Week 6

James Restell

Exercise 4.1

(A)

We first create our values and samples based on the Exponential distribution with parameter θ.

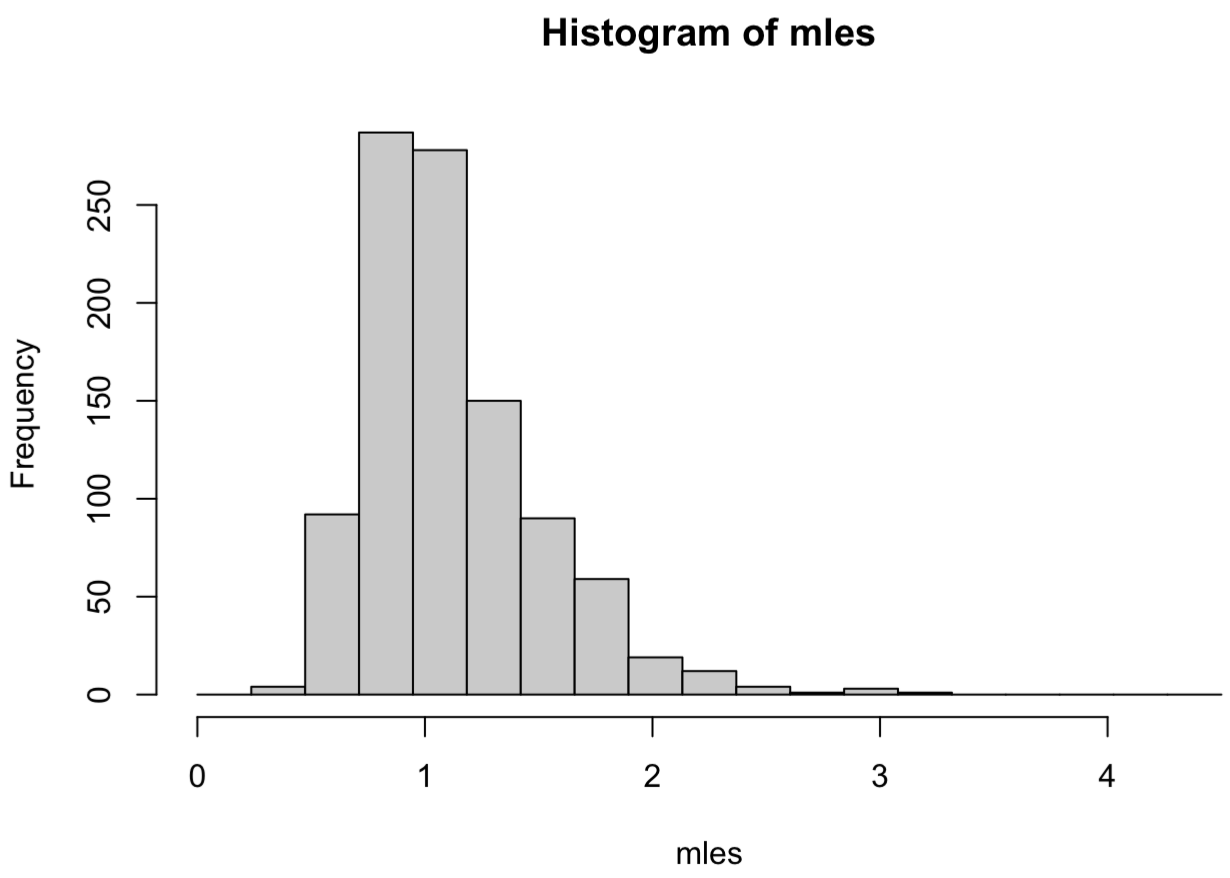
```
xvalues = rexp(10000, rate=1)
xsamples = matrix(xvalues, nrow=1000)
```

Now, treating each row of the matrix as independent samples from the distribution, we can calculate $\hat{\theta} = 1/\bar{x}$ as follows:

```
sums = rowSums(xsamples)
rowAverages = sums/10
mles = 1/rowAverages
```

We can now plot these MLEs.

```
hist(mles, breaks=seq(from=0.0, to=4.5, length.out=20), right=FALSE)
```



(B)

Initialise two lists that will store the medians we calculate in two different ways.

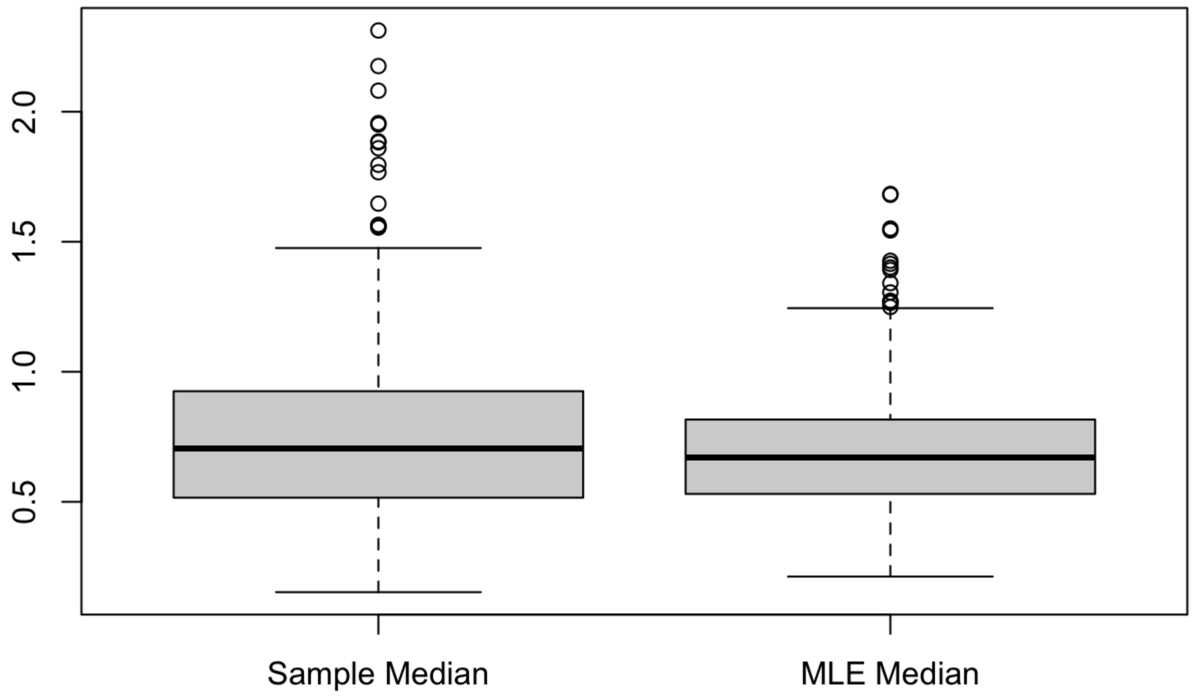
```
sampleMedians = seq(0,0,1000)
MLEmedians = seq(0,0,1000)
```

For each row in the matrix, calculate both the sample median and MLE median and add them to their corresposnding lists.

```
for (row in 1:nrow(xsamples))
{
  MLEmedians[row] = log(2) / mles[row]
  sampleMedians[row] = median(xsamples[row,])
}
```

We can use a boxplot to visualise the difference between the two methods.

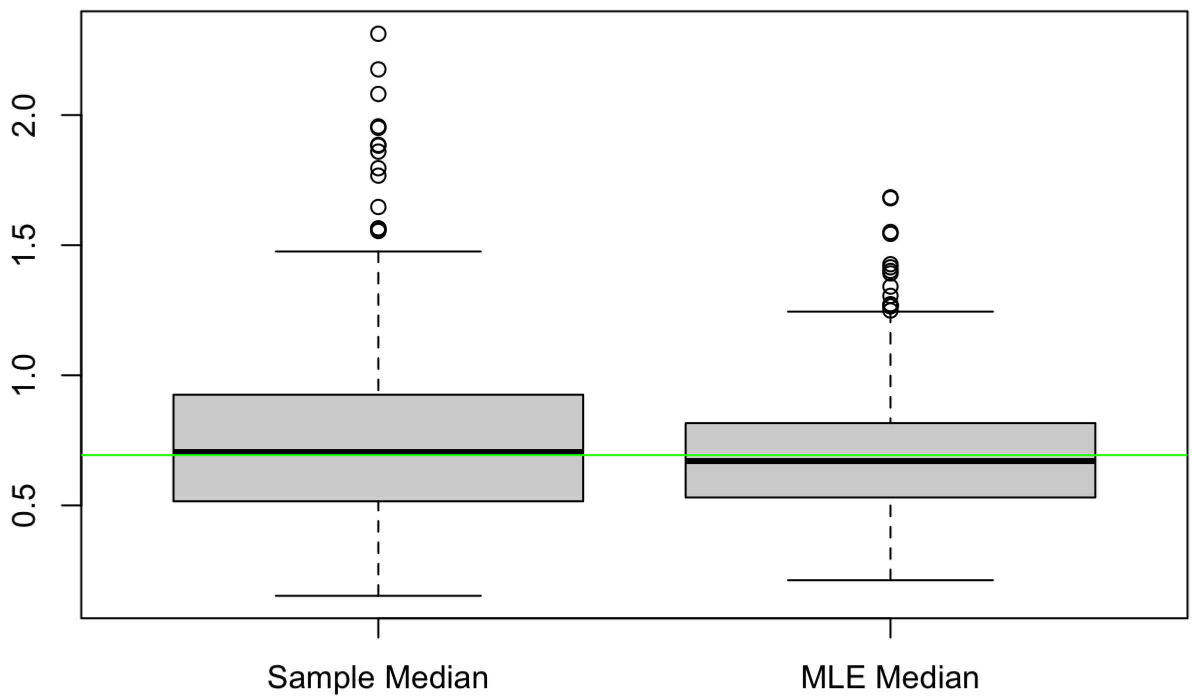
```
boxplot(sampleMedians, MLEmedians,
        names = c("Sample Median", "MLE Median"))
```



(C)

To show the true median on this plot, we can use the following code:

```
boxplot(sampleMedians, MLEmedians,
        names = c("Sample Median", "MLE Median"))
abline(h=log(2), col='green')
```



(D)

For the sample medians we have

```
sampleMedianMean = mean(sampleMedians)
sampleMedianVariance = var(sampleMedians)
sprintf("Mean: %f, Variance: %f", sampleMedianMean, sampleMedianVariance)
```

```
## [1] "Mean: 0.743649, Variance: 0.095725"
```

For the MLE medians we have

```
MLEmedianMean = mean(MLEmedians)
MLEmedianVariance = var(MLEmedians)
sprintf("Mean: %f, Variance: %f", MLEmedianMean, MLEmedianVariance)
```

```
## [1] "Mean: 0.687247, Variance: 0.047502"
```

With these values we can now calculate the Bias and the Mean Squared Error (MSE) of the two estimators.

```
sprintf("For the sample median, Bias: %f and MSE: %f", sampleMedianMean - log(2),
       sampleMedianVariance + (sampleMedianMean - log(2))^2)
```

```
## [1] "For the sample median, Bias: 0.050501 and MSE: 0.098275"
```

```
sprintf("For the MLE median, Bias: %f and MSE: %f", MLEmedianMean - log(2),
       MLEmedianVariance + (MLEmedianMean - log(2))^2)
```

```
## [1] "For the MLE median, Bias: -0.005901 and MSE: 0.047537"
```

So from these values we can concluded that the MLE estimate of the median is likely to be the better estimator for the true median.