



Master of Statistics and Data Science

Principles of Statistical Inference

Home Work I

2023-2024

**Harvey Nyerovwo Okotie (2262068)**

*Submission Date:* 17-10-2023

**Lecturer:**

Prof. dr. Anneleen Verhasselt

---

Let  $X$  be a random binomial variable with  $n = 10$  and  $p \in (0; 1)$  to be estimated. The following 15 independent observations were taken from this distribution:

7, 1, 5, 1, 3, 3, 3, 3, 6, 5, 4, 5, 3, 2, 4

1. Find the method of moment estimate for  $p$ .

**Answer:**

Given:

$$f(x|p) = \binom{n}{x} P^x (1 - P)^{n-x}$$

The analytical approach of the Method of Moment for Binomial distribution can be expressed as:

$$E(x) = \hat{\mu}_1 = \bar{X} = \sum_{i=1}^n \frac{x_i}{n}$$

Therefore, the method of moment estimate of  $p$  is  $\hat{p} = \bar{X}$

```
n <- 10
sample <- c(7, 1, 5, 1, 3, 3, 3, 3, 6, 5, 4, 5, 3, 2, 4)
MoM_estimate <- mean(sample)
MoM_estimate
[1] 3.666667
```

The numerical approach of the method of moment using bootstrap to sample the data from a Binomial distribution with  $n =$  the number of sample and  $p =$  the probability of success. For each sample, we calculate the mean of the bootstrap sample as follows:

```
set.seed(1111)
# calculate the probability
p <- (MoM_estimate/n)
n_sim <- 1000
np_Mom <- as.vector(n_sim)

for (i in 1:n_sim){
```

---

```
sample_sim <- rbinom(n_sim, n, p)
np_Mom[i] <- mean(sample_sim)
}
```

```
mean_Mom <- mean(np_Mom)
mean_Mom
[1] 3.664637
```

It can be observed that the estimate result between the analytical and numerical method are quite close.

2. Find the standard error for your method of moment estimate using bootstrap.

**Answer:**

The standard error for the Method of Moment estimate can be expressed as taking the standard deviation of the bootstrap method of moment estimates.

```
# Standard Error for the Method of Moment
SE_Mom <- sd(np_Mom)
SE_Mom
[1] 0.04887175
```

3. Plot the log likelihood function as a function of p.

**Answer:** The (log)likelihood can be expressed as the sum of the log of the marginal densities. Which involves the combination of the observed data and the probability for each trial. To calculate the likelihood, we take the product of Binomial distribution as follows:

```
# likelihood function
log_lk <- function(prob, sample, n){
  sum(dbinom(sample, size = n, prob = prob, log = TRUE))
}

# probability values
prob_values <- seq(0, 1, 0.01)
count_values <- length(prob_values)
```

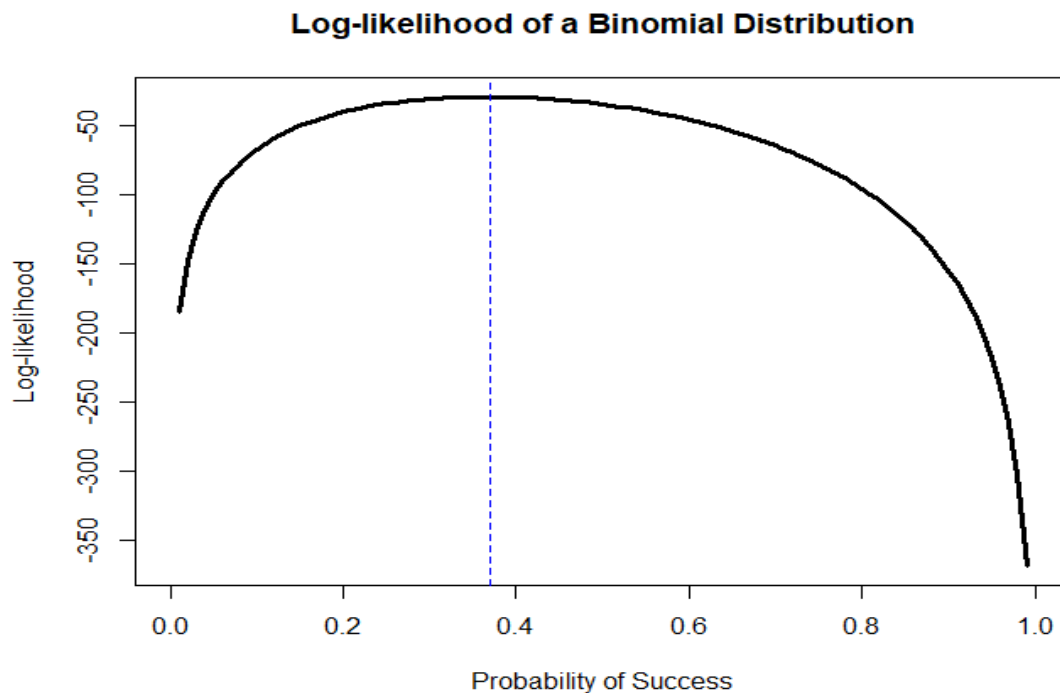
---

```
ll_values <- as.vector(count_values)

for (i in 1:count_values){
  ll_values[i] <- log_lk(prob_values[i], sample, n)
}

plot(prob_values, ll_values, xlab = "Probability of Success",
     ylab = "Log-likelihood",
     main = "Log-likelihood of a Binomial Distribution", type =
       "l", lwd = 3)

# Find the MLE for the parameter P and mark it on the plot
prob_MLE <- prob_values[which.max(ll_values)]
abline(v = prob_MLE, col = "blue", lty = 2)
```



- 
4. Find the maximum likelihood estimate of  $p$ . (You can do this in R, so calculations should not be done by hand.)

**Answer:** The numerical approach was used to find the Maximum Likelihood estimate using the bootstrap method to sample the data from a Binomial distribution. To initiate the process, we defined the essential parameters, including the number of samples, the probability estimate, and the number of trials. These parameters are fundamental for estimating the Maximum Likelihood. A log-likelihood function was created to calculate the log-likelihood of the observed data with respect to the specified parameter. We continued by generating a large number of bootstrap samples (with replacement). For each iteration within the bootstrap sampling, we use the function (negative) log likelihood and the data from the bootstrap sample. We furtherly stored the result in a vector after identifying the Maximum Likelihood Estimation (MLE) using the "mle" function to find the optimal parameter values that maximize the likelihood.

```
library(stats4)
set.seed(1111)

n <- 10
n_sim <- 1000
np_MLE <- as.vector(n_sim)

# Define the likelihood function
nLL <- function(p_new) {
  -sum(dbinom(sample_sim, size = n, prob = p_new, log = TRUE))
}

for (i in 1:n_sim) {
  # Generate a random sample of size 'n' for each iteration
  sample_sim <- rbinom(n, size = n, prob = p)

  # Apply the mle using the log-likelihood function and
  # estimate p
  fit <- mle(nLL, start = list(p_new = 0.5))
}
```

---

```

# Extract the parameter estimate
np_MLE[i] <- coef(fit)["p_new"]
}

# Calculate the mean of the MLE estimates
mean_MLE <- mean(np_MLE)
mean_MLE

```

5. Find an (approximate) standard error for the maximum likelihood estimate using the Fisher information.

**Answer:** Recall the mass function:

Where  $p$  is the probability of success and  $n$  is the number of trials.

$$f(x|p) = \binom{n}{x} p^x (1-p)^{n-x} \quad (1)$$

The approximation of the standard error for the Binomial maximum likelihood estimates using the Fisher Information could be calculated as:

$$ASE(\hat{p}) = \sqrt{\left(\frac{1}{nI(\hat{p})}\right)} \quad (2)$$

We could derive the Fisher Information as:

$$I(\hat{p}) = E \left[ \left( \frac{d}{dp} \log f(x|p) \right)^2 \right] \quad (3)$$

Therefore;

---


$$\log f(x|p) = \log \binom{n}{x} + \log(p^x) + \log((1-p)^{n-x}) \quad (4)$$

$$\frac{d}{dp} \log f(x|p) = \frac{x}{p} + \frac{n-x}{1-p} \quad (5)$$

$$I(\hat{p}) = E \left[ \frac{x - n\hat{p}}{\hat{p}(1-\hat{p})} \right]^2 \quad (6)$$

$$I(\hat{p}) = \frac{1}{\hat{p}^2(1-\hat{p})^2} E[(x - n\hat{p})]^2 \quad (7)$$

Recall;

$$Var(x) = E[(x - n\hat{p})]^2 = n\hat{p}(1 - \hat{p}) \quad (8)$$

So,

$$I(\hat{p}) = \frac{1}{\hat{p}^2(1-\hat{p})^2} n\hat{p}(1 - \hat{p}) \quad (9)$$

Simplifying:

$$I(\hat{p}) = \frac{n}{\hat{p}(1-\hat{p})} \quad (10)$$

And the standard error:

$$ASE(\hat{p}) = \frac{1}{\sqrt{\frac{n^2}{\hat{p}(1-\hat{p})}}} \quad (11)$$

Plugging in the values of  $n = 10$  and  $p = 0.3666667$ :

$$ASE(\hat{p}) = \frac{1}{\sqrt{\frac{10^2}{0.3666667(1-0.3666667)}}} = 0.048189 \quad (12)$$

6. Give a bootstrap confidence interval for the maximum likelihood estimate.

**Answer:** The approximate confidence interval (CI) for the maximum likelihood estimate can be calculated by using quantiles from the bootstrap maximum likelihood estimates.

---

$$CI(\alpha) = (\theta_{\alpha/2}^*, \theta_{1-\alpha/2}^*)$$

Where:  $\theta^*$  is the bootstrap estimates

The 95% CI for  $\alpha = 0.05$ , the approximate CI is done by finding the lower quantile (0.025) and upper quantile (0.975) as shown below.

```
np_MLE_CI <- c(quantile(np_MLE, 0.025), quantile(np_MLE, 0.975))
np_MLE_CI
      2.5%      97.5%
0.2700008 0.4700001
```



---

## Appendix

The Complete R-code

```
# Since it is from a binomial distribution
# n = 10, p = (0, 1) , 15 independent observations
sample <- c(7, 1, 5, 1, 3, 3, 3, 3, 6, 5, 4, 5, 3, 2, 4)

# Question 1
# calculating the mean
np <- mean(sample)
n <- 10
np

# calculating the probability
p <- (np/n)

# Question 2
set.seed(1111)
n_sim <- 1000
np_Mom <- as.vector(n_sim)

for (i in 1:n_sim){
  sample_sim <- rbinom(n_sim, n, p)
  np_Mom[i] <- mean(sample_sim)
}

mean_Mom <- mean(np_Mom)
mean_Mom

# Standard Error for the Method of Moment
SE_Mom <- sd(np_Mom)
SE_Mom

# Question 3
# Plot the log-likelihood as a product of p
```

---

```

# likelihood function
log_lik <- function(prob, sample, n){
  sum(dbinom(sample, size = n, prob = prob, log = TRUE))
}

# probability values
prob_values <- seq(0, 1, 0.01)
count_values <- length(prob_values)

ll_values <- as.vector(count_values)

for (i in 1:count_values){
  ll_values[i] <- log_lik(prob_values[i], sample, n)
}

plot(prob_values, ll_values, xlab = "Probability of Success", ylab
     = "Log-likelihood",
     main = "Log-likelihood of a Binomial Distribution", type = "l"
     , lwd = 3)

# Find the MLE for the parameter P and mark it on the plot
prob_MLE <- prob_values[which.max(ll_values)]
abline(v = prob_MLE, col = "blue", lty = 2)

# Question 4
# Find the MLE of p

library(stats4)

set.seed(1111)
n_sim <- 1000
n <- 10
np_MLE <- as.vector(n_sim)

```

---

```

# Define the likelihood function
nLL <- function(p_new) {
  -sum(dbinom(sample_sim, size = n, prob = p_new, log = TRUE))
}

for (i in 1:n_sim) {
  # Generate a random sample of size 'n' for each iteration
  sample_sim <- rbinom(n, size = n, prob = p)

  # Fit the model and estimate p
  fit <- mle(nLL, start = list(p_new = 0.5))

  # Extract the parameter estimate
  np_MLE[i] <- coef(fit)["p_new"]
}

# Calculate the mean of the MLE estimates
mean_MLE <- mean(np_MLE)
mean_MLE

# Question 5
# Find an (approximate) standard error for the maximum
# likelihood estimate using the Fisher information.

{Solved analytically}

# Question 6
# Give a bootstrap confidence interval for the maximum likelihood
# estimate.

np_MLE_CI <- c(quantile(np_MLE, 0.025), quantile(np_MLE, 0.975))
np_MLE_CI

```