Practical Session 2 in R — Probability Laws

Instructor: Dr. Chennaf Bouchra

Objectives of the Lab

- Apply probability laws using `R`.
- Visualize discrete and continuous distributions (Bernoulli, Binomial, Normal).
- Calculate probabilities and conditional probabilities (Bayes' theorem).

Exercise 1: Bayes' Theorem and Conditional Probability

A medical test detects a rare disease present in 1% of the population. The test is positive 99% of the time when the person is sick (sensitivity) and 5% of the time when the person is healthy (false positive).

1. Define the following probabilities:

$$P(M) = 0.01, \quad P(T^+|M) = 0.99, \quad P(T^+|\neg M) = 0.05$$

2. Calculate the probability that a person who tested positive is actually sick, i.e., $P(M|T^+)$.
3. Simulate a large number of individuals and estimate $P(M|T^+)$ empirically.

R Code with comments

```
# Given probabilities
P_M <- 0.01           # Probability of being sick
P_notM <- 1 - P_M     # Probability of being healthy
P_T_given_M <- 0.99   # Test positive if sick
P_T_given_notM <- 0.05 # Test positive if healthy

# Apply Bayes' theorem:
# P(M|T+) = [P(T+|M) * P(M)] / [P(T+|M)*P(M) + P(T+|¬M)*P(¬M)]
P_M_given_T <- (P_T_given_M * P_M) /
               (P_T_given_M * P_M + P_T_given_notM * P_notM)
cat("P(M|T+) =", P_M_given_T, "\n")

# Empirical simulation
set.seed(2025)
n <- 100000  # simulated population
sick <- rbinom(n, 1, P_M)
test_pos <- ifelse(sick == 1,
                   rbinom(n, 1, P_T_given_M),
                   rbinom(n, 1, P_T_given_notM))

# Empirical estimation of P(M|T+)
P_empirical <- mean(sick[test_pos == 1])
cat("P(M|T+) (simulation) =", P_empirical, "\n")
```

Interpretation: Even with a highly accurate test, since the disease is rare, the probability that a person who tests positive is actually sick remains low (prevalence effect).
—

Exercise 2: Bernoulli Distribution

A seed has a probability $p = 0.9$ of germinating.

1. Define the probability of success p and failure $1 - p$ in R.
2. Compute the mean and variance of the Bernoulli distribution.
3. Simulate 20 random observations and display the distribution as a bar chart.

R Code with comments

```
# Define probability of success
p <- 0.9

# Calculate mean and variance of Bernoulli distribution
E <- p
Var <- p * (1 - p)
cat("Mean =", E, "Variance =", Var, "\n")

# Simulate 20 random trials (0 = failure, 1 = success)
set.seed(1)  # reproducible results
x <- rbinom(20, size = 1, prob = p)

# Display frequency table
table(x)

# Barplot representation
barplot(table(x), col="skyblue",
        main="Bernoulli Distribution (p=0.9)",
        names.arg=c("Failure","Success"))
```

—

Exercise 3: Binomial Distribution

We plant $n = 10$ seeds, each with probability $p = 0.8$ of germinating.

1. Compute $P(X = 8)$ where $X \sim \text{Binomial}(10, 0.8)$.
2. Plot the full distribution of X.
3. Simulate 1000 experiments and compare empirical frequencies to the theoretical distribution.

R Code with comments

```
# Binomial parameters
n <- 10
p <- 0.8

# Possible values of X
k <- 0:n

# Compute theoretical probabilities
prob <- dbinom(k, size=n, prob=p)

# Probability P(X=8)
P8 <- dbinom(8, size=n, prob=p)
cat("P(X=8) =", P8, "\n")

# Barplot of theoretical distribution
barplot(prob, names.arg=k, col="lightgreen",
        main="Binomial Distribution B(10,0.8)",
        xlab="k", ylab="P(X=k)")

# Simulation of 1000 trials
set.seed(123)
sim <- rbinom(1000, size=n, prob=p)

# Histogram and comparison with theoretical probabilities
hist(sim, breaks=seq(-0.5,10.5,1), col="orange",
     main="Simulation (1000 trials)", xlab="Number of successes", freq=FALSE)
points(k, prob, col="blue", pch=19)  # blue points = theoretical values
```

—

Exercise 4: Normal Distribution

Apple weights follow $X \sim \mathcal{N}(200, 25^2)$.

1. Compute $P(175 \leq X \leq 225)$.
2. Simulate a sample of 500 weights and plot a histogram.
3. Overlay the theoretical normal density curve.

R Code with comments

```
# Mean and standard deviation
mu <- 200
sigma <- 25

# Compute probability P(175 <= X <= 225)
P <- pnorm(225, mean=mu, sd=sigma) - pnorm(175, mean=mu, sd=sigma)
cat("P(175 <= X <= 225) =", P, "\n")

# Simulate 500 random weights
set.seed(42)
x <- rnorm(500, mean=mu, sd=sigma)

# Histogram (density)
hist(x, breaks=20, freq=FALSE, col="lightgray",
     main="Apple weights ~ N(200,25^2)",
     xlab="Weight (g)")

# Overlay theoretical density
curve(dnorm(x, mean=mu, sd=sigma), add=TRUE, col="red", lwd=2)
```