Math 275 Project

Math 275, St Clair

# Introduction:

The problem that we are exploring in this project is understanding estimation options for a “half normal” distribution. We are using distance data from Bergen lake. The idea is that the further out a mussle is from the transect, the lower its probability of detection. This is what the “half normal” distribution models.

# Methods:

### Derivation Results:

Sampling Distribution of :

MoM Estimator for :

Bias of MLE and MoM estimators of :

$$Bias(\sigma\_{MLE}^2) = 0 \\ Bias(\sigma\_{MoM}^2) = \frac{\pi \sigma^2 - 2 \sigma ^2}{2n}$$

Find the exact 95% confidence interval of :

$$L(x)=\frac{\bar{X}^2}{q\_{0.975}} \\ U(x) = \frac{\bar{X}^2}{q\_{0.025}}$$

q = quantiles from

### Simulation:

> summary(lake$distance)

Min. 1st Qu. Median Mean 3rd Qu. Max.   
 0.0000 0.1800 0.3500 0.3851 0.5400 0.9800

> sd(lake$distance)

[1] 0.2449113

> distance <- lake$distance

> simsize <- 100 # simulations being run  
> n <- 30 #sample size  
> sigma <- 0.5 # value of sigma for ML estimate  
> N <- 1000 #bootstrap  
> mle\_boot\_lower <- mle\_lower <- rep(NA,simsize)  
> mle\_boot\_upper <- mle\_upper <- rep(NA,simsize)  
> mom\_boot\_lower <- mle\_lower <- rep(NA,simsize)  
> mom\_boot\_upper <- mle\_upper <- rep(NA,simsize)  
> x\_boot\_MLE <- rep(NA,N)  
> x\_boot\_MOM <- rep(NA,N)  
>   
> for (i in 1:simsize)  
+ {  
+ x <- abs(rnorm(n,0,sigma)) # generating half normal random sample, repeat this  
+ est\_mle <- sum(x^2)/n # MLE formula  
+ est\_mom <- (pi\*mean(x)^2)/2 # MoM formula  
+   
+ mle\_lower[i] <- (sum(x^2)/n)/qgamma(0.975, n/2, n/2) # based on exact MLE CI (part d)  
+ mle\_upper[i] <- (sum(x^2)/n)/qgamma(0.0275, n/2, n/2)# based on exact MLE CI (part d)  
+   
+ for (j in 1:N)  
+ {  
+ x\_boot <- sample(x, replace=TRUE)  
+ x\_boot\_MLE[j] <- sum(x\_boot^2)/n  
+ x\_boot\_MOM[j] <- (pi\*mean(x\_boot)^2)/2  
+ }  
+   
+ mle\_boot\_lower[i] <- (mean(x)-quantile(x\_boot\_MLE,0.975)\*sqrt(var(x)/n)) # CI based on bootstrapping the MLE  
+ mle\_boot\_upper[i] <- (mean(x)-quantile(x\_boot\_MLE,0.025)\*sqrt(var(x)/n))  
+ mom\_boot\_lower[i] <- (mean(x)-quantile(x\_boot\_MOM,0.975)\*sqrt(var(x)/n))# CI based on bootstrapping the MoM  
+ mom\_boot\_upper[i] <- (mean(x)-quantile(x\_boot\_MOM,0.025)\*sqrt(var(x)/n))  
+ }  
>   
> mean(mle\_boot\_lower)

[1] 0.3774137

> mean(mle\_boot\_upper)

[1] 0.390985

> mean(mom\_boot\_lower)

[1] 0.3758497

> mean(mom\_boot\_upper)

[1] 0.3911691

> mle\_bias <- mean(est\_mle) - sigma^2   
> perc\_bias <- 100\*mle\_bias/sigma^2 # bias as a percentage of sigma^2  
> mse <- mean((est\_mle - sigma^2)^2) # mean of squared deviations (est - truth)^2  
>   
> mom\_bias <- mean(est\_mom) - sigma^2   
> perc\_bias <- 100\*mom\_bias/sigma^2 # bias as a percentage of sigma^2  
> mse <- mean((est\_mom - sigma^2)^2) # mean of squared deviations (est - truth)^2

> mle\_cover <- mean((mle\_boot\_lower <= sigma^2) & (sigma^2 <= mle\_boot\_upper))  
> mle\_mean.length <- mean(mean(mle\_boot\_upper) - mean(mle\_boot\_lower))  
>   
> mom\_cover <- mean((mom\_boot\_lower <= sigma^2) & (sigma^2 <= mom\_boot\_upper))  
> mom\_mean.length <- mean(mean(mom\_boot\_upper) - mean(mom\_boot\_lower))

### Estimation for Bergen

> mle\_berg <- (sum(distance^2))/length(distance)  
> mle\_berg\_lower <- sum(distance^2)/qgamma(0.975, shape = length(distance)/2,rate = 1/2)  
> mle\_berg\_upper <- sum(distance^2)/qgamma(0.025, shape = length(distance)/2,rate = 1/2)  
>   
> #Distance of 0.5 meters  
> #lower  
> exp(-(0.5^2)/(2\*mle\_berg\_lower^2))

[1] 0.00546522

> #upper  
> exp(-(0.5^2)/(2\*mle\_berg\_upper^2))

[1] 0.2320951

> #p value  
> exp(-(0.5^2)/(2\*mle\_berg^2))

[1] 0.05489094

> #Distance of 1m  
> #lower  
> exp(-(1^2)/(2\*mle\_berg\_lower^2))

[1] 8.921349e-10

> #upper  
> exp(-(1^2)/(2\*mle\_berg\_upper^2))

[1] 0.002901778

> #p value  
> exp(-(1^2)/(2\*mle\_berg^2))

[1] 9.078264e-06

# Results: