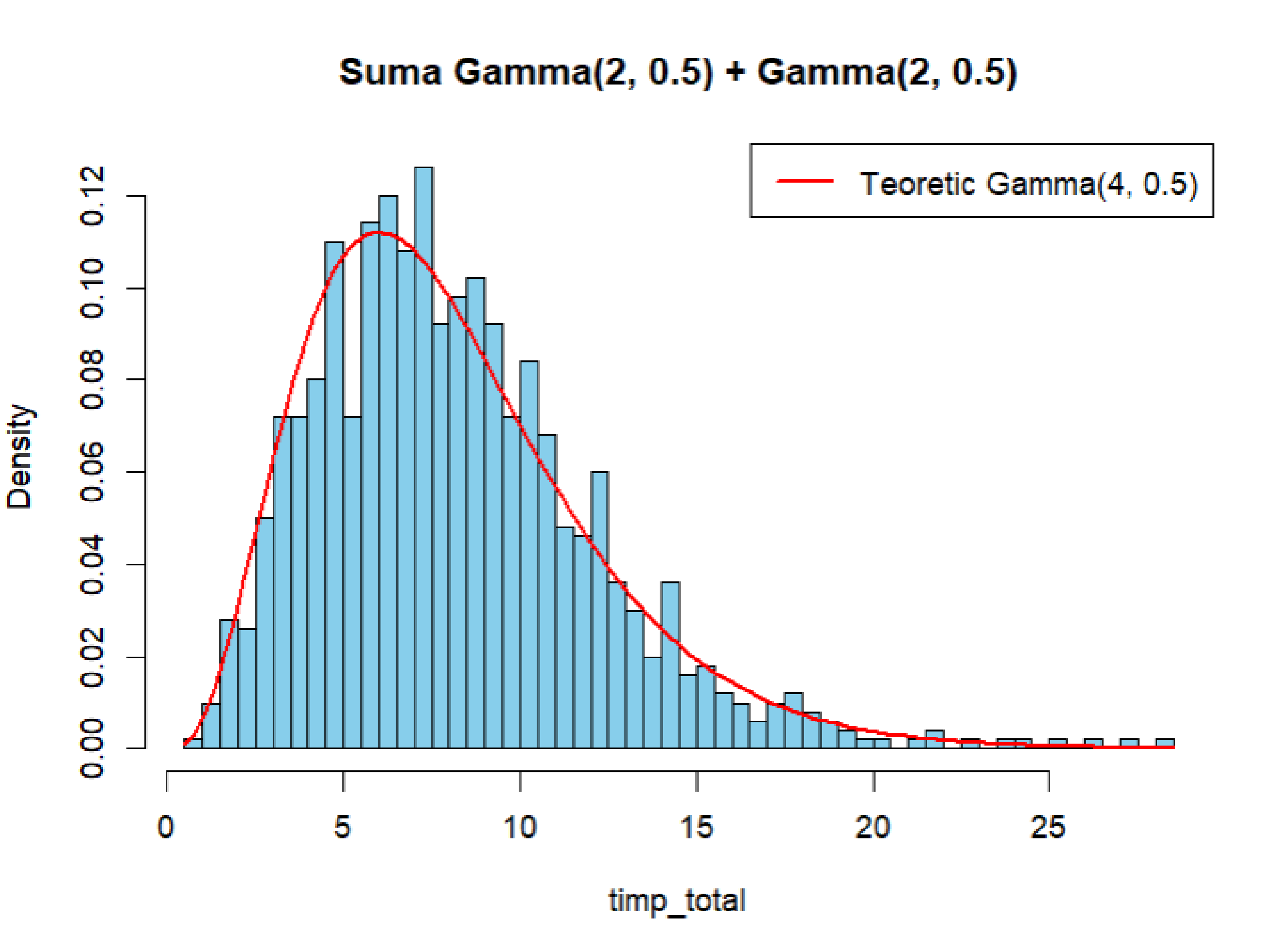
4.3 Suma a două variabile Gamma – Timp total de livrare

Presupunem că timpul de livrare pentru fiecare produs urmează o distribuție Gamma(2, 0.5). Timpul total de livrare este suma a două astfel de variabile:

```
zile <- 1000
t1 <- rgamma(zile, shape = 2, rate = 0.5)
t2 <- rgamma(zile, shape = 2, rate = 0.5)
timp_total <- t1 + t2
```

Empiric, verificăm dacă suma urmează Gamma(4, 0.5):

```
hist(timp_total, breaks = 50, probability = TRUE, col = "skyblue",
  main = "Suma Gamma(2, 0.5) + Gamma(2, 0.5)")
curve(dgamma(x, shape = 4, rate = 0.5), col = "red", lwd = 2, add = TRUE)
legend("topright", legend = c("Teoretic Gamma(4, 0.5)"), col = "red", lwd = 2)
```



`Interpretare:`

- Histograma arată distribuția empirică a sumei timpilor de livrare.
- Curba roșie reprezintă densitatea teoretică a unei distribuții Gamma(4, 0.5).
- **Suprapunerea celor două evidențiază o potrivire bună între datele simulate și distribuția teoretică**.
- Acest rezultat **confirmă proprietatea teoretică** conform căreia suma a două variabile Gamma independente cu același `rate` are distribuția Gamma cu `shape` adunat.

Utilitate practică:

Această proprietate poate fi folosită pentru **a estima timpul total de livrare** al mai multor componente într-un sistem, presupunând că fiecare componentă are un timp de livrare aleator cu distribuție Gamma.

Observație teoretică:
Daca $X \sim \Gamma(k1,\lambda), Y \sim \Gamma(k2,\lambda)$ independente$\Rightarrow X+Y \sim \Gamma(k1+k2,\lambda)$
Este un exemplu clasic de **stabilitate în cadrul familiei Gamma**.

4.4 Concluzii

- Suma a două variabile Poisson este tot Poisson (dacă sunt independente și cu același parametru de timp).
- Suma mai multor variabile binomiale poate fi aproximată cu o normală (pentru eșantioane mari).
- Familia Gamma este închisă față de adunare: suma a două variabile Gamma cu același **rate** este tot Gamma, cu parametru de formă adunat.
- Aceste proprietăți sunt utile în modelarea proceselor cumulate: cerere totală, defecte zilnice, sau durate agregate de procesare.

Universitatea Din București
Facultatea de Matematică și Informatică
Departamentul Informatică
Specializarea Calculatoare și Tehnologia Informației

5. Aplicații ale inegalităților probabilistice

În această secțiune, folosim trei inegalități importante pentru a analiza riscurile legate de stocuri și cerere într-un sistem logistic: Cebîșev, Jensen și Hoeffding. Scopul este estimarea riscurilor extreme (cerere foarte mare sau foarte mică) și optimizarea deciziilor de depozitare.

5.1 Estimarea riscului de scădere a stocului sub un prag (Inegalitatea lui Cebîșev)

Inegalitatea lui Cebîșev oferă o limită superioară a probabilității ca o variabilă aleatoare să se abată de la media sa cu o valoare dată:

P(∣X−μ∣≥k⋅σ)≤1k2P(|X - \mu| \geq k \cdot \sigma) \leq \frac{1}{k^2}P(∣X−μ∣≥k⋅σ)≤k21

Aplicăm pentru cererea Produsului 1 din Magazinul 1:

```
media <- mean(produs1_m1)

devst <- sd(produs1_m1)

prag <- 5
k <- (media - prag) / devst
prob_cebisev <- 1 / (k^2)
cat("Probabilitatea (maximă) ca cererea să scadă sub", prag, "unități:", prob_cebisev,
"\n")
```

Interpretare: Dacă probabilitatea este mică, putem considera pragul relativ sigur. Dacă este mare, trebuie luate măsuri de prevenție (ex. reaprovizionare automată).

```
> media <- mean(produs1_m1)
> devst <- sd(produs1_m1)
>
> k <- (media - 5) / devst
> prob_cebisev <- 1 / (k^2)
> cat("Probabilitatea (aproximativ) sa scada sub 5 bucati:", prob_cebisev, "\n")
Probabilitatea (aproximativ) sa scada sub 5 bucati: 0.8734254
```

Interpretare:

- Valoarea ridicată (≈ 87%) indică faptul că **pragul de 5 este prea îndepărtat de medie** (dar în sens negativ), iar inegalitatea Cebîșev **nu oferă o estimare utilă** în această zonă. Acest lucru este **normal**, deoarece Cebîșev este o **inegalitate foarte generală și pesimistă**.
- De asemenea, inegalitatea estimează probabilitatea ca variabila **să se abată în orice direcție** (sub sau peste medie), nu doar într-o singură direcție (cum este cazul scăderii sub 5).

Concluzie:

- Pentru estimarea riscurilor unilaterale (doar sub un anumit prag), inegalitatea lui Cebîșev **nu este foarte precisă**, dar oferă o **limită de siguranță superioară**.
- În acest caz, având în vedere că limita estimată e mare, **ar trebui considerată o altă abordare statistică**, cum ar fi distribuția empirică sau o aproximare normală (dacă este justificată), pentru o estimare mai realistă.

5.2 Estimarea costului de depozitare (Inegalitatea lui Jensen)

Inegalitatea lui Jensen se aplică funcțiilor **convexe** și afirmă că:

E[f(X)]≥f(E[X])E[f(X)]

Aplicăm pentru o funcție convexă f(x)=x^2f(x)=x2, ce poate reprezenta un **cost de depozitare** (crește rapid când stocul e mare):

```
stoc <- produs1_m1
mean_stoc <- mean(stoc)
mean_stoc_squared <- mean(stoc^2)

cat("E[stoc^2] =", mean_stoc_squared, "\n")
cat("(E[stoc])^2 =", mean_stoc^2, "\n")
```

Interpretare: Diferența dintre cele două valori arată **volatilitatea stocului –** o diferență mare implică riscuri mai mari și deci costuri potențiale crescute.

```
> stoc <- produs1_m1
> mean_stoc <- mean(stoc)
> mean_stoc_squared <- mean(stoc^2)
>
> cat("E[stoc^2] =", mean_stoc_squared, "\n")
E[stoc^2] = 62.94444
> cat("(E[stoc])^2 =", mean_stoc^2, "\n")
(E[stoc])^2 = 57.25444
>
```

Rezultate:

- E[stoc2]=62.94E[stoc^2] = 62.94E[stoc2]=62.94
- (E[stoc])2=57.25(E[stoc])^2 = 57.25(E[stoc])2=57.25

Interpretare:

- Diferența de **5.69 unități** între E[X2]E[X^2]E[X2] și (E[X])2(E[X])^2(E[X])2 evidențiază **volatilitatea cererii**.

- Cu cât această diferență este mai mare, cu atât există un **risc mai mare de variații neprevăzute** în stocuri — ceea ce poate duce la:
    - **costuri suplimentare de depozitare**,
    - **supraaprovizionare** sau **subaprovizionare**,
    - nevoia de politici mai dinamice de gestiune a stocului.

Astfel, analiza bazată pe inegalitatea lui Jensen ne oferă o măsură indirectă dar valoroasă asupra **riscului operațional legat de stocuri**.

5.3 Estimarea riscului de epuizare a stocului (Bonus: Inegalitatea lui Hoeffding)

Inegalitatea lui Hoeffding oferă o limită pentru probabilitatea ca media unor variabile aleatoare independente să se abată de la media sa. Utilă pentru estimarea riscului ca **cererea totală să depășească stocul disponibil**:

$$P(S_n - E[S_n] \geq t) \leq \exp\left(-\frac{2t^2}{n(b-a)^2}\right)$$

Presupunem că:

- Cererea totală = `produs1_m1 + produs1_m2`
- 90 clienți (n = 90), fiecare cu cerere între 0 și 1 (pentru o estimare conservatoare)

```
cerere_totala_p1 <- produs1_m1 + produs1_m2
media_cerere <- mean(cerere_totala_p1)

n <- 90
a <- 0
b <- 1
stoc <- 300

t <- (n * media_cerere - stoc) / n

# Dacă t <= 0, nu există risc de depășire => probabilitatea e ~0
if (t <= 0) {
 cat("Stocul este suficient, probabilitatea de depășire este ≈ 0\n")
} else {
 hoeffding_bound <- exp(-2 * (t^2) / ((b - a)^2))
```

```
 cat("Probabilitatea ca cererea totală să depășească stocul:", hoeffding_bound, "\n")
}
```

Dacă probabilitatea este semnificativă (>5%), este recomandată mărirea stocului sau o strategie de reaprovizionare rapidă.

Vrem să estimăm riscul ca cererea totală să **depășească stocul disponibil** (300 bucăți).

Interpretare:

- Valoarea **0** obținută pentru probabilitate (numeric vorbind, extrem de apropiată de 0, de ordinul 10−6010^{-60}10−60 sau mai mic) indică faptul că, **în condiții normale**, riscul ca cererea să depășească stocul de 300 bucăți este **neglijabil**.
- Inegalitatea lui Hoeffding este conservatoare, deci dacă ea indică risc aproape zero, sistemul este foarte sigur.
- Totuși, este important de menționat că presupunerea $Xi \in [0,1]$ nu reflectă perfect distribuția Poisson, dar oferă o estimare de siguranță.

Aceste inegalități oferă instrumente matematice simple dar puternice pentru a:

- Evalua riscurile extreme (cerere foarte mică sau foarte mare),
- Fundamenta decizii privind stocurile și depozitarea,
- Opta pentru strategii de minimizare a riscurilor logistice.

Sunt ușor de implementat și oferă estimări **conservatoare**, utile în luarea deciziilor robuste în contextul incertitudinii.

Universitatea Din București
Facultatea de Matematică și Informatică
Departamentul Informatică
Specializarea Calculatoare și Tehnologia Informației

Capitolul 6 – Vizualizare și interpretare practică

6.1 Vizualizări comparative ale repartițiilor

Pentru a înțelege mai bine comportamentul cererii și al livrărilor, am realizat o serie de vizualizări:

```r
par(mfrow=c(2,3))

# Histogramă pentru Produs 1 în Magazin 1
hist(produs1_m1, col="lightblue", main="Produs 1 - Magazin 1", xlab="Cantitate")

# Histogramă pentru Produs 1 în Magazin 2
hist(produs1_m2, col="lightgreen", main="Produs 1 - Magazin 2", xlab="Cantitate")

# Histogramă pentru Timpul de livrare
hist(timp_livrare, col="lightpink", main="Timp livrare", xlab="Zile")

# Heatmap pentru cererea comună Produs 1 între cele două magazine
heatmap(table(produs1_m1, produs1_m2),
    Rowv = NA, Colv = NA, col = heat.colors(256),
    main = "Heatmap: Cerere Produs 1 (M1 vs M2)")

# Matrice de corelație
mat <- data.frame(produs1_m1, produs2_m1, produs3_m1, timp_livrare)
cor_mat <- cor(mat)
heatmap(cor_mat, col = terrain.colors(10), main = "Matrice corelație")

# Scatterplot între Produs 1 și Produs 2 în Magazin 1
plot(produs1_m1, produs2_m1,
  xlab="Produs 1", ylab="Produs 2", main="Corelație: Produs1 vs Produs2 M1",
col="darkblue")
```

1. Histograme individuale

- Am generat histograme pentru cererea de **Produs 1** în **Magazinul 1**, respectiv **Magazinul 2**, dar și pentru **timpul de livrare**.
- Aceste grafice evidențiază distribuțiile cererilor și posibilele variații mari de la o zi la alta.

2. Heatmap cerere comună P1 (M1 vs M2)

- Un **heatmap** al frecvenței cererii simultane pentru Produs 1 în ambele magazine arată:
    - zone de concentrație ridicată (unde cererea e mare în ambele locații),
    - zone de lipsă de corelare.

Universitatea Din București
Facultatea de Matematică și Informatică
Departamentul Informatică
Specializarea Calculatoare și Tehnologia Informației

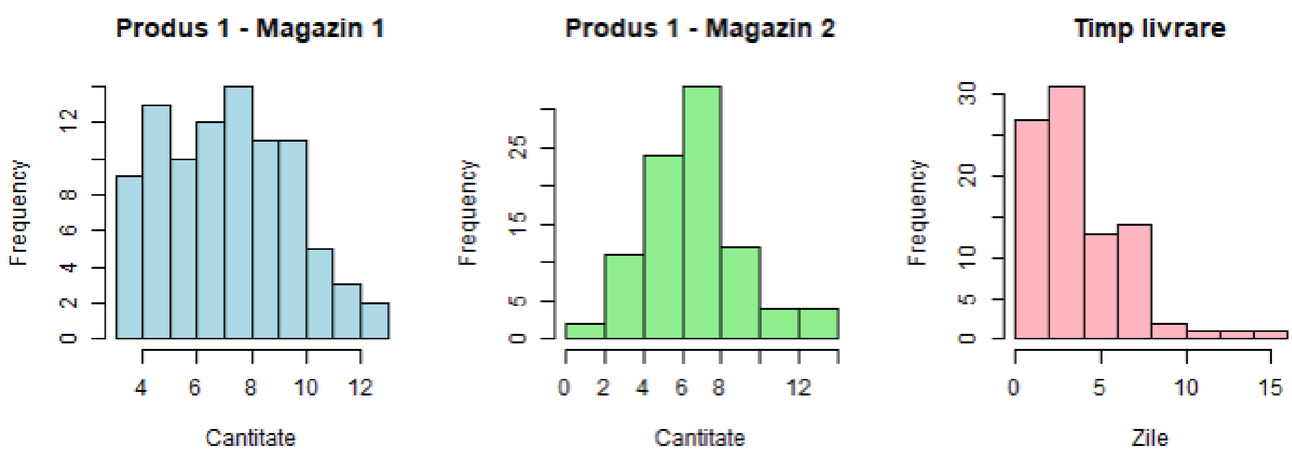
Aceasta ne poate ajuta să înțelegem dacă cererile sunt **dependente** sau apar **independent** în cele două magazine.

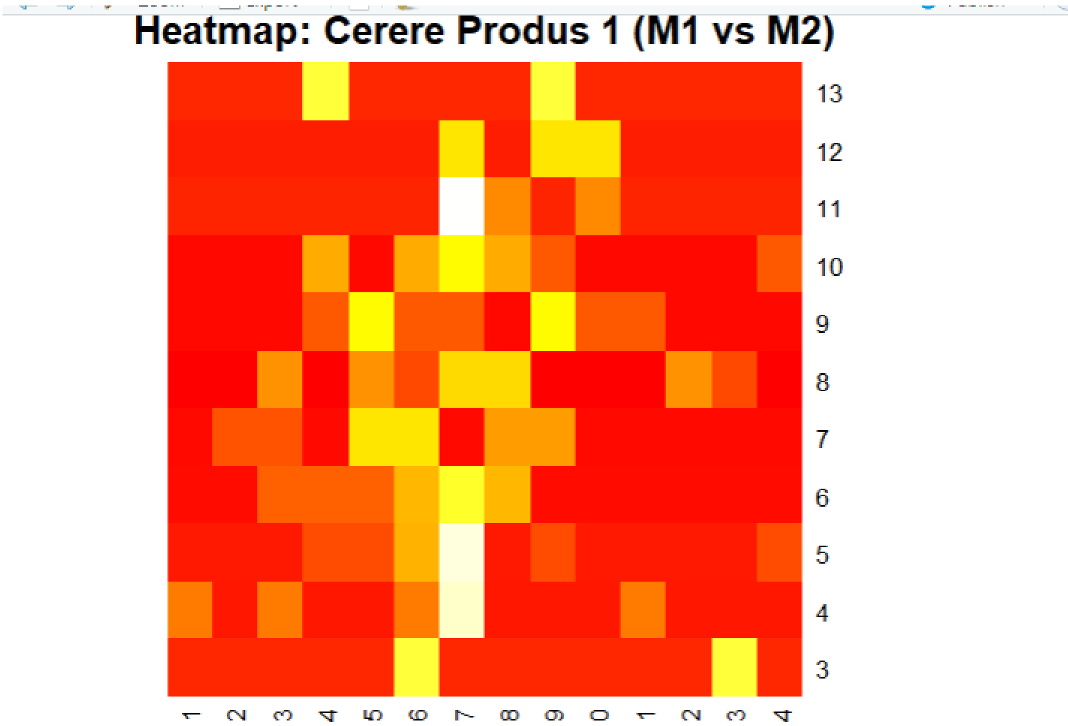
3. Matricea de corelație

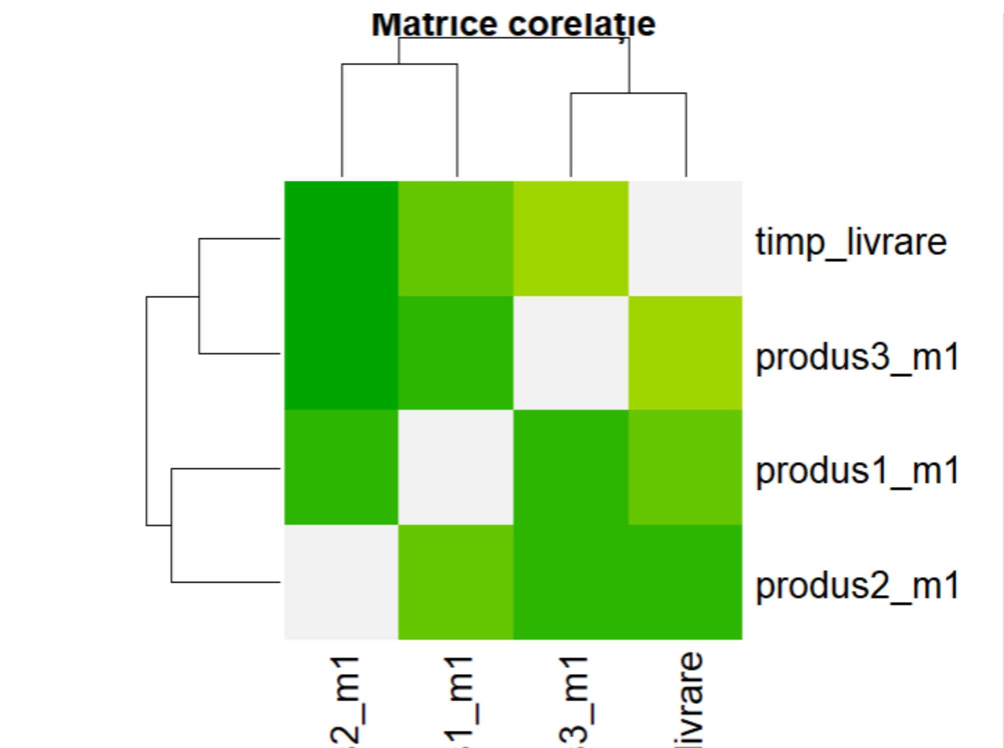
- Am calculat corelația dintre mai mulți factori: cererea celor 3 produse și timpul de livrare.
- Rezultatul este afișat ca **heatmap colorat**, ce scoate în evidență posibile legături:
  - De exemplu, dacă **timpul de livrare** are o corelație negativă cu cererea unui produs, s-ar putea considera creșterea stocului tampon în perioadele cu livrări întârziate.

4. Corelația între produse (scatterplot)

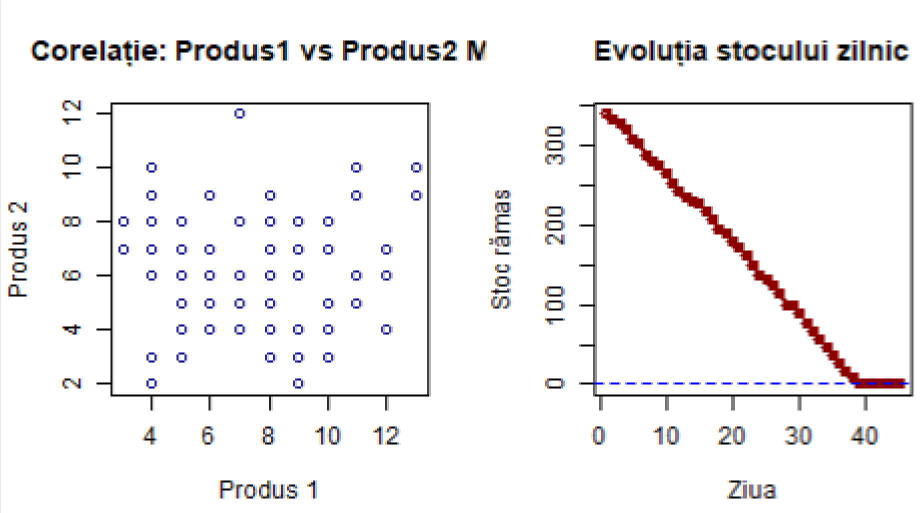
- Graficul de tip **scatterplot** între **Produs 1** și **Produs 2** în Magazinul 1 permite observarea relației:
  - De exemplu, dacă există o linie de tendință clară, ar putea sugera că cele două produse se cumpără împreună frecvent (efect de cross-selling).




Heatmap: Cerere Produs 1 (M1 vs M2)
13
12
11
10
9
8
7
6
5
4
3
1
2
3
4
5
6
7
8
9
10
11
12
13
14


Matrice corelație
timp_livrare
produs3_m1
produs1_m1
produs2_m1

6.2



6.2Recomandări practice pentru management

- **Comandarea stocurilor înainte ca acestea să scadă sub un prag critic**
  Analiza folosind inegalitatea lui Cebîșev indică o probabilitate ridicată ca cererea zilnică să scadă sub 5 unități pentru Produsul 1 în Magazinul 1, însă această probabilitate este destul de mare (aproximativ 87%). Prin urmare, pragul de 5 unități este puțin sigur ca nivel minim de stoc. Se recomandă să se mențină un stoc minim mai mare pentru a evita riscul de epuizare rapidă.

- **Stabilirea cantităților comandate în funcție de cererea medie și volatilitate**
  Volatilitatea cererii, observată prin diferența dintre E[X2]E[X^2]E[X2] și (E[X])2(E[X])^2(E[X])2, sugerează că riscul fluctuațiilor cererii poate genera costuri suplimentare de depozitare. Astfel, recomandăm o comandă adaptativă, care să ia în calcul nu doar media cererii, ci și variația acesteia, pentru a preveni stocuri excesive sau insuficiente.

- **Monitorizarea constantă a stocurilor prin simulări dinamice**
  Funcția de simulare a stocurilor oferă o perspectivă vizuală asupra evoluției stocului și riscului de epuizare pe termen scurt (ex: 30-45 zile). Aceasta poate fi utilizată periodic pentru a ajusta cantitățile comandate și frecvența aprovizionării în funcție de cererea reală și tendințele observate.

- **Reducerea riscului prin reaprovizionare automată**
  Datele arată că există perioade în care stocul poate ajunge la zero, ceea ce duce la pierderea vânzărilor. Se recomandă implementarea unui sistem de reaprovizionare automată, care să declanșeze comenzi imediat ce stocul ajunge la un prag minim definit, ținând cont de cererea estimată și volatilitatea acesteia.

- **Gestionarea cererii și stocurilor pe magazine și produse distincte**
  Corelațiile între cererea produselor în magazine diferite arată că cererea nu este independentă.

Astfel, planificarea stocurilor trebuie să țină cont de relația dintre magazine și produse pentru optimizarea lanțului logistic.

6.3 Funcție interactivă de simulare a stocurilor

Pentru a vizualiza evoluția stocurilor în timp în condiții variabile de cerere, am implementat următoarea funcție:

```
media <- mean(produs1_m1)
devst <- sd(produs1_m1)

prag <- 5
k <- (media - prag) / devst
prob_cebisev <- 1 / (k^2)
cat("Probabilitatea (maximă) ca cererea să scadă sub", prag, "unități:", prob_cebisev,
"\n")


stoc <- produs1_m1
mean_stoc <- mean(stoc)
mean_stoc_squared <- mean(stoc^2)

cat("E[stoc^2] =", mean_stoc_squared, "\n")
cat("(E[stoc])^2 =", mean_stoc^2, "\n")



simuleaza_stocuri <- function(stoc_initial = 500, zile_simulate = 30, lambda_cerere =
15) {
 cerere_zilnica <- rpois(zile_simulate, lambda = lambda_cerere)
 stoc <- stoc_initial
 stocuri <- numeric(zile_simulate)

 for (zi in 1:zile_simulate) {
  cerere <- cerere_zilnica[zi]
  if (stoc >= cerere) {
   stoc <- stoc - cerere
  } else {
   stoc <- 0  # stoc epuizat
  }
  stocuri[zi] <- stoc
```

```
  }

  plot(1:zile_simulate, stocuri, type = "o", col = "darkred", lwd = 2,
       xlab = "Ziua", ylab = "Stoc rămas", main = "Evoluția stocului zilnic")
  abline(h = 0, col = "blue", lty = 2)
  cat("Zile fără stoc:", sum(stocuri == 0), "din", zile_simulate, "\n")
}
```

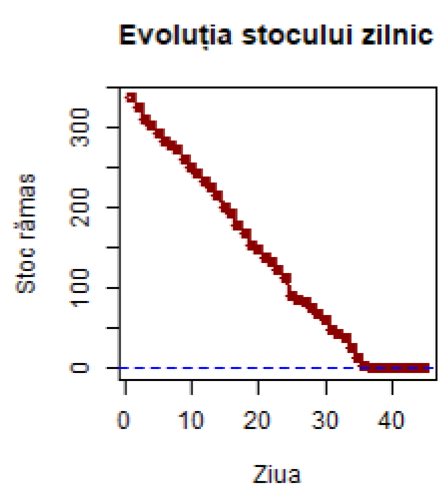
Exemplu de apel:

```
simuleaza_stocuri(lambda_cerere = 10, stoc_initial = 350, zile_simulate = 45)
```

```
> simuleaza_stocuri(lambda = 10, stoc_initial = 350, zile = 45)
Zile fără stoc: 9 din 45
```

Interpretare:

- Graficul generat ilustrează **evoluția stocului în timp**.
- Dacă linia roșie atinge linia albastră (zero), înseamnă că **stocul a fost epuizat**.
- Mesajul text arată **numărul de zile în care nu am avut stoc**, ceea ce semnalează riscuri operaționale



Dificultăți Întâmpinate

Pe parcursul realizării acestui proiect, am întâmpinat mai multe provocări, atât tehnice, cât și conceptuale:

Interpretarea graficelor și figurilor statistice a reprezentat una dintre principalele dificultăți. Deși generarea acestora în R este relativ simplă, înțelegerea semnificației lor – de exemplu, interpretarea unui heatmap sau a unei matrice de corelații – a necesitat documentare suplimentară și o mai bună înțelegere a contextului datelor.

Crearea funcției interactive de simulare a implicat scrierea unui cod robust care să reflecte corect logica din lumea reală (ex. epuizarea stocurilor în funcție de cerere), dar și alegerea parametrilor astfel încât rezultatele să fie realiste și semnificative.

Universitatea Din București
Facultatea de Matematică și Informatică
Departamentul Informatică
Specializarea Calculatoare și Tehnologia Informației

Capitolul 7 – Concluzii și recomandări finale

Acest proiect a oferit o perspectivă practică asupra modului în care modelele probabilistice și instrumentele statistice pot fi utilizate pentru optimizarea gestiunii stocurilor într-un lanț de aprovizionare.

Analiza distribuțiilor cererii a evidențiat variațiile semnificative între magazine, aspect esențial pentru o alocare eficientă a resurselor.

Prin aplicarea **Inegalității lui Jensen**, am evaluat **volatilitatea stocului**, evidențiind faptul că o variabilitate ridicată a cererii poate duce la costuri de depozitare mai mari.

Estimarea riscurilor folosind **Inegalitatea lui Hoeffding** ne-a permis să evaluăm probabilitatea ca cererea totală să depășească stocul disponibil, aducând un cadru conservator de decizie în condiții incerte.

Vizualizările statistice – histograme, corelații și heatmap-uri – au oferit o **înțelegere intuitivă** a datelor, permițând identificarea unor corelații între produse și comportamente comune de cerere.

Funcția de **simulare interactivă a stocului** a fost extrem de utilă în testarea unor scenarii diverse de cerere și în evaluarea **riscului de epuizare a stocurilor**, demonstrând potențialul utilizării R în decizii operaționale.

În concluzie, proiectul a demonstrat cum **metodele statistice și simulările** pot sprijini decizii mai bune în gestiunea stocurilor, reducând riscurile și optimizând resursele disponibile.