СЕМ практикум Домашно задание

Теодор Костадинов 4MI0600097

2024-01-10

Задача 1

На първия етаж на административна сграда 7 души чакат асансьора. Всеки от тях отива в някой от офисите в сградата. Сградата има 16 етажа и на всеки етаж има равен брой офиси (на първия етаж няма офиси). С помощта на симулации отговорете:

а) Каква е вероятността поне двама от чакащите да отиват на един и същи етаж?

б) Ако Иван е един от седемте, каква е вероятността поне един от останалите шест да отива на етажа, на който отива Иван?

Решение:

а)

Симулираме взимане с връщане на елемент от вектор с етажите за всеки един от хората. Проверяваме дали има дупликати във взетите данни. Повтаряме опита много пъти и взимаме пропорцията.

```
set.seed(777)

floors = 2:16
people = 7
Nrep = 100000

task17.experiment.a = function(){
  floors.choosen = sample(floors, people, replace = T)
  anyDuplicated(floors.choosen) > 0
}

task14.prob.a = function(){
  output = replicate(Nrep, task17.experiment.a())
  sum(output) / length(output)
}

task14.prob.a()
```

```
## [1] 0.80859
```

Алтернативно можем да сметнем реалната пропорция:

```
1 - (factorial(15)/factorial(15-7))/15^7
```

```
## [1] 0.8101807
```

б)

Отново симулираме кой на кой етаж отива. Избираме Иван да е първият симулиран етаж и търсим елемент равен на първия в останалата част от генерирани етажи.

```
task17.experiment.b = function(){
  floors = sample(floors, people, replace = T)
  sum(floors[1] == floors) > 1
}

task14.prob.b = function(){
  output = replicate(Nrep, task17.experiment.b())
  sum(output) / length(output)
}

task14.prob.b()
```

```
## [1] 0.33963
```

Алтернативно решение без симулации - от останалите 6 човека поне един да е при нас:

```
1 - dbinom(0, 6, 1/15)
```

```
## [1] 0.3389708
```

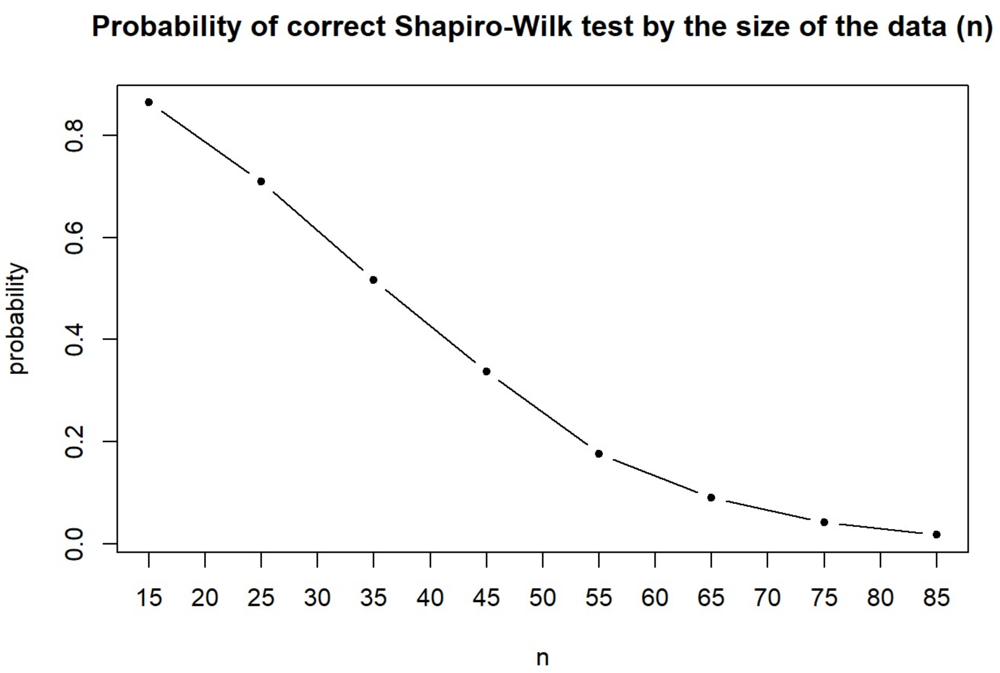
Задача 2

Генерирайте данни *x1, x2, …, xn* от равномерно разпределение в интервала *(4, 5)*. Проверете хипотезата, че данните са от нормално разпределение с помощта на теста на Шапиро–Уилк (Shapiro-Wilk). Повторете *10000* пъти за *n = 15, 25, 35, 45, 55, 65, 75, 85*. Колко често заключението на теста е вярно? Представете чрез подходяща графика честотата на вярно заключение в зависимост от *n*.

Решение:

- Генерирането на данни от равномерно разпределение за дадено *n* и проверката на хипотезата се изпълняват от функцията *task2.experiment*.
- Повторението на единичния експетимент за конкретно *n* се случва вфункцията *task2.replicate*.
- Изчисляването на резултатите за множество от стойности *n* при големи повторения се случва в *task2.graph*. След като резултатите бъдат получени, функцията визуализира данните чрез подходяща графика.

```
a = 4
b = 5
alpha = 0.05
Nrep = 10^4
n.values = seq(15, 85, 10)

task2.experiment = function(n) {
  x = runif(n, a, b)
  shapiro.test(x)$p.value > alpha
}

task2.replicate = function(n) {
  result = replicate(Nrep, task2.experiment(n))
  sum(result) / Nrep
}

task2.graph = function(n.values) {
  probabilities = sapply(n.values, task2.replicate)
  df = data.frame(n = n.values, probability = probabilities)
  plot(df, type = 'b', pch = 20,
       main = "Probability of correct Shapiro-Wilk test by the size of the data (n)",)
  axis(1, at = n.values, labels = n.values)
  df
}

task2.graph(n.values)
```



```
##    n probability
## 1 15      0.8638
## 2 25      0.7087
## 3 35      0.5173
## 4 45      0.3375
## 5 55      0.1767
## 6 65      0.0900
## 7 75      0.0421
## 8 85      0.0178
```

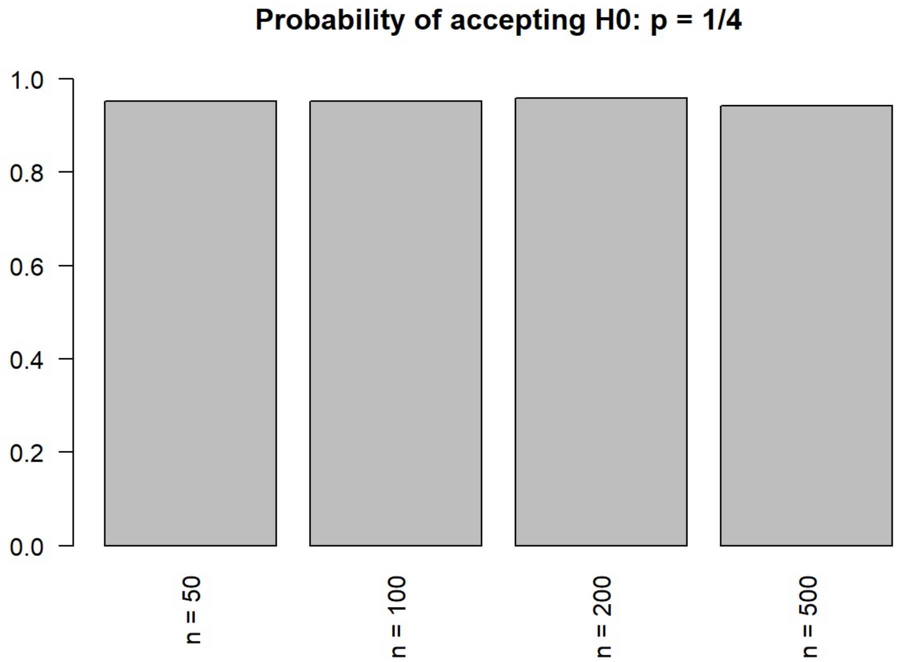
Задача 3

В кутия има 2 зелени и 6 бели топки. Симулирайте *n* теглениа на топка с връщане. Използвайте генерираните данни, за да проверите хипотезата, че вероятността да изтеглим зелена топка е *1/4*. Повторете 10000 пъти за *n = 50, 100, 200, 500*. Колко често заключението на теста е вярно? Представете чрез подходяща графика честотата на вярно заключение в зависимост от *n*.

Решение:

- Генерираме *n* тегления от кутията с връщане. Проверяваме кои от тях са желаните зелени - функцията *task3.experiment*.
- Изпълняваме *z-test* за пропорция на получената извадка с функцията *task3.test*. Проверяваме дали нулевата хипотеза може да бъде отхвърлена.
- Повтаряме проверката на хипотезата 10000 пъти с функцията *task3.prob* и изчисляваме пропорцията на приета хипотеза спрямо всички опити.
- Във функцията *task3.graph* използваме множество от стойности, за да проверим резултатите при намирането на пропорцията за различни *n*, подадени на първоначалната функция *task3.experiment*. Получените резултати показваме с подходяща графика.

```r
n.values = c(50, 100, 200, 500)
Nrep = 10000
p0 = 1/4

task3.experiment = function(n) {
  balls.all = c(1, 1, 0, 0, 0, 0, 0, 0)
  balls.drawn = sample(balls.all, n, replace = T)
  sum(balls.drawn == 1)
}

task3.test = function(n) {
  x = task3.experiment(n)
  z.obs = (x/n - p0)/sqrt(p0*(1 - p0)/n)
  p.value = 2*(1 - pnorm(abs(z.obs)))
  p.value > 0.05
}

task3.prob = function(n) {
  results = replicate(Nrep, task3.test(n))
  sum(results) / Nrep
}

task3.graph = function(n.values) {
  proportions = sapply(n.values, task3.prob)
  names(proportions) = paste("n =", n.values)

  barplot(proportions, ylim = c(0,1), las = 2,
          main = "Probability of accepting H0: p = 1/4")
  proportions
}

task3.graph(n.values)
```




```
##  n = 50 n = 100 n = 200 n = 500
##  0.9519  0.9523  0.9591  0.9429
```