

Домашно задание по СЕМ практикум

Задача 1

В тесте от карти има маркирани. Тестето е разбъркано и раздадено на играчи. Чрез подходяща симулация намерете приближение на вероятността на събитието “на всеки играч се пада една маркирана карта”.

```
simulate_round <- function() {  
  cards <- c(rep(1, 4), rep(0, 28))  
  cards <- sample(cards)  
  piles <- split(cards, rep(1:4, each = 8))  
  result <- sapply(piles, function(pile) sum(pile) == 1)  
  return(all(result))  
}  
  
n <- 10000  
successes <- 0  
for (i in 1:n) {  
  if (simulate_round()) {  
    successes <- successes + 1  
  }  
}  
  
probability <- successes / n  
print(probability)
```

```
R 4.2.2 ~/  
+ # Check if each pile contains one ace  
+ result <- sapply(piles, function(pile) sum(pile) == 1)  
+ # Return TRUE if all piles contain one ace, FALSE otherwise  
+ return(all(result))  
+ }  
> # Set the number of simulations to run  
> n <- 10000  
>  
> # Initialize a counter for successful rounds  
> successes <- 0  
>  
> # Run the simulations  
> for (i in 1:n) {  
+   if (simulate_round()) {  
+     successes <- successes + 1  
+   }  
+ }  
>  
> # Calculate the probability of success  
> probability <- successes / n  
> print(probability)  
[1] 0.1095
```

Задача 2

В кутия има 5 зелени, 7 сини и 7 жълти топки. Симулирайте n тегления на топка с връщане. Проверете хипотезата, че в кутията има еднакъв брой топки от трите цвята. Повторете пъти 10000 за $n = 50, 100, 200, 500, 1000$. Колко често заключението на теста е вярно? Представете чрез подходяща графика честотата на вярно заключение в зависимост от n .

```
install.packages("tidyverse")
install.packages("broom")

library(tidyverse)
library(broom)

simulate_chi_square <- function(n, green, blue, yellow) {
  colors <- c(rep("green", green), rep("blue", blue), rep("yellow", yellow))

  counts <- table(sample(colors, n, replace = TRUE))

  test_result <- chisq.test(counts)

  p_value <- test_result$p.value

  return(p_value)
}

simulate_chi_square_repeatedly <- function(n, green, blue, yellow, repetitions) {

  p_values <- rep(NA, repetitions)

  for (i in 1:repetitions) {
    p_values[i] <- simulate_chi_square(n, green, blue, yellow)
  }

  frequency <- mean(p_values > 0.05)

  return(frequency)
}

repetitions <- 10000

green <- 5
blue <- 7
yellow <- 7
```

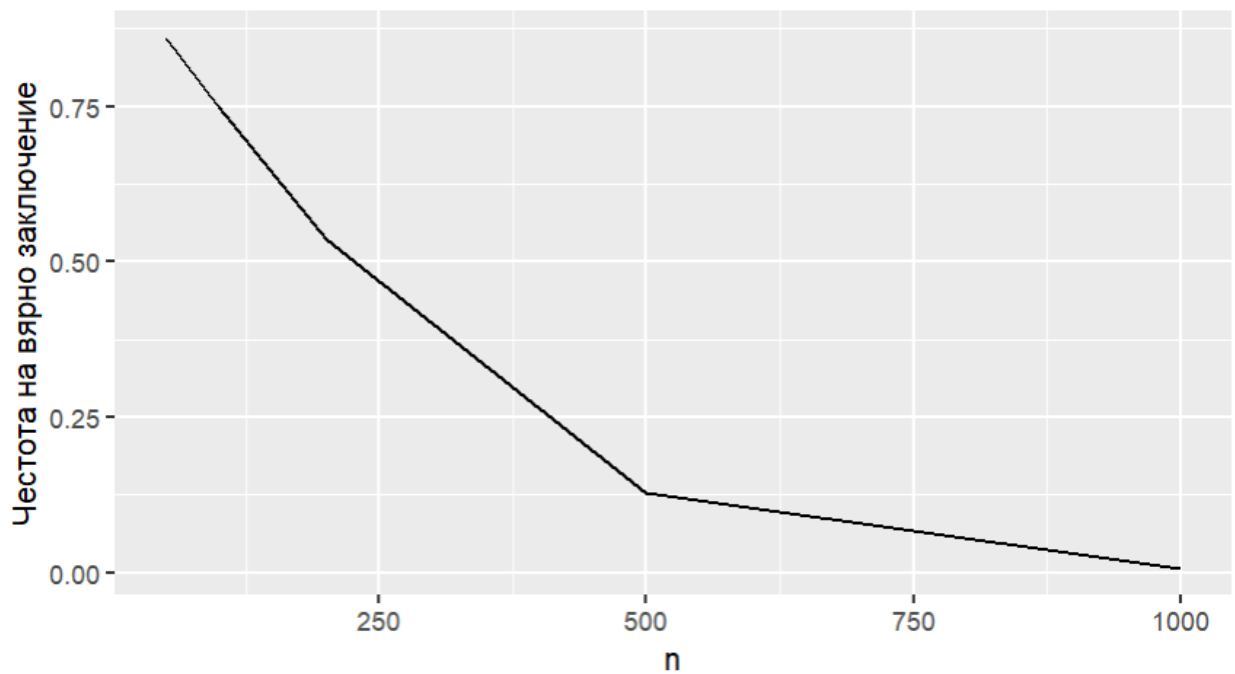
```
ns <- c(50, 100, 200, 500, 1000)

frequencies <- rep(NA, length(ns))

for (i in 1:length(ns)) {
  frequencies[i] <- simulate_chi_square_repeatedly(ns[i], green, blue, yellow, repetitions)
}

results <- data.frame(n = ns, frequency = frequencies)

ggplot(results, aes(x = n, y = frequency)) +
  geom_line() +
  xlab("n") +
  ylab("Честота на вярно заключение")
```



Задача 3

В кутия има 5 зелени, 5 сини и 5 жълти топки. Симулирайте n тегления на топка с връщане. Проверете хипотезата, че в кутията има еднакъв брой топки от трите цвята. Повторете пъти 10000 за $n = 50, 100, 200, 500, 1000$. Колко често заключението на теста е вярно? Представете чрез подходяща графика честотата на вярно заключение в зависимост от n .

```
install.packages("tidyverse")
install.packages("broom")

library(tidyverse)
library(broom)

simulate_chi_square <- function(n, green, blue, yellow) {
  colors <- c(rep("green", green), rep("blue", blue), rep("yellow", yellow))

  counts <- table(sample(colors, n, replace = TRUE))

  test_result <- chisq.test(counts)

  p_value <- test_result$p.value

  return(p_value)
}

simulate_chi_square_repeatedly <- function(n, green, blue, yellow, repetitions) {

  p_values <- rep(NA, repetitions)

  for (i in 1:repetitions) {
    p_values[i] <- simulate_chi_square(n, green, blue, yellow)
  }

  frequency <- mean(p_values > 0.05)

  return(frequency)
}

repetitions <- 10000

green <- 5
blue <- 5
yellow <- 5
```

```

ns <- c(50, 100, 200, 500, 1000)

frequencies <- rep(NA, length(ns))

for (i in 1:length(ns)) {
  frequencies[i] <- simulate_chi_square_repeatedly(ns[i], green, blue, yellow, repetitions)
}

results <- data.frame(n = ns, frequency = frequencies)

ggplot(results, aes(x = n, y = frequency)) +
  geom_line() +
  xlab("n") +
  ylab("Честота на вярно заключение")

```

