

# Stat 3202 Lab 5

Nathan Johnson.9254

2023-02-20

**Problem 1: a.**

$$L(\mu|x_1, \dots, x_n, \sigma^2) = (2\pi)^{\frac{n}{2}} e^{-\frac{1}{2} \sum_{i=1}^n (x_i - \mu)^2}$$

**b.**

```
normaldist = function(mu, data) {  
  n = length(data)  
  likelihood = (2*pi)^(-n/2) * exp(-1/2 * sum((data-mu)^2))  
  return(likelihood)  
}
```

**c.**

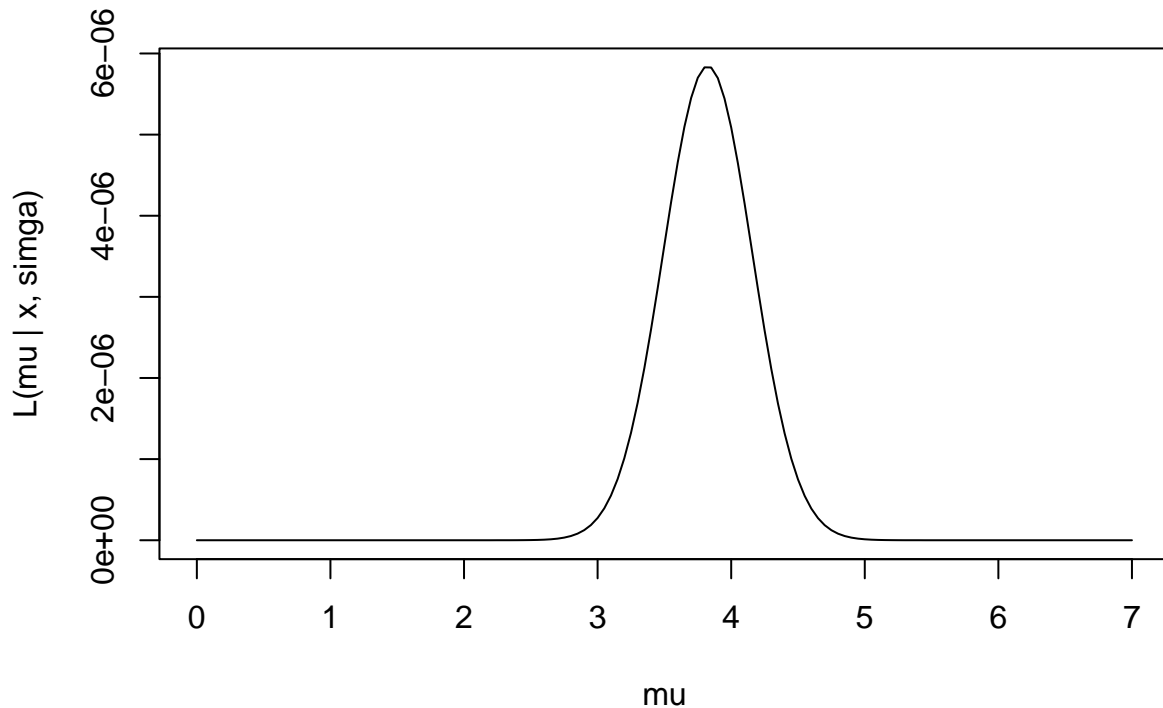
```
frogs = c(4.57, 4.73, 2.24, 4.16, 4.53, 3.79, 2.48, 4.74, 3.18)  
  
xbar = mean(frogs)  
  
print(xbar)
```

```
## [1] 3.824444
```

**d.**

```
mus = seq(0, 7, by = 0.05)  
likelihood.mus = c()  
  
for(i in 1:length(mus)) {  
  likelihood.mus[i] = normaldist(mus[i], frogs)  
}  
  
plot(mus, likelihood.mus, type = 'l', xlab = "mu", ylab = "L(mu | x, sigma)", main = "Normal Likelihood")
```

## Normal Likelihood of Frog Weights



e.

```
optimize(f = normaldist,  
         interval = c(0,7),  
         data = frogs,  
         maximum = TRUE  
         )
```

```
## $maximum  
## [1] 3.824445  
##  
## $objective  
## [1] 5.844802e-06
```

f. The `$maximum` is the  $\mu$  value that maximizes the likelihood function. The `$objective` is the value of the likelihood function at the `$maximum`.

g.

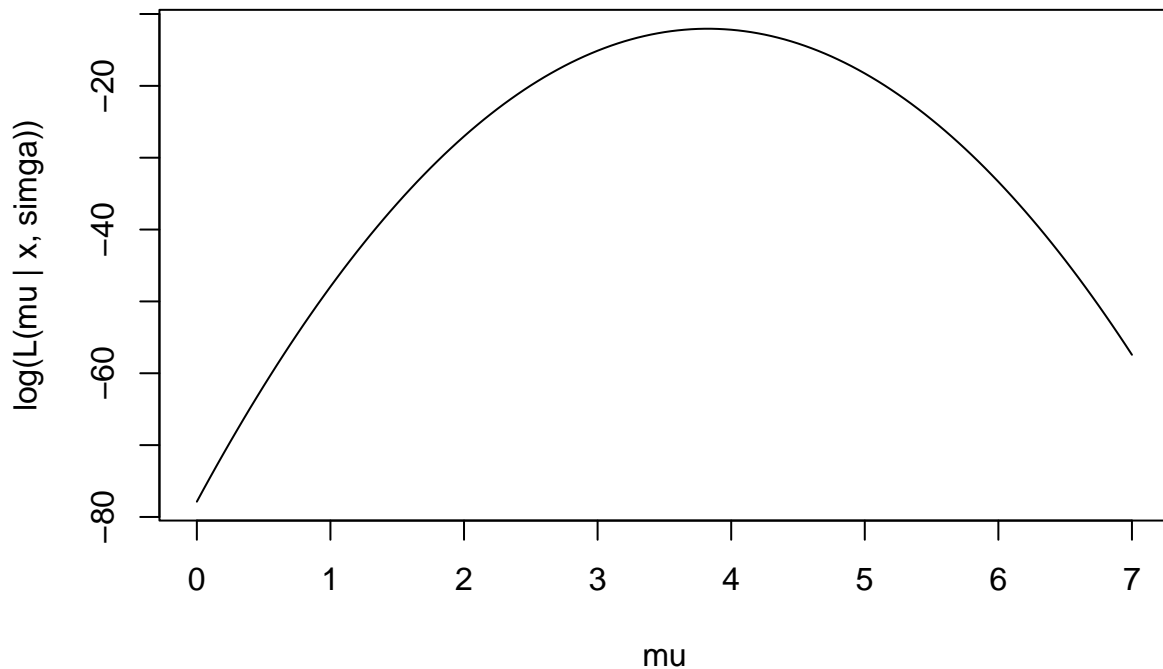
```
normaldist.log = function(mu, data) {  
  n = length(data)  
  likelihood = (2*pi)^(-n/2) * exp(-1/2 * sum((data-mu)^2))  
  return(log(likelihood))  
}  
  
mus = seq(0, 7, by = 0.05)
```

```
log.likelihood.mus = c()

for(i in 1:length(mus)) {
  log.likelihood.mus[i] = normaldist.log(mus[i], frogs)
}

plot(mus, log.likelihood.mus, type = 'l', xlab = "mu", ylab = "log(L(mu | x, sigma))", main = "Normal L.
```

## Normal Likelihood of Frog Weights



h.

```
optimize(f = normaldist.log,
  interval = c(0,7),
  data = frogs,
  maximum = TRUE
)
```

```
## $maximum
## [1] 3.824444
##
## $objective
## [1] -12.04996
```

i. The shapes are different because, well obviously, they are different functions. One is just the function. The other is the log of the function. The maximum are the same because where the functions increase hasn't changed, so the maximum will be the same.

Problem 2: a.

$$L(\lambda|x_1, \dots, x_n) = \lambda^n e^{-\lambda \sum_{i=1}^n x_i}$$

b.

```
expdist = function(lambda, data) {  
  n = length(data)  
  likelihood = (lambda^n)*exp(-lambda*sum(data))  
  return(likelihood)  
}
```

c.

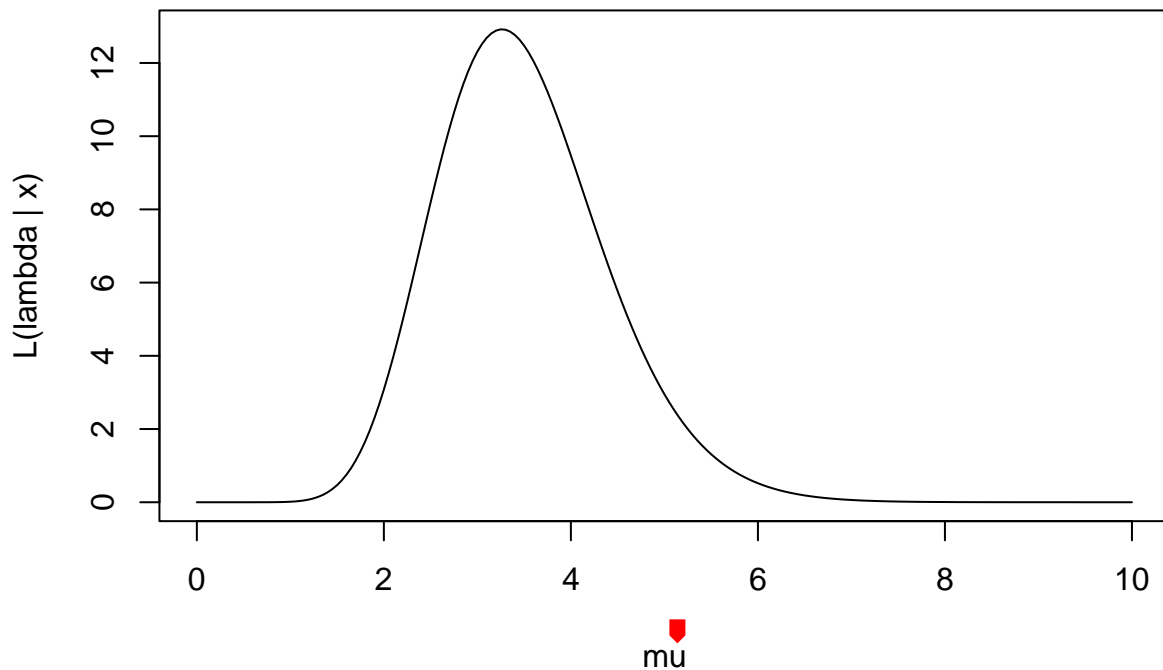
```
mercury = c(0.51,0.02,0.15,0.46,0.11,0.04,0.39,0.52,0.2,0.17,0.01,0.02,0.32,1.37)  
  
xbar = mean(mercury)  
  
print(xbar)
```

```
## [1] 0.3064286
```

d.

```
mus = seq(0, 10, by = 0.05)  
likelihood.mus = c()  
  
for(i in 1:length(mus)) {  
  likelihood.mus[i] = expdist(mus[i], mercury)  
}  
  
plot(mus, likelihood.mus, type = 'l', xlab = "mu", ylab = "L(lambda | x)", main = "Likelihood of Mercury")
```

## Likelihood of Mercury Concentrations



The data looks approximately normal with an ever so slight right skew. The MLE seems to be around 3.5 which is contrary to the statement that the MLE for  $\lambda$  is  $\bar{x}$ . As the next part, this is because  $\lambda = \frac{1}{x}$

e.

```
optimize(expdist, c(0,10), data = mercury, maximum = TRUE)
```

```
## $maximum
## [1] 3.263401
##
## $objective
## [1] 12.92015
```

f.

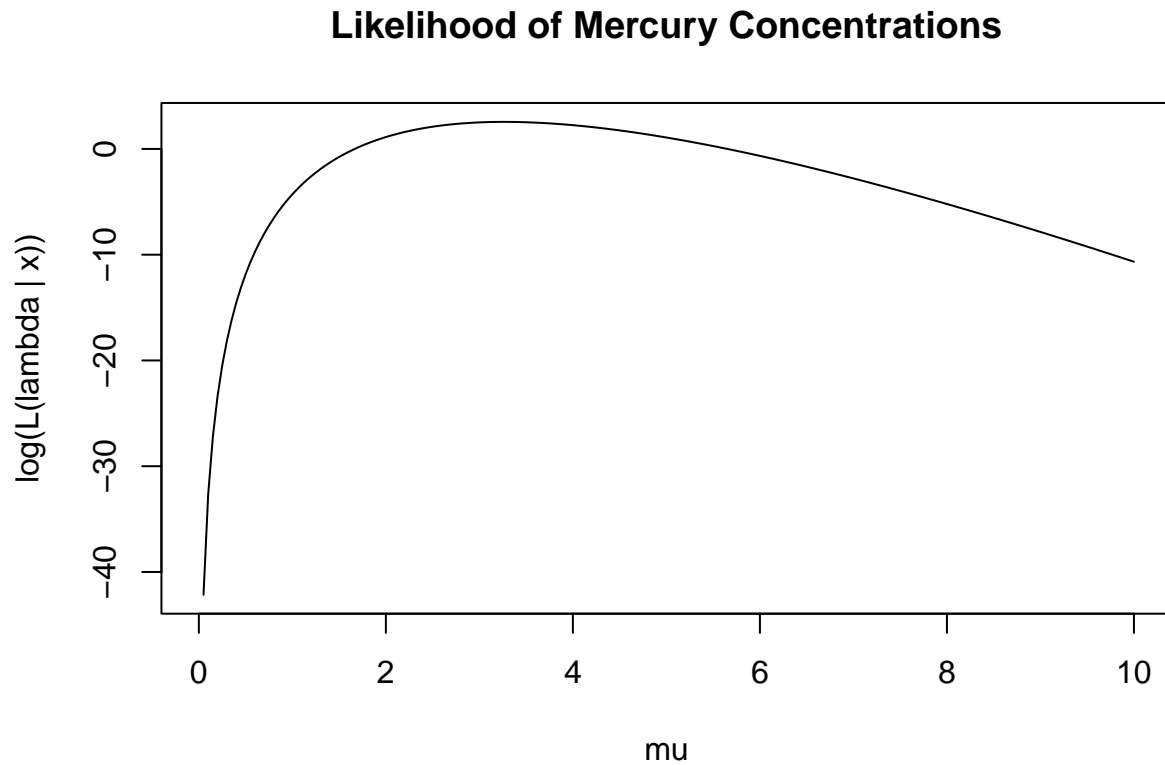
```
expdist.log = function(lambda, data) {
  n = length(data)
  likelihood = log((lambda^n)*exp(-lambda*sum(data)))
  return(likelihood)
}

mus = seq(0, 10, by = 0.05)
likelihood.mus = c()

for(i in 1:length(mus)) {
  likelihood.mus[i] = expdist.log(mus[i], mercury)
```

```
}
```

```
plot(mu, likelihood.mu, type = 'l', xlab = "mu", ylab = "log(L(lambda | x))", main = "Likelihood of M
```



```
optimize(expdist.log, c(0,10), data = mercury, maximum = TRUE)
```

```
## $maximum
## [1] 3.263402
##
## $objective
## [1] 2.558788
```

The location of the maximum is at 3.263402.

### Problem 3:

$$L(p|k_1, \dots, k_n) = p^{\sum_{i=1}^n k_i} (1-p)^{n-\sum_{i=1}^n k_i}$$

```
river = c("Safe", "Safe",
"Safe", "Safe", "Safe", "Safe", "Safe", "Safe", "Flood", "Safe", "Safe", "Safe",
"Flood", "Safe", "Safe", "Flood", "Flood", "Safe", "Safe", "Flood", "Safe", "Safe",
"Safe", "Safe", "Safe", "Safe", "Safe", "Flood", "Safe")
river.binary = c()
for(i in 1:length(river)) {
  if(river[i] == "Safe") {
```

```

    river.binary[i] = 1
  } else {
    river.binary[i] = 0
  }
}

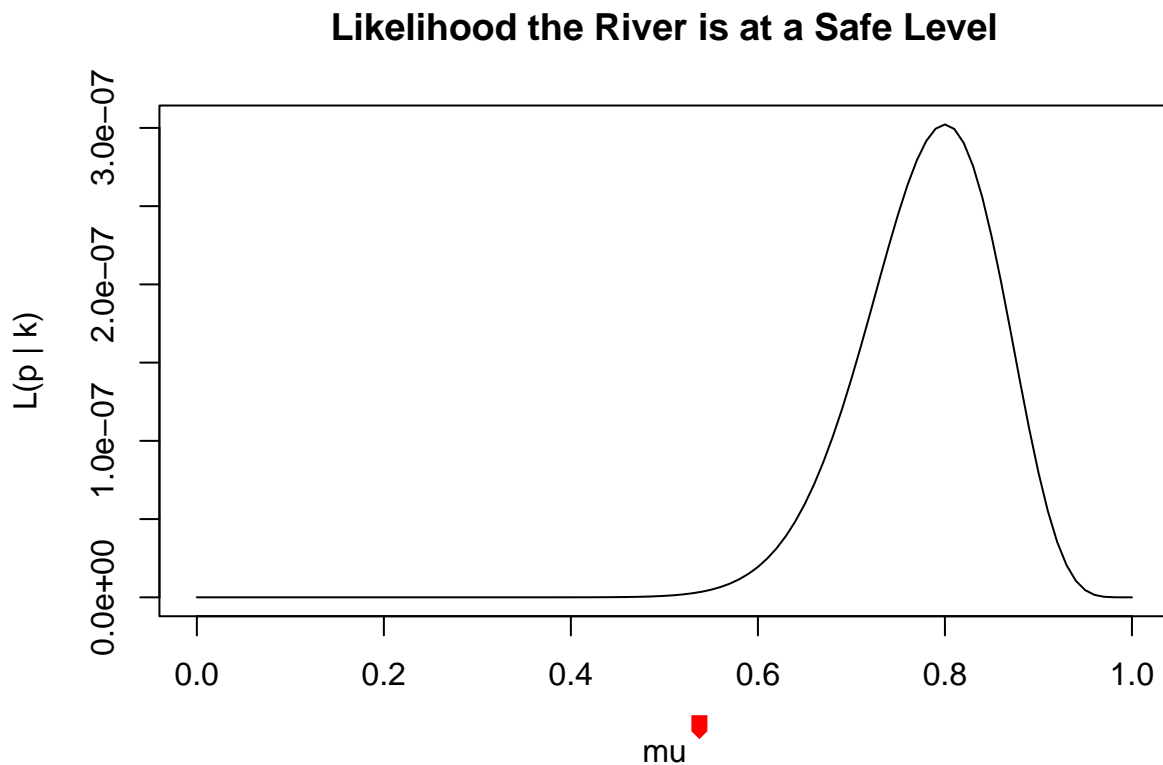
berndist = function(p, data) {
  n = length(data)
  likelihood = (p^(sum(data)))*(1-p)^(n-sum(data))
  return(likelihood)
}

mus = seq(0, 1, by = 0.01)
likelihood.mus = c()

for(i in 1:length(mus)) {
  likelihood.mus[i] = berndist(mus[i], river.binary)
}

plot(mus, likelihood.mus, type = 'l', xlab = "mu", ylab = "L(p | k)", main = "Likelihood the River is a

```



```

optimize(berndist, c(0,1), data = river.binary, maximum = TRUE)

```

```

## $maximum
## [1] 0.7999833

```

```
##
## $objective
## [1] 3.022314e-07

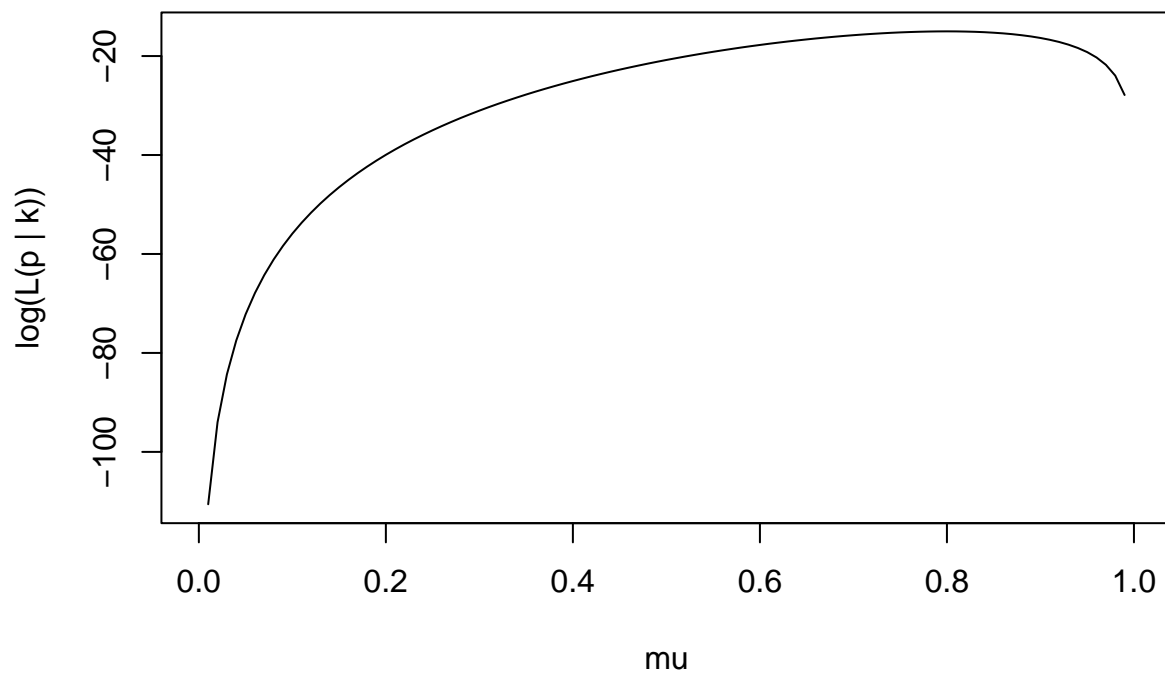
berndist.log = function(p, data) {
  n = length(data)
  likelihood = log((p^(sum(data)))*(1-p)^(n-sum(data)))
  return(likelihood)
}

mus.log = seq(0, 1, by = 0.01)
likelihood.mus.log = c()

for(i in 1:length(mus.log)) {
  likelihood.mus.log[i] = berndist.log(mus.log[i], river.binary)
}

plot(mus.log, likelihood.mus.log, type = 'l', xlab = "mu", ylab = "log(L(p | k))", main = "Likelihood t
```

### Likelihood the River is at a Safe Level



```
optimize(berndist.log, c(0,1), data = river.binary, maximum = TRUE)
```

```
## $maximum
## [1] 0.7999832
##
## $objective
## [1] -15.01207
```



The answer seems reasonable since  $24/30$  of the days the river is at a safe level which is  $4/5$  and the maximum is roughly at  $4/5$ ths. The location of the maximum is  $0.7999832$ .