1. A group of researchers is running an experiment over the course of 30 months, with a single observation collected at the end of each month. Let $X_1, ..., X_{30}$ denote the observations for each month. From prior studies, the researchers know that

$$X_i \sim f_X(x),$$

but the mean μ_X is unknown, and they wish to conduct the following test

$$H_0 : \mu_X = 0$$
$$H_a : \mu_X > 0.$$

At month k, they have accumulated data $X_1, ..., X_k$ and they have the t-statistic

$$T_k = \frac{\bar{X} - 0}{S_k/\sqrt{n}}.$$

The initial plan was to test the hypotheses after all data was collected (at the end of month 30), at level $\alpha = 0.05$. However, conducting the experiment is expensive, so the researchers want to "peek" at the data at the end of month 20 to see if they can stop it early. That is, the researchers propose to check whether t_{20} provides statistically discernible support for the alternative. If it does, they will stop the experiment early and report support for the researcher's alternative hypothesis. If it does not, they will continue to month 30 and test whether t_{30} provides statistically discernible support for the alternative.

(a) What values of t_{20} provide statistically discernible support for the alternative hypothesis?

```
t.val = qt(.95, df = 19)
```

Values > 1.73

(b) What values of t_{30} provide statistically discernible support for the alternative hypothesis?

```
t.val2 = qt(.95, df = 29)
```

Values > 1.70

(c) Suppose $f_X(x)$ is a Laplace distribution with $a = 0$ and $b = 4.0$. Conduct a simulation study to assess the Type I error rate of this approach.

```
library(VGAM)

## Loading required package:  stats4
## Loading required package:  splines

type1.cnt = 0
for(i in 1:1000){
dat = rlaplace(n = 30, scale = 4)
p.20 = t.test(dat[1:20], alternative = "greater")
p.30 = t.test(dat, alternative = "greater")
if(p.20$statistic > t.val){
  type1.cnt = type1.cnt + 1
}
else if(p.30$statistic>t.val2)
{
  type1.cnt = type1.cnt + 1
}
}
(proportion.type1 = type1.cnt/1000)

## [1] 0.079
```

Note: You can use the `rlaplace()` function from the `VGAM` package for `R` (Yee, 2010).

(d) **Optional Challenge:** Can you find a value of $\alpha < 0.05$ that yields a Type I error rate of 0.05?

2. Perform a simulation study to assess the robustness of the T test. Specifically, generate samples of size $n = 15$ from the Beta(10,2), Beta(2,10), and Beta(10,10) distributions and conduct the following hypothesis tests against the actual mean for each case (e.g., $\frac{10}{10+2}$, $\frac{2}{10+2}$, and $\frac{10}{10+10}$).

(a) What proportion of the time do we make an error of Type I for a left-tailed test?

```
 #Beta (10,2)
actual.mean1 = 10/(10+2)
actual.mean2 = 2/(2+10)
actual.mean3 = 10/(10+10)
total.b1 = 0
for(i in 1:1000){
  dat.b1 = rbeta(n = 15, shape1 = 10, shape2 = 2)
  b1 = t.test(dat.b1, alternative = "less", mu = actual.mean1)
  if(b1$p.value < .05){ #Because we know that the null should't be rejected
total.b1 = total.b1 + 1
  }
}
#Beta (2,10)
(type1.error.b1 = total.b1/1000) #Proportion of type1 errors
```

```
## [1] 0.023
```

```
total.b2 = 0
for(i in 1:1000){
  dat.b2 = rbeta(n = 15, shape1 = 2, shape2 = 10)
  b2 = t.test(dat.b2, alternative = "less", mu = actual.mean2)
  if(b2$p.value < .05){
total.b2 = total.b2 + 1
  }
}
(type1.error.b2 = total.b2/1000)
```

```
## [1] 0.102
```

```
#Beta (10,10)
total.b3 = 0
for(i in 1:1000){
  dat.b3 = rbeta(n = 15, shape1 = 10, shape2 = 10)
  b3 = t.test(dat.b3, alternative = "less", mu = actual.mean3)
  if(b3$p.value < .05){
total.b3 = total.b3 + 1
  }
}
(type1.error.b3 = total.b3/1000)
```

```
## [1] 0.056
```

(b) What proportion of the time do we make an error of Type I for a right-tailed test?

```
#Beta(10,2)
total.a1 = 0
for(i in 1:1000){
  dat.a1 = rbeta(n = 15, shape1 = 10, shape2 = 2)
  a1 = t.test(dat.a1, alternative = "greater", mu = actual.mean1)
  if(a1$p.value < .05){
total.a1 = total.a1 + 1
  }
}
(type1.error.a1 = total.a1/1000)
```

```
## [1] 0.086
```

```
#Beta(2,10)
total.a2 = 0
for(i in 1:1000){
  dat.a2 = rbeta(n = 15, shape1 = 2, shape2 = 10)
  a2 = t.test(dat.a2, alternative = "greater", mu = actual.mean2)
  if(a2$p.value < .05){
total.a2 = total.a2 + 1
  }
}
(type1.error.a2 = total.a2/1000)
```

```
## [1] 0.024
```

```
#Beta(10,10)
total.a3 = 0
for(i in 1:1000){
  dat.a3 = rbeta(n = 15, shape1 = 10, shape2 = 10)
  a3 = t.test(dat.a3, alternative = "greater", mu = actual.mean3)
  if(a3$p.value < .05){
total.a3 = total.a3 + 1
  }
}
(type1.error.a3 = total.a3/1000)
```

```
## [1] 0.057
```

(c) What proportion of the time do we make an error of Type I for a two-tailed test?

```
############
#2.c

#Beta(10,2)
totalc1 = 0
for(i in 1:1000){
  datc1 = rbeta(n = 15, shape1 = 10, shape2 = 2)
  c1 = t.test(datc1, alternative = "two.sided", mu = actual.mean1)
  if(c1$p.value < .05){
totalc1 = totalc1 + 1
  }
}
(type1.errorc1 = totalc1/1000)

## [1] 0.064

#Beta(2,10)
total.c2 = 0
for(i in 1:1000){
  dat.c2 = rbeta(n = 15, shape1 = 2, shape2 = 10)
  c2 = t.test(dat.c2, alternative = "two.sided", mu = actual.mean2)
  if(c2$p.value < .05){
total.c2 = total.c2 + 1
  }
}
(type1.error.c2 = total.c2/1000)

## [1] 0.076

#Beta(10,10)
total.c3 = 0
for(i in 1:1000){
  dat.c3 = rbeta(n = 15, shape1 = 10, shape2 = 10)
  c3 = t.test(dat.c3, alternative = "two.sided", mu = actual.mean3)
  if(c3$p.value < .05){
total.c3 = total.c3 + 1
  }
}
(type1.error.c3 = total.c3/1000)

## [1] 0.05
```

(d) How does skewness of the underlying population distribution effect Type I error across the test types?

It impacts the Type1 error for one sided tests. If a population distribution is skewed to the right, then there will be a cluster of data on the left which causes a greater type 1 error for a left sided t-test. The same applies for a right sided t.test if the data is skewed left. For two sided tests, the type 1 error isn't impacted because both sides are measured which ultimately causes the same overall p-value and type 1 error.

References

Yee, T. W. (2010). The VGAM package for categorical data analysis. *Journal of Statistical Software*, 32(10):1–34.