Problem2

MDS202203

2022-11-15

Problem 2 : Simulation Study to Understand Sampling Distribution

Part A Suppose $X_1, X_2, \cdots, X_n \stackrel{iid}{\sim} Gamma(\alpha, \sigma)$, with pdf as

$$f(x|\alpha,\sigma) = \frac{1}{\sigma^{\alpha}\Gamma(\alpha)} e^{-x/\sigma} x^{\alpha-1}, \quad 0 < x < \infty,$$

1. Write a `function` in `R` which will compute the MLE of $\theta = \log(\alpha)$ using `optim` function in `R`. You can name it `MyMLE` 2. Choose `n=20`, and `alpha=1.5` and `sigma=2.2` (i) Simulate $\{X_1, X_2, \cdots, X_n\}$ from `rgamma(n=20,shape=1.5,scale=2.2)` (ii) Apply the `MyMLE` to estimate θ and append the value in a vector (iii) Repeat the step (i) and (ii) 1000 times (iv) Draw histogram of the estimated MLEs of θ. (v) Draw a vertical line using `abline` function at the true value of θ. (vi) Use `quantile` function on estimated θ's to find the 2.5 and 97.5-percentile points. 3. Choose `n=40`, and `alpha=1.5` and repeat the (2). 4. Choose `n=100`, and `alpha=1.5` and repeat the (2). 5. Check if the gap between 2.5 and 97.5-percentile points are shrinking as sample size `n` is increasing?

Hint: Perhaps you should think of writing a single `function` where you will provide the values of `n`, `sim_size`, `alpha` and `sigma`; and it will return the desired output.

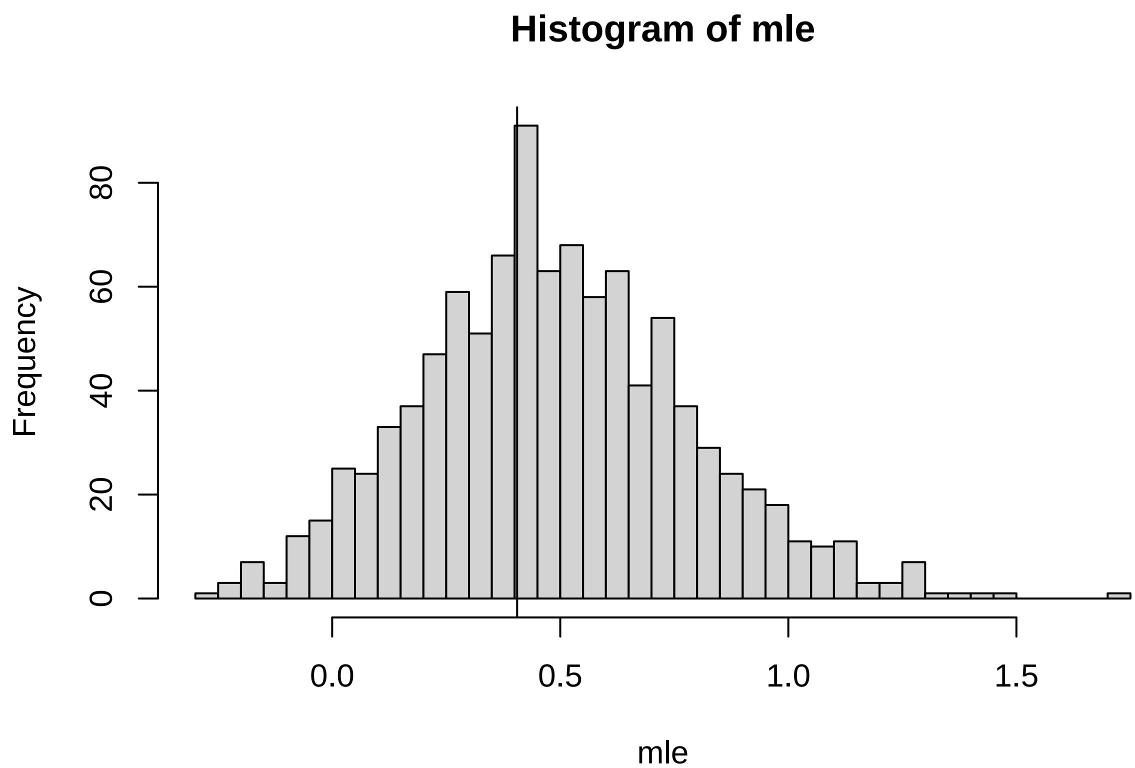
```
our_function= function(n, sim_size,alpha1, sigma1){
  mle=c()
  for(i in 1:sim_size){
    X = rgamma(n,alpha1,scale= sigma1)
    total=sum(log(X))
    sigma2= var(X)/mean(X)
    alpha2=mean(X)^2/var(X)
    n= length(X)
    MyMLE=function(parametes){
    alpha= parametes[1]
    sigma= parametes[2]
    log_likli= -n*alpha*log(sigma)-n*log(gamma(alpha))-sum(X)/sigma +(alpha-1)*total
    return(-log_likli)
    }
    fit= optim(c(alpha2,sigma2), MyMLE)
    mle=append(mle,log(fit$par[1]))
  }
  hist(mle, breaks=50)
  abline(v=log(alpha1))
  ml= data.frame(mle)
  quantile(ml, na.rm= T, probs= c(0.025,0.975))

}
print(our_function(20,1000,1.5,2.2))
```

```
## Warning in log(gamma(alpha)): NaNs produced
```

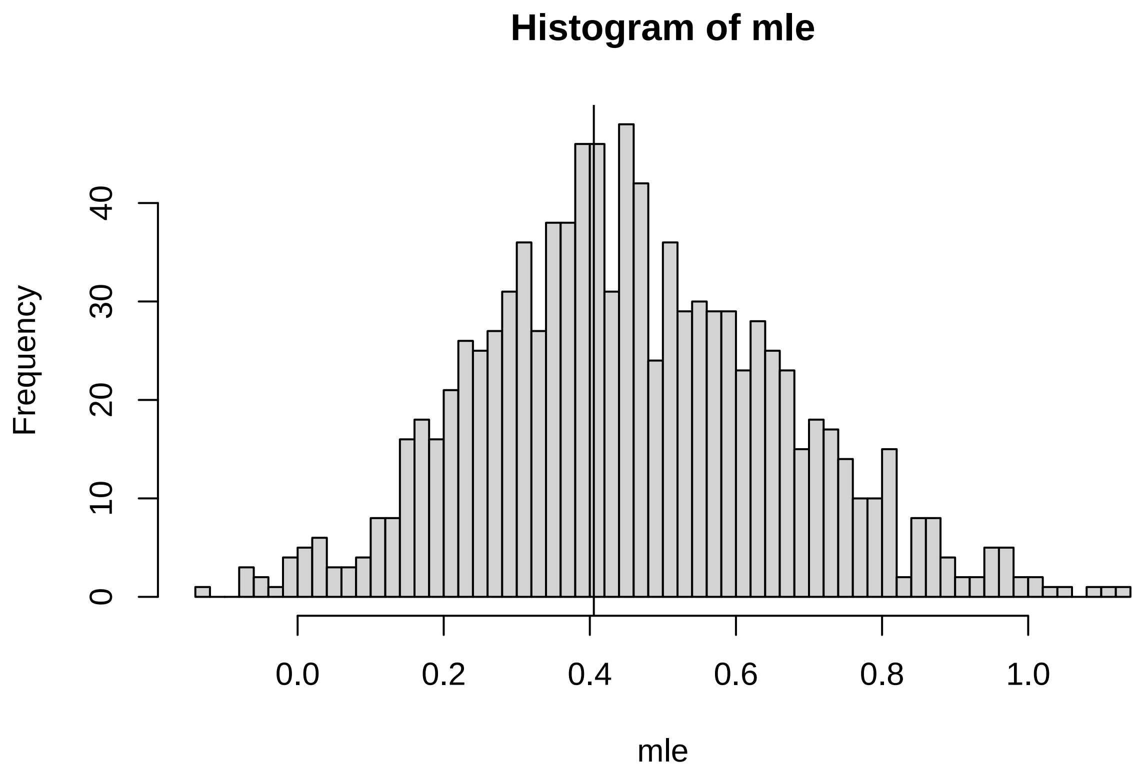
```
## Warning in log(gamma(alpha)): NaNs produced

## Warning in log(sigma): NaNs produced
```



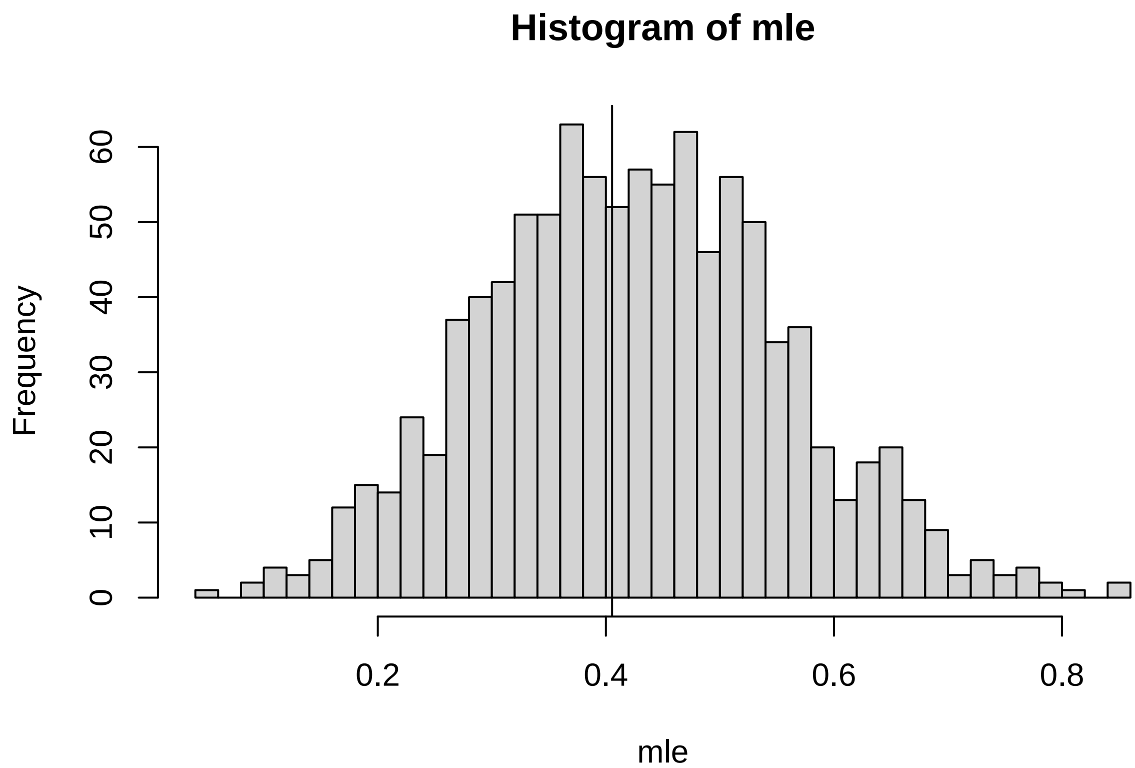

```
##        2.5%       97.5%
## -0.05126846  1.11111377
```

```
print(our_function(40,1000,1.5,2.2))
```




```
##       2.5%      97.5%
## 0.06702688 0.89259169
```

```
print(our_function(100,1000,1.5,2.2))
```




```
##      2.5%     97.5%
## 0.1773821 0.6865549
```

As we increase the sample size we observe that distance between 2.5 percentile and 97.5 percentile decreases.