

Lesson 08: Statistical Inference, Estimation for Single Populations

References

- Black, Chapter 8 Statistical Inference: Estimation for Single Populations (pp. 231-256)
- Davies, Chapter 17 Sampling Distribution and Confidence (pp. 378-384)
- Stowell, Chapter 5 Summary Statistics for Continuous Variables (pp. 70-71), Chapter 6 Tabular Data (pp.84-86)

Exercises:

- 1) Assume a random sample of size 100 is drawn from a normal distribution with variance 1. The average value of the sample is 50. Find a 95% confidence interval for the mean.

```
n <- 100      # sample size
mean <- 50    # mean of sample
sd <- sqrt(1) # standard deviation of population

margin.of.error <- qnorm(1-(0.05/2)) * sd/sqrt(n)

conf.int <- c(mean - margin.of.error, mean + margin.of.error)
conf.int
```

```
## [1] 49.804 50.196
```

- 2) Assume the standard deviation for a normal distribution is equal to 100 units. Also assume we want to estimate the unknown mean with a 95% confidence interval of total width 8 units. Calculate the sample size required.

```
z_score <- qnorm(0.025, mean = 0, sd = 1, lower.tail = FALSE)
sample_size <- (z_score*100.0/4.0)**2
round(sample_size)
```

```
## [1] 2401
```

- 3) A random sample of 1600 registered voters are contacted and asked a variety of questions. For one question, 60% of the voters expressed approval and 40% disapproval. Calculate a 95% confidence interval for the proportion expressing approval.

```
# Here we will use a built-in R function for testing a proportion.
# prop.test(x, n, p = NULL,
#   alternative = c("two.sided", "less", "greater"),
#   conf.level = 0.95, correct = TRUE
# x a vector of counts of successes, a one-dimensional table with two entries,
# or a two-dimensional table (or matrix) with 2 columns, giving the counts of
# successes and failures, respectively.
# n a vector of counts of trials; ignored if x is a matrix or a table.
# p a vector of probabilities of success. The length of p must be the same
# as the number of groups specified by x, and its elements must be greater
# than 0 and less than 1.
# alternative: a character string specifying the alternative hypothesis,
# must be one of "two.sided" (default), "greater" or "less". You can specify
# just the initial letter. Only used for testing the null that a single proportion
```

```

# equals a given value, or that two proportions are equal; ignored otherwise.
# conf.level: confidence level of the returned confidence interval.
# Must be a single number between 0 and 1. Only used when testing the
# null that a single proportion equals a given value, or that two proportions
# are equal; ignored otherwise.
# correct: a logical indicating whether Yates' continuity correction should be
# applied where possible.
prop.test(x = 1600 * 0.6, n = 1600,
          alternative = "two.sided", conf.level = 0.95)

```

```

##
## 1-sample proportions test with continuity correction
##
## data: 1600 * 0.6 out of 1600, null probability 0.5
## X-squared = 63.601, df = 1, p-value = 1.524e-15
## alternative hypothesis: true p is not equal to 0.5
## 95 percent confidence interval:
## 0.5754686 0.6240461
## sample estimates:
## p
## 0.6

```

```

# If we were to store prop.test in an R object, we could examine its structure.
prop_test_object <- prop.test(x = 1600 * 0.6, n = 1600,
                             alternative = "two.sided", conf.level = 0.95)
print(str(prop_test_object))

```

```

## List of 9
## $ statistic : Named num 63.6
## .. attr(*, "names")= chr "X-squared"
## $ parameter : Named int 1
## .. attr(*, "names")= chr "df"
## $ p.value : num 1.52e-15
## $ estimate : Named num 0.6
## .. attr(*, "names")= chr "p"
## $ null.value : Named num 0.5
## .. attr(*, "names")= chr "p"
## $ conf.int : atomic [1:2] 0.575 0.624
## .. attr(*, "conf.level")= num 0.95
## $ alternative: chr "two.sided"
## $ method : chr "1-sample proportions test with continuity correction"
## $ data.name : chr "1600 * 0.6 out of 1600, null probability 0.5"
## - attr(*, "class")= chr "htest"
## NULL

```

```

# Notice that this object is a list and the confidence interval lower
# and upper limits themselves can be extracted from the object as
as.numeric(prop_test_object$conf.int)

```

```
## [1] 0.5754686 0.6240461
```

- 5) A random sample of consumers are presented with two beverages in random order and asked which they prefer most. All the consumers expressed a preference. One beverage was preferred 85% of the time. Use this number to determine how large a sample of consumers would be needed to generate a 95% confidence interval with an overall width just less than 2% (i.e. from 84% to 86%)?

```
p <- 0.85
z_score <- qnorm(0.025, mean = 0, sd = 1, lower.tail = FALSE)
sample_size <- (z_score**2)*p*(1-p)/(0.01)**2
round(sample_size)
```

```
## [1] 4898
```

```
sample_size <- (z_score**2)*(0.25)/(0.01)**2
round(sample_size)
```

```
## [1] 9604
```

Data Set: hot_dogs.csv (Original source: Consumer Reports, June 1986, pp. 366-367)

Description: Results of a laboratory analysis of calories and sodium content of major hot dog brands. Researchers for Consumer Reports analyzed three types of hot dog: beef, poultry, and meat (mostly pork and beef, but up to 15% poultry meat). Fifty four observations are reported.

Variable Names:

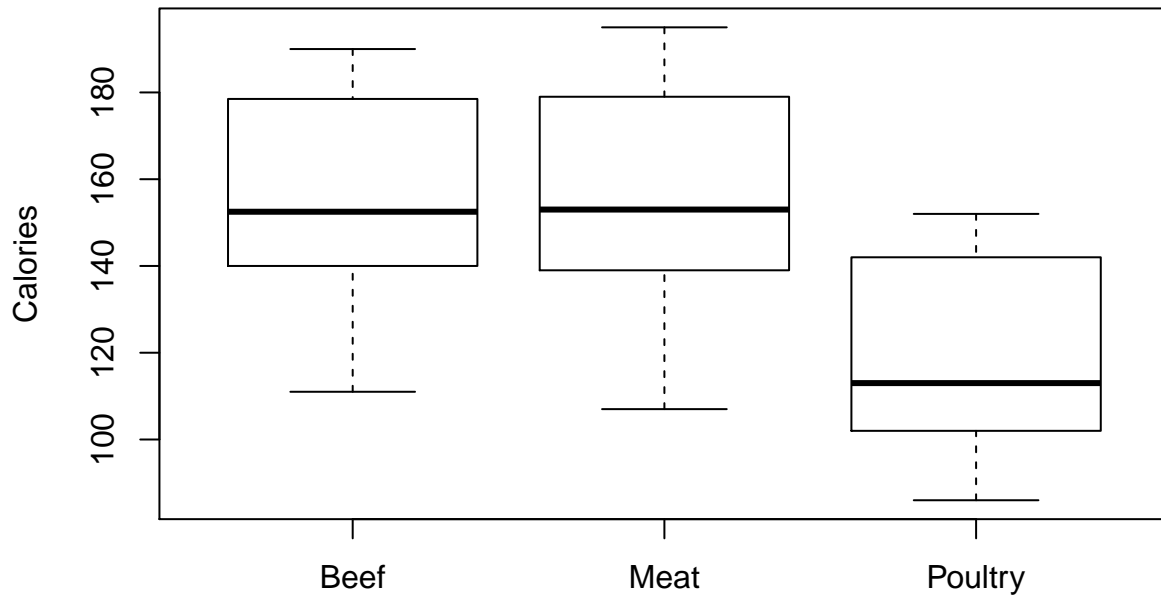
1. Type = Type of hotdog (beef, meat, or poultry)
 2. Calories = Calories per hot dog
 3. Sodium = Milligrams of sodium per hot dog
- 1) Create boxplots and find 95% confidence intervals for the mean amount of calories in each Type of hot dog: beef, meat and poultry. Construct 99% one-sided lower confidence intervals for the mean amount of calories in each Type of hot dog: beef, meat and poultry.

```
# Read the comma-delimited text file creating a data frame object in R,
# then examine its structure:
hotdogs <- read.csv("hot_dogs.csv")
str(hotdogs)
```

```
## 'data.frame': 54 obs. of 3 variables:
## $ Type : Factor w/ 3 levels "Beef","Meat",...: 1 1 1 1 1 1 1 1 1 1 ...
## $ Calories: int 186 181 176 149 184 190 158 139 175 148 ...
## $ Sodium : int 495 477 425 322 482 587 370 322 479 375 ...

with(hotdogs, boxplot(Calories ~ Type, main = "Calories, by hotdog type",
ylab = "Calories"))
```

Calories, by hotdog type



```
# Reading ahead in this lesson, we see that most of the questions  
# concern subsets of the data by hotdog type. To respond to the  
# questions, we will create three subset data frames.
```

```
beef <- subset(hotdogs, subset = (Type == "Beef"))  
meat <- subset(hotdogs, subset = (Type == "Meat"))  
poultry <- subset(hotdogs, subset = (Type == "Poultry"))
```

```
# Here we know of an R function to compute the confidence interval.  
with(beef, t.test(Calories)$conf.int)
```

```
## [1] 146.2532 167.4468  
## attr("conf.level")  
## [1] 0.95
```

```
with(meat, t.test(Calories)$conf.int)
```

```
## [1] 145.7308 171.6809  
## attr("conf.level")  
## [1] 0.95
```

```
with(poultry, t.test(Calories)$conf.int)
```

```
## [1] 107.1698 130.3596  
## attr("conf.level")  
## [1] 0.95
```

```
# Construct 99% one-sided lower confidence intervals for the mean amount of  
# calories in each Type of hot dog: beef, meat and poultry.
```

```
# One-sided confidence intervals can also be created.
t.test(beef$Calories, alternative = "less", conf.level = 0.99)
```

```
##
## One Sample t-test
##
## data: beef$Calories
## t = 30.98, df = 19, p-value = 1
## alternative hypothesis: true mean is less than 0
## 99 percent confidence interval:
##      -Inf 169.7072
## sample estimates:
## mean of x
##      156.85
```

```
t.test(meat$Calories, alternative = "less", conf.level = 0.99)
```

```
##
## One Sample t-test
##
## data: meat$Calories
## t = 25.93, df = 16, p-value = 1
## alternative hypothesis: true mean is less than 0
## 99 percent confidence interval:
##      -Inf 174.5183
## sample estimates:
## mean of x
##      158.7059
```

```
t.test(poultry$Calories, alternative = "less", conf.level = 0.99)
```

```
##
## One Sample t-test
##
## data: poultry$Calories
## t = 21.714, df = 16, p-value = 1
## alternative hypothesis: true mean is less than 0
## 99 percent confidence interval:
##      -Inf 132.8951
## sample estimates:
## mean of x
##      118.7647
```

- 2) Find a 95% confidence interval for the variance in the amount of calories found for each type of hotdog: beef, meat and poultry.

```
# Here, we set up a user-defined function.
# Note that this is a chi-square test... so we use qchisq() function
# qchisq(p, df, ncp = 0, lower.tail = TRUE, log.p = FALSE) # quantiles
var.conf.int = function(x, conf.level = 0.95) {
  df <- length(x) - 1
  chilower <- qchisq((1 - conf.level)/2, df, lower.tail = TRUE)
  chiupper <- qchisq((1 - conf.level)/2, df, lower.tail = FALSE)
  v <- var(x)
  c(df * v/chiupper, df * v/chilower)
}
```

```
with(beef, var.conf.int(Calories))
```

```
## [1] 296.495 1093.643
```

```
with(meat, var.conf.int(Calories))
```

```
## [1] 353.2469 1475.1049
```

```
with(poultry, var.conf.int(Calories))
```

```
## [1] 282.0926 1177.9754
```