Homework 8

Zachary Foster

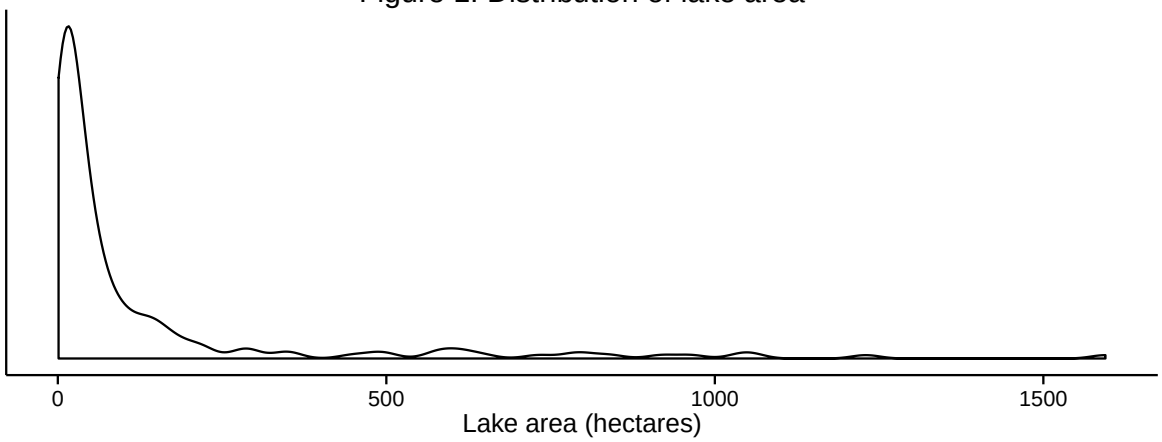
12/02/2014

```
library(boot)
library(ggplot2)
data <- read.table("emap.build08.txt", header = TRUE)
```

The distribution of lake area is highly skewed and irregular (Fig. 1), with many small lakes and a few very large lakes.

```
ggplot(data, aes(x = LK.HA)) +
  geom_density() +
  labs(title = paste0("Figure 1: Distribution of lake area"),
       x = "Lake area (hectares)",
       y = "") +
  theme_classic() +
  theme(axis.text.y = element_blank(),
        axis.ticks.y = element_blank())
```

Figure 1: Distribution of lake area



To test the effect this distribution has on the variance of median estimates, I generated a distribution of 100,000 bootstrapped medians.

```
count <- 1000
get_median <- function(data, index) median(data[index])
boot_info <- boot(data$LK.HA, get_median, count)
boot_info
```

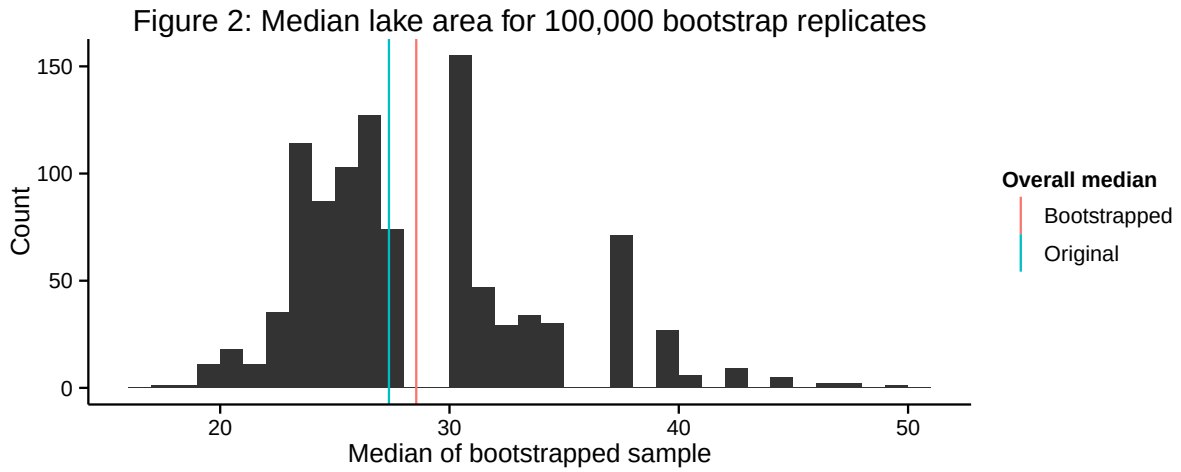
```
##
## ORDINARY NONPARAMETRIC BOOTSTRAP
##
##
## Call:
```

```
## boot(data = data$LK.HA, statistic = get_median, R = count)
##
##
## Bootstrap Statistics :
##     original  bias    std. error
## t1*    27.36   1.191        5.19
```

```
original_median <- median(data$LK.HA)
boot_median <- mean(boot_info$t)
boot_bias <- boot_median - original_median
```

The original median is 27.36 and the mean median of 100,000 bootstrap replicates is 28.551, indicating a bias of 1.191 (fig 2).

```
graph_data <- data.frame(median = boot_info$t)
vline_data <- data.frame(label = c("Original", "Bootstrapped"),
                         value = c(original_median, boot_median))
ggplot(graph_data, aes(x = median)) +
  geom_histogram(binwidth = 1) +
  geom_vline(data = vline_data,
             aes(xintercept = value, color = label),
             show_guide = TRUE) +
  labs(title = "Figure 2: Median lake area for 100,000 bootstrap replicates",
       x = "Median of bootstrapped sample",
       y = "Count",
       color = "Overall median") +
  theme_classic()
```



Next, I calculated a 95% confidence interval using two techniques: the percentile method and the bias-corrected and accelerated (BCa) bootstrap.

```
boot.ci(boot_info, type = c("perc", "bca"))
```

```
## BOOTSTRAP CONFIDENCE INTERVAL CALCULATIONS
## Based on 1000 bootstrap replicates
##
```

```
## CALL :
## boot.ci(boot.out = boot_info, type = c("perc", "bca"))
##
## Intervals :
## Level     Percentile            BCa
## 95%   (20.64, 40.00 )   (20.64, 40.01 )
## Calculations and Intervals on Original Scale
```

The BCa method produced a slightly smaller interval. This is probably because BCa corrects for bias and skew, making for a more conservative confidence interval. Another method of calculating a confidence interval of the median is to calculate the confidence interval for the mean of logged values and back-transform the resulting limits.

```
log_lake_area <- log(data$LK.HA + 1)
error <- qnorm(0.975)*sd(log_lake_area)/sqrt(length(log_lake_area))
ci <-  c(mean(log_lake_area) - error, mean(log_lake_area) + error)
exp(1)^ci - 1
```

```
## [1] 27.73 44.08
```

This results in higher limits. One of the assumptions of this method is that the logged distribution is symmetrical. Although the logged distribution is much more symmetrical than the original one (Fig. 2), it is still significantly skewed (Fig. 3). This could explain the disagreement between the results of this and the bootstrap methods.

```
data$LK.HA <- log(data$LK.HA + 1)
ggplot(data, aes(x = LK.HA)) +
  geom_density() +
  labs(title = paste0("Figure 3: Distribution of ln(lake area + 1)"),
       x = "ln(Lake area + 1)",
       y = "") +
  theme_classic() +
  theme(axis.text.y = element_blank(),
        axis.ticks.y = element_blank())
```