Assignment 1 Rui Qin 30874157

Question 1 (4 marks)

1. Creating an image of a gorilla riding a motorcycle: **Generative AI systems**
2. Discovering the shopping preferences of customers at Coles: **Recommendation systems**
3. Predicting the amount of Panadol that will purchased in the next week: **Forecasting**
4. Using genomic and lifestyle factors to determine the chance of an individual contracting diabetes in the next two years: **Risk prediction**

Question 2 (6 marks)

1. Marginal probability of winning (P(W = 1)):
   P(W = 1)
   = P(W = 1, H = 0, P = 0) + P(W = 1, H = 0, P = 1) + P(W = 1, H = 1, P = 0) + P(W = 1, H = 1, P = 1)
   = 0.588

2. Probability of winning a game given that the team lost their previous game (P(W = 1 | P = 0)):

   P(W = 1 | P = 0)

   = (P(W = 1, H = 0, P = 0) + P(W = 1, H = 1, P = 0)) / (P(P = 0, H = 0) + P(P = 0, H = 1))

   = (0.235+0.058) / (0.176+0.235+0.117+0.058)

   = 0.5

3. Probability of winning a game given that the team won their previous game (P(W = 1 | P = 1)):

   P(W = 1 | P = 1)

   = (P(W = 1, H = 0, P = 1) + P(W = 1, H = 1, P = 1)) / (P(P = 1, H = 0) + P(P = 1, H = 1))

   = (0.117 + 0.178) / (0.060 + 0.117 + 0.059 + 0.178)

   = 0.713

4. Yes. We can compare the probabilities in questions 2 and 3:

   P(W = 1 | P = 1) > P(W = 1 | P = 0)

   It shows that the team is more likely to win after winning their previous game compared to winning after losing their previous game.

5. Probability of winning next home game after previous win:

   $P(W = 1 \mid P = 1, H = 1)$

   $= P(W = 1, P = 1, H = 1) / (P(W = 1, H = 1, P = 1) + P(W = 0, H = 1, P = 1))$

   $= 0.178 / (0.178+0.059)$

   $= 0.75$

   Probability of winning next game after previous home game win:

   $P(W = 1 \mid P = 1, H = 0)$

   $= P(W = 1, P = 1, H = 0) / (P(W = 1, P = 1, H = 0) + P(W = 0, P = 1, H = 0))$

   $= 0.117 / 0.177$

   $= 0.661$

   Probability of winning next two games:

   $P(W = 1 \mid P = 1, H = 1) \cdot P(W = 1 \mid P = 1, H = 0)$

   $= 0.75 \cdot 0.661$

   $= 0.496$

Question 3 (8 marks)

1. Expected value of S:

   $E[S]$

   $= 2 \cdot E[X1] - E[Y1]$

   $= 2 \cdot ((1 + 2 + 3 + 4 + 5 + 6) / 6) - ((1 + 2 + 3 + 4) / 4)$

   $= 4.5$

2. Variance of S:

   $Var[S]$

   $= E[S^2] - E[S]^2$

   $= E[(2 \cdot X1 - Y1)^2] - 20.25$

   $= E[4 \cdot X1^2 - 4 \cdot X1Y1 + Y1^2] - 20.25$

   $= 4 \cdot E[X1^2] - 4 \cdot E[X1Y1] + E[Y1^2] - 20.25$

   - $E[X1^2] = (1^2 + 2^2 + 3^2 + 4^2 + 5^2 + 6^2) / 6 = 91/6$
   - $E[Y1^2] = (1^2 + 2^2 + 3^2 + 4^2) / 4 = 7.5$
   - $E[X1 \cdot Y1] = 3.5 \cdot 2.5=8.75$

   $= 12.917$

3. Probability distribution of S:

   For S = −2:

   - P(S = −2) = P(2·X1 − Y1 = −2) if:
     - X1 = 1, Y1 = 4
   - P(S = −2) = 1/6 · 1/4 = 1/24

   For S = −1:

   - P(S = −1) = P(2·X1 − Y1 = −1) if:
     - X1 = 1, Y1 = 3
   - P(S = −1) = 1/24

   … …

   The final result:

   For P(S = s) = 1/24: s ∈ {−2, −1, 10, 11}

   For P(S = s) = 1/12: s ∈ {0, 1, 2, 3, 4, 5, 6, 7, 8, 9}

4. Expected value of S^3:

   E(S^3)

   = Σ [P(S) · S^3]

   = (−2)^3 · P(S=−2) + (−1)^3 · P(S=−1) + 0^3 · P(S=0) + 1^3 · P(S=1) +…+ 11^3 · P(S=11)

   = (−2)^3 · 1/24 + (−1)^3 · 1/24 + 1^3 · 1/12 + 2^3 · 1/12 + 3^3 · 1/12 + 4^3 · 1/12 + 5^3 · 1/12 + 6^3 · 1/12 + 7^3 · 1/12 + 8^3 · 1/12 + 9^3 · 1/12 + 10^3 · 1/24 + 11^3 · 1/24

   = 265.5

5. Approximate value of E(S^3) using the Taylor series procedure:

   f(x) = x^3, f′(x) = 3x^2, f′′(x) = 6x, f′′′(x) = 6

   E[S] = 4.5, E[X1] = 3.5, E[Y1] = 2.5

   E[S^3]

   = E[f(4.5) + f′(4.5) · (S−4.5) + f′′(4.5) / 2! · (S−4.5)^2 + f′′′(4.5) / 3! · (S−4.5)^3]

   = 91.125 + 60.75 · E[S − 4.5] + 27 / 2! · E[(S − 4.5)^2] + 6 / 3! · E[(S − 4.5)^3]

   ∵

   Var[S] = E[(S−E[S])^2] = 12.917

   E[S − 4.5] = E[S] − 4.5 = 0

   ∴

   = 91.125 + 60.75 · 0 + 13.5 · 12.917+ E[(S − 4.5)^3]

= 91.125 + 60.75 · 0 + 13.5 · 12.917+ E[(S − 4.5)^2] · 0

= 265.5045

6. Expected value of (2X1 − Y1 + 2Y2)^2:

   (2X1 − Y1 + 2Y2)^2

   = 4X1^2 − 4X1Y1 + 8 X1Y2 + Y1^2 − 4Y1Y2 + 4Y2^2

   E[(2X1 − Y1 + 2Y2)^2] = 4 · E[X1^2] − 4 · E[X1 · Y1] + 8 · E[X1 · Y2] + E[Y1^2] − 4 · E[Y1 · Y2] + 4 · E[Y2^2]

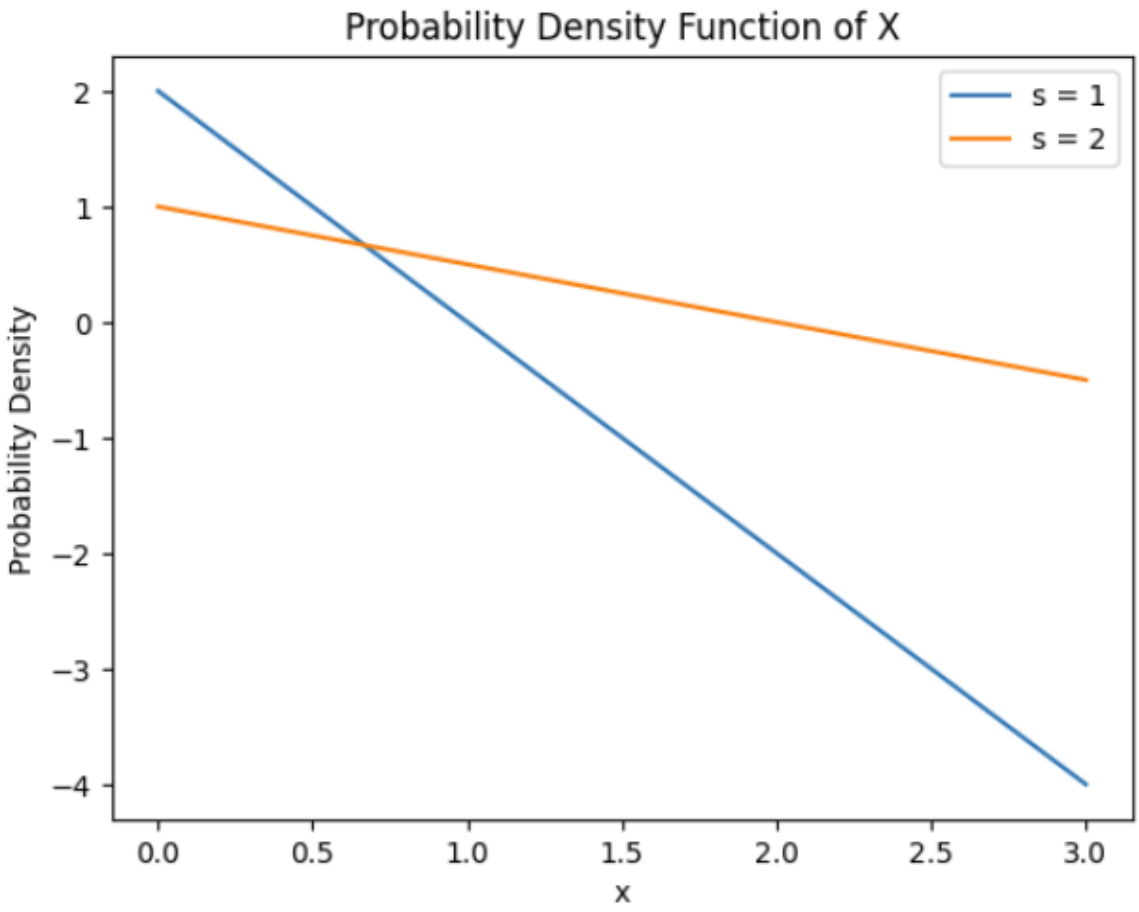
   - E [X1^2] = 91/6
   - E [Y1^2] = E [Y2^2] = E[Y1 · Y2] = 7.5
   - E[X1 · Y1] = E[X1 · Y2] = E[X1] · E[Y1]=3.5 · 2.5=8.75

   = 4 · 91/6 − 4 · 8.75 + 8 · 8.75 + 7.5 − 4 · 7.5+4 · 7.5

   ≈ 103.167

Question 4 (7 marks)

1. Plotting the Probability Density Function (PDF):



2. Expected Value of X (E[X]):
   E[X]
   = ∫[0,s] x · 2(s − x) / s^2 dx
   = s/3

3. Expected Value of √x:
   E[√x]
   = ∫[0,s] √x · 2(s − √x) / s^2 dx
   = 4√s / 3 − 1

4. Variance of X (V[X]):
   Var[x] = E[(x − E[x])^2]
   E[X^2]
   = ∫[0,s] x^2 · 2(s−x)/s^2 dx
   = s^2/6
   V[x]
   = s^2/6 – (s/3)^2
   = s^2/18

5. Median of X:

To find the median of the X with the given probability density function, we need to use cumulative distribution function = 0.5.

CDF(x | s) = ∫[−∞,x] p(X = t|s) dt = 0.5

∵ p(X = t|s) = 2(s−x)/s^2

∴ CDF(x | s)

= ∫[0,x] 2(s−t)/s^2 dt

= (2sx – x^2) / s^2 = 0.5

∴ x^2 – 2sx + 0.5s^2 = 0

x1 = ½ (√2s+2s)

x2 = ½ (-√2s+2s)

∵ 0 < x < s, ½ (√2s+2s) > S

∴ Median (x) = ½ (-√2s+2s)

Question 5 (7 marks)

1. It takes around 15.556 days for a patient to recover from COVID-19 base on Poisson distribution

```
# Question 1
library(MASS)
fit <- fitdistr(df$Days, "Poisson")
estimated_lambda <- fit$estimate["lambda"]
print(estimated_lambda)
```

```
> print(estimated_lambda)
lambda
15.556
```

2.

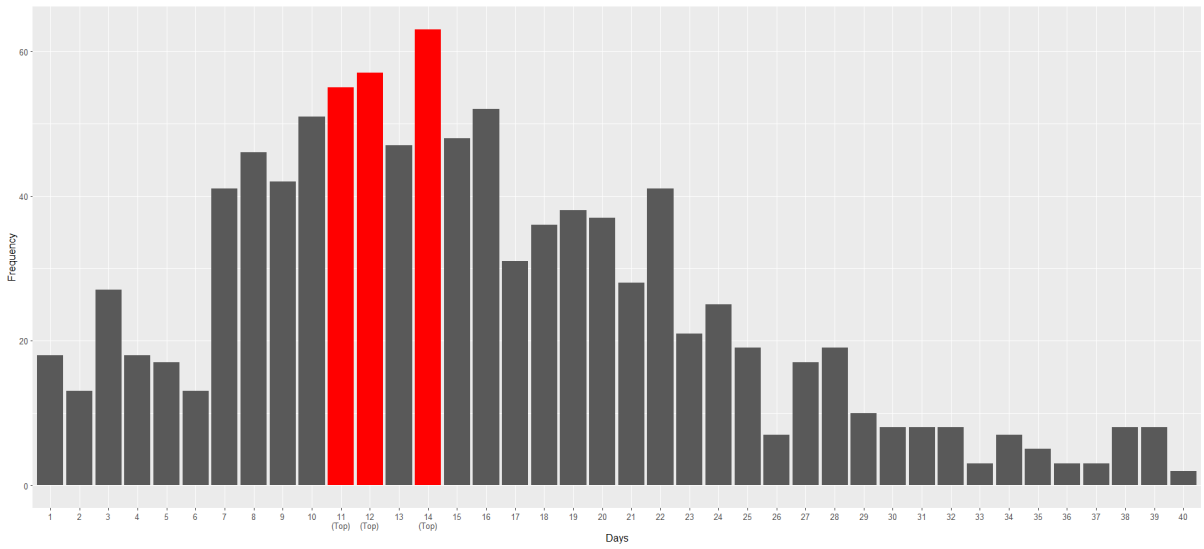
   a. The possibility can be calculated with R, and final answer is 0.0938464

```
# Question 2A
lambda <- 15.556
probability_10_or_less <- ppois(10, lambda)
print(probability_10_or_less)
```

```
> print(probability_10_or_less)
[1] 0.0938464
```

   b. We can know though the graph that, 11, 12 and 14 are the three most likely number of days it takes a patient to recover

```
#Question 2B
freq_df <- data.frame(table(df$Days))
top_values <- head(freq_df[order(-freq_df$Freq), ], 3)$Var1

ggplot(df, aes(x = factor(Days))) +
  geom_bar(stat = "count") +
  geom_bar(data = subset(df, Days %in% top_values),
           aes(x = factor(Days)),
           stat = "count", fill = "red") +
  scale_x_discrete(labels = function(x)
    ifelse(x %in% top_values, paste(x, "\n(Top)", sep = ""),
as.character(x))) +
  labs(x = "Days", y = "Frequency")
```



c. We can use function P(X = k) = (λ^k * e^(-λ)) / k! to solve this problem in R, and the result is 0.6115397

```
# Question 2C
# Function to calculate Poisson probability base on PMF
poisson_prob_range <- function(lambda_f, k_min, k_max) {
  probabilities <- numeric(k_max - k_min + 1)
  for (k in k_min:k_max) {
    probabilities[k - k_min + 1] <- (lambda_f^k * exp(-lambda_f)) /
factorial(k)
  }
  return(sum(probabilities))
}
# Total rate parameter for five individuals, Range is 60 to 80 days
probability <- poisson_prob_range(5 * lambda, 60, 80)
print(probability)
```

```
> print(probability)
[1] 0.6115397
```

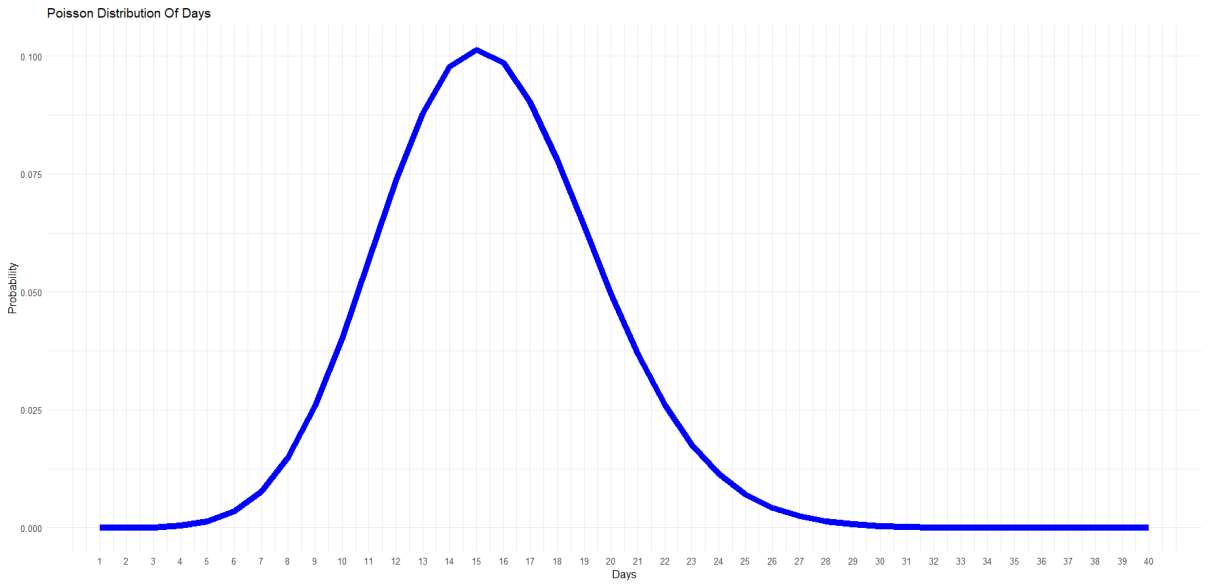
d. Using binomial distribution since we have multiple patients and based on the formula: P(X >= 3) = P(X = 3) + P(X = 4) + P(X = 5), the result is 0.8205065

```
# Question 2D

prob_after_14 <- 1 - ppois(13, lambda)

probability_three_or_more <- sum(dbinom(3:5, 5, prob_after_14))

print(probability_three_or_more)
```

```
> print(probability_three_or_more)
[1] 0.8205065
```

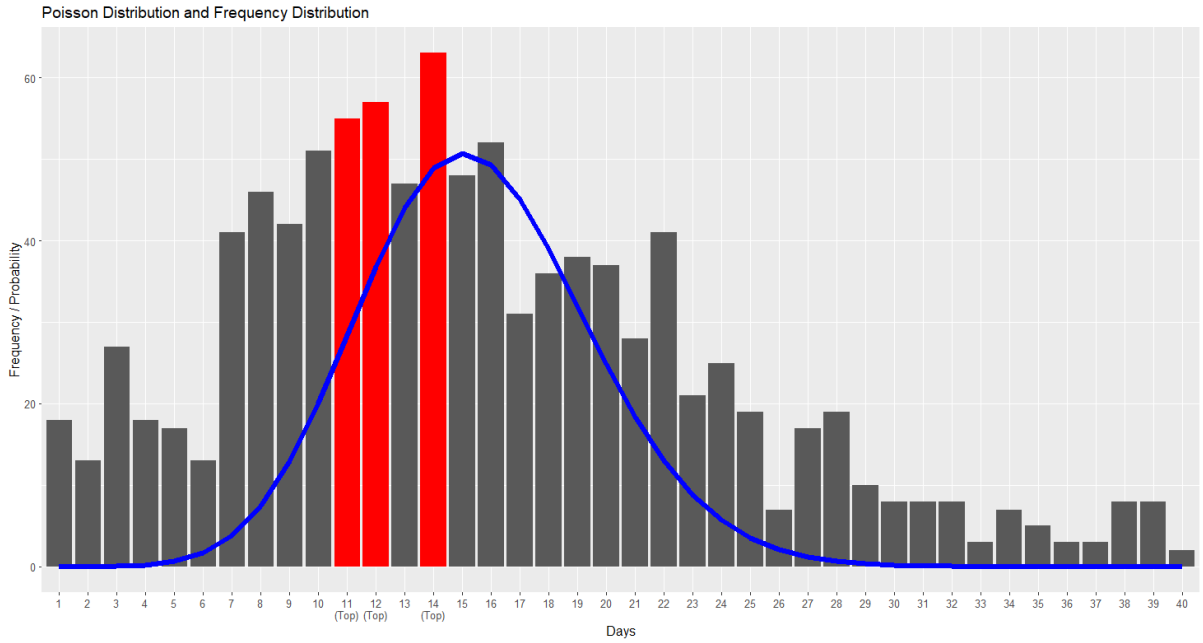
3. Start with drawing Poisson Distribution graph

```
# Question 3
ggplot() +
  geom_line(aes(x = df$Days, y = dpois(df$Days, lambda)), color = "blue", size = 3) +
  labs(title = "Poisson Distribution Of Days",
       x = "Days",
       y = "Probability") +
  scale_x_continuous(breaks = seq(min(df$Days), max(df$Days), by = 1))
```



We can force the Poisson distribution to scale and then merge it with the data distribution map

```
ggplot(df, aes(x = factor(Days))) +
  geom_bar(stat = "count") +
  geom_bar(data = subset(df, Days %in% top_values),
           aes(x = factor(Days)),
           stat = "count", fill = "red") +
  geom_line(aes(x = df$Days,
                y = 500*dpois(df$Days, lambda)),
            color = "blue", size = 2) +
  scale_x_discrete(labels = function(x)
    ifelse(x %in% top_values, paste(x, "\n(Top)", sep = ""), as.character(x))) +
  labs(title = "Poisson Distribution and Frequency Distribution",
       x = "Days",
       y = "Frequency / Probability")
```



As we can see, the Poisson distribution does not provide a perfect fit to the data, most people recover after 11-14 days but the top point of Poisson distribution is 15. However, the two distributions are generally similar, and the Poisson distribution does capture the general trend of the data. Therefore, I believe that the Poisson distribution is an appropriate model for the COVID recovery data.