Worksheet6

STAT414

2024-10-19

```
# #This worksheet, is based on the Module06 material, but also draws on your knowledge from
# previous modules, especially Module05. Main event here is the introduction of a super useful and
# widely used technique called "Bootstrapping" to estimate standard errors (and in general, sampling
# distributions) of estimators, without making any distributional assumptions on the population.
# After the simulation technique you learned in Module03, bootstrapping is perhaps the most important
# skill you are picking up in this course. Happy bootstrapping!

library(EnvStats)
```

```
##
## Attaching package: 'EnvStats'

## The following objects are masked from 'package:stats':
##
##     predict, predict.lm
```

```
TcCB <- EPA.94b.tccb.df
head(TcCB)
```

```
##   TcCB.orig TcCB Censored      Area
## 1      0.22 0.22    FALSE Reference
## 2      0.23 0.23    FALSE Reference
## 3      0.26 0.26    FALSE Reference
## 4      0.27 0.27    FALSE Reference
## 5      0.28 0.28    FALSE Reference
## 6      0.28 0.28    FALSE Reference
```

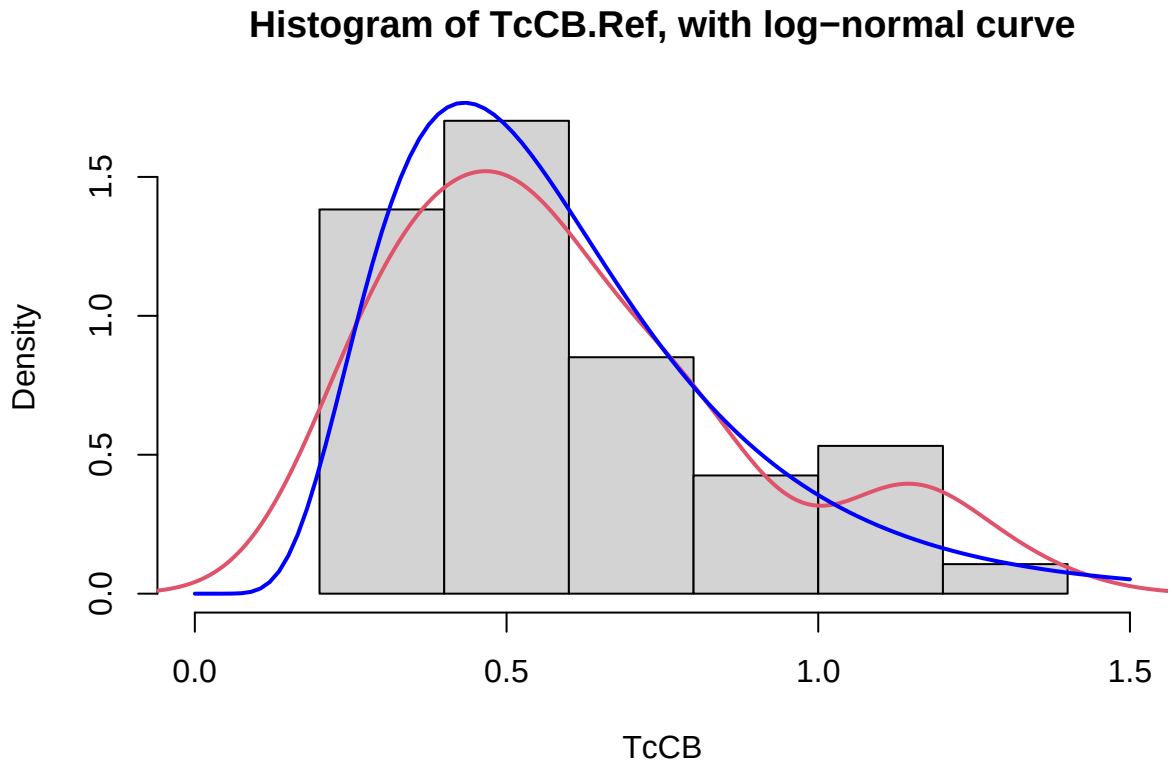
```
TcCB.Ref <- TcCB[TcCB$Area == "Reference",]
TcCB.Cleanup <- TcCB[TcCB$Area == "Cleanup",]

head(TcCB.Ref)
```

```
##   TcCB.orig TcCB Censored      Area
## 1      0.22 0.22    FALSE Reference
## 2      0.23 0.23    FALSE Reference
## 3      0.26 0.26    FALSE Reference
## 4      0.27 0.27    FALSE Reference
## 5      0.28 0.28    FALSE Reference
## 6      0.28 0.28    FALSE Reference
```

```
head(TcCB.Cleanup)
```

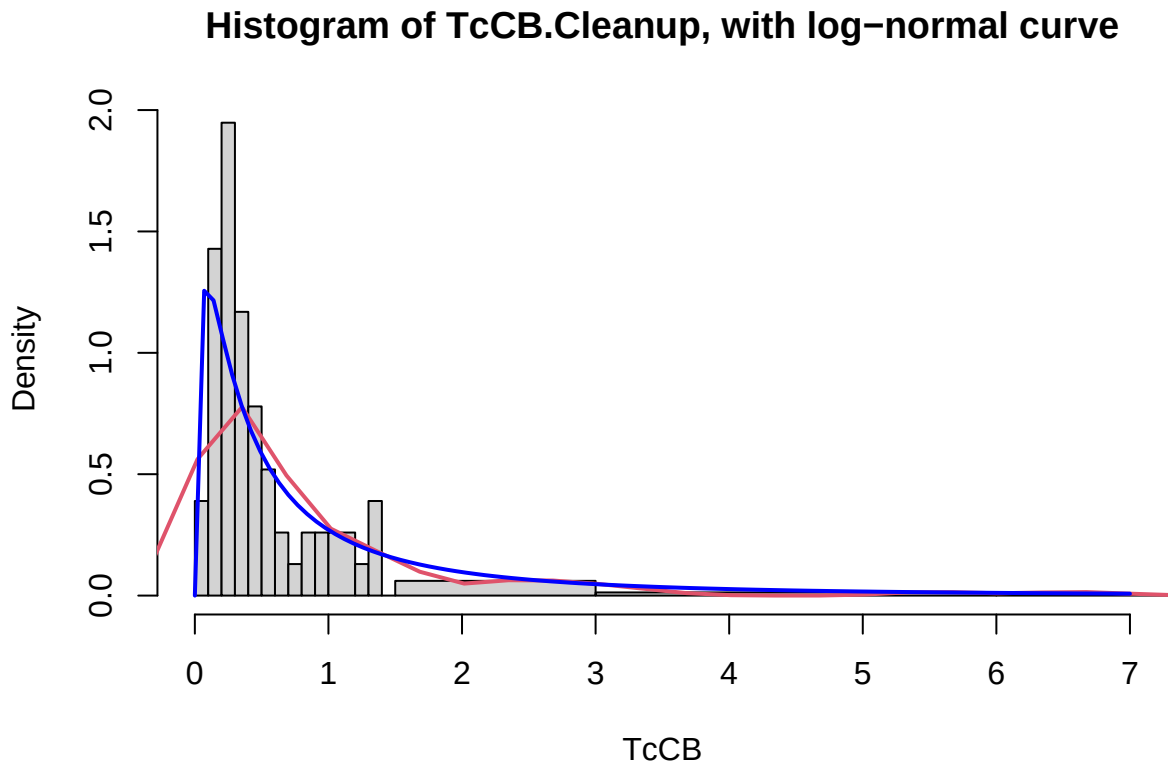
```
##    TcCB.orig TcCB Censored    Area
## 48     <0.09 0.09     TRUE Cleanup
## 49      0.09 0.09    FALSE Cleanup
## 50      0.09 0.09    FALSE Cleanup
## 51      0.12 0.12    FALSE Cleanup
## 52      0.12 0.12    FALSE Cleanup
## 53      0.14 0.14    FALSE Cleanup
```

```
# 1. Determine if we can comfortably assume a theoretical model for each of the above variables.

# (a) Plot the histogram of the data, and overlay a log-normal curve.

hist(TcCB.Ref$TcCB, probability=TRUE, main = "Histogram of TcCB.Ref, with log-normal curve",
     xlab="TcCB", xlim=c(0,1.5))
lines(density(TcCB.Ref$TcCB), col = 2, lwd = 2)
#log-normal = lnorm
curve(dlnorm(x, mean=mean(log(TcCB.Ref$TcCB)), sd=sd(log(TcCB.Ref$TcCB))),
      lwd=2, col="blue", add=TRUE)
```




```
#Ignore the extreme outliers in the TcCB.Cleanup data by setting xlim=c(0,7), which
#would otherwise greatly imbalance the histogram to be unreadable
hist(TcCB.Cleanup$TcCB, probability=TRUE, main = "Histogram of TcCB.Cleanup, with log-normal curve",
```

```
     xlab="TcCB", xlim=c(0,7), breaks=c(0,0.1,0.2,0.3,0.4,0.5,0.6,0.7,0.8,0.9,
                                        1,1.2,1.3,1.4,1.5,3,6,9,12,15,200))
lines(density(TcCB.Cleanup$TcCB), col = 2, lwd = 2)
#log-normal = lnorm
curve(dlnorm(x, mean=mean(log(TcCB.Cleanup$TcCB)), sd=sd(log(TcCB.Cleanup$TcCB))),
      lwd=2, col="blue", add=TRUE)
```




```
# (b) Use the function gofTest() of EnvStats to test the log-normality assumption of the
# data for each variable.
gofTest(TcCB.Ref$TcCB, dist = "lnorm")
```

```
##
## Results of Goodness-of-Fit Test
## -------------------------------
##
## Test Method:                     Shapiro-Wilk GOF
##
## Hypothesized Distribution:       Lognormal
##
## Estimated Parameter(s):          meanlog = -0.6195712
##                                  sdlog   =  0.4679530
##
## Estimation Method:               mvue
##
## Data:                            TcCB.Ref$TcCB
```

```
##
## Sample Size:                     47
##
## Test Statistic:                  W = 0.9786379
##
## Test Statistic Parameter:        n = 47
##
## P-value:                         0.5371935
##
## Alternative Hypothesis:          True cdf does not equal the
##                                  Lognormal Distribution.
```

```
gofTest(TcCB.Cleanup$TcCB, dist = "lnorm")
```

```
##
## Results of Goodness-of-Fit Test
## -------------------------------
##
## Test Method:                     Shapiro-Wilk GOF
##
## Hypothesized Distribution:       Lognormal
##
## Estimated Parameter(s):          meanlog = -0.5474262
##                                  sdlog   =  1.3604488
##
## Estimation Method:               mvue
##
## Data:                            TcCB.Cleanup$TcCB
##
## Sample Size:                     77
##
## Test Statistic:                  W = 0.8708372
##
## Test Statistic Parameter:        n = 77
##
## P-value:                         1.284008e-06
##
## Alternative Hypothesis:          True cdf does not equal the
##                                  Lognormal Distribution.
```

```
# (c) State your conclusions.
cat("The histograms are mostly inconclusive when comparing the density curves to
    the superimposed log-normal curves. They are both similar in pattern and shape
    to the log-normal curves, but the margins are not correct and there are many
    noticeable devations. However, from the gofTest for both datasets for the log-normal
    distribution, choose a significance level of 0.05, and the calculated p-values
    for Ref and Cleanup are: 0.5371935 and ~0.0000013, repspectively.
    0.5371935 is mucher greater than .05, and 0.0000013 is much smaller than .05,
    therefore we can conclude that there is strong evidence to suggest that the
    TcCB data for entries that have Area==Reference do not follow the log-normal
    distribution, and that there is strong support that the TcCB data for entries that
    have Area==Cleanup do follow log-normal distribution.")
```

```
## The histograms are mostly inconclusive when comparing the density curves to
##     the superimposed log-normal curves. They are both similar in pattern and shape
##     to the log-normal curves, but the margins are not correct and there are many
##     noticeable devations. However, from the gofTest for both datasets for the log-normal
##     distribution, choose a significance level of 0.05, and the calculated p-values
##     for Ref and Cleanup are: 0.5371935 and ~0.0000013, repspectively.
##     0.5371935 is mucher greater than .05, and 0.0000013 is much smaller than .05,
##     therefore we can conclude that there is strong evidence to suggest that the
##     TcCB data for entries that have Area==Reference do not follow the log-normal
##     distribution, and that there is strong support that the TcCB data for entries that
##     have Area==Cleanup do follow log-normal distribution.
```

```
# 2. Apply the log transformation to each of the above variable and create
ln.TcCB.Ref <- log(TcCB.Ref$TcCB)
ln.TcCB.Cleanup <- log(TcCB.Ref$TcCB)

# (a) Use the appropriately modified version of the Bootstrapping code provided to obtain
# estimates of the mean [= mu = E(log(X))] for each variable and corresponding bootstrap
# standard errors.

cat("The basic steps of the bootstrap are:
      1. Estimate the parameter based on the data.
      2. Sample the data with replacement B times, and each
          time estimate the parameter based on this bootstrap
          sample.
      3. Use the estimated parameter created in Step 1 and the
          bootstrap estimate of the sampling distribution of the
          estimator created in Step 2 to obtain the standard
          error for the parameter.
")
```

```
## The basic steps of the bootstrap are:
##       1. Estimate the parameter based on the data.
##       2. Sample the data with replacement B times, and each
##           time estimate the parameter based on this bootstrap
##           sample.
##       3. Use the estimated parameter created in Step 1 and the
##           bootstrap estimate of the sampling distribution of the
##           estimator created in Step 2 to obtain the standard
##           error for the parameter.
```

```
#Step 1 - Estimate population - Already done above
#Step 2 - Perform bootstrapping, draw samples from data using original sampling method

set.seed(1)
strap_size <- 10000

results_ref_means <- c()
results_cleanup_means <- c()

#Replace sample to generate bootstraps
for(i in 1:strap_size){
  refsample <- sample(ln.TcCB.Ref, size=length(ln.TcCB.Ref), replace=TRUE)
```

```
  cleanupsample <- sample(ln.TcCB.Cleanup, size=length(ln.TcCB.Cleanup), replace=TRUE)

  results_ref_means[i] <- mean(refsample)
  results_cleanup_means[i] <- mean(cleanupsample)
}

#Calculate mean of each and the bootstrap standard errors (SD)
refmean <- mean(results_ref_means)
refse <- sd(results_ref_means)
cat("Bootstrap mean of ln.TcCB.ref:", refmean)
```

```
## Bootstrap mean of ln.TcCB.ref: -0.6215979
```

```
cat("Bootstrap standard error of ln.TcCB.ref:", refse)
```

```
## Bootstrap standard error of ln.TcCB.ref: 0.06687884
```

```
cleanupmean <- mean(results_cleanup_means)
cleanupse <- sd(results_cleanup_means)
cat("Bootstrap mean of ln.TcCB.cleanup:", cleanupmean)
```

```
## Bootstrap mean of ln.TcCB.cleanup: -0.6195984
```

```
cat("Bootstrap standard error of ln.TcCB.cleanup:", cleanupse)
```

```
## Bootstrap standard error of ln.TcCB.cleanup: 0.06712359
```

```
# (b) Compare the above SE's to the corresponding s/sqrt(n), and comment on how close they are.
ref_normal_se <- sd(ln.TcCB.Ref) / sqrt(length(ln.TcCB.Ref))
cleanup_normal_se <- sd(ln.TcCB.Cleanup) / sqrt(length(ln.TcCB.Cleanup))

cat("Normal ln.TcCB.Reference SE:", ref_normal_se)
```

```
## Normal ln.TcCB.Reference SE: 0.06825795
```

```
cat("Normal ln.TcCB.Cleanup SE:", cleanup_normal_se)
```

```
## Normal ln.TcCB.Cleanup SE: 0.06825795
```

```
cat("The normal standard errors of both datasets are very close to their bootstrap
    standard errors, within a ~0.03 margin. This suggests that the log-transformed
    TcCB datasets are approximately following a normal distribution, which we
    were able to calculate without any knowledge of underlying distributions, using
    bootstrapping.")
```

```
## The normal standard errors of both datasets are very close to their bootstrap
##     standard errors, within a ~0.03 margin. This suggests that the log-transformed
##     TcCB datasets are approximately following a normal distribution, which we
##     were able to calculate without any knowledge of underlying distributions, using
##     bootstrapping.
```