SA1

Paul Joaquin M. Delos Santos
Kiana Marie D. Dizon

2024-03-10

```
knitr::opts_chunk$set(echo = TRUE)
library(ggplot2)
```

```
## Warning: package 'ggplot2' was built under R version 4.3.3
```

Question (3)

By generating 10,000 searches in R, carry out a simulation experiment for a search engine going through a list of sites for a given key phrase, until the key phrase is found. You should allow your program to input the probability p that any site will contain the key phrase.

a. Plot the simulated pdf and calculate its mean and variance.

b. Obtain the simulated conditional distribution of searches when three searches have been carried out without success. Calculate its mean and variance, and satisfy yourself that they are equivalent to the simulated distribution of the complete set.

As test data assume each site has a 60% chance of containing the key phrase. To satisfy yourself that the Markov memoryless property holds, obtain estimates of

a. P(X = 4|X > 3) and P(X = 1)

b. P(X = 5|X > 3) and P(X = 2)

where X is the number of searches to the first success

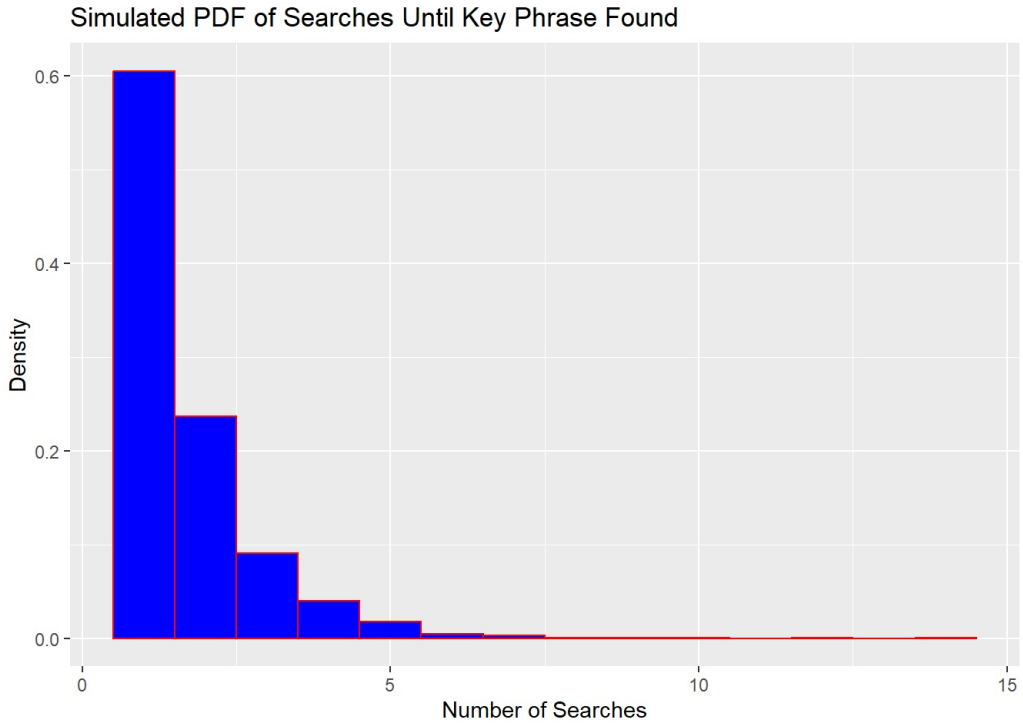
```
# Correct simulation using the geometric distribution
set.seed(42) # For reproducibility
n <- 10000 # Number of simulations
p <- 0.6 # Probability a site contains the key phrase

# Simulate 10,000 searches with the geometric distribution
# Adding 1 because rgeom starts counting at 0
searches <- rgeom(n, p) + 1
```

(A) Simulated PDF, Mean, and Variance

Plotting the PDF

```
# Plotting the simulated PDF
ggplot(data.frame(searches), aes(x=searches)) +
  geom_histogram(aes(y=after_stat(density)), binwidth = 1, color="red", fill="blue") +
  ggtitle("Simulated PDF of Searches Until Key Phrase Found") +
  xlab("Number of Searches") +
  ylab("Density")
```



Calculating Mean and Variance

```
mean_searches <- mean(searches)
var_searches <- var(searches)

mean_searches
```

```
## [1] 1.6627
```

```
var_searches
```

```
## [1] 1.129442
```

(B) Simulated Conditional Distribution After Three Unsuccessful Searches

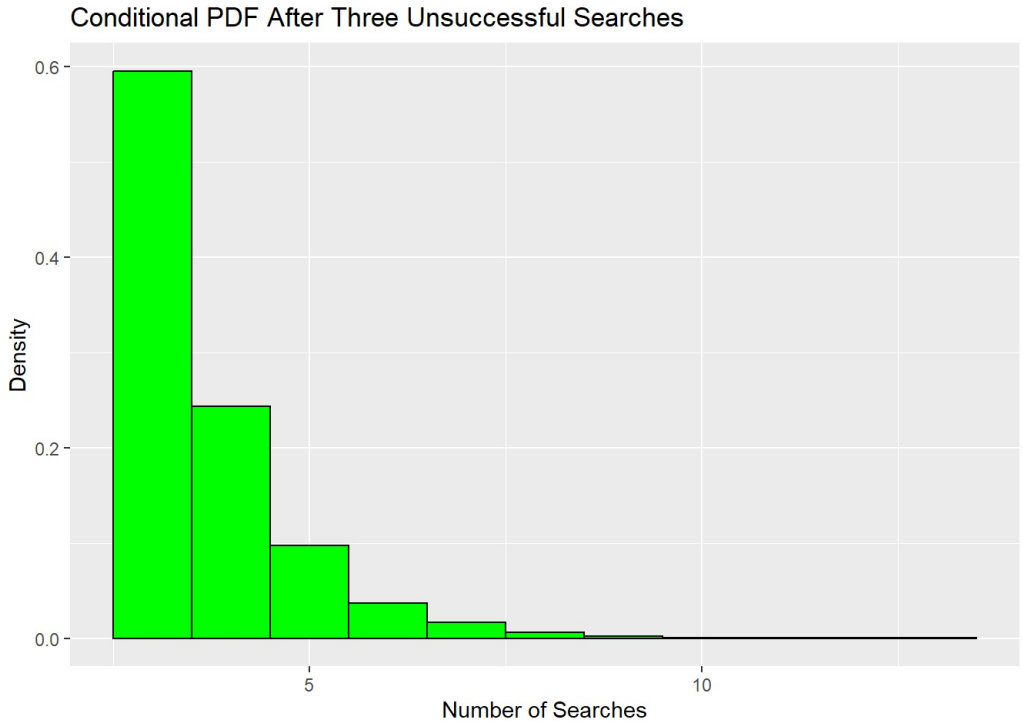
Adjusting the Simulation

```
# Simulating with the condition of 3 unsuccessful searches
searches_after_three_fails <- numeric(n)

for (i in 1:n) {
  # Use rgeom for the conditional scenario after 3 unsuccessful searches
  searches_after_three_fails[i] <- rgeom(1, p) + 3
}
```

Plotting the Conditional PDF

```
# Plotting the PDF for the conditional scenario
ggplot(data.frame(searches_after_three_fails), aes(x=searches_after_three_fails)) +
  geom_histogram(aes(y=after_stat(density)), binwidth = 1, color="black", fill="green") +
  ggtitle("Conditional PDF After Three Unsuccessful Searches") +
  xlab("Number of Searches") +
  ylab("Density")
```



Calculating Mean and Variance

```
mean_searches_conditional <- mean(searches_after_three_fails, na.rm = TRUE)
var_searches_conditional <- var(searches_after_three_fails, na.rm = TRUE)

mean_searches_conditional
```

```
## [1] 3.675
```

```
var_searches_conditional
```

```
## [1] 1.124287
```

Verifying the Markov Memoryless Property

Calculating Probabilities

(a) P(X = 4|X > 3) and P(X = 1)

```
# For X = 4|X > 3, it's essentially the probability of success on the first trial
p_x_equals_4_given_greater_than_3 <- p # P(X=4|X>3)

# For X = 1, since the first success is with probability p
p_x_equals_1 <- p # P(X=1)

p_x_equals_4_given_greater_than_3
```

```
## [1] 0.6
```

```
p_x_equals_1
```

```
## [1] 0.6
```

(b) P(X = 5|X > 3) and P(X = 2)

```
# For X = 5|X > 3, it's also the probability of success on the first trial after 3 failures
p_x_equals_5_given_greater_than_3 <- p # P(X=5|X>3)

# For X = 2, the probability of first failure then success
p_x_equals_2 <- p * (1 - p) # P(X=2)

p_x_equals_5_given_greater_than_3
```

```
## [1] 0.6
```

```
p_x_equals_2
```

```
## [1] 0.24
```