Due Date: 15th October 2021
Instructions:

1. Answer ALL questions in the spaces allocated.
2. In this assignment, you are required to show all your working.
3. Your answers must be written in the spaces provided. You can adjust the spaces allocated for the answers if you need more space. You can type your answers if you wish.
4. *The lecturer maintains the right to call students in individually and ask them questions on the assignments. This may result in an adjustment of the final assignment grade.*
5. *Upload (i) Your R code (ii) Your Data and (3) A softcopy of your assignment on myelearning as a pdf. In Dropbox 1. DO NOT SUBMIT AS A SINGLE ZIP FILE with all the documents.*

1. QUESTION 1 Suppose that $X_1, X_2, ..., X_n$ are i.i.d. Poisson random variables whose distribution is a mixture of Poisson random variables with parameters $\lambda \wedge \mu$. We observe a Poisson random variable with mean λ with probability θ and a Poisson variable with mean μ with a probability $1 - \theta$.

   (a) Derive the EM Algorithm to estimate the parameters [10]

   This problem requires derivation of the EM algorithm for a mixture model of two Poisson. Suppose:

$$Y_1 \sim Poisson(\lambda_1) \text{ and } Y_2 \sim Poisson(\lambda_2)$$

   Let w be a Bernoulli random variable independent of Y_1 and Y_2 with probability of success π. The random variable X observed is therefore:

$$X = (1 - W)Y_1 + WY_2$$

   Let the parameter vector $\theta = (\lambda_1, \lambda_2, \pi)$. The PDF of the mixture of random variable X is:

$$\begin{gathered} f(x) = (1 - \pi)f_1(x) + \pi f_2(x) \\ -\infty < x < \infty \\ 0 \leq \pi \leq 1 \end{gathered}$$

   The unobserved data are r.v. identifying the distribution memberships, for $i = 1, .., n$ constitute a Bernoulli r.v.:

$$w_i = \begin{cases} 0 & \text{if } x_i \text{ has PDF } f_1(x) \\ 1 & \text{if } x_i \text{ has PDF } f_2(x) \end{cases} \tag{1}$$

   The complete likelihood function is:

$$\mathcal{L}^c(\theta|x, w) = \prod_{w_i=0} f_1(x_i) \prod_{w_i=1} f_2(x_i)$$

   Hence the log- of the complete likelihood is:

$$l^c(\theta|x, w) = \sum_{i=1}^{n} [(1 - w_i) \log f_1(x_i) + w_i \log f_2(x_i)]$$

For the **E**-step, we need the conditional expectation of w_i given $\tilde{x}, \theta_0$:

$$E_{\theta_0}\left[w|\theta_0, \tilde{x}\right] = P\left[w_i = 1|\theta_0, \tilde{x}\right]$$

In order to estimate this probability, Bayes' rule is used:

$$\gamma_i = \frac{\hat{\pi} f_{2,\theta_0}(x_i)}{(1-\hat{\pi}) f_{1,\theta_0}(x_i) + \hat{\pi} f_{2,\theta_0}(x_i)}$$

Replacing w_i with γ_i in the log of the complete likelihood:

$$Q(\theta|\theta_0, \tilde{x}) = \sum_{i=1}^{n} \left[(1-\gamma_i)\log f_1(x_i) + \gamma_i \log f_2(x_i)\right]$$

Re-substituting $f(x) = \frac{e^{\lambda}\lambda^x}{x!}$ into the above:

$$\begin{aligned}
Q(\theta|\theta_0, \tilde{x}) &= \sum_{i=1}^{n} \left[(1-\gamma_i)\log \frac{e^{-\lambda_1}\lambda_1^{x_i}}{x!} + \gamma_i \log \frac{e^{-\lambda_2}\lambda_2^{x_i}}{x!}\right] \\
&= \sum_{i=1}^{n} (1-\gamma_i)\left[-\lambda_1 + x_i \log \lambda_1 - \log x_i!\right] + \gamma_i \left[-\lambda_2 + x_i \log \lambda_2 - \log x_i!\right] \\
&= \left[-\lambda_i + x_i \log \lambda_1 - \log x_i!\right] - \lambda_i \left[-\lambda_1 + x_i \log \lambda_1 - \log x_i!\right] + \gamma_i \left[-\lambda_2 + x_i \log \lambda_2 - \log x_i!\right]
\end{aligned}$$

From the above, differentiate with respect to λ_i and set to zero:

$$\begin{aligned}
\frac{\delta Q}{\delta \lambda_1} &= \sum_{i=1}^{n} \left[-1 + \frac{x_i}{\lambda_1} + \gamma_i \left(1 - \frac{x_i}{\lambda_1}\right)\right] = 0 \\
\frac{\delta Q}{\delta \lambda_2} &= \sum_{i=1}^{n} \gamma_i \left[-1 + \frac{x_i}{\lambda_2}\right] = 0
\end{aligned}$$

Solving the above results in the following maximization equations, solving for λ_1:

$$\begin{aligned}
\sum_{i=1}^{n} \left[-1 + \frac{x_i}{\lambda_1} + \gamma_i \left(1 - \frac{x_i}{\lambda_1}\right)\right] &= 0 \\
\Rightarrow \sum_{i=1}^{n} \left[\frac{1}{\lambda_1}(x_i - x_i\gamma_i) + (\gamma_i - 1)\right] &= 0 \\
\frac{1}{\lambda_i} \sum_{i}^{n} x_i(1-\gamma_i) &= \sum_{i=1}^{n} (1 - x_i) \\
\therefore \lambda_1 &= \frac{\sum_i^n x_i(1-\gamma_i)}{\sum_i^n (1-\gamma_i)}
\end{aligned}$$

Similarly, solving for λ_2:

$$\begin{aligned}
\sum_{i}^{n} -\gamma_i \left[-\gamma_i + \frac{x_i}{\lambda_2}\right] &= 0 \\
\therefore \lambda_2 &= \frac{\sum_i^n \gamma_i x_i}{\sum_i^n \gamma_i}
\end{aligned}$$

(b) Suppose the following data are observed $1, 2, 3, 8$ and 12 perform one iteration (BY HAND!!) of the EM algorithm to estimate the parameters. You can use an ordinary calculator OR the calculator facilities in R. [10]

Using the above derived equations, the following table lists the results of the E-M algorithm for a single iteration, choosing $\theta = 0.5$, $\lambda = 2$ and $\mu = 8$ as the starting values. Firstly solving the expectation step:

$$\gamma_i = \frac{\hat{\pi} f_{2,\lambda_0}(x_i)}{(1-\hat{\pi}) f_{1,\mu_0}(x_i) + \hat{\pi} f_{2,\lambda_0}(x_i)}$$

Setting $\theta = 0.5$ and finding γ_0 for all values of x:

$$\gamma_0 = \frac{0.5 f_{2,\lambda_0}(x_i)}{(1-0.5) f_{1,\mu_0}(x_i) + 0.5 f_{2,\lambda_0}(x_i)} = 2.821$$

The new value for $\hat{\theta}$ is now $\frac{\gamma_0}{n} = \frac{2.82}{5} = 0.564$ Performing the maximization step:

$$\lambda_1 = \frac{\sum_i^n x_i(1-\gamma_i)}{\sum_i^n (1-\gamma_i)} = 1.968$$

$$\mu_1 = \frac{\sum_i^n \gamma_i x_i}{\sum_i^n \gamma_i} = 9.385$$

(c) Write R code to perform the EM Algorithm and include in the R code an appropriate stopping tolerance level. [10]

```
e_step <- function(x, mean_1, mean_2, pi) {
    return(
        (pi * dpois(x, mean_2, log=FALSE))/
        ((1-pi)*dpois(x, mean_1, log=FALSE) + pi*dpois(x, mean_2, log=FALSE))
    )
}

m_step <- function(x, gamma) {
    mean_1 = sum(x*(1-gamma))/sum(1-gamma)
    mean_2 = sum(x*gamma)/sum(gamma)

    return(c(mean_1, mean_2))
}

e_m <- function(x, mean_1_0, mean_2_0, pi_0, tol=1e-10, max_iter=1000) {
    # specifies change of means at every iteration
    delta <- Inf
    # wrap both means into vector
    means <- c(mean_1_0, mean_2_0)
    pi <- pi_0
    iter <- 1


    while (delta > tol) {
        # update new gamma values in expectation step
        gamma_prev <- e_step(x, means[1], means[2], pi)
        # use updated gamma to maximize both means in  maximization step
        new_means <- m_step(x, gamma_prev)
        means <- new_means
        # update pi (proportion)
        pi <- sum(gamma_prev)/length(x)

        # calculate change in old and new estimates
        delta <- abs(new_means[1]-means[1] + new_means[2]-means[2])
        iter <- iter + 1
```

```
        # if gradients explode, return error
        if (is.na(delta)) {
            return("ERROR: EXPLOSION")
        }

        # if more than max_iter or change is below tolerance, stop and return
        if (iter >= max_iter | delta < tol) {
            return(new_means)
        }

    }
    return(new_means)
}
```

(d) Generate 100 values from a Poisson distribution with mean 5 and 900 values from a Poisson distribution with mean 10. Amalgamate those 1000 values into a single column. Then, run the R code you constructed to see how well your algorithm is able to estimate the parameters. [10]

```
set.seed(12020569)
# Set means for both distributions
means <- c(5, 10)

# let number of values generated be n
n <- c(100, 900)

# generate 1000 samples from distributions in one column
x <- c(rpois(n[1], means[1]), rpois(n[2], means[2]))

e_m(x, 2, 7, 0.5)
```

The output from above yielded the following results after 100 iterations

Parameter	Actual	Estimated
λ_1	5	3.70450652859229
λ_2	10	9.95064324920369
π	0.9	0.932335254876982

2. QUESTION 2.

(a) Show that the function illustrated in Table 1 represents the joint probability mass function of the two discrete random variables X and Y? Justify your answer [2 marks]

f(x,y)		x		
		1	**2**	**3**
y	**1**	1/20	2/20	1/20
	2	4/20	7/20	5/20

In order to show that the above represents a joint PMF, the individual probabilities must sum to unity. From the above:

$$\frac{1}{20} + \frac{2}{20} + \frac{1}{20} + \frac{4}{20} + \frac{7}{20} + \frac{5}{20} = \frac{20}{20} = 1$$

(b) Find $E(XY)$ by hand. Leave your answer in exact form. [3]

Let $g(X,Y) = XY$ be a function of of the joint PMF. In order to find the expectation:

$$\begin{aligned} E(XY) = E[g(X,Y)] &= \sum_x \sum_y g(x,y)p(x,y) \\ \Rightarrow E[XY] &= \sum_x \sum_y xy \cdot p(x,y) \\ &= (1)(1)\frac{1}{20} + (1)(2)\frac{4}{20} + (2)(1)\frac{2}{20} + (2)(2)\frac{7}{20} + (3)(1)\frac{1}{20} + (3)(2)\frac{5}{20} \\ &= \frac{1}{20} + \frac{8}{20} + \frac{4}{20} + \frac{28}{20} + \frac{3}{20} + \frac{30}{20} \\ &= \frac{74}{20} \end{aligned}$$

(c) Write an appropriate algorithm to generate pairs of random variables X and Y from the joint PMF in Table 1. [5]

In order to sample from the joint PMF as described in table 1, a random number is generated from the Uniform distribution in the interval $[0,1]$. An index k is deterermined such that

$$\sum_{j=1}^{k-1} p_j \le U < \sum_{j=1}^{k} p_j$$

Marginalizing first for X then Y:

$$\begin{aligned} P_X(x) &= \sum_{y_j \in R_Y} P_{XY}(x, y_j) \\ \Rightarrow P_X(1) &= P_{XY}(1,1) + P_{XY}(1,2) \\ &= \frac{1}{20} + \frac{4}{20} = \frac{5}{20} \end{aligned}$$

Similarly:

$$P_X(x) = \begin{cases} \frac{5}{20} & x = 1 \\ \frac{9}{20} & x = 2 \\ \frac{6}{20} & x = 3 \end{cases}$$

Marginalising for Y results in the following summation:

$$\begin{aligned} P_Y(y) &= \sum_{x_i \in R_X} P_{XY}(x_i, y) \\ \Rightarrow P_Y(1) &= P_{XY}(1,1) + P_{XY}(2,1) + P_{XY}(3,1) \\ &= \frac{1}{20} + \frac{2}{20} + \frac{1}{20} = \frac{4}{20} \end{aligned}$$

Similarly:

$$P_Y(y) = \begin{cases} \frac{4}{20} & y = 1 \\ \frac{16}{20} & y = 2 \end{cases}$$

The cumulative probabilities are calculated for both X and Y, where $F(x) = P(X \le x)$, for any $x \in \mathbb{R}$. Given this definition:

$$F_X(x) = \begin{cases} \frac{5}{20} & x = 1 \\ \frac{14}{20} & x = 2 \\ \frac{20}{20} & x = 3 \end{cases}$$

Similarly for Y:

$$F_Y(y) = \begin{cases} \frac{4}{20} & y = 1 \\ \frac{20}{20} & y = 2 \end{cases}$$

In order to sample from the joint distribution, random samples are drawn from a Uniform distribution in the range $(0, 1)$. Depending on the interval within which random variables of this uniform lie, the corresponding x and y values are produced.

This is known as **Inverse Transform Sampling**

(d) i) Implement the algorithm in R and generate 100,000 pairs of random variables (X,Y) from the distribution. (PASTE R CODE) [5]

```
set.seed(46692)

N = 100000
# generate N Uniform Samples
uniform_X <- runif(N, min=0, max=1)
uniform_Y <- runif(N, min=0, max=1)

# create matrix to store resulting samples
joint <- matrix(nrow=N, ncol=2)


for (i in seq_len(N)) {
    u_X = uniform_X[i]
    u_Y = uniform_Y[i]

    # getting the X value, using the CDF to determine which X value to 'choose'
    # if the value from the uniform is less than 5/20, choose X=1
    if (u_X < 5/20) {
        joint[i, 1] = 1
    }
    # if the value from the uniform is greater than 5/20 and less than 14/20, c
    else if (u_X >= 5/20 && u_X < 14/20) {
        joint[i, 1] = 2
    }
    # if the value from the uniform is greather than 14/20, choose X=3
    else {
       joint[i, 1] = 3
    }

    # getting the Y value, using the CDF to determine which Y value to 'choose'
    # if the value from the uniform is less than 4/20, choose Y=1
    if (u_Y < 4/20) {
        joint[i, 2] = 1
    }
    # if the value from the uniform is greater than 5/20, choose Y=2
    else {
        joint[i, 2] = 2
    }
}
```

ii) Estimate $E(X, Y)$ from the 100,000 pairs of values you simulated (USE R!). [3]

```
set.seed(46692)

P <- matrix(nrow=2, ncol=3)
P[1, ] <- c(1/20,2/20,1/20)
P[2, ] <- c(4/20, 7/20, 5/20)
```

```
# using joint probabilities to find expectation
#  this is the sample mean
expectation = 0
for (idx in seq_len(N)) {
    x <- joint[idx, 1]
    y <- joint[idx, 2]
    P_xy = P[y,x]
    expectation = expectation + x * y
}
expectation = expectation/N
expectation
```

iii) Compare the estimated mean of $E(XY)$ with the true mean you found in (b) [2]
The estimated expectation found using Inverse Transform sampling was 3.68761, while the expectation found using analytic methods was 3.7. This difference of around 0.012 may be due to the number of samples derived. As the number of samples (for the uniform) approach infinity, the estimated expected value is presumed to approach the actual expected value owing to the central limit theorem, where the standard error tends to zero as the number of samples increase.