1. When conducting the work of Lab 11, we conducted the test that uses the Central Limit Theorem even though the sample size was "small" (i.e., $n < 30$). It turns out, that how "far off" the t-test is can be computed using a first-order Edgeworth approximation for the error. Below, we will do this for the the further observations.

   (a) Boos and Hughes-Oliver (2000) note that

$$P(T \leq t) \approx F_Z(t) + \underbrace{\frac{\text{skew}}{\sqrt{n}} \frac{(2t^2+1)}{6} f_Z(t)}_{\text{error}},$$

   where $f_Z(\cdot)$ and $F_Z(\cdot)$ are the Gaussian PDF and CDF and skew is the skewness of the data. What is the potential error in the computation of the p-value when testing $H_0 : \mu_X = 0; H_a : \mu_X < 0$ using the zebra finch further data?

```
#read the data file
dat.finches <- read_csv("zebrafinches.csv")
#separate further column
dat.further <- dat.finches$further

mu0 = 0
#compute t-statistics to get the t-value for the further data
t_furth <- t.test(dat.further, mu = mu0, alternative = "less")
t <- t_furth$statistic #extract t
n <- length(dat.further) #get n

#calculate skewness of data
skew <- skewness(dat.further)

#use Gaussian pdf and calculate the error
pdf.finches <- dnorm(t, mean = 0, sd = 1)
potential.error <- (skew/sqrt(n))*((2*t^2 +1)/6)*pdf.finches

potential.error

##             t
## -1.226006e-13
```

   The potential error in the computation of the p-value when testing $H_0 : \mu_X = 0; H_a : \mu_X < 0$ using the zebra finch further data is $-1.2260063 \times 10^{-13}$, which is extremely close to 0. Therefore, the t-test results on the zebra finch further data can be trusted even though the sample size is small. The Central Limit Theorem approximation works very well for the data. The error is negative, which means the true p-value is slightly lower than we assumed.

   (b) Compute the error for t statistics from -10 to 10 and plot a line that shows the error across t. Continue to use the skewness and the sample size for the zebra finch further data.

```
#generate a vector of t-values and compute pdf for each of them
t.vals <- seq(from = -10, to = 10, length.out = 1000)
pdf.t.vals <- dnorm(t.vals, mean = 0, sd = 1)

#calculate the error across of all t
t.potential.error <- (skew/sqrt(n))*((2*t.vals^2 +1)/6)*pdf.t.vals

#create a tibble to plot the errors for t-statistics
dat.error.plot <- tibble(t.vals, t.potential.error)

#plot the error for the t-statistics
error.plot <- ggplot(dat.error.plot)+
geom_line(aes(x= t.vals, y = t.potential.error))+ #plot the line for the errors
theme_bw()+
labs(title = "Edgeworth approximation for the error for p-value",
     x = "t-statistics",
     y = "Potential error")
```

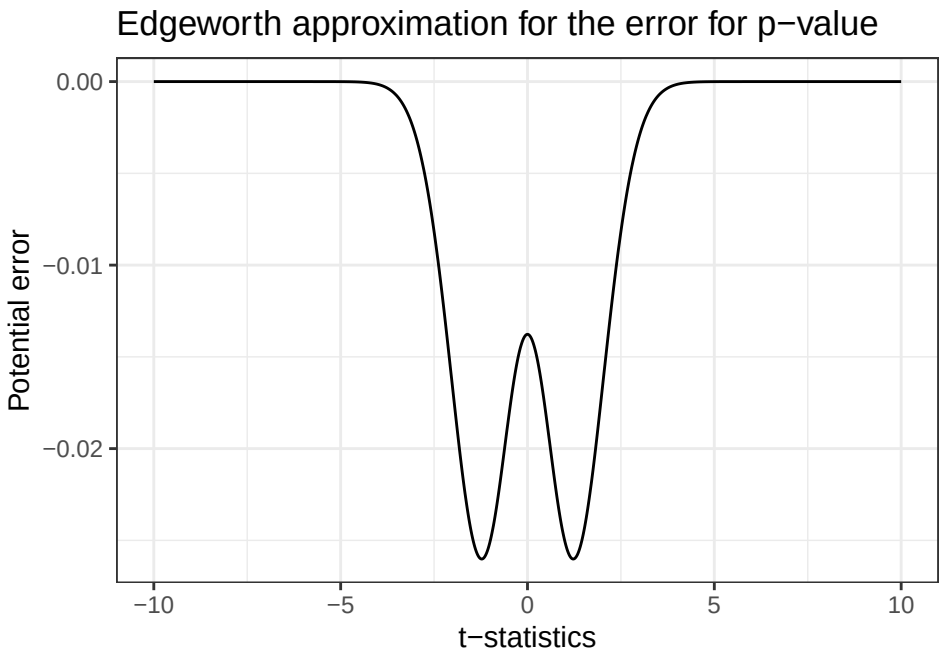

Figure 1: Edgeworth approximation for the error for p-value across t

The potential error in p-value estimation is minimal across all t values, which supports the validity of t-test approximation. As t gets closer to -10 and 10, the potential error is close to 0. The error becomes larger as t approaches 0. However, when t is very close to 0, the error decreases.

(c) Suppose we wanted to have a tail probability within 10% of the desired $\alpha = 0.05$. Recall we did a left-tailed test using the further data. How large of a sample size would we need? That is, we need to solve the error formula equal to 10% of the desired left-tail probability:

$$0.10\alpha \overset{set}{=} \underbrace{\frac{\text{skew}}{\sqrt{n}} \frac{(2t^2+1)}{6} f_Z(t)}_{\text{error}},$$

which yields

$$n = \left(\frac{\text{skew}}{6(0.10\alpha)} (2t^2+1) f_Z(t) \right)^2.$$

```
alpha <- 0.05
#calculate the critical value for the left-tailed test
t.crit <- qnorm(alpha, mean = 0, sd = 1)

#calculate pdf
pdf.sample <- dnorm(t.crit, mean = 0, sd = 1)

#calculate n
n.sample <- (skew/(6*(0.10*alpha))*(2*t.crit^2+1)*pdf.sample)^2
n.sample <- ceiling(n.sample) #round n to the closest positive integer
n.sample

## [1] 521
```

We would need a sample size of $n = 521$ to have a tail probability within 10% of the desired $\alpha = 0.05$. This large sample size compensates for the skewness in the data, controlling the approximation error.

2. Complete the following steps to revisit the analyses from lab 11 using the bootstrap procedure.

(a) Now, consider the zebra finch data. We do not know the generating distributions for the closer, further, and difference data, so perform resampling to approximate the sampling distribution of the T statistic:

$$T = \frac{\bar{x}_r - 0}{s/\sqrt{n}},$$

where $\bar{x}_r$ is the mean computed on the rth resample and s is the sample standard deviation from the original samples. At the end, create an object called `resamples.null.closer`, for example, and store the resamples shifted to ensure they are consistent with the null hypotheses at the average (i.e., here ensure the shifted resamples are 0 on average, corresponding to $t = 0$, for each case).

```
n.resamples <- 10000
mu0 <- 0

#separate closer and difference data
dat.closer <- dat.finches$closer
dat.diff <- dat.finches$diff

#get standard deviation
dat.closer.sd <- sd(dat.closer)
dat.further.sd <- sd(dat.further)
dat.diff.sd <- sd(dat.diff)

#store the approximation of T-statistics
t.stat.storage <- tibble(closer = rep(NA, n.resamples),
                         further = rep(NA, n.resamples),
                         diff = rep(NA, n.resamples))

#perform resampling and calculate T-statistics
for (i in 1:n.resamples){
#resample
resample.closer <- sample(dat.closer, size = n, replace = T)
resample.further <- sample(dat.further, size = n, replace = T)
resample.diff <- sample(dat.diff, size = n, replace = T)
#calculate T
t.stat.closer <- mean(resample.closer)/(dat.closer.sd/sqrt(n))
t.stat.further <- mean(resample.further)/(dat.further.sd/sqrt(n))
t.stat.diff <- mean(resample.diff)/(dat.diff.sd/sqrt(n))
#store the statistics
t.stat.storage$closer[i] = t.stat.closer
t.stat.storage$further[i] = t.stat.further
t.stat.storage$diff[i] = t.stat.diff
}

#store the shifted resamples in an object and shift the mean of the data to be 0 under null hypothesis
resamples.null.closer <- t.stat.storage$closer - mean(t.stat.storage$closer) + mu0
resamples.null.further <- t.stat.storage$further - mean(t.stat.storage$further) + mu0
resamples.null.diff <- t.stat.storage$diff - mean(t.stat.storage$diff) + mu0

#calculate the mean of resamples - should be 0 on average
mean.resample.closer <- mean(resamples.null.closer)
mean.resample.further <- mean(resamples.null.further)
mean.resample.diff <- mean(resamples.null.diff)

mean.resample.closer

## [1] -8.069767e-16

mean.resample.further

## [1] -9.716672e-17

mean.resample.diff

## [1] -4.621636e-16
```

We performed 10,000 resamples with replacement and computed t-statistics for each resample. Then, we shifted the resamples to make sure that they reflect the null hypothesis. We verified that the shifted resamples were properly centered around 0 by checking that their means were approximately zero: `mean.resample.closer` $= -8.0697671 \times 10^{-16}$, `mean.resample.further` $= -9.7166719 \times 10^{-17}$, and `mean.resample.diff` $= -4.6216364 \times 10^{-16}$.

(b) Compute the bootstrap p-value for each test using the shifted resamples. How do these compare to the t-test p-values?

(c) What is the 5^{th} percentile of the shifted resamples under the null hypothesis? Note this value approximates $t_{0.05,n-1}$. Compare these values in each case.

(d) Compute the bootstrap confidence intervals using the resamples. How do these compare to the t-test confidence intervals?

3. Complete the following steps to revisit the analyses from lab 11 using the randomization procedure.

(a) Now, consider the zebra finch data. We do not know the generating distributions for the closer, further, and difference data, so perform the randomization procedure

(b) Compute the randomization test p-value for each test.

(c) Compute the randomization confidence interval by iterating over values of μ_0.
Hint: You can "search" for the lower bound from Q_1 and subtracting by 0.0001, and the upper bound using Q_3 and increasing by 0.0001. You will continue until you find the first value for which the two-sided p-value is greater than or equal to 0.05.

4. **Optional Challenge:** In this lab, you performed resampling to approximate the sampling distribution of the T statistic using

$$T = \frac{\bar{x}_r - 0}{s/\sqrt{n}}.$$

I'm curious whether it is better/worse/similar if we computed the statistics using the sample standard deviation of the resamples (s_r), instead of the original sample (s)

$$T = \frac{\bar{x}_r - 0}{s_r/\sqrt{n}}.$$

(a) Perform a simulation study to evaluate the Type I error for conducting this hypothesis test both ways.

(b) Using the same test case(s) as part (a), compute bootstrap confidence intervals and assess their coverage – how often do we 'capture' the parameter of interest?

References

Boos, D. D. and Hughes-Oliver, J. M. (2000). How large does n have to be for z and t intervals? *The American Statistician*, 54(2):121–128.