STAT432 Assignment 2

Rory Sarten 301005654

21 August 2020

Question 1

a)

These are independent binomial processes (despite happening in sequence), so the three trapping occasions can be denoted:

$n_1 \sim Binomial(N, \theta) = \binom{N}{n_1}\theta^{n_1}(1-\theta)^{N-n_1}$

$n_2 \sim Binomial(N-n_1, \theta) = \binom{N-n_1}{n_2}\theta^{n_2}(1-\theta)^{N-n_1-n_2}$

$n_3 \sim Binomial(N-n_1-n_2, \theta) = \binom{N-n_1-n_2}{n_3}\theta^{n_3}(1-\theta)^{N-n_1-n_2-n_3}$

So the probability of catching $n_1 + n_2 + n_3$ rats is given by

$Binomial(N, \theta) \times Binomial(N-n_1, \theta) \times Binomial(N-n_1-N-2, \theta)$

b)

$$\begin{aligned}
\ell(N,\theta) = &\ln[\frac{N!}{n_1(N-n_1)!}\theta^{n_1}(1-\theta)^{N-n_1} \\
&\times \frac{(N-n_1)!}{n_2(N-n_1-n_2)!}\theta^{n_2}(1-\theta)^{N-n_1-n_2} \\
&\times \frac{(N-n_1-n_2)!}{n_3(N-n_1-n_2-n_3)!}\theta^{n_3}(1-\theta)^{N-n_1-n_2-n_3}]
\end{aligned}$$

$$\begin{aligned}
\ell(N,\theta) = &\ln N! - [\ln n_1 + \ln(N-n_1)] + n_1 \ln\theta + (N-n_1)\ln(1-\theta) + \\
&\ln(N-n_1)! - [\ln n_2 + \ln(N-n_1-n_2)] + n_2\ln\theta + (N-n_1-n_2)\ln(1-\theta) + \\
&\ln(N-n_1-n_2)! - [\ln n_3 + \ln(N-n_1-n_2-n_3)] + n_3\ln\theta + (N-n_1-n_2-n_3)\ln(1-\theta)
\end{aligned}$$

We can reduce the components

$n_1\ln\theta + n_2\ln\theta + n_3\ln\theta = (n_1+n_2+n_3)\ln\theta$

$(N-n_1)\ln(1-\theta) + (N-n_1-n_2)\ln(1-\theta) + (N-n_1-n_2-n_3)\ln(1-\theta) = (3N-3n_1-2n_2-n_3)\ln(1-\theta)$

$\ln N! + \ln(N-n_1)! ...$

Gives

$$\ln N! - \ln(N-n_1-n_2-n_3)! + (n_1+n_2+n_3)\ln\theta + (3N-3n_1-2n_2-n_3)\ln(1-\theta)$$

finish that bit of algebra somehow

c)

```
llfunc <- function(par, n1, n2, n3) {
  N <- par[1]
  theta <- par[2]
  lfactorial(N) - lfactorial(N - n1 - n2 - n3) +
    sum(n1, n2, n3)*log(theta) +
    (3*N - 3*n1 - 2*n2 - n3)*log(1 - theta)
}
```

```r
n1 <- 82L
n2 <- 54L
n3 <- 23L
## since N >= n1 + n2 + n3
N_0 <- n1 + n2 + n3
par_start <- c(N_0, 0.5)

optim_fit <- optim(par = par_start,
                   fn = llfunc,
                   n1 = n1, n2 = n2, n3 = n3,
                   lower = c(N_0, 1e-4),
                   upper = c(Inf, 1),
                   method = "L-BFGS-B",
                   control = list(fnscale = -1),
                   hessian = TRUE)

MLE <- optim_fit$par
SE <- sqrt(diag(solve(-optim_fit$hessian)))
LowerBound <- MLE - qnorm(0.975) * SE
UpperBound <- MLE + qnorm(0.975) * SE
results <- cbind(MLE, SE, LowerBound, UpperBound) %>% round(2)
rownames(results) <- c("N", "theta")
results
```

```
##            MLE    SE LowerBound UpperBound
## N       189.43 13.23     163.49     215.36
## theta     0.45  0.06       0.34       0.57
```

The maximum likelihood point $\hat{N} = 189.43$ with confidence interval $(163.49, 215.36)$

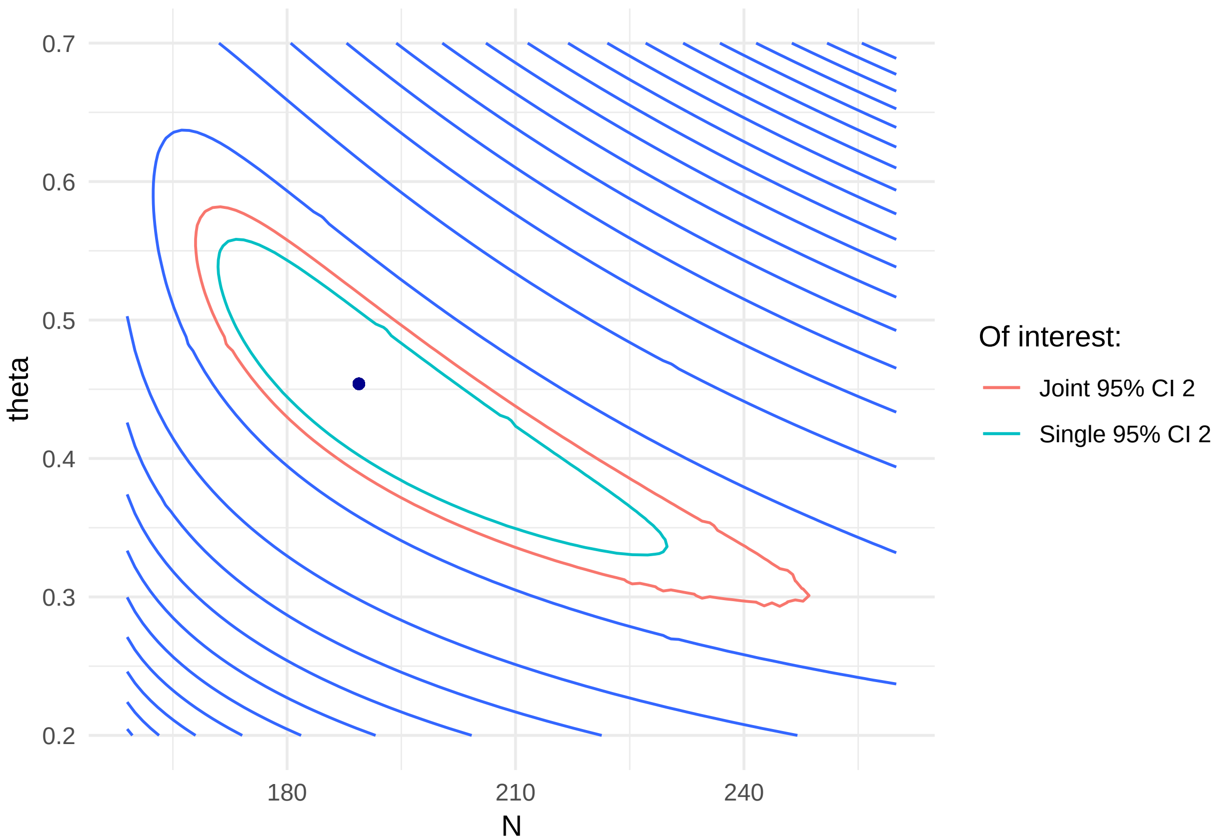
The maximum likelihood point $\hat{\theta} = 0.45$ with confidence interval $(0.34, 0.57)$

d)

Contour map of (N, θ) surface

```r
n_vals <- seq(from = n1+n2+n3, to = 260, length = 100)
theta_vals <- seq(from = 0.2, to = 0.7, length = 100)

combos <- expand.grid(n_vals, theta_vals)
surface <- apply(expand.grid(n_vals, theta_vals), 1, llfunc, n1 = n1, n2 = n2, n3 = n3) %>% round(2)
outcome <- cbind(combos, surface)
names(outcome) <- c("N", "theta", "value")

ggplot(data = outcome, mapping = aes(N, theta)) +
  geom_contour(aes(z = value),
             breaks = seq(from = min(outcome$value), to = max(outcome$value), by = 12)) +
  geom_contour(aes(z = value,
                   colour = factor(..level.. == max(surface) - 1.92,
                                   levels = c(F, T),
                                   labels = c("Single 95% CI "))),
             breaks = max(surface) - 1.92) +
  geom_contour(aes(z = value,
                   colour = factor(..level.. == max(surface) - 3,
                                   levels = c(F, T),
                                   labels = c("Joint 95% CI "))),
```

```
               breaks = max(surface) - 3) +
  geom_point(mapping = aes(x = MLE[1], y = MLE[2]), colour = "darkblue") +
  labs(colour = "Of interest:") +
  #ggtitle("Contour map of (N, \u03B8) surface") +
  #ylab("\u03B8") +
  theme_minimal()
```



e)

$$\frac{\partial \ell}{\partial \theta} = \frac{n_1 + n_2 + n_3}{\theta} - \frac{3N - 3n_1 - 2n_2 - n_3}{1 - \theta}$$

$$= \frac{(n_1 + n_2 + n_3) - (n_1 + n_2 + n_3)\theta - (3N - 3n_1 - 2n_2 - n_3)\theta}{\theta(1 - \theta)}$$

Setting this to 0

$$(n_1 + n_2 + n_3) - [(n_1 + n_2 + n_3) + (3N - 3n_1 - 2n_2 - n_3)]\theta = 0$$

$$[(n_1 + n_2 + n_3) + (3N - 3n_1 - 2n_2 - n_3)]\theta = (n_1 + n_2 + n_3)$$

$$\theta = \frac{(n_1 + n_2 + n_3)}{(n_1 + n_2 + n_3) + (3N - 3n_1 - 2n_2 - n_3)} = \frac{(n_1 + n_2 + n_3)}{3N - 3n_1 + n_1 - 2n_2 + n_2 - n_3 + n_3}$$

$$\hat{\theta} = \frac{n_1 + n_2 + n_3}{3N - 2n_1 - n_2}$$

f)

We have

$$\hat{\theta} = \frac{n_1 + n_2 + n_3}{3N - 2n_1 - n_2}$$

and

$$1 - \hat{\theta} = \frac{3N - 2n_1 - n_2}{3N - 2n_1 - n_2} - \frac{n_1 + n_2 + n_3}{3N - 2n_1 - n_2} = \frac{3N - 3n_1 - 2n_2 - n_3}{3N - 2n_1 - n_2}$$

So we can rewrite the original kernel of the log likelihood in terms of N as:

$$\ell(N) = \ln N! - \ln(N - n_1 - n_2 - n_3)! + (n_1 + n_2 + n_3) \ln\left(\frac{n_1 + n_2 + n_3}{3N - 2n_1 - n_2}\right) + \\ (3N - 3n_1 - 2n_2 - n_3) \ln\left(\frac{3N - 3n_1 - 2n_2 - n_3}{3N - 2n_1 - n_2}\right)$$

$$\begin{aligned}\ell(N) = {} & \ln N! - \ln N! - \ln(N - n_1 - n_2 - n_3)! + \\ & (n_1 + n_2 + n_3) \ln(n_1 + n_2 + n_3) - (n_1 + n_2 + n_3) \ln(3N - 2n_1 - n_2) + \\ & (3N - 3n_1 - 2n_2 - n_3) \ln(3N - 3n_1 - 2n_2 - n_3) - (3N - 3n_1 - 2n_2 - n_3) \ln(3N - 2n_1 - n_2)\end{aligned}$$

$$\begin{aligned}\ell(N) = {} & \ln N! - \ln N! - \ln(N - n_1 - n_2 - n_3)! + \\ & (3N - 3n_1 - 2n_2 - n_3) \ln(3N - 3n_1 - 2n_2 - n_3) - \\ & (n_1 + n_2 + n_3 + 3N - 3n_1 - 2n_2 - n_3) \ln(3N - 2n_1 - n_2)\end{aligned}$$

Therefore

$$\begin{aligned}\ell(N) = {} & \ln N! - \ln N! - \ln(N - n_1 - n_2 - n_3)! + \\ & (3N - 3n_1 - 2n_2 - n_3) \ln(3N - 3n_1 - 2n_2 - n_3) - \\ & (3N - 2n_1 - n_2) \ln(3N - 2n_1 - n_2)\end{aligned}$$

g)

```r
llfunc_N <- function(N, n1, n2, n3) {
  lfactorial(N) - lfactorial(N - n1 - n2 - n3) +
    (3*N - 3*n1 - 2*n2 - n3)*log(3*N - 3*n1 - 2*n2 - n3) -
    (3*N - 2*n1 - n2)*log(3*N - 2*n1 - n2)
}

n1 <- 82L
n2 <- 54L
n3 <- 23L
## since N >= n1 + n2 + n3
N_0 <- n1 + n2 + n3
N_start <- c(N_0)

optim_fit <- optim(par = N_start,
                   fn = llfunc_N,
                   n1 = n1, n2 = n2, n3 = n3,
                   lower = c(N_0, 1e-4),
                   upper = c(Inf, 1),
                   method = "L-BFGS-B",
                   control = list(fnscale = -1),
                   hessian = TRUE)

N_hat <- optim_fit$par
log_max <- optim_fit$value

## calculate 95% CI
crit_point <- log_max - 0.5 * qchisq(0.95, df = 1)

ci_func <- function(critical_value, ...) {
  llfunc_N(...) - critical_value
}

lower_ci <- uniroot(ci_func, interval = c(N_0, N_hat),
                    n1 = n1, n2 = n2, n3 = n3,
                    critical_value = crit_point)$root
upper_ci <- uniroot(ci_func, interval = c(N_hat, 300),
                    n1 = n1, n2 = n2, n3 = n3,
                    critical_value = crit_point)$root

cbind(N_hat, lower_ci, upper_ci)
```

```
##         N_hat lower_ci upper_ci
## [1,] 189.4255  170.918 229.9337
```

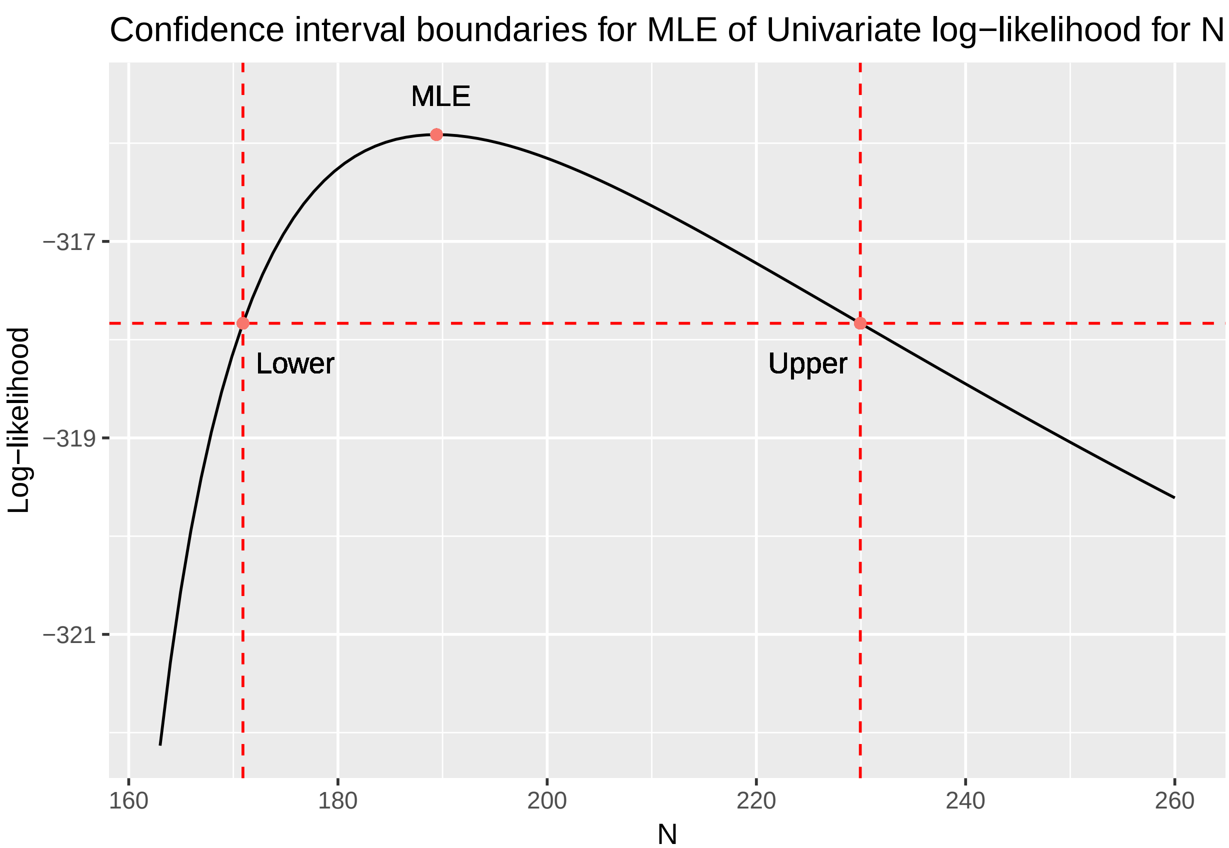
The univariate log-likelihood estimate for $\hat{N} = 189.4254905$

with 95% confidence interval (170.92, 229.93).

Compared with the confidence interval in (c), this interval is similar in size, but is no longer symmetric around the MLE.

```r
n_vals <- seq(from = 163, to = 260, length = 100)
data <- data.frame(x = n_vals,
                   y = llfunc_N(n_vals, n1 = n1, n2 = n2, n3 = n3))
```

```r
ggplot(data = data, mapping = aes(x = x, y = y)) +
  geom_line() +
  geom_point(aes(x = N_hat, y = log_max, color = "red")) +
  geom_text(aes(x = N_hat + 0.4, y = log_max + 0.4, label = "MLE")) +
  geom_hline(yintercept=crit_point, linetype="dashed", color = "red") +
  geom_vline(xintercept=lower_ci, linetype="dashed", color = "red") +
  geom_vline(xintercept=upper_ci, linetype="dashed", color = "red") +
  geom_point(aes(x = lower_ci, y = crit_point, color = "red")) +
  geom_point(aes(x = upper_ci, y = crit_point, color = "red")) +
  geom_text(aes(x = lower_ci + 5, y = crit_point - 0.4, label = "Lower")) +
  geom_text(aes(x = upper_ci - 5, y = crit_point - 0.4, label = "Upper")) +
  ggtitle("Confidence interval boundaries for MLE of Univariate log-likelihood for N") +
  xlab("N") + ylab("Log-likelihood") +
  theme(legend.position="none")
```



h)

Given the 95% profile liklihood-ratio for N calculated above, and that we know a total of 159 rats were caught over the three occasions, we can our confidence interval for R to be $(lower - 159, upper - 159) = (171 - 159, 230 - 159) = (12, 71)$ (with rounding to the nearest whole rat).

Question 2

a)

$$
\begin{aligned}
\ell(\lambda, x) &= \log\left(\prod_{i=1}^{n}\left(\frac{e^{-\lambda}\lambda^{x_i}}{x_i!}\right)\right) \\
&= \sum_{i=1}^{n}(-\lambda + x_i \ln\lambda - \ln(x_i!)) \\
&= -n\lambda + \left(\sum_{i=1}^{n} x_i\right)\ln\lambda - \sum_{i=1}^{n}\ln(x_i!)
\end{aligned}
$$

The last term can be dropped, giving the kernel of the log likelihood

$$
-n\lambda + \left(\sum_{i=1}^{n} x_i\right)\ln\lambda
$$

Differentiating gives

$$
\frac{d\ell}{d\lambda} = -n + \frac{\sum_{i=1}^{n} x_i}{\lambda}
$$

Setting this to 0 gives

$$
\hat{\lambda} = \frac{\sum_{i=1}^{n} x_i}{n}
$$

We can calculte the standard error by taking the second derivative of the log-likelihood at the MLE and calculating the negative inverse

$$
\frac{d^2\ell}{d\lambda^2} = -\frac{\sum_{i=1}^{n} x_i}{\lambda^2}
$$

$$
\left.\frac{d^2\ell}{d\lambda^2}\right|_{\hat{\lambda}} = -\left[\frac{n}{\sum_{i=1}^{n} x_i}\right]^2 \times \sum_{i=1}^{n} x_i = -\frac{n^2}{\sum_{i=1}^{n} x_i}
$$

The negative inverse is

$$
\frac{\sum_{i=1}^{n} x_i}{n^2}
$$

So we can calculate the standard error as

$$
SE[\hat{\lambda}] = \frac{\sqrt{\sum_{i=1}^{n} x_i}}{n}
$$

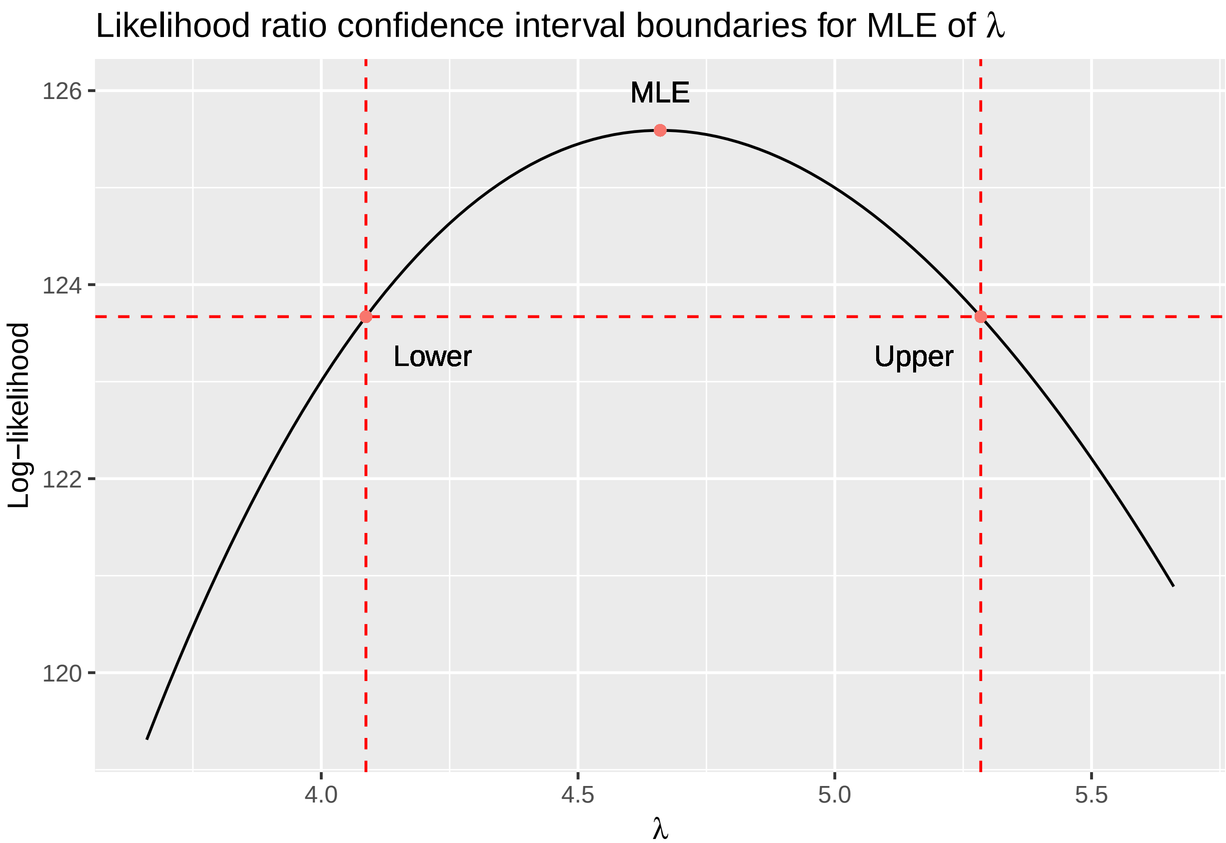
```r
counts <- readr::read_rds("counts.rds")
MLE <- sum(counts)/length(counts) # or mean(counts)
SE <- (sqrt(sum(counts))/length(counts)) %>% round(digits = 2)
LOWER <- (MLE - SE * 1.96) %>% round(digits = 2)
UPPER <- (MLE + SE * 1.96) %>% round(digits = 2)
cbind(MLE, SE, LOWER, UPPER)
```

```
##       MLE   SE LOWER UPPER
## [1,] 4.66 0.31  4.05  5.27
```

```r
poisson_llfunc <- function(lambda, counts) {
  -(length(counts) * lambda) + sum(counts) * log(lambda)
}

optim_fit <- optim(par = 1,
                   fn = poisson_llfunc,
                   counts = counts,
                   lower = 1e-3,
                   upper = Inf,
                   method = "L-BFGS-B",
                   control = list(fnscale = -1),
                   hessian = TRUE)
MLE <- optim_fit$par

n_vals <- seq(from = MLE - 1, to = MLE + 1, length = 100)
data <- data.frame(x = n_vals,
                   y = poisson_llfunc(n_vals, counts))

log_max <- optim_fit$value
crit_point <- optim_fit$value - 0.5 * qchisq(0.95, df = 1)
ci_func <- function(critical_point, ...) poisson_llfunc(...) - critical_point

lower_ci <- uniroot(ci_func,
                    interval = c(0, MLE),
                    counts = counts,
                    critical_point = crit_point)$root
upper_ci <- uniroot(ci_func,
                    interval = c(MLE, MLE*5),
                    counts = counts,
                    critical_point = crit_point)$root
cbind(MLE, lower_ci, upper_ci)
```

```
##       MLE lower_ci upper_ci
## [1,] 4.66 4.086982 5.284223
```

```r
ggplot(data = data, mapping = aes(x = x, y = y)) +
  geom_line() +
  geom_point(aes(x = MLE, y = log_max, color = "red")) +
  geom_text(aes(x = MLE, y = log_max + 0.4, label = "MLE")) +
  geom_hline(yintercept=crit_point, linetype="dashed", color = "red") +
  geom_vline(xintercept=lower_ci, linetype="dashed", color = "red") +
  geom_vline(xintercept=upper_ci, linetype="dashed", color = "red") +
  geom_point(aes(x = lower_ci, y = crit_point, color = "red")) +
  geom_point(aes(x = upper_ci, y = crit_point, color = "red")) +
  geom_text(aes(x = lower_ci + 0.13, y = crit_point - 0.4, label = "Lower")) +
  geom_text(aes(x = upper_ci - 0.13, y = crit_point - 0.4, label = "Upper")) +
  #ggtitle("Likelihood ratio confidence interval boundaries for MLE of", expression(lambda)) +
  ggtitle(expression(paste("Likelihood ratio confidence interval boundaries for MLE of ", lambda))) +
  xlab(expression(lambda)) + ylab("Log-likelihood") +
  theme(legend.position="none")
```



c)

d)

$$\ell(\boldsymbol{\lambda}, \boldsymbol{\pi}|\boldsymbol{x}) = \sum_{i=1}^{n} \ln \left(\sum_{g=1}^{G} \pi_g \frac{e^{-\lambda}\lambda^{x_i}}{x_i!} \right)$$

$$\ell_C(\boldsymbol{\lambda}, \boldsymbol{\pi}|\boldsymbol{x}, Z) = \sum_{i=1}^{n} \ln \left(\sum_{g=1}^{G} z_{ig}\pi_g \frac{e^{-\lambda}\lambda^{x_i}}{x_i!} \right) \text{ where } z_{ig} \text{ has value 0 or 1}$$

ii)

If we are looking at each $g = 1, ..., G$ then we can ignore the π_g term, as we are not concerned with the proportion in the group g. We can also exploit that z_{ig} is 0 or 1, but treating it as an exponent.

$$\begin{aligned}
\ell_C(\boldsymbol{\lambda}, \boldsymbol{\pi}|\boldsymbol{x}, Z) &= \sum_{i=1}^{n} \ln \left(z_i \frac{e^{-\lambda}\lambda^{x_i}}{x_i!} \right) \\
&= \sum_{i=1}^{n} \ln \left(\frac{e^{-\lambda}\lambda^{x_i}}{x_i!} \right)^{z_i} \\
&= \sum_{i=1}^{n} z_i \ln \left(\frac{e^{-\lambda}\lambda^{x_i}}{x_i!} \right) \\
&= \sum_{i=1}^{n} z_i \left(-\lambda + x_i \ln \lambda - \ln x_i! \right) \\
&= -\lambda \sum_{i=1}^{n} z_i + \sum_{i=1}^{n} z_i x_i \ln \lambda - \sum_{i=1}^{n} z_i \ln x_i!
\end{aligned}$$

$$\frac{\partial \ell_C}{\partial \lambda} = -\sum_{i=1}^{n} z_i + \frac{\sum_{i=1}^{n} z_i x_i}{\lambda}$$

Setting equal to 0 and solving

$$\frac{\sum_{i=1}^{n} z_i x_i}{\lambda} = \sum_{i=1}^{n} z_i$$

therefore for each group g

$$\hat{\lambda_g} = \frac{\sum_{i=1}^{n} z_{ig} x_i}{\sum_{i=1}^{n} z_{ig}}$$

iv)