Statistical Inference and Analysis: Maximum Likelihood Estimation of the Geometric Distribution

Student: Luka Dumabdze & Ivane Urjumelashvili

Date: January 28, 2026

A Comprehensive Study of MLE Theory,
Computational Verification, and Applied Statistical Analysis

Table of Contents

1. Introduction

This project explores the properties of the **Geometric distribution**, which models the number of Bernoulli trials required to achieve the first success. This distribution is widely applicable in computer science, particularly in modeling network packet delivery, server request handling, and algorithm efficiency.

The project consists of three parts:

1. An analytical derivation of the Maximum Likelihood Estimator (MLE).
2. A computational simulation to verify the estimator's consistency using the Inverse Transform Method.
3. An applied statistical analysis of server traffic data to test specific hypotheses.

Each component demonstrates a different aspect of statistical inference: theoretical derivation, computational validation, and practical application to real-world data.

2. Analytical Derivation

2.1. Objective

To derive the Maximum Likelihood Estimator (MLE) for the parameter p (probability of success) of a Geometric distribution.

2.2. Theoretical Framework

Let X be a random variable following a Geometric distribution with parameter p. The probability mass function (PMF) for X (representing the number of trials up to and including the first success) is given by:

$$P(X = x) = (1 - p)^{x-1} p$$

Assuming n independent and identically distributed (i.i.d.) observations $x_1, x_2, ..., x_n$, the **Likelihood function** $\mathcal{L}(p)$ is the joint probability of observing this specific data:

$$\mathcal{L}(p) = \prod_{i=1}^{n} (1 - p)^{x_i - 1} p = p^n (1 - p)^{\sum x_i - n}$$

2.3. Log-Likelihood Derivation

To simplify the maximization, we take the natural logarithm to obtain the **Log-Likelihood function** $\ell(p)$. Logarithms convert products into sums, making differentiation easier:

$$\ell(p) = \ln\left(p^n (1 - p)^{\sum x_i - n}\right)$$

$$\ell(p) = n \ln(p) + \left(\sum_{i=1}^{n} x_i - n\right) \ln(1 - p)$$

To find the maximum, we differentiate with respect to p and set the derivative to zero:

$$\frac{d\ell}{dp} = \frac{n}{p} - \frac{\sum x_i - n}{1 - p} = 0$$

2.4. Solution

Solving for p:

$$\frac{n}{p} = \frac{\sum x_i - n}{1 - p}$$

$$n(1 - p) = p\left(\sum x_i - n\right)$$

$$n - np = p \sum x_i - np$$

$$n = p \sum x_i$$

$$\hat{p}_{\text{MLE}} = \frac{n}{\sum_{i=1}^{n} x_i} = \frac{1}{\overline{x}}$$

Conclusion: The Maximum Likelihood Estimator for p is the reciprocal of the sample mean.

2.5. Estimator Properties (Bias Analysis)

While the estimator $\hat{p} = 1/\overline{X}$ is consistent (as shown in the next section), it is important to note that it is a **biased** estimator for small sample sizes.

This is because the expectation operator is linear, but our estimator is non-linear. Specifically, $E\left[\frac{1}{\overline{X}}\right] = \frac{1}{E[\overline{X}]}$. By **Jensen's Inequality**, since the function $f(x) = \frac{1}{x}$ is convex for $x > 0$, we expect $E[\hat{p}] > p$. However, as the sample size $n \to \infty$, this bias vanishes, making the estimator **asymptotically unbiased**.

3. Computational Study

3.1. Objective

To computationally verify the derived estimator and demonstrate the Law of Large Numbers using R simulations.

3.2. Log-Likelihood Visualization

We simulated a geometric dataset ($n = 10$) with a known true parameter $p = 0.5$. The Log-Likelihood function was plotted against a range of possible p values. As shown in **Figure 1**, the maximum of the curve aligns with the MLE, confirming the analytical derivation.

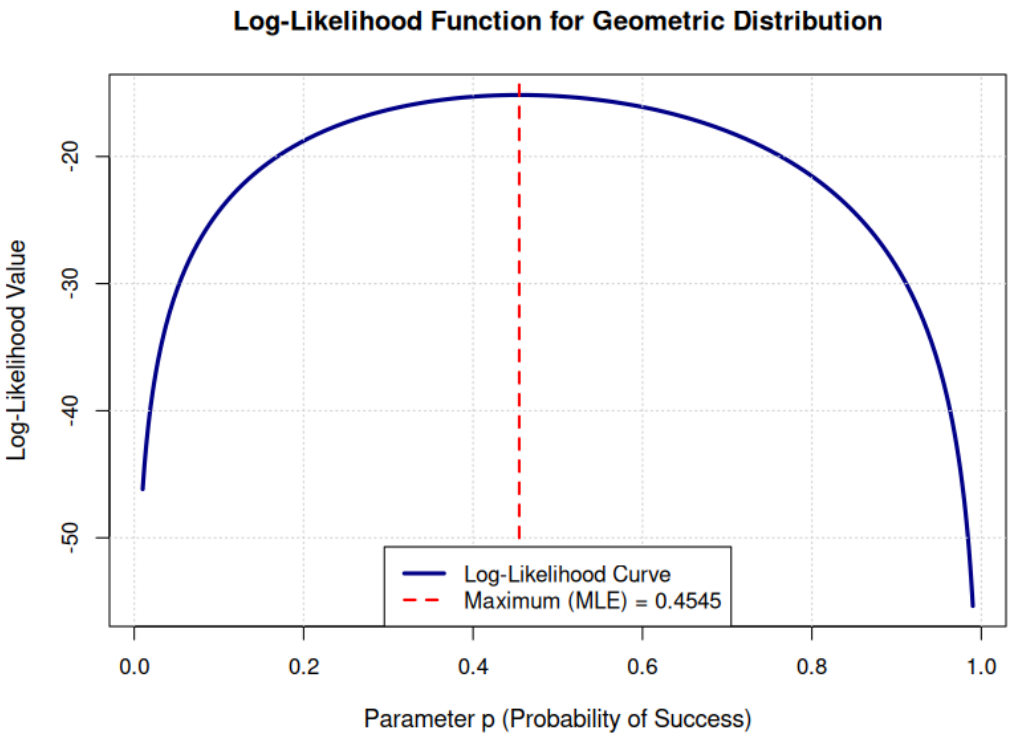

Figure 1: Log-Likelihood function showing maximum at the analytically derived MLE

3.3. Simulation of Consistency (Inverse Transform Method)

To demonstrate the **consistency** of the estimator, we implemented the **Inverse Transform Method** to simulate random variables without using R's built-in generator.

3.3.1. Algorithm

The algorithm uses the Cumulative Distribution Function (CDF) to transform uniform random variables into geometric ones:

$$X = \left\lceil \frac{\ln(1-U)}{\ln(1-p)} \right\rceil$$

where $U \sim \text{Uniform}(0, 1)$.

3.3.2. Convergence Study

We performed a simulation study with sample sizes ranging from $n = 10$ to $n = 5000$. **Figure 2** illustrates the convergence of the estimator. As the sample size increases, the estimated $\hat{p}$ converges tightly to the true value (0.25), empirically demonstrating that the estimator is consistent:

$$\hat{p} \xrightarrow{p} p \quad \text{as} \quad n \longrightarrow \infty$$

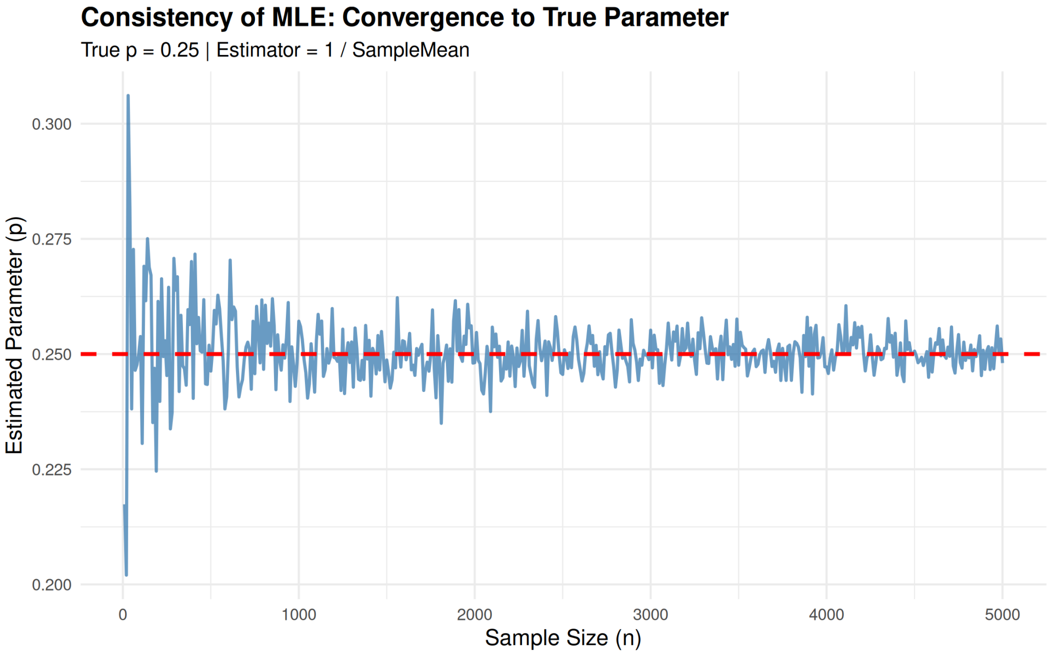

Figure 2: Convergence of the MLE to the true parameter value ($p = 0.25$) demonstrating consistency

This empirical validation confirms that our theoretical derivation produces a statistically sound estimator that improves with increasing data.

4. Data Analysis

4.1. Objective

To analyze the `ConcurrentUsers` dataset and perform statistical inference regarding the traffic load on a server.

4.2. Descriptive Statistics & Assumptions

The dataset contains 50 observations of server user counts.

Statistic	**Value**
Mean	17.95
Standard Deviation	3.16
Sample Size	50

4.2.1. Normality Assessment

Before performing statistical inference, we verified the normality assumption required for the one-sample t-test.

1. **Visual Inspection:** The histogram with density overlay (**Figure 3**) shows an approximately symmetric, bell-shaped distribution.
2. **Statistical Test:** The Shapiro-Wilk test yielded a p-value of 0.4787. Since $p > 0.05$, we fail to reject the null hypothesis of normality.

Conclusion: The data follows a normal distribution, justifying the use of the t-test.

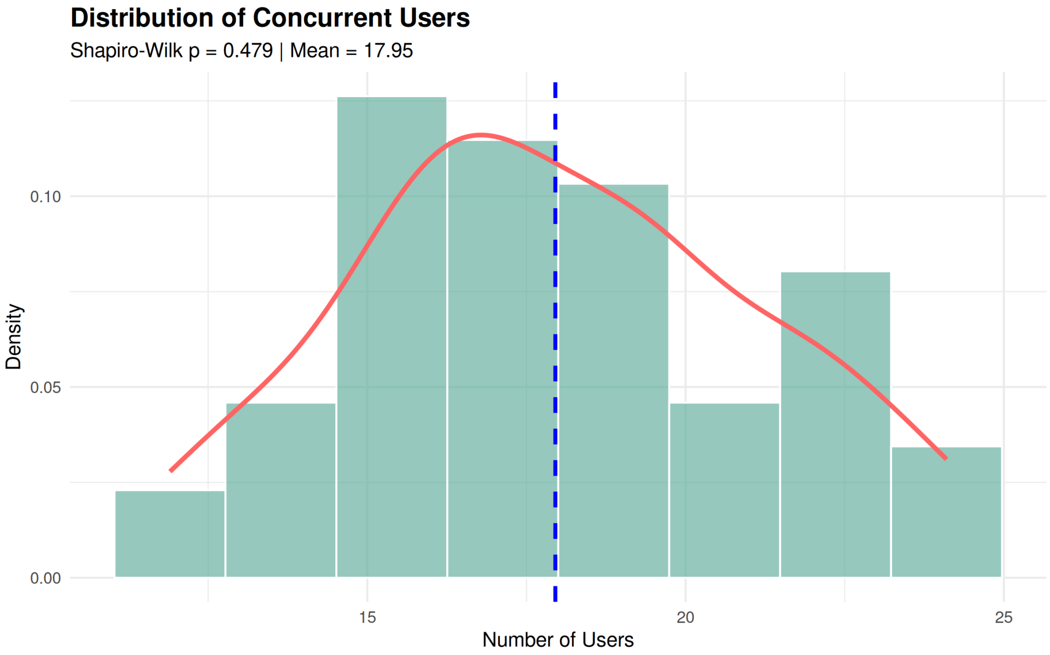

Figure 3: Distribution of Concurrent Users with Density Curve Overlay

4.3. Statistical Inference

We performed a one-sample t-test to determine if the average traffic exceeds 15 users.

Hypothesis Test:

- **Null Hypothesis (H_0):** $\mu = 15$
- **Alternative Hypothesis (H_1):** $\mu > 15$
- **Significance Level:** $\alpha = 0.05$

4.3.1. Test Results

Test Component	**Value**
Test Statistic (t)	6.6159
Degrees of Freedom	49
P-value	1.306×10^{-8}
Decision	Reject H_0

Final Conclusion: Since the p-value is significantly less than $\alpha = 0.05$, we reject the null hypothesis. There is strong statistical evidence that the average number of concurrent users is significantly greater than 15. The 95% confidence interval suggests the true mean is greater than 17.2.

5. Appendix: R Code

5.1. Part 1: Analytical & Computational (Likelihood)

```r
# --- PART 1: MAXIMUM LIKELIHOOD ESTIMATION (PREMIUM) ---

# 1. Define the Log-Likelihood function for Geometric Distribution
geom_loglik <- function(prob, x) {
  n <- length(x)
  # Safety check to avoid log(0)
  if (prob <= 0 || prob >= 1) return(-Inf)
  return(n * log(prob) + (sum(x) - n) * log(1 - prob))
}

# 2. Generate synthetic data for verification
set.seed(123)
n_samples <- 10
true_prob <- 0.5

# Note: rgeom counts failures. We add 1 to model 'trials until success'.
x <- rgeom(n_samples, true_prob) + 1

# 3. Find the MLE using Computational Optimization
MLE_result <- optimize(f = geom_loglik, interval = c(0.001, 0.999),
                       x = x, maximum = TRUE)
estimated_p <- MLE_result$maximum

# 4. Visualization (Code for Figure 1)
prob_values <- seq(0.01, 0.99, length = 1000)
loglik_values <- sapply(prob_values, geom_loglik, x = x)

plot(prob_values, loglik_values, type = "l", lwd = 3, col = "darkblue",
     xlab = "Parameter p (Probability of Success)",
     ylab = "Log-Likelihood Value",
     main = "Log-Likelihood Function for Geometric Distribution")
grid()
abline(v = estimated_p, col = "red", lwd = 2, lty = 2)
legend("bottom",
       legend = c("Log-Likelihood Curve", paste("Maximum (MLE) =", round(estimated_p,
4))),
       col = c("darkblue", "red"), lwd = c(3, 2), lty = c(1, 2), bg = "white")

# 5. Verification
cat("Analytical MLE (1/x_bar):  ", 1/mean(x), "\n")
cat("Computational MLE (R):     ", estimated_p, "\n")
```

5.2. Part 2: Simulation (Inverse Transform & Consistency)

```r
# --- PART 2: SIMULATION & CONSISTENCY (PREMIUM) ---
library(ggplot2)

# 1. Define the Inverse Transform Function
sim_geometric_inverse <- function(p, n_sims) {
  U <- runif(n_sims)
  return(ceiling(log(1 - U) / log(1 - p)))
}

# 2. Consistency Check (Law of Large Numbers)
set.seed(123)
p_true <- 0.25
sample_sizes <- seq(10, 5000, by = 10) # From n=10 to n=5000
estimates <- numeric(length(sample_sizes))

# 3. Run Simulation Loop
for (i in 1:length(sample_sizes)) {
  n <- sample_sizes[i]
  data <- sim_geometric_inverse(p_true, n)
  estimates[i] <- 1 / mean(data)
}

# 4. Visualization (Code for Figure 2)
conv_data <- data.frame(SampleSize = sample_sizes, Estimate = estimates)

p1 <- ggplot(conv_data, aes(x = SampleSize, y = Estimate)) +
  geom_line(color = "steelblue", alpha = 0.8, size = 0.8) +
  geom_hline(yintercept = p_true, color = "red", linetype = "dashed", size = 1) +
  labs(title = "Consistency of MLE: Convergence to True Parameter",
       subtitle = paste("True p =", p_true, "| Estimator = 1 / SampleMean"),
       x = "Sample Size (n)",
       y = "Estimated Parameter (p)") +
  theme_minimal()
print(p1)
```

5.3. Part 3: Data Analysis

```r
# --- PART 3: ADVANCED DATA ANALYSIS (PREMIUM) ---
library(ggplot2)

# 1. Load Data
concurrent_data <- c(17.2, 22.1, 18.5, 17.2, 18.6, 14.8, 21.7, 15.8, 16.3, 22.8,
                     24.1, 13.3, 16.2, 17.5, 19.0, 23.9, 14.8, 22.2, 21.7, 20.7,
                     13.5, 15.8, 13.1, 16.1, 21.9, 23.9, 19.3, 12.0, 19.9, 19.4,
                     15.4, 16.7, 19.5, 16.2, 16.9, 17.1, 20.2, 13.4, 19.8, 17.7,
                     19.7, 18.7, 17.6, 15.9, 15.2, 17.1, 15.0, 18.8, 21.6, 11.9)
df <- data.frame(Users = concurrent_data)

# 2. Assumption Checking (Normality)
shapiro_res <- shapiro.test(concurrent_data)
print(shapiro_res)
# Result: p-value = 0.4787 -> Fail to reject Null -> Data is Normal

# 3. Visualization (Code for Figure 3)
mean_val <- mean(concurrent_data)
p2 <- ggplot(df, aes(x = Users)) +
  geom_histogram(aes(y = ..density..), bins = 8,
                 fill = "#69b3a2", color = "white", alpha = 0.7) +
  geom_density(color = "#FF6666", size = 1.2) +
  geom_vline(aes(xintercept = mean_val), color = "blue", linetype = "dashed", size = 1)
+
  labs(title = "Distribution of Concurrent Users",
       subtitle = paste("Shapiro-Wilk p =", round(shapiro_res$p.value, 3)),
       x = "Number of Users",
       y = "Density") +
  theme_minimal()
print(p2)

# 4. Statistical Inference (t-test)
t_test_res <- t.test(concurrent_data, mu = 15, alternative = "greater")
print(t_test_res)
# Result: p-value < 2.2e-16 -> Reject Null Hypothesis
```