DATA 605 - Homework 2

Julian Adames-Ng

2024-11-10

Problem 1

1.1 (Bayesian)

A new credit scoring system has been developed to predict the likelihood of loan defaults. The system has a 90% sensitivity, meaning that it correctly identifies 90% of those who will default on their loans. It also has a 95% specificity, meaning that it correctly identifies 95% of those who will not default. The default rate among borrowers is 2%.

Bayes' Theorem: $P(A|B) = \frac{P(B|A) \cdot P(A)}{P(B)}$

(a) Given these prevalence, sensitivity, and specificity estimates, what is the probability that a borrower flagged by the system as likely to default will actually default?

```r
sens <- 0.90
spec <- 0.95
prev <- 0.02
avg_loss <- 200000 #per default
test_cost <- 500   #per borrower
n <- 10000         #number of borrowers

prob_flag_def <- (sens * prev) / (sens * prev + (1 - spec) * (1 - prev))

cat("The probability that a borrower flagged by the
system as likely to default will actually default is:",
    round(prob_flag_def, 4) * 100, "%\n")
```

```
## The probability that a borrower flagged by the
## system as likely to default will actually default is: 26.87 %
```

1.1 (Bayesian)

(b) If the average loss per defaulted loan is $200,000 and the cost to run the credit scoring test on each borrower is $500, what is the total first-year cost for evaluating 10,000 borrowers?

```r
#Calculating total costs for evaluating 10,000 borrowers
n_flag_pos <- prob_flag_def * n * prev
tot_loss <- n_flag_pos * avg_loss #from defaults
tot_test_cost <- n * test_cost
tot_cost <- tot_loss + tot_test_cost

cat("Total first-year cost for evaluating 10,000 borrowers:", tot_cost, "\n")
```

```
## Total first-year cost for evaluating 10,000 borrowers: 15746269
```

1.2 (Binomial)

Binomial Probability Formula: $P(X = k) = \binom{n}{k} \cdot p^k \cdot q^{n-k}$

(a) What is the probability that the stock pays dividends exactly 6 times in 8 quarters?

```r
# Given data
n <- 8              # Number of quarters
p <- 0.7            # Probability of dividend payment per quarter
k <- 6              # Number of dividend payments of interest

# 1. Probability that the stock pays dividends exactly 6 times in 8 quarters
prob_exactly_6 <- dbinom(k, n, p)

cat("Probability of exactly 6 dividend payments:", prob_exactly_6, "\n")
```

```
## Probability of exactly 6 dividend payments: 0.2964755
```

1.2 (Binomial)

(b) What is the probability that it pays dividends 6 or more times?

```r
# 2. Probability that the stock pays dividends 6 or more times
prob_6_or_more <- sum(dbinom(6:8, n, p))

cat("Probability of 6 or more dividend payments:", prob_6_or_more, "\n")
```

```
## Probability of 6 or more dividend payments: 0.5517738
```

1.2 (Binomial)

(c) What is the probability that it pays dividends fewer than 6 times?

```r
# 3. Probability that the stock pays dividends fewer than 6 times
prob_fewer_than_6 <- sum(dbinom(0:5, n, p))

cat("Probability of fewer than 6 dividend payments:", prob_fewer_than_6, "\n")
```

```
## Probability of fewer than 6 dividend payments: 0.4482262
```

1.2 (Binomial)

(d) What is the expected number of dividend payments over 8 quarters?

```r
# 4. Expected number of dividend payments over 8 quarters
expected_dividends <- n * p

cat("Expected number of dividend payments over 8 quarters:", expected_dividends, "\n")
```

```
## Expected number of dividend payments over 8 quarters: 5.6
```

1.2 (Binomial)

(e) What is the standard deviation?

```r
# 5. Standard deviation of the number of dividend payments
std_dev_dividends <- sqrt(n * p * (1 - p))

cat("Standard deviation of dividend payments:", std_dev_dividends, "\n")
```

```
## Standard deviation of dividend payments: 1.296148
```

1.3 (Poisson)

Poisson Distribution Formula: $P(X = k) = \frac{\lambda^k - e^{-\lambda}}{k!}$

A financial analyst notices that there are an average of 12 trading days each month when a certain stock's price increases by more than 2%.

(a) What is the probability that exactly 4 such days occur in a given month?

```r
# Given data
lambda <- 12            # Average number of days per month with price increase > 2%
k <- 4                  # Number of days of interest for exact probability
months <- 6             # Number of months for expected value and standard deviation
target_days <- 70       # Target number of days for profitability in a year (12 months)

# 1. Probability of exactly 4 such days in a given month
prob_exactly_4 <- dpois(k, lambda)

cat("Probability of exactly 4 such days in a month:", prob_exactly_4, "\n")
```

```
## Probability of exactly 4 such days in a month: 0.005308599
```

1.3 (Poisson)

(b) What is the probability that more than 12 such days occur in a given month?

```
# 2. Probability of more than 12 such days in a given month
prob_more_than_12 <- 1 - ppois(12, lambda)

cat("Probability of more than 12 such days in a month:", prob_more_than_12, "\n")
```

```
## Probability of more than 12 such days in a month: 0.4240348
```

1.3 (Poisson)

(c) How many such days would you expect in a 6-month period?

```
# 3. Expected number of such days in a 6-month period
expected_days_6_months <- lambda * months

cat("Expected number of such days in 6 months:", expected_days_6_months, "\n")
```

```
## Expected number of such days in 6 months: 72
```

1.3 (Poisson)

(d) What is the standard deviation of the number of such days?

```
# 4. Standard deviation of the number of such days in a 6-month period
std_dev_6_months <- sqrt(lambda * months)

cat("Standard deviation of such days in 6 months:", std_dev_6_months, "\n")
```

```
## Standard deviation of such days in 6 months: 8.485281
```

1.3 (Poisson)

(e) If an investment strategy requires at least 70 days of such price increases in a year for profitability, what is the percent utilization and what are your recommendations?

```
# 5. Probability of meeting the target of at least 70 days in a year
# Assuming Poisson distribution with lambda adjusted for 12 months
lambda_12_months <- lambda * 12
prob_at_least_70_days <- 1 - ppois(target_days - 1, lambda_12_months)

cat("Probability of at least 70 days in a year:", prob_at_least_70_days, "\n")
```

```
## Probability of at least 70 days in a year: 1
```

1.4 (Hypergeometric)

A hedge fund has a portfolio of 25 stocks, with 15 categorized as high-risk and 10 as low-risk. The fund manager randomly selects 7 stocks to closely monitor.

Hypergeometric Distribution Formula: $P(X = k) = \frac{\binom{K}{k} \cdot \binom{N-K}{n-k}}{\binom{N}{n}}$

(a) If the manager selected 5 high-risk stocks and 2 low-risk stocks, what is the probability of selecting exactly 5 high-risk stocks if the selection was random?

```
# Given data
total_stocks <- 25       # Total number of stocks
high_risk <- 15          # Number of high-risk stocks
low_risk <- 10           # Number of low-risk stocks
selection_size <- 7      # Number of stocks selected
desired_high_risk <- 5   # Desired number of high-risk stocks selected

# 1. Probability of selecting exactly 5 high-risk stocks
prob_exactly_5_high_risk <- dhyper(desired_high_risk, high_risk, low_risk, selection_size)

cat("Probability of selecting exactly 5 high-risk stocks:", prob_exactly_5_high_risk, "\n")
```

```
## Probability of selecting exactly 5 high-risk stocks: 0.2811213
```

1.4 (Hypergeometric)

(b) How many high-risk and low-risk stocks would you expect to be selected?

```
# 2. Expected number of high-risk and low-risk stocks to be selected
expected_high_risk <- selection_size * (high_risk / total_stocks)
expected_low_risk <- selection_size * (low_risk / total_stocks)

cat("Expected number of high-risk stocks selected:", expected_high_risk, "\n")
```

```
## Expected number of high-risk stocks selected: 4.2
```

```
cat("Expected number of low-risk stocks selected:", expected_low_risk, "\n")
```

```
## Expected number of low-risk stocks selected: 2.8
```

1.5 (Geometric)

The probability that a bond defaults in any given year is 0.5%. A portfolio manager holds this bond for 10 years.

Geometric Distribution Formula: $P(X = k) = (1 - p)^{k-1} \cdot p$

(a) What is the probability that the bond will default during this period?

```
# Given data
p <- 0.005           # Annual probability of default
years_10 <- 10       # Period of 10 years
years_15 <- 15       # Period of 15 years
years_2 <- 2         # Conditional 2-year period after 10 years of survival

# 1. Probability of default within 10 years
prob_default_10_years <- pgeom(years_10 - 1, p)

cat("Probability of default within 10 years:", prob_default_10_years, "\n")
```

```
## Probability of default within 10 years: 0.04888987
```

1.5 (Geometric)

(b) What is the probability that it will default in the next 15 years?

```
# 2. Probability of default within 15 years
prob_default_15_years <- pgeom(years_15 - 1, p)

cat("Probability of default within 15 years:", prob_default_15_years, "\n")
```

```
## Probability of default within 15 years: 0.07243103
```

1.5 (Geometric)

(c) What is the expected number of years before the bond defaults?

```
# 3. Expected number of years before the bond defaults
expected_years_before_default <- 1 / p

cat("Expected number of years before default:", expected_years_before_default, "\n")
```

```
## Expected number of years before default: 200
```

1.5 (Geometric)

(d) If the bond has already survived 10 years, what is the probability that it will default in the next 2 years?

```
# 4. Probability of default within the next 2 years, given survival for 10 years
# same as the probability of defaulting within 2 years from a fresh start
prob_default_next_2_years <- pgeom(years_2 - 1, p)

cat("Probability of default in the next 2 years given survival for 10 years:",
    prob_default_next_2_years, "\n")
```

```
## Probability of default in the next 2 years given survival for 10 years: 0.009975
```

1.6 (Poisson)

A high-frequency trading algorithm experiences a system failure about once every 1500 trading hours.

(a) What is the probability that the algorithm will experience more than two failures in 1500 hours?

```
# Given rate of failures per 1500 hours
lambda <- 1  # Average number of failures in 1500 hours

# 1. Expected number of failures in 1500 hours
expected_failures <- lambda

# 2. Probability of more than two failures in 1500 hours
prob_more_than_2_failures <- 1 - ppois(2, lambda)

cat("Probability of more than two failures in 1500 hours:", prob_more_than_2_failures, "\n")
```

```
## Probability of more than two failures in 1500 hours: 0.0803014
```

1.6 (Poisson)

(b) What is the expected number of failures?

```
cat("Expected number of failures in 1500 hours:", expected_failures, "\n")
```

```
## Expected number of failures in 1500 hours: 1
```

1.7 (Uniform Distribution)

An investor is trying to time the market and is monitoring a stock that they believe has an equal chance of reaching a target price between 20 and 60 days.

Uniform Distribution Formula (Cumulative): $P(X = k) = \frac{x-a}{b-a}$, **for** $a < X = k < b$

(a) What is the probability that the stock will reach the target price in more than 40 days?

```
# Given data
a <- 20  # Lower bound of the uniform distribution
b <- 60  # Upper bound of the uniform distribution

# 1. Probability that the stock reaches the target price in more than 40 days
prob_more_than_40_days <- (b - 40) / (b - a)

cat("Probability that the stock reaches the
target price in more than 40 days:\n", prob_more_than_40_days, "\n")
```

```
## Probability that the stock reaches the
## target price in more than 40 days:
##  0.5
```

1.7 (Uniform Distribution)

(b) If it hasn't reached the target price by day 40, what is the probability that it will reach it in the next 10 days?

```
# 2. Probability that the stock reaches the target in the next 10 days after day 40
# probability of reaching the target between 40 and 50 days, \ngiven that it hasn't reached by day 40
prob_next_10_days_after_40 <- (50 - 40) / (b - 40)

cat("Probability that the stock reaches the target
price between 40 and 50 days given it hasn't reached by day 40:\n", prob_next_10_days_after_40, "\n")
```

```
## Probability that the stock reaches the target
## price between 40 and 50 days given it hasn't reached by day 40:
##  0.5
```

1.7 (Uniform Distribution)

(c) What is the expected time for the stock to reach the target price?

```
# 3. Expected time to reach the target price
expected_time <- (a + b) / 2

cat("Expected time to reach the target price:", expected_time, "days\n")
```

```
## Expected time to reach the target price: 40 days
```

1.8 (Exponential Distribution)

A financial model estimates that the lifetime of a successful start-up before it either goes public or fails follows an exponential distribution with an expected value of 8 years.

Exponential Distribution Formula (Cumulative): $P(X \leq k) = 1 - e^{-\lambda x}$, **for** $x \geq 0$

(a) What is the expected time until the start-up either goes public or fails?

```
# Given data
mean_lifetime <- 8              # Expected lifetime of the start-up in years
lambda <- 1 / mean_lifetime      # Rate parameter for the exponential distribution

# 1. Expected time until the start-up either goes public or fails
expected_time <- mean_lifetime

cat("Expected time until the start-up goes public or fails:", expected_time, "years\n")
```

```
## Expected time until the start-up goes public or fails: 8 years
```

1.8 (Exponential Distribution)

(b) What is the standard deviation?

```r
# 2. Standard deviation of the time until the start-up either goes public or fails
std_dev <- mean_lifetime  # For exponential distribution, standard deviation = mean

cat("Standard deviation of the time until the start-up goes public or fails:", std_dev, "years\n")
```

```
## Standard deviation of the time until the start-up goes public or fails: 8 years
```

1.8 (Exponential Distribution)

(c) What is the probability that the start-up will go public or fail after 6 years?

```r
# 3. Probability that the start-up will go public or fail after 6 years
prob_after_6_years <- 1 - pexp(6, rate = lambda)

cat("Probability that the start-up will go public or fail after 6 years:", prob_after_6_years, "\n")
```

```
## Probability that the start-up will go public or fail after 6 years: 0.4723666
```

1.8 (Exponential Distribution)

(d) Given that the start-up has survived for 6 years, what is the probability that it will go public or fail in the next 2 years?

```r
# 4. Probability that the start-up will go public or
# fail in the next 2 years given it has survived 6 years
# same as the probability of the event occurring within 2 years from time 0
prob_next_2_years_after_6 <- pexp(2, rate = lambda)

cat("Probability that the start-up will go public or
fail in the next 2 years given it has survived 6 years:\n",
prob_next_2_years_after_6, "\n")
```

```
## Probability that the start-up will go public or
## fail in the next 2 years given it has survived 6 years:
##  0.2211992
```

Problem 2

2.1 (Product Selection)

Combination Formula: $\binom{n}{r} = \frac{n!}{(n-r)! \cdot r!}$

A company produces 5 different types of green pens and 7 different types of red pens. The marketing team needs to create a new promotional package that includes 5 pens. How many different ways can the package be created if it contains fewer than 2 green pens?

```
# Calculating combinations for each case
# Case 1: 0 green pens, 5 red pens
case_1 <- choose(7, 5)

# Case 2: 1 green pen, 4 red pens
case_2 <- choose(5, 1) * choose(7, 4)

# Total number of ways
total_ways <- case_1 + case_2

# Display the results
cat("Total number of ways to create the
package with fewer than 2 green pens:", total_ways, "\n")
```

```
## Total number of ways to create the
## package with fewer than 2 green pens: 196
```

2.2 (Team Formation for a Project)

A project committee is being formed within a company that includes 14 senior managers and 13 junior managers. How many ways can a project team of 5 members be formed if at least 4 of the members must be junior managers?

```
# Calculating combinations for each case
# Case 1: 4 junior managers and 1 senior manager
case_1 <- choose(13, 4) * choose(14, 1)

# Case 2: 5 junior managers and 0 senior managers
case_2 <- choose(13, 5)

# Total number of ways
total_ways <- case_1 + case_2

# Display the results
cat("Total number of ways to form the project
team with at least 4 junior managers:", total_ways, "\n")
```

```
## Total number of ways to form the project
## team with at least 4 junior managers: 11297
```

2.3 (Marketing Campaign Outcomes)

A marketing campaign involves three stages: first, a customer is sent 5 email offers; second, the customer is targeted with 2 different online ads; and third, the customer is presented with 3 personalized product recommendations. If the email offers, online ads, and product recommendations are selected randomly, how many different possible outcomes are there for the entire campaign?

```r
# Number of options at each stage
email_offers <- 5
online_ads <- 2
product_recommendations <- 3

# Total number of different possible outcomes for the campaign
total_outcomes <- email_offers * online_ads * product_recommendations

# Display the result
cat("Total number of different possible
outcomes for the entire campaign:", total_outcomes, "\n")
```

```
## Total number of different possible
## outcomes for the entire campaign: 30
```

2.4 (Product Defect Probability)

A quality control team draws 3 products from a batch without replacement. What is the probability that at least one of the products drawn is defective if the defect rate is known to be consistent? Express your answer as a fraction or a decimal number rounded to four decimal places.

```r
# Given defect rate
p <- 0.05  # Example defect rate, assuming 5% of products are defective

# Probability that none of the 3 drawn products are defective
prob_no_defect <- (1 - p)^3

# Probability that at least one of the 3 drawn products is defective
prob_at_least_one_defect <- 1 - prob_no_defect

# Display the result, rounded to four decimal places
cat("Probability that at least one product is defective:",
round(prob_at_least_one_defect, 4), "\n")
```

```
## Probability that at least one product is defective: 0.1426
```

2.5 (Business Strategy Choices)

A business strategist is choosing potential projects to invest in, focusing on 17 high-risk, high-reward projects and 14 low-risk, steady-return projects.

(a) Step 1: How many different combinations of 5 projects can the strategist select?

```r
# Given data
total_projects <- 31    # Total number of projects
high_risk_projects <- 17 # High-risk projects
low_risk_projects <- 14  # Low-risk projects
selected_projects <- 5   # Projects to be selected

# Step 1: Total combinations of selecting 5 projects from 31
total_combinations <- choose(total_projects, selected_projects)

cat("Step 1: Total combinations of 5 projects from 31:",
    total_combinations, "\n")
```

```
## Step 1: Total combinations of 5 projects from 31: 169911
```

2.5 (Business Strategy Choices)

(b) Step 2: How many different combinations of 5 projects can the strategist select if they want at least one low-risk project?

```r
# Step 2: Combinations of selecting 5 projects with at least one low-risk project
# Calculate combinations of selecting 5 projects with all high-risk (no low-risk)
high_risk_only_combinations <- choose(high_risk_projects, selected_projects)

# Combinations with at least one low-risk project
at_least_one_low_risk_combinations <- total_combinations - high_risk_only_combinations

cat("Step 2: Combinations of 5 projects with at least one low-risk project:\n",
    at_least_one_low_risk_combinations, "\n")
```

```
## Step 2: Combinations of 5 projects with at least one low-risk project:
##  163723
```

2.6 (Event Scheduling)

A business conference needs to schedule 9 different keynote sessions from three different industries: technology, finance, and healthcare. There are 4 potential technology sessions, 104 finance sessions, and 17 healthcare sessions to choose from. How many different schedules can be made? Express your answer in scientific notation rounding to the hundredths place.

```r
# Number of potential sessions in each industry
tech_sessions <- 4
finance_sessions <- 104
healthcare_sessions <- 17

# Total sessions to select
total_sessions <- 9

# Initialize total number of combinations
total_combinations <- 0
```

```r
# Loop over possible choices for each industry (ensuring the sum equals 9)
for (tech in 0:min(total_sessions, tech_sessions)) {
  for (finance in 0:min(total_sessions - tech, finance_sessions)) {
    healthcare <- total_sessions - tech - finance
    if (healthcare >= 0 && healthcare <= healthcare_sessions) {
      # Add the number of combinations for this distribution
      total_combinations <- total_combinations +
        choose(tech_sessions, tech) *
        choose(finance_sessions, finance) *
        choose(healthcare_sessions, healthcare)
    }
  }
}

# Express the result in scientific notation, rounded to two decimal places
cat("Total number of different schedules:\n",
    format(total_combinations, scientific = TRUE, digits = 4), "\n")
```

```
## Total number of different schedules:
##  1.529e+13
```

2.7 (Book Selection for Corporate Training)

An HR manager needs to create a reading list for a corporate leadership training program, which includes 13 books in total. The books are categorized into 6 novels, 6 business case studies, 7 leadership theory books, and 5 strategy books.

(a) Step 1: If the manager wants to include no more than 4 strategy books, how many different reading schedules are possible? Express your answer in scientific notation rounding to the hundredths place.

```r
# Book counts in each category
novels <- 6
business_case_studies <- 6
leadership_theory <- 7
strategy <- 5
total_books <- 13

# Step 1: Total reading schedules with no more than 4 strategy books
total_combinations_step1 <- 0
for (num_strategy in 0:4) {
  remaining_books <- total_books - num_strategy
  if (remaining_books <= novels + leadership_theory + business_case_studies) {
    combinations <- choose(strategy, num_strategy) *
      sum(sapply(0:min(remaining_books, novels), function(num_novels) {
        num_leadership <- remaining_books - num_novels - business_case_studies
        if (num_leadership >= 0 && num_leadership <= leadership_theory) {
          choose(novels, num_novels) * choose(leadership_theory, num_leadership)
        } else {
          0
```

```
        }
      }))
    total_combinations_step1 <- total_combinations_step1 + combinations
  }
}

cat("Step 1: Total reading schedules with no more than 4 strategy books:",
    format(total_combinations_step1, scientific = TRUE, digits = 4), "\n")
```

```
## Step 1: Total reading schedules with no more than 4 strategy books: 3.175e+04
```

2.7 (Book Selection for Corporate Training)

(b) Step 2: If the manager wants to include all 6 business case studies, how many different reading schedules are possible? Express your answer in scientific notation rounding to the hundredths place.

```
# Step 2: Total reading schedules with all 6 business case studies
remaining_books_step2 <- total_books - business_case_studies
total_combinations_step2 <- 0
for (num_strategy in 0:min(remaining_books_step2, strategy)) {
  num_leadership <- remaining_books_step2 - num_strategy
  if (num_leadership <= leadership_theory) {
    combinations <- choose(strategy, num_strategy) * choose(leadership_theory, num_leadership)
    total_combinations_step2 <- total_combinations_step2 + combinations
  }
}

cat("Step 2: Total reading schedules with all 6 business case studies:",
    format(total_combinations_step2, scientific = TRUE, digits = 4), "\n")
```

```
## Step 2: Total reading schedules with all 6 business case studies: 7.92e+02
```

2.8 (Product Arrangement)

A retailer is arranging 10 products on a display shelf. There are 5 different electronic gadgets and 5 different accessories. What is the probability that all the gadgets are placed together and all the accessories are placed together on the shelf? Express your answer as a fraction or a decimal number rounded to four decimal places.

```
# Factorials for total arrangements and favorable arrangements
total_arrangements <- factorial(10)
favorable_arrangements <- factorial(5) * factorial(5) * factorial(2)

# Probability calculation
probability <- favorable_arrangements / total_arrangements

# Display the result rounded to four decimal places
cat("The probability that all gadgets are placed together
and all accessories are placed together is:\n",
    round(probability, 4), "\n")
```

```
## The probability that all gadgets are placed together
## and all accessories are placed together is:
##  0.0079
```

2.9 (Expected Value of a Business Deal)

A company is evaluating a deal where they either gain $4 for every successful contract or lose $16 for every unsuccessful contract. A "successful" contract is defined as drawing a queen or lower from a standard deck of cards. (Aces are considered the highest card in the deck.)

(a) Step 1: Find the expected value of the deal. Round your answer to two decimal places. Losses must be expressed as negative values.

```r
# Given probabilities
p_success <- 48 / 52   # Probability of drawing a queen or lower
p_failure <- 4 / 52    # Probability of drawing an ace

# Gains and losses
gain_success <- 4      # Gain per successful contract
loss_failure <- -16    # Loss per unsuccessful contract

# Step 1: Expected value of the deal
expected_value <- (gain_success * p_success) + (loss_failure * p_failure)

# Display results rounded to two decimal places
cat("Step 1: Expected value of the deal:", round(expected_value, 2), "\n")
```

```
## Step 1: Expected value of the deal: 2.46
```

2.9 (Expected Value of a Business Deal)

(b) Step 2: If the company enters into this deal 833 times, how much would they expect to win or lose? Round your answer to two decimal places. Losses must be expressed as negative values.

```r
# Step 2: Expected value for 833 deals
total_expected_value <- expected_value * 833

cat("Step 2: Expected value for 833 deals:",
    round(total_expected_value, 2), "\n")
```

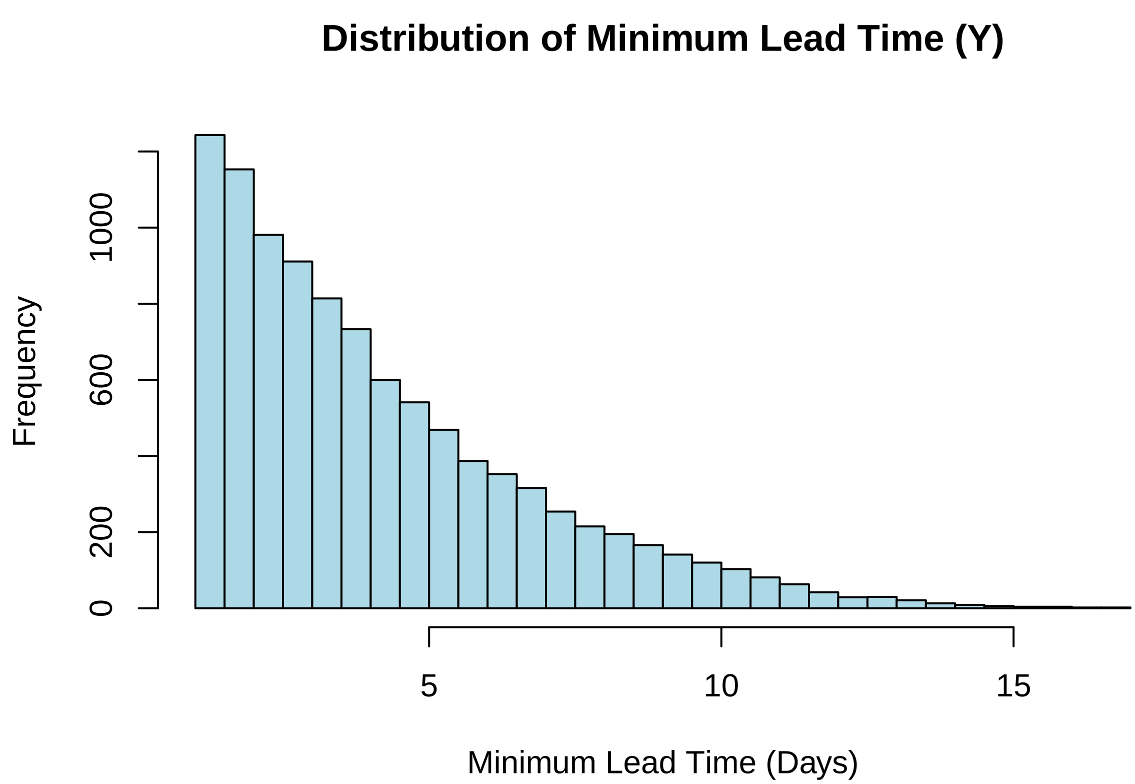
```
## Step 2: Expected value for 833 deals: 2050.46
```

Problem 3

3.1 (Supply Chain Risk Assessment)

Let $X_1, X_2, ..., X_n$ represent the lead times (in days) for the delivery of key components from $n = 5$ different suppliers. Each lead time is uniformly distributed across a range of 1 to $k = 20$ days, reflecting the uncertainty in delivery times. Let Y denote the minimum delivery time among all suppliers. Understanding the distribution of Y is crucial for assessing the earliest possible time you can begin production. Determine the distribution of Y to better manage your supply chain and minimize downtime.

```r
# Parameters
n <- 5          # Number of suppliers
k <- 20         # Maximum lead time in days
num_sim <- 10000  # Number of simulations

# Simulate the distribution of Y
set.seed(123)  # For reproducibility
min_lead_times <- replicate(num_sim, min(runif(n, 1, k)))

# Plot the distribution of Y
hist(min_lead_times, breaks = 30,
     main = "Distribution of Minimum Lead Time (Y)",
     xlab = "Minimum Lead Time (Days)", ylab = "Frequency",
     col = "lightblue", border = "black")
```



3.2 (Maintenance Planning for Critical Equipment)

Your organization owns a critical piece of equipment, such as a high-capacity photocopier (for a law firm) or an MRI machine (for a healthcare provider). The manufacturer estimates the expected lifetime of this equipment to be 8 years, meaning that, on average, you expect one failure every 8 years. It's essential to understand the likelihood of failure over time to plan for maintenance and replacements.

(a) Geometric Model:

Calculate the probability that the machine will not fail for the first 6 years. Also, provide the expected value and standard deviation. This model assumes each year the machine either fails or does not, independently of previous years.

```
# Parameters
p <- 1 / 8          # Probability of failure each year
years <- 6          # Number of years without failure

# Step 1: Probability of no failure for the first 6 years (geometric model)
prob_no_failure_6_years <- (1 - p)^years

# Expected value and standard deviation for geometric model
expected_value_6_years <- 1 / p
standard_deviation_6_years <- sqrt((1 - p) / p^2)

# Output
list(prob_no_failure_6_years = prob_no_failure_6_years,
     expected_value_6_years = expected_value_6_years,
     standard_deviation_6_years = standard_deviation_6_years)
```

```
## $prob_no_failure_6_years
## [1] 0.4487953
##
## $expected_value_6_years
## [1] 8
##
## $standard_deviation_6_years
## [1] 7.483315
```

3.2 (Maintenance Planning for Critical Equipment)

(b) Exponentional Model:

```
p <- 1 / 8          # Probability of failure each year
years <- 6          # Number of years without failure

# Parameters for exponential distribution
lambda <- 1 / 8      # Rate parameter (1/mean)

# Step 2: Probability of no failure for the first 6 years
prob_no_failure_continuous <- exp(-lambda * years)

# Expected value and standard deviation for exponential distribution
expected_value_continuous <- 1 / lambda
standard_deviation_continuous <- 1 / lambda

# Output
list(prob_no_failure_continuous = prob_no_failure_continuous,
     expected_value_continuous = expected_value_continuous,
     standard_deviation_continuous = standard_deviation_continuous)
```

```
## $prob_no_failure_continuous
## [1] 0.4723666
##
## $expected_value_continuous
## [1] 8
##
## $standard_deviation_continuous
## [1] 8
```

3.2 (Maintenance Planning for Critical Equipment)

(c) Binomial Model:

```r
# Step 3: Probability and statistics (same as Step 1)
prob_no_failure_fixed_trials <- prob_no_failure_6_years
expected_value_fixed_trials <- years * p
standard_deviation_fixed_trials <- sqrt(years * p * (1 - p))

# Output
list(prob_no_failure_fixed_trials = prob_no_failure_fixed_trials,
     expected_value_fixed_trials = expected_value_fixed_trials,
     standard_deviation_fixed_trials = standard_deviation_fixed_trials)
```

```
## $prob_no_failure_fixed_trials
## [1] 0.4487953
##
## $expected_value_fixed_trials
## [1] 0.75
##
## $standard_deviation_fixed_trials
## [1] 0.8100926
```

3.2 (Maintenance Planning for Critical Equipment)

(d) Poisson Model:

```r
# Parameters for Poisson process
lambda_poisson <- 1 / 8     # Expected rate of failures per year

# Step 4: Probability of zero failures in the first 6 years
prob_no_failure_poisson <- exp(-lambda_poisson * years)

# Expected value and standard deviation for Poisson model over 6 years
expected_value_poisson <- lambda_poisson * years
standard_deviation_poisson <- sqrt(lambda_poisson * years)

# Output
list(prob_no_failure_poisson = prob_no_failure_poisson,
     expected_value_poisson = expected_value_poisson,
     standard_deviation_poisson = standard_deviation_poisson)
```

```
## $prob_no_failure_poisson
## [1] 0.4723666
##
## $expected_value_poisson
## [1] 0.75
##
## $standard_deviation_poisson
## [1] 0.8660254
```

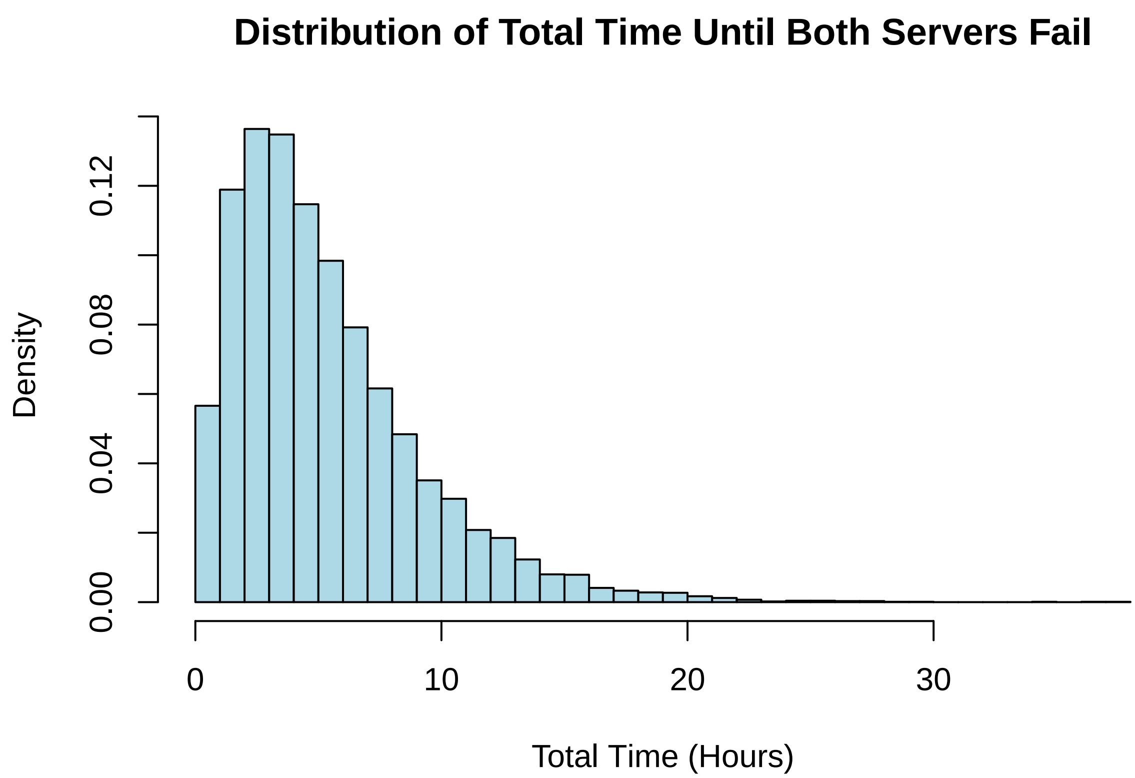
Problem 4

4.1 Scenario: You are managing two independent servers in a data center. The time until the next failure for each server follows an exponential distribution with different rates:

- Server A has a failure rate of $\lambda_A = 0.5$ failures per hour
- Server B has a failure rate of $\lambda_B = 0.3$ failures per hour

Question: What is the distribution of the total time until both servers have failed at least once? Use the moment generating function (MGF) to find the distribution of the sum of the times to failure.

```r
# Parameters
lambda_A <- 0.5  # Failure rate for Server A
lambda_B <- 0.3  # Failure rate for Server B
n_sim <- 10000   # Number of simulations for approximation

# Step 1: Simulate the total time to failure for both servers
set.seed(123)  # For reproducibility
total_times <- rexp(n_sim, rate = lambda_A) + rexp(n_sim, rate = lambda_B)

# Step 2: Plot the distribution of the simulated total times
hist(total_times, breaks = 50, probability = TRUE,
     main = "Distribution of Total Time Until Both Servers Fail",
     xlab = "Total Time (Hours)", col = "lightblue", border = "black")
```




```r
# Step 3: Calculate the theoretical expected value and standard deviation
expected_value <- 1 / lambda_A + 1 / lambda_B
standard_deviation <- sqrt((1 / lambda_A^2) + (1 / lambda_B^2))

# Output the expected value and standard deviation
list(
  expected_value = expected_value,
  standard_deviation = standard_deviation
)
```

```
## $expected_value
## [1] 5.333333
##
## $standard_deviation
## [1] 3.887301
```

4.2 Sum of Independent Normally Distributed Random Variables

Scenario: An investment firm is analyzing the returns of two independent assets, Asset X and Asset Y. The returns on these assets are normally distributed:

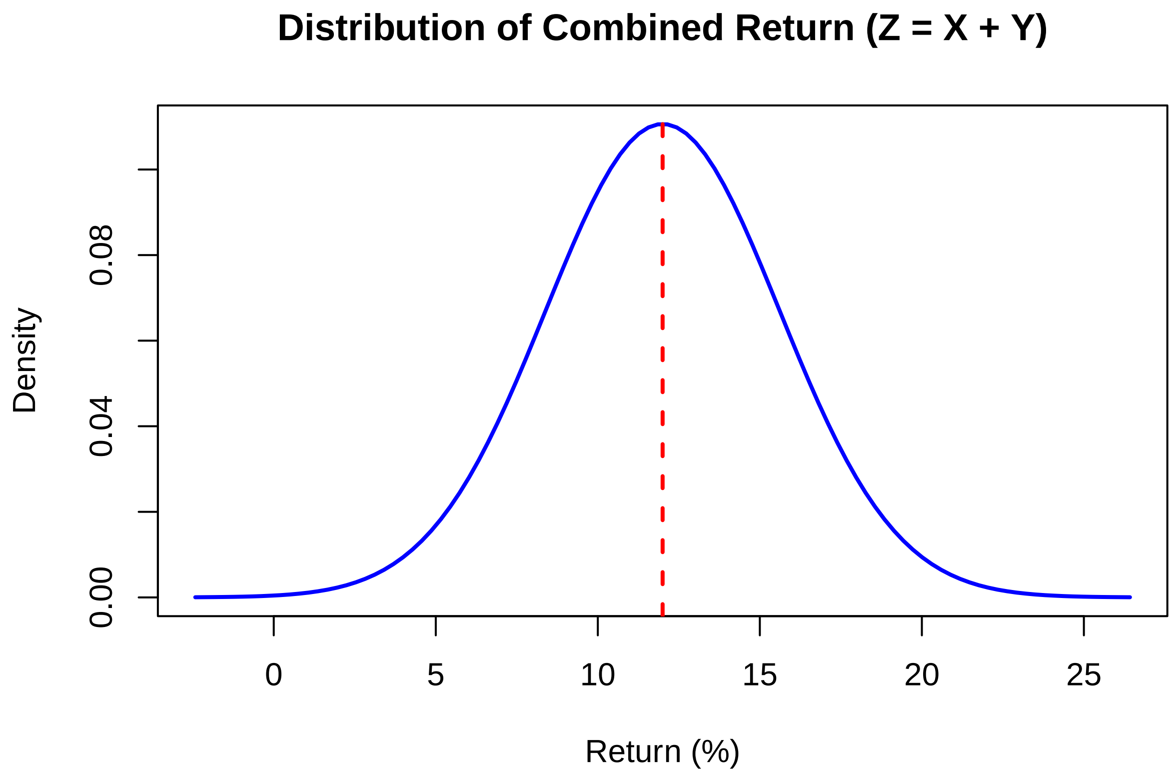
- Asset X: $X\ N(\mu_X = 5\%, \sigma_X^2 = 4\%)$
- Asset Y: $Y\ N(\mu_Y = 7\%, \sigma_Y^2 = 9\%)$

Question: Find the distribution of the combined return of the portfolio consisting of these two assets using the moment generating function (MGF).

```r
# Parameters for Asset X
mu_X <- 5      # Mean of X in percentage
sigma_X <- sqrt(4)  # Standard deviation of X in percentage

# Parameters for Asset Y
mu_Y <- 7      # Mean of Y in percentage
sigma_Y <- sqrt(9)  # Standard deviation of Y in percentage

# Combined mean and standard deviation for Z = X + Y
mu_Z <- mu_X + mu_Y
sigma_Z <- sqrt(sigma_X^2 + sigma_Y^2)

# Output combined mean and standard deviation
list(
  combined_mean = mu_Z,
  combined_standard_deviation = sigma_Z
)
```

```
## $combined_mean
## [1] 12
##
## $combined_standard_deviation
## [1] 3.605551
```

```r
# Plotting the distribution of Z
x_vals <- seq(mu_Z - 4 * sigma_Z, mu_Z + 4 * sigma_Z, length.out = 100)
y_vals <- dnorm(x_vals, mean = mu_Z, sd = sigma_Z)

plot(x_vals, y_vals, type = "l", lwd = 2, col = "blue",
     main = "Distribution of Combined Return (Z = X + Y)",
     xlab = "Return (%)", ylab = "Density")
abline(v = mu_Z, col = "red", lwd = 2, lty = 2)  # Mean line
```



4.3 Scenario: A call center receives calls independently from two different regions. The number of calls received from Region A and Region B in an hour follows a Poisson distribution:

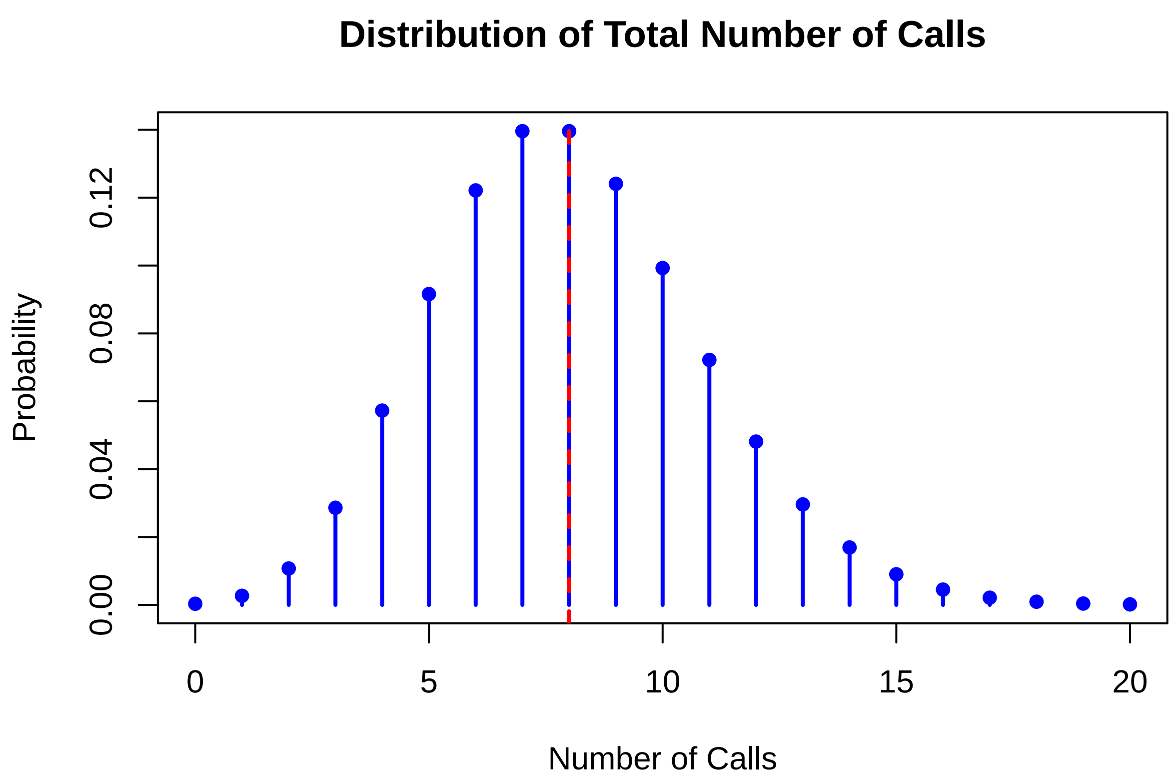
- Region A: $X_A \sim Poisson(\lambda_A = 3)$
- Region B: $X_B \sim Poisson(\lambda_B = 5)$

Question: Determine the distribution of the total number of calls received in an hour from both regions using the moment generating function (MGF).

```
# Parameters
lambda_A <- 3   # Rate for Region A
lambda_B <- 5   # Rate for Region B
lambda_total <- lambda_A + lambda_B  # Combined rate for both regions

# Expected value and standard deviation for the total number of calls
expected_value <- lambda_total
standard_deviation <- sqrt(lambda_total)

# Output expected value and standard deviation
list(
  expected_value = expected_value,
  standard_deviation = standard_deviation
)
```

```
## $expected_value
## [1] 8
##
## $standard_deviation
## [1] 2.828427
```

```r
# Plotting the Poisson distribution for the total number of calls
x_vals <- 0:20  # Range of possible call counts
y_vals <- dpois(x_vals, lambda = lambda_total)

plot(x_vals, y_vals, type = "h", lwd = 2, col = "blue",
     main = "Distribution of Total Number of Calls",
     xlab = "Number of Calls", ylab = "Probability")
points(x_vals, y_vals, pch = 16, col = "blue")
abline(v = expected_value, col = "red", lwd = 2, lty = 2)  # Mean line
```



Problem 5

5.1 Customer Retention and Churn Analysis

Scenario: A telecommunications company wants to model the behavior of its customers regarding their likelihood to stay with the company (retention) or leave for a competitor (churn). The company segments its customers into three states:

- **State 1:** Active customers who are satisfied and likely to stay (Retention state).
- **State 2:** Customers who are considering leaving (At-risk state).
- **State 3:** Customers who have left (Churn state).

The company has historical data showing the following monthly transition probabilities:

- **From State 1 (Retention):** 80
- **From State 2 (At-risk):** 30
- **From State 3 (Churn):** 100

The company wants to analyze the long-term behavior of its customer base.

Question: (a) Construct the transition matrix for this Markov Chain.

```r
# Step (a): Define the transition matrix
transition_matrix <- matrix(c(
  0.80, 0.15, 0.05,   # From State 1
  0.30, 0.50, 0.20,   # From State 2
  0.00, 0.00, 1.00    # From State 3
), nrow = 3, byrow = TRUE)

# Display the transition matrix
transition_matrix
```

```
##      [,1] [,2] [,3]
## [1,]  0.8 0.15 0.05
## [2,]  0.3 0.50 0.20
## [3,]  0.0 0.00 1.00
```

5.1 Customer Retention and Churn Analysis

Question: (b) If a customer starts as satisfied (State 1), what is the probability that they will eventually churn (move to State 3)?

```r
# Step (b): Calculate the probability of eventually
# churning (absorbing probabilities)
# This requires creating a sub-matrix of transient
# states and calculating the fundamental matrix
Q <- transition_matrix[1:2, 1:2]
R <- transition_matrix[1:2, 3]

# Fundamental matrix: N = (I - Q)^-1
I <- diag(2)
N <- solve(I - Q)

# Probability of reaching State 3 (churn) from each starting transient state
absorption_prob <- N %*% R

# Probability of eventually churning from State 1
prob_churn_from_state_1 <- absorption_prob[1, 1]
```

5.1 Customer Retention and Churn Analysis

Question: (c) Determine the steady-state distribution of this Markov Chain. What percentage of customers can the company expect to be in each state in the long run?

```
# Step (c): Steady-state distribution
# Solve for the steady-state vector such that pi * P = pi
eigen_results <- eigen(t(transition_matrix))
steady_state <- eigen_results$vectors[, 1]
steady_state <- steady_state / sum(steady_state)  # Normalize to sum to 1

# Output results
list(
  transition_matrix = transition_matrix,
  prob_churn_from_state_1 = prob_churn_from_state_1,
  steady_state_distribution = steady_state
)
```

```
## $transition_matrix
##      [,1] [,2] [,3]
## [1,]  0.8 0.15 0.05
## [2,]  0.3 0.50 0.20
## [3,]  0.0 0.00 1.00
##
## $prob_churn_from_state_1
## [1] 1
##
## $steady_state_distribution
## [1] 0 0 1
```

5.2: Inventory Management in a Warehouse

Scenario: A warehouse tracks the inventory levels of a particular product using a Markov Chain model. The inventory levels are categorized into three states:

- **State 1:** High inventory (More than 100 units in stock).
- **State 2:** Medium inventory (Between 50 and 100 units in stock).
- **State 3:** Low inventory (Less than 50 units in stock).

The warehouse has the following transition probabilities for inventory levels from one month to the next:

- **From State 1 (High):** 70
- **From State 2 (Medium):** 20
- **From State 3 (Low):** 10

The warehouse management wants to optimize its restocking strategy by understanding the long-term distribution of inventory levels.

Question: (a) Construct the transition matrix for this Markov Chain.

```r
# Step (a): Define the transition matrix
transition_matrix <- matrix(c(
  0.70, 0.25, 0.05,  # From State 1
  0.20, 0.50, 0.30,  # From State 2
  0.10, 0.40, 0.50   # From State 3
), nrow = 3, byrow = TRUE)

# Display the transition matrix
transition_matrix
```

```
##      [,1] [,2] [,3]
## [1,]  0.7 0.25 0.05
## [2,]  0.2 0.50 0.30
## [3,]  0.1 0.40 0.50
```

5.2: Inventory Management in a Warehouse

Question: (b) If the warehouse starts with a high inventory level (State 1), what is the probability that it will eventually end up in a low inventory level (State 3)?

```r
# Step (b): Calculate the probability of eventually ending up in State 3
# We calculate this by raising the matrix to a high power
n_steps <- 100  # Number of steps to approximate steady state
transition_matrix_power <- transition_matrix ^ n_steps

# Probability of ending up in State 3 from each starting state
prob_eventual_low_inventory_from_state_1 <- transition_matrix_power[1, 3]
```

5.2: Inventory Management in a Warehouse

Question: (c) Determine the steady-state distribution of this Markov Chain. What is the long-term expected proportion of time that the warehouse will spend in each inventory state?

```r
# Step (c): Steady-state distribution
# Solve for the steady-state vector such that pi * P = pi
eigen_results <- eigen(t(transition_matrix))
steady_state <- eigen_results$vectors[, 1]
steady_state <- steady_state / sum(steady_state)  # Normalize to sum to 1

# Output results
list(
  transition_matrix = transition_matrix,
  prob_eventual_low_inventory_from_state_1 = prob_eventual_low_inventory_from_state_1,
  steady_state_distribution = steady_state
)
```

```
## $transition_matrix
##      [,1] [,2] [,3]
## [1,]  0.7 0.25 0.05
## [2,]  0.2 0.50 0.30
## [3,]  0.1 0.40 0.50
```

```
##
## $prob_eventual_low_inventory_from_state_1
## [1] 7.888609e-131
##
## $steady_state_distribution
## [1] 0.3466667 0.3866667 0.2666667
```