Lab 7 Rhinestones

By Musa Rasheed

Excercises

The weight of rhinestones used at a jewelry store is uniformly distributed between 1 and 5 grams. You want to estimate the true mean using an interval estimator (point estimator with confidence intervals).

1) Simulate a sample of size 20 from the above distribution

1. Compute your 95% confidence interval for the mean, assuming variance is known.
2. Compute your 95% confidence interval for the mean, assuming variance is unknown.
3. For each of the two previous problems, repeat the process 10,000 times, and count how many times your confidence interval contains the true mean. Does it match up with what you expect?

2) Simulate a sample of size 100 from the above distribution.

1. Compute your 95% confidence interval for the mean, assuming variance is known.
2. Compute your 95% confidence interval for the mean, assuming variance is unknown.
3. For each of the two previous problems, repeat the process 10,000 times, and count how many times your confidence interval contains the true mean. Does it match up with what you expect?

Answers to 1 (sample size 20)

1.1 Known Variance

Here, we assume the variance is known. Therefore, we use the Z-score for the test statistic. Let's start by generating the data and defining paramters.

In [1]:
```
### Generating the sample of size 20 ###
samp = runif(20, 1, 5)

mu = (1+5)/2
n = 20
sigma = ((5-1)^2)/12
alpha = 0.05
```

For this question, I'll assume that I don't know the true mean (which should be 3) but I DO know the true variance (which is $\frac{4}{3}$)

In [2]:
```
avg = mean(samp)
```

To build the confidence interval, I first need the Z-score for a 95% confidence interval:

$$Z_{\frac{0.05}{2}} = 1.96$$

Afterwards, I simply plug it into the following formula:

$$Z_{\frac{0.05}{2}} * \frac{\sigma}{\sqrt{n}}$$

Then we will have the following interval estimator:

$$\bar{X} \pm Z_{\frac{0.05}{2}} \frac{\sigma}{\sqrt{n}}$$

In [3]:
```
Z = qnorm(1-(alpha/2))
CIK = Z * (sigma/sqrt(n)) #CIK stands for Confidence Interval Known

maxk = avg+CIK #Known variance maximum
mink = avg-CIK #Known variance minimum

print(c(maxk, mink))

if(mu < maxk && mu > mink){
    print("True mean is within the maximum and minimum")
} else if (mu > maxk || mu < mink){
    print("True mean is NOT within the interval")
}
```

```
[1] 3.565134 2.396438
[1] "True mean is within the maximum and minimum"
```

1.2 Variance Unknown

If the variance is unknown, we have to resort to the t-distribution, and rely on the sample variance as well as a t-table (or R in this case).

First, get the value from R depending on the confidence level and the degrees of freedom:

$$t_{\frac{\alpha}{2},n-1}$$

Then get the sample variance and take the product:

$$t_{\frac{\alpha}{2},n-1} * \frac{s}{\sqrt{n}}$$

We get the sample variance using the following formula

$$s = \sqrt{\frac{1}{n-1} * \sum_{i=1}^{n}(x_i - \bar{x})^2}$$

easy!

In [4]:
```
t = qt(alpha/2, n-1, lower.tail = FALSE)
s = sqrt((1/(n-1)) * sum((samp - mean(samp))^2))

CIU = t * s / (sqrt(n))

maxu = avg + CIU
minu = avg - CIU

print(c(maxu, minu))

if(mu < maxu && mu > minu){
    print("True mean is within the maximum and minimum")
} else if (mu > maxu || mu < minu){
    print("True mean is NOT within the interval")
}
```

```
[1] 3.535111 2.426461
[1] "True mean is within the maximum and minimum"
```

1.3 Do it 10,000 times!

Now we have to see if our 95% cofidence intervals really work 95% of the time. 10,000 is large enough to see the true average.

I'll set up a for loop to take a random sample from the uniform distribution 10,000 times. I'll then see how many times the mean of that sample falls within each of the confidence intervals I created. I'll then add the known and unknown confidence intervals to those averages, and see how many times the true mean (3) falls into that range.

In [5]:
```
Z_success = 0
Z_failure = 0

T_success = 0
T_failure = 0

for(i in 1:10000){
    samp = runif(20, 1, 5)
    avg = mean(samp) #Sample Average

    maxk = avg + CIK
    mink = avg - CIK

    maxu = avg + CIU
    minu = avg - CIU


    if(mu < maxk && mu > mink){
    Z_success = Z_success + 1
    } else if (mu > maxk || mu < mink){
    Z_failure = Z_failure + 1
    }

    if(mu < maxu && mu > minu){
    T_success = T_success + 1
    } else if (mu > maxu || mu < minu){
    T_failure = T_failure + 1
    }

}
```

In [6]:
```
paste(round(Z_success / 10000, 4)*100,'% Success rate for the known variance', sep = "")
paste(round(T_success / 10000, 4)*100,'% Success rate for the unknown variance', sep = "")
```

'97.79% Success rate for the known variance'

'97.01% Success rate for the unknown variance'

Now this does NOT match up with what I expected. It could be that I made some error I don't yet understand, but this seems to be close to a 97% or 98% confidence interval rather than a 95%. I don't fully understand what I did wrong because I followed the steps pretty closley. It may have something to do with the 0.025 number since $100 - 2.5 \approx 97.5$ but where that mistake is made I do not know. It could be that the "random" numbers that R is using is less random and that has something to do with it?

Anyways, on to the next question!

Answers to 2 (sample size 100)

2.1 Known Variance

Since the steps are going to be the same as before, I won't do much explaining.

In [7]:
```
### Generating the sample of size 100 ###
samp = runif(100, 1, 5)

mu = (1+5)/2
n = 100
sigma = ((5-1)^2)/12
alpha = 0.05
avg = mean(samp)
```

In [8]:
```
### Known Variance ###
Z = qnorm(1-(alpha/2))
CIK = Z * (sigma/sqrt(n)) #CIK stands for Confidence Interval Known

maxk = avg+CIK #Known variance maximum
mink = avg-CIK #Known variance minimum

print(c(maxk, mink))

if(mu < maxk && mu > mink){
    print("True mean is within the maximum and minimum")
} else if (mu > maxk || mu < mink){
    print("True mean is NOT within the interval")
}
```

```
[1] 3.422294 2.899637
[1] "True mean is within the maximum and minimum"
```

2.2 uknown Variance

In [9]:
```
### Unknown Variance ###
t = qt(alpha/2, n-1, lower.tail = FALSE)
s = sqrt((1/(n-1)) * sum((samp - mean(samp))^2))

CIU = t * s / (sqrt(n))

maxu = avg + CIU
minu = avg - CIU

print(c(maxu, minu))

if(mu < maxu && mu > minu){
    print("True mean is within the maximum and minimum")
} else if (mu > maxu || mu < minu){
    print("True mean is NOT within the interval")
}
```

```
[1] 3.391305 2.930626
[1] "True mean is within the maximum and minimum"
```

2.3 Do it 10,000 times!

In [10]:
```
Z_success = 0
Z_failure = 0

T_success = 0
T_failure = 0

for(i in 1:10000){
    samp = runif(100, 1, 5)
    avg = mean(samp) #Sample Average

    maxk = avg + CIK
    mink = avg - CIK

    maxu = avg + CIU
    minu = avg - CIU


    if(mu < maxk && mu > mink){
    Z_success = Z_success + 1
    } else if (mu > maxk || mu < mink){
    Z_failure = Z_failure + 1
    }

    if(mu < maxu && mu > minu){
    T_success = T_success + 1
    } else if (mu > maxu || mu < minu){
    T_failure = T_failure + 1
    }

}
```

In [11]:
```
paste(round(Z_success / 10000, 4)*100,'% Success rate for the known variance', sep = "")
paste(round(T_success / 10000, 4)*100,'% Success rate for the unknown variance', sep = "")
```

'97.68% Success rate for the known variance'

'95.39% Success rate for the unknown variance'

Similair result for the known variance CI, but the unknown variance CI seems to be acting normally! At least for the unkown variance result, the result is what I expect!