Estadística Inferencial con R

Arturo Pérez

1. Prueba t de student

Prueba estadística que nos permite hacer la comparación de las medias de 2 grupos y poder identificar si un grupo es diferente con respecto a otro.

```
## 1. Reading data
### control
poblacionctrl <- xlsx::read.xlsx("./data/Experimento 1.xlsx",
                                 sheetIndex = 1, colIndex = 1:15, startRow = 4, header = TRUE)
### experimental
poblacionexp <- xlsx::read.xlsx("./data/Experimento 1.xlsx",
                                sheetIndex = 3, colIndex = 1:15, startRow = 4, header = TRUE)
## 2. Filtrando datos y arreglando la tabla
### control
poblacionctrl <- poblacionctrl[, c("Slice", "Count", "Total.Area", "Circularity", "Solidity")]
poblacionctrl$Circularity <- as.numeric(poblacionctrl$Circularity)
poblacionctrl$Solidity <- as.numeric(poblacionctrl$Solidity)
###experimental
poblacionexp <- poblacionexp[, c("Slice", "Count", "Total.Area", "Circularity", "Solidity")]
poblacionexp$Circularity <- as.numeric(poblacionexp$Circularity)
poblacionexp$Solidity <- as.numeric(poblacionexp$Solidity)
## 3. Quitando valores perdidos
poblacionctrl <- na.omit(poblacionctrl)
poblacionexp <- na.omit(poblacionexp)
## 4. Variable Conteos
control <- poblacionctrl[grepl("FBS",poblacionctrl$Slice ),]
experimental <- poblacionexp[grepl("FBS",poblacionctrl$Slice ),]
conteos.ctrl <- control$Count
conteos.exp <- experimental$Count
poblacion <- c(conteos.ctrl,conteos.exp)
```

Ahora construimos los estadísticos para la prueba t. La fórmula es: $t = \frac{\mu_x - \mu_y}{SED}$

```
N <- length(conteos.exp)
N
```

```
## [1] 19
```

```
obs <- mean(conteos.ctrl)-mean(conteos.exp)
obs
```

```
## [1] 19.57895
```

```r
se <- sqrt(var(conteos.ctrl)/N + var(conteos.exp)/N)
t.stat <- obs/se
t.stat
```

```
## [1] 2.59305
```

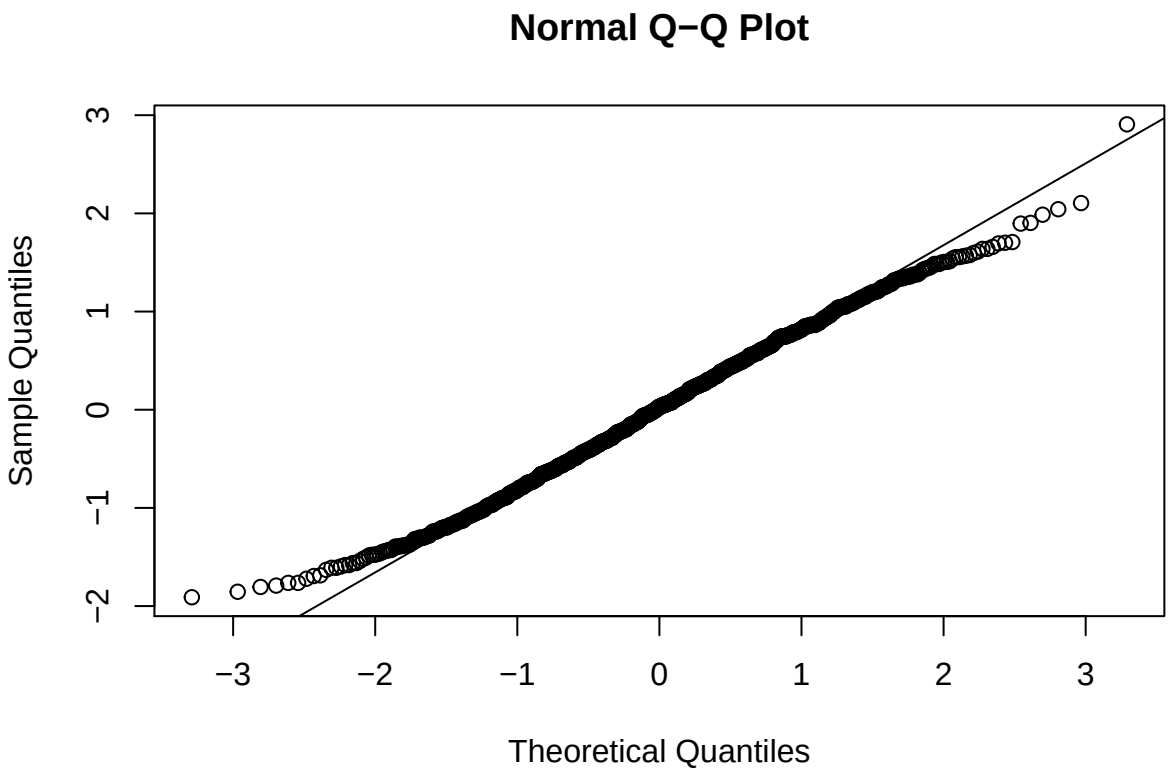
Asumiendo que nuestra distribució es normal de H0 con media 0 y sd 1 podemos sacar el valor p con `1-pnorm()` pero para dos colas. Es decir **ACEPTAR O RECHAZAR LA H0**.

```r
2*(1-pnorm(t.stat))
```

```
## [1] 0.009512896
```

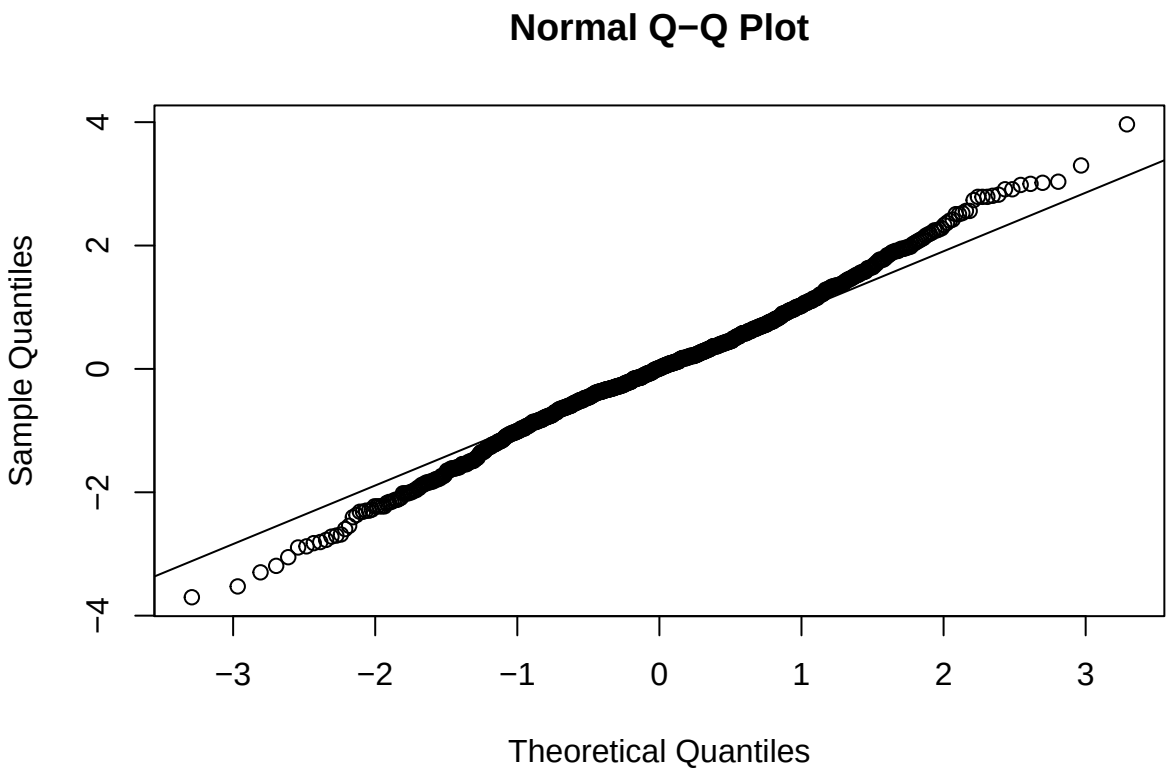
Para saber si la aproximación es buena lo comparamos con la población (SI TUVIERAMOS ACCESO). Haremos el loop, pero ahora dividimos los *nulls* sobre el *error estandar estimado*, básicamente haremos una prueba t varias veces, pero con una muestra más pequeña.

```r
nulls <- numeric(1000)
for (i in 1:1000) {
     control <- sample(poblacion, N)
     experimental <- sample(poblacion, N)
     se <- sqrt(var(conteos.ctrl)/N +
                           var(conteos.exp)/N)
     nulls[i] <- (mean(control)- mean(experimental))/se
}
##BUENA APROXIMACIÓN DEL VALOR P PARA EL ESTADISTICO T
qqnorm(nulls)
qqline(nulls)
```



Si nuestra muestra es más pequeña, usualmente la distribución normal deja de funcionar, este no es el caso, pero aumentar la N nos ayuda a que las colas no se alarguen tanto

```
nulls <- numeric(1000)
for (i in 1:1000) {
     control <- sample(poblacion, 3)
     experimental <- sample(poblacion, 3)
     se <- sqrt(var(conteos.ctrl)/3 +
                           var(conteos.exp)/3)
     nulls[i] <- (mean(control)- mean(experimental))/se
}
qqnorm(nulls)
qqline(nulls)
```



Ahora, podemos realizar la prueba t de una manera más sencilla con la función `t.test()`

```
t.stat
```

```
## [1] 2.59305
```

```
t.test(conteos.ctrl, conteos.exp)
```

```
##
##  Welch Two Sample t-test
##
## data:  conteos.ctrl and conteos.exp
## t = 2.593, df = 28.853, p-value = 0.01478
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##   4.13293 35.02496
## sample estimates:
## mean of x mean of y
##  33.52632  13.94737
```

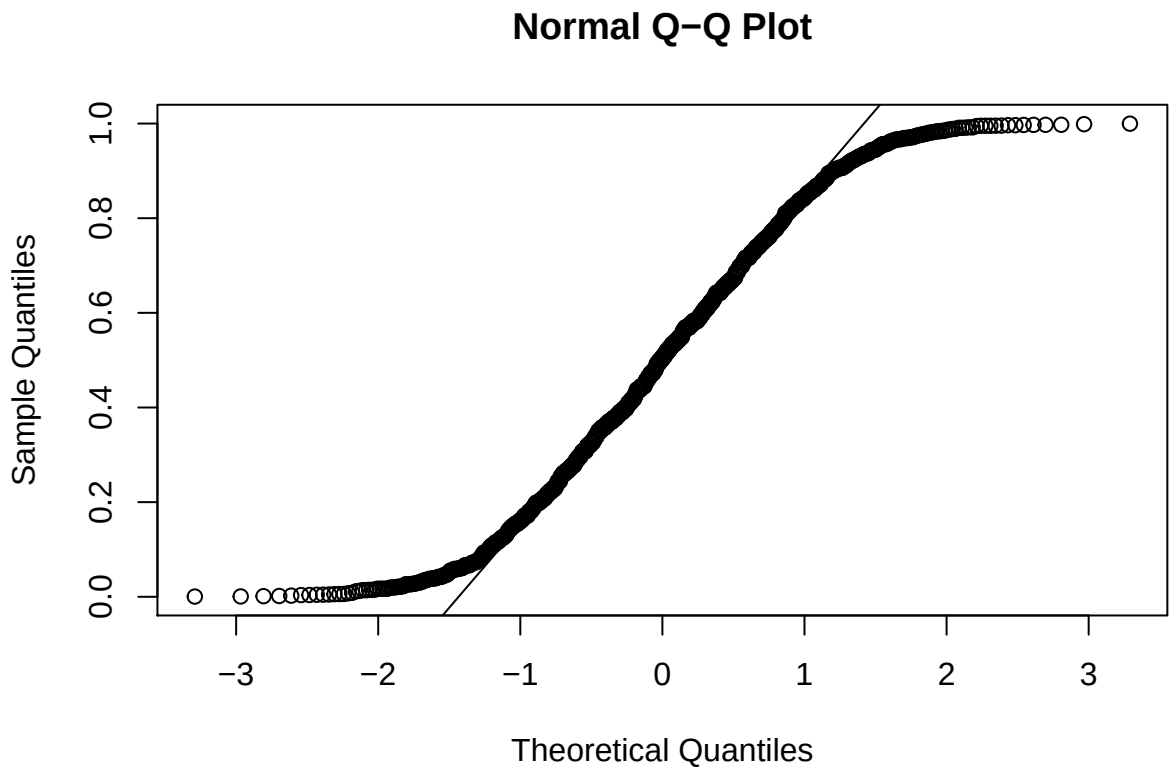
```
2*(1-pnorm(t.stat))
```

```
## [1] 0.009512896
```

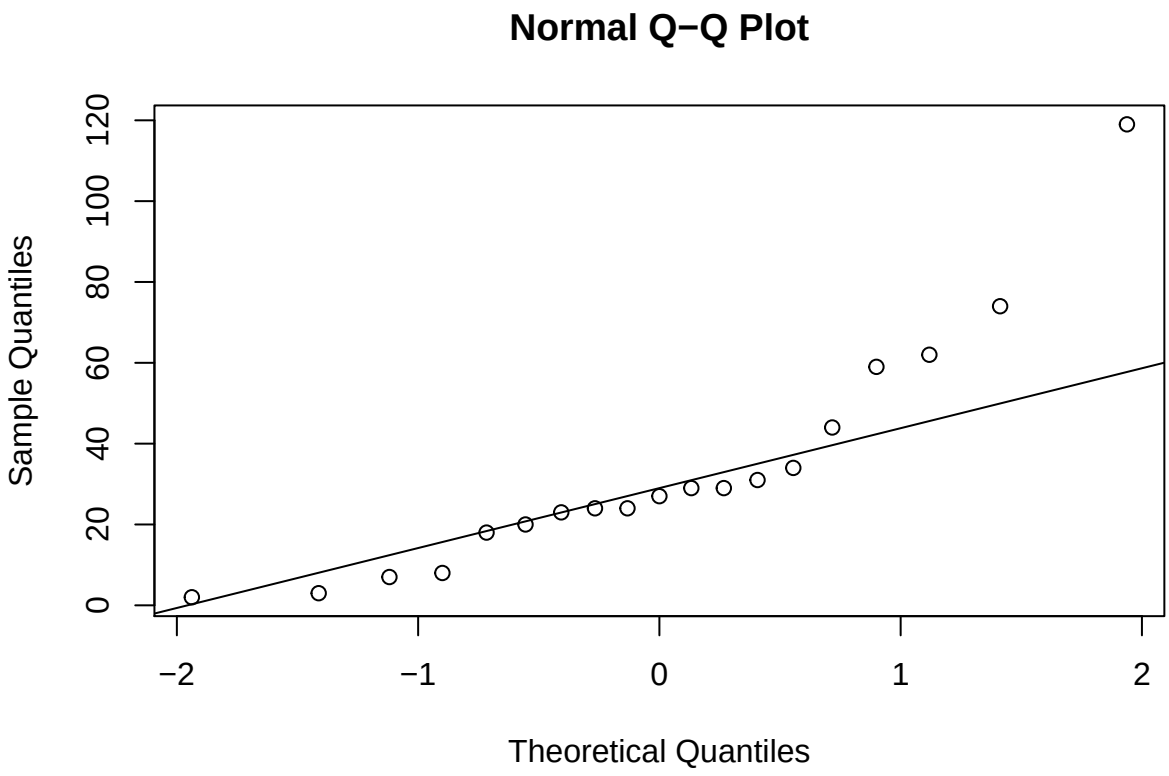
Como podemos observar, el valor p siguiendo el TEOREMA DEL LÍMITE CENTRAL (es decir, normalidad dentro de nuestros datos) es mucho más pequeño que el que nos da la prueba t. La razón de esto es porque la función `t.test()` no asume que la distribución normal aplica para el estadístico t, por lo tanto sigue una aproximación para la distribución t (tiene colas más grandes) por lo tanto este valor p será diferente.

Podemos utilizar un Q-Q plot para ver si nuestra distribución es aceptable para utilizar el valor p de la dist normal o de la dist. t.

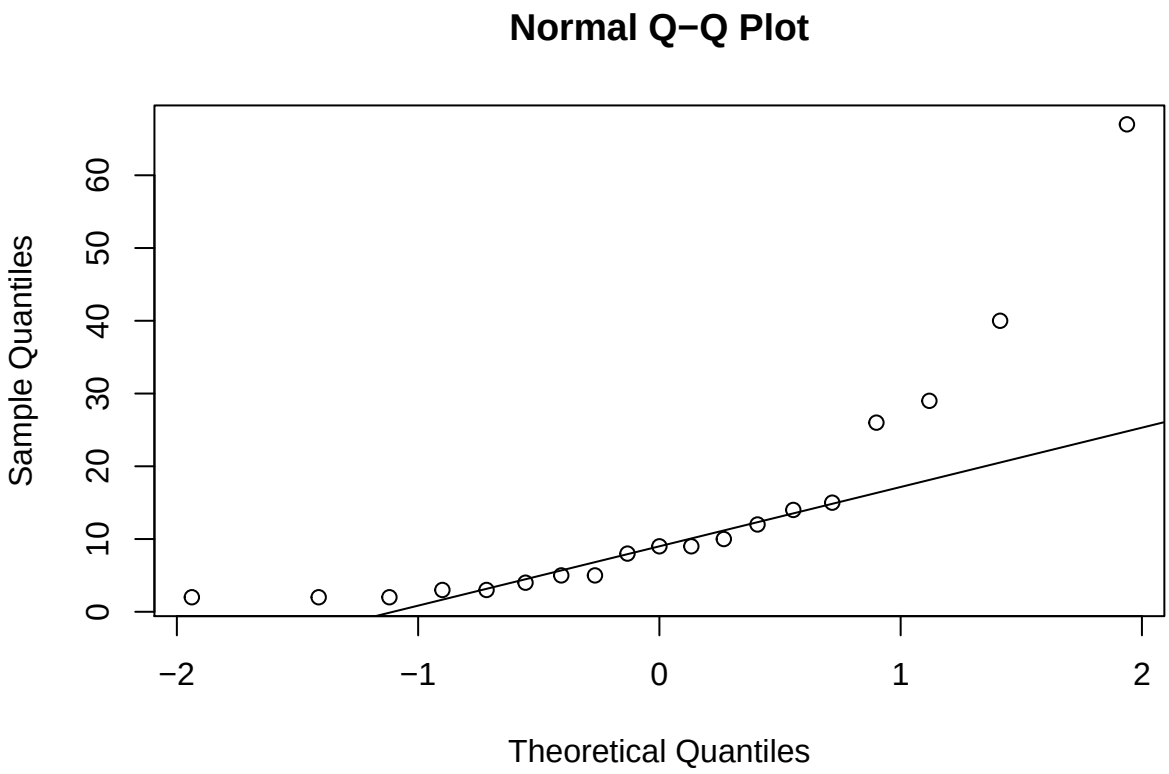
```
##primero así se ve una dist t comparada con una dist. normal
qqnorm(pt(nulls, 28.853))
qqline(pt(nulls, 28.853))
```




```
##mis datos
qqnorm(conteos.ctrl)
qqline(conteos.ctrl)
```




```
qqnorm(conteos.exp)
qqline(conteos.exp)
```



Con esto podemos asumir que es mejor utilizar el valor p de la distribución t a utilizar el valor p de la dist. normal

2. Valores p e intervalo de confianza

Reportar valores p nos da una historia incompleta de un descubrimiento. Por ejemplo, con muestras grandes, diferencias insignificantes pueden darnos un valor p muy pequeño, esto quiere decir qué reportar un valor p como único resumen estadístico no es útil. Una significancia estadística no garantiza significancia científica. Una diferencia sólo es una fracción, un porcentaje, no hay **tamaño del efecto** puesto que no hay población, así que una alternativa es reportar el intervalo de confianza.

El intervalo de confianza nos da info. acerca del tamaño del efecto estimado y la incertidumbre asociada con ese estimado.

```
set.seed(1)
population <- rnorm(150,15,2.5)
mediapop <- mean(population)## parametro real a estimar no existe en la vida real
mediapop
```

```
## [1] 15.05441
```

```
muestra <- sample(population, 30)
mediamuestra <- mean(muestra)
mediamuestra;mediapop
```

```
## [1] 14.60612
```

```
## [1] 15.05441
```

```
##error estandar de la muestra asumiendo que es normal
SE <- sd(muestra)/sqrt(30)
```

En teoría son la misma media, por lo tanto, cada que hagamos una muestra de una población, ese valor caerá el 95% de las veces entre -Q y Q.

El valor real es el de pa población

```
## 1=100%, 0.05 es nuestro alfa
## se divide entre 2 porque 2 colas
Q <- qnorm(1-0.05/2)
Q
```

```
## [1] 1.959964
```

```
##calculamos el intervalo en donde estarán el 95% de las obs
## de mi muestra (esto es lo que se reporta!!!)
intervalo <- c(mediamuestra-Q*SE, mediamuestra+Q*SE)
intervalo ; mean(population); mediamuestra
```

```
## [1] 13.95710 15.25515
```

```
## [1] 15.05441
```

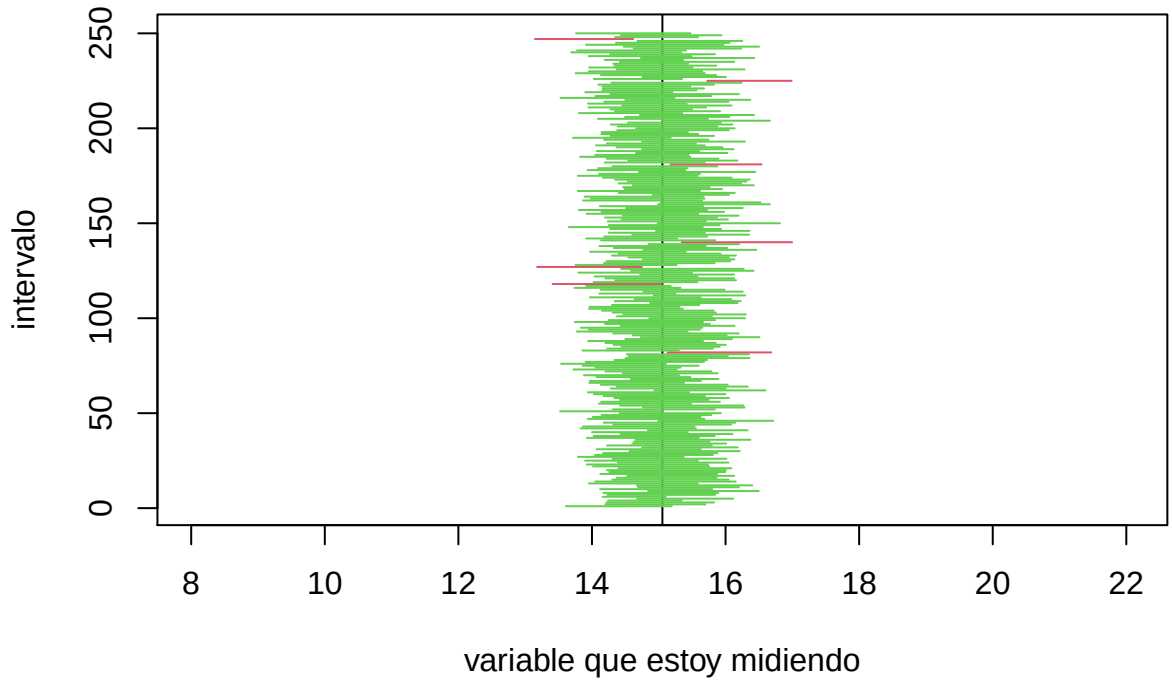
```
## [1] 14.60612
```

```
## vemos si esta condicion es verdadera
intervalo[1] < mediapop & intervalo[2] > mediapop
```

```
## [1] TRUE
```

Ahora, comprobemos que cada que hagamos una muestra de la población obtendremos una media dentro del intervalo de confianza el 95% de las veces.

```
plot(mediapop+c(-7,7),c(1,1), type="n",
                xlab="variable que estoy midiendo",
                ylab="intervalo", ylim=c(1,250))
abline(v=mediapop)
covered2 <- logical(250)
for (i in 1:250) {
     muestra <- sample(population,30)
     se <- sd(muestra)/sqrt(30)
     interval <- c(mean(muestra)-Q*se, mean(muestra)+Q*se)
     covered <- mediapop <= interval[2] & mediapop >= interval[1]
     covered2[i] <- covered
     color <- ifelse(covered,3,2)
     lines(interval, c(i,i),col=color)
}
```

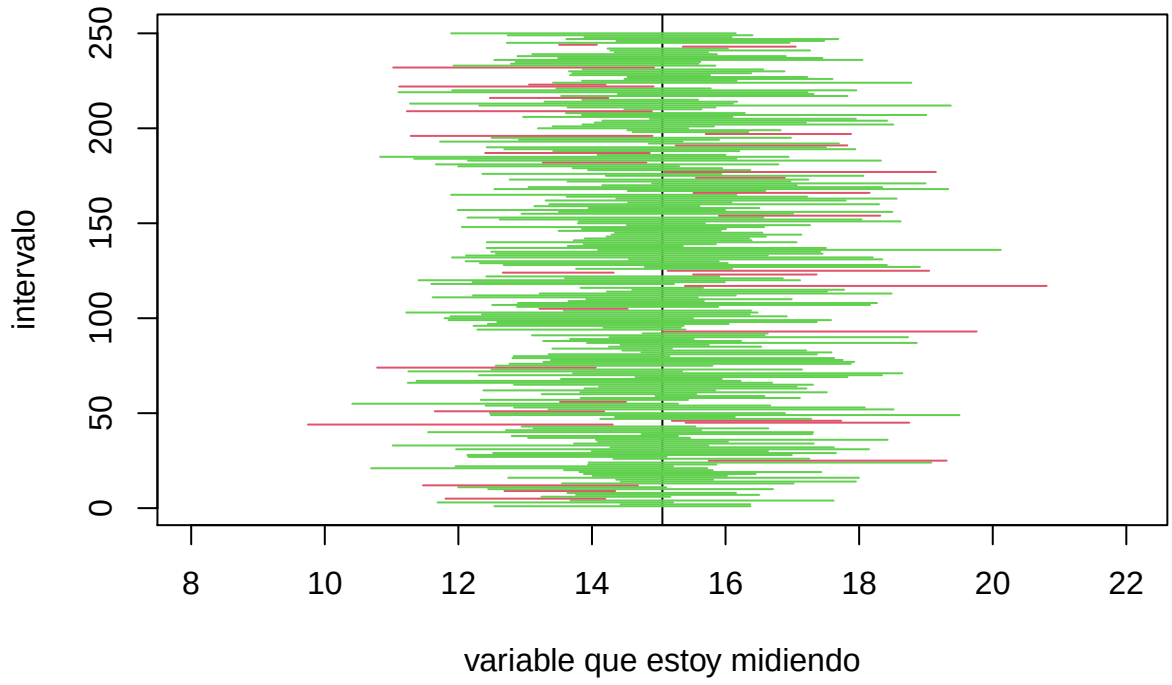



```
mean(covered2)
```

```
## [1] 0.972
```

En muestras pequeñas el intervalo de confianza no es correcto, utilizamos una distribución t. Si lo hacemos basandonos en el TLM, es decir, es variable aleatoria normalmente disribuida, vemos que ese 5% se rebasa.

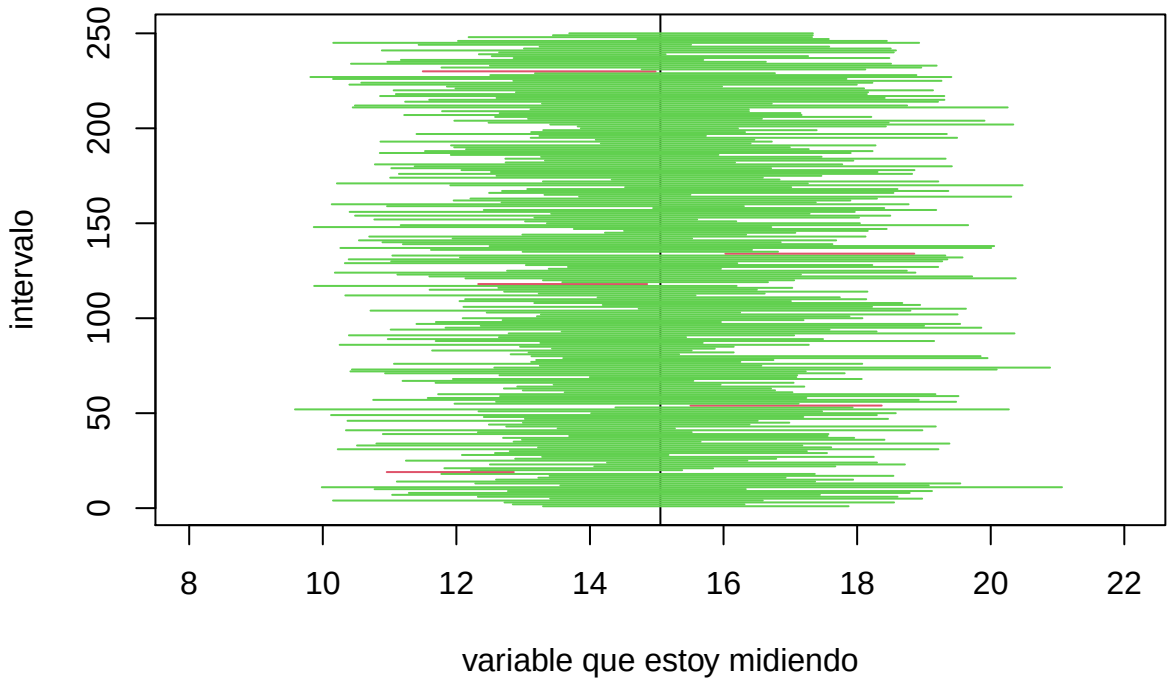
```
## primero con un intervalo para una dist normal
Q <- qnorm(1-0.05/2)
plot(mediapop+c(-7,7),c(1,1), type="n",
                xlab="variable que estoy midiendo",
                ylab="intervalo", ylim=c(1,250))
abline(v=mediapop)
for (i in 1:250) {
     muestra <- sample(population,5)
     se <- sd(muestra)/sqrt(5)
     interval <- c(mean(muestra)-Q*se, mean(muestra)+Q*se)
     covered <- mediapop <= interval[2] & mediapop >= interval[1]
     covered2[i] <- covered
     color <- ifelse(covered,3,2)
     lines(interval, c(i,i),col=color)
}
```




```
mean(covered2==TRUE)
```

```
## [1] 0.872
```

```
## intervalo con distribución t
Q <- qt(1-.05/2, df=4)
plot(mediapop+c(-7,7),c(1,1), type="n",
                xlab="variable que estoy midiendo",
                ylab="intervalo", ylim=c(1,250))
abline(v=mediapop)
for (i in 1:250) {
     muestra <- sample(population,5)
     se <- sd(muestra)/sqrt(5)
     interval <- c(mean(muestra)-Q*se, mean(muestra)+Q*se)
     covered <- mediapop <= interval[2] & mediapop >= interval[1]
     covered2[i] <- covered
     color <- ifelse(covered,3,2)
     lines(interval, c(i,i),col=color)
}
```




```
mean(covered2)
```

```
## [1] 0.98
```

Saquemos el intervalo de confianza para una muestra de datos reales:

```
Q1 <- qnorm(1-0.05/2)
Q2 <- qt(1-0.05/2, 18)
SE <- sd(conteos.ctrl)/sqrt(19)
intervaloQ1 <- c(mean(conteos.ctrl)-Q1*SE, mean(conteos.ctrl)+Q1*SE)
intervaloQ1;mean(conteos.ctrl)
```

```
## [1] 20.72005 46.33258
```

```
## [1] 33.52632
```

```
intervaloQ2 <- c(mean(conteos.ctrl)-Q2*SE, mean(conteos.ctrl)+Q2*SE)
intervaloQ2;mean(conteos.ctrl)
```

```
## [1] 19.79904 47.25359
```

```
## [1] 33.52632
```

3. Poder estadístico

Nos da una estimación de el porcentaje de veces que rechazaríamos la hipótesis nula, mientras más poder, más significancia entre la diferencia entre 2 grupos.

```
#1. calcular medias de ambos grupos a comparar y ver la diferencia.
media.conteos.ctrl <- mean(conteos.ctrl)
media.conteos.exp <- mean(conteos.exp)
diferencia <- media.conteos.ctrl - media.conteos.exp
diferencia
```

```
## [1] 19.57895
```

```
#2. Ver el porcentaje del incremento (o decremento) del grupo ctrl vs exp.
(media.conteos.ctrl-media.conteos.exp) / media.conteos.ctrl*100
```

```
## [1] 58.39874
```

```
#3. tomar muestra pequeña en cada grupo
set.seed(1)
muestra.ctrl <- sample(conteos.ctrl, 5)
muestra.exp <- sample(conteos.exp, 5)
```

Corremos una prueba t para las muestras de ambos grupos. Si la p no sale significativa usualmente diríamos que es porque no hay suficiente evidencia y no existen esas diferencias (donde probablemente si las haya)(ERROR TIPO II). Esto se debe a que tenemos muestras pequeñas, si las incrementamos la p va haciéndose más pequeña. Cometer ERROR TIPO II es muy común con muestras pequeñas.

```
#4. correr prueba t (más muestra, el valor p es más pequeño)
muestra.ctrl <- sample(conteos.ctrl, 12)
muestra.exp <- sample(conteos.exp, 12)
t.test(muestra.ctrl, muestra.exp)$p.value
```

```
## [1] 0.05980872
```

Ahora estableceremos un alfa para rechazar la hipótesis nula y haremos una función que corra un N número de veces múltiples tests dandonos un valor lógico que nos diga si rechazamos H0 en favor de H1

```
rechazo <- function(N, alfa=0.05){
     exp <- sample(conteos.exp, N)
     ctrl <- sample(conteos.ctrl, N)
     pval <- t.test(exp, ctrl)$p.value
     pval <= alfa
}
## Nos va a dar FALSE si la muestra es pequeña (error tipo 2)
rechazo(3)
```

```
## [1] FALSE
```

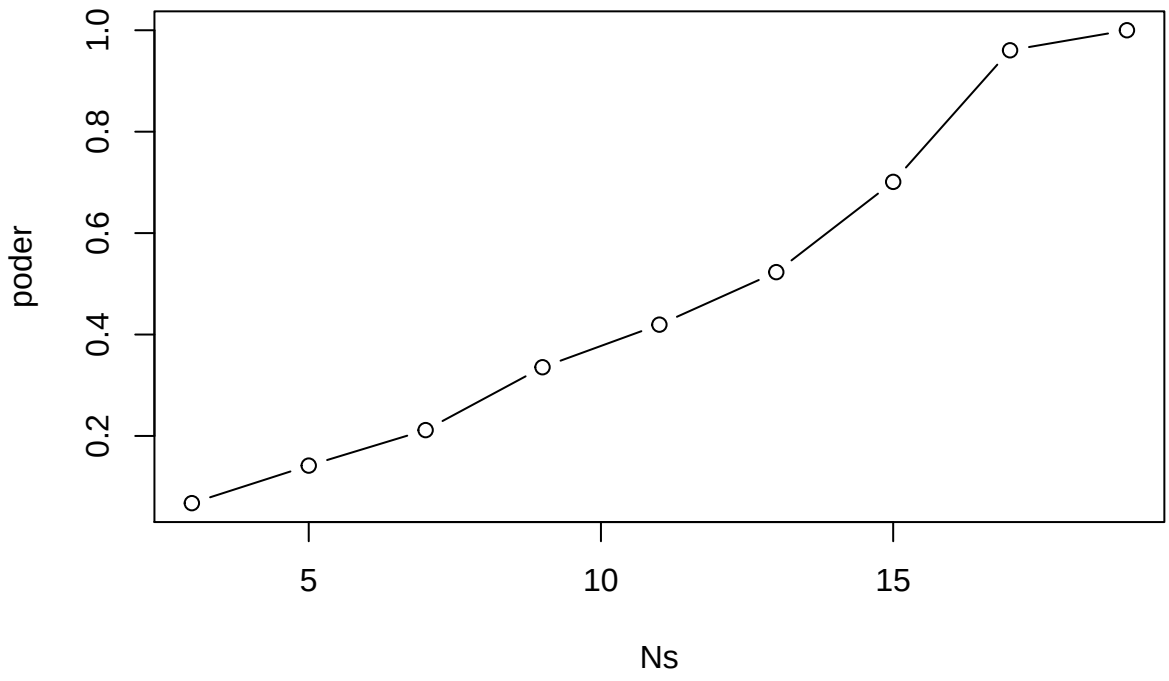
```r
## Hagamos esto 2000 veces para una N de 5 y saquemos la proporcion
## de veces que si hubo diferenicias
rechazosH1 <- replicate(2000, rechazo(5))
mean(rechazosH1)
```

```
## [1] 0.1155
```

El 13 % de las veces se rechazó H0, ESTE ES EL PODER ESTADÍSTICO y es un poder muy bajo en este caso.

Ahora observemos qué pasa cuando incremento la n de mi muestra

```r
Ns <- seq(3,19,2)
##repito lo mismo de arriba pero con tamaños de muestra diferentes
poder <- sapply(Ns, function(N){
     rechazosH1 <- replicate(2000, rechazo(N))
     mean(rechazosH1)
})
plot(Ns, poder, type = "b")
```



4. Permutaciones

Lo anteriormente visto es muy ideal, ya que en la gran mayoría de ocasiones no tenemos acceso a los valores de toda la población. No podemos hacer una simulación como anteriormente, es decir, obtener el poder

estadístico y no estamos seguros de que nuestro intervalo de confianza sea el de la población real, por lo tanto hacemos uso de las permutaciones.

Tenemos un escenario con 2 muestras de 19 sujetos cada una

```
conteos.ctrl; conteos.exp
```

```
##  [1] 119  29  24  44  31   3   2   8  23  24  27  29  34  18  59  20   7  74  62
```

```
##  [1]  8  9 67  9 12 40 29  4 26  2 14  5  2  3  2  5  3 10 15
```

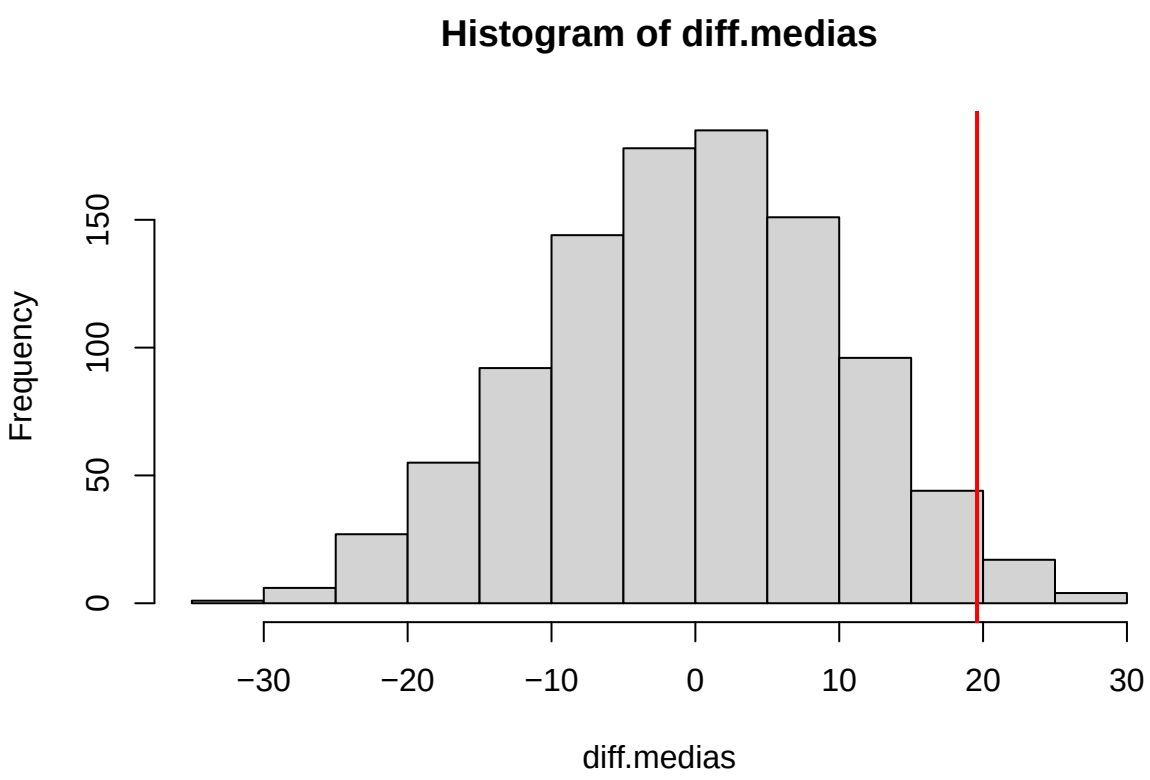
```
diferencia <- media.conteos.ctrl - media.conteos.exp
diferencia
```

```
## [1] 19.57895
```

Lo anteriormente visto son aproximaciones paramétricas que nos ayudarán a determinar si la diferencia observada es significativa. Las permutaciones toman ventaja del hecho de que si barajeamos los casos y controles, entonces H0 es verdadera.

Entonces juntamos casos y controles y la distribución se aproxima a la distribución nula. Loas barajeamos 1000 veces

```
N <- 12
diff.medias <- replicate(1000, {
     todo <- sample(c(conteos.ctrl, conteos.exp))
     nuevos.ctrl <- todo[1:N]
     nuevos.exp <- todo[(N+1):(2*N)]
     return(mean(nuevos.ctrl)-mean(nuevos.exp))
})
hist(diff.medias)
##Ahí podría estar el valor p para rechazar H0
abline(v=diferencia, col="red", lwd=2)
```



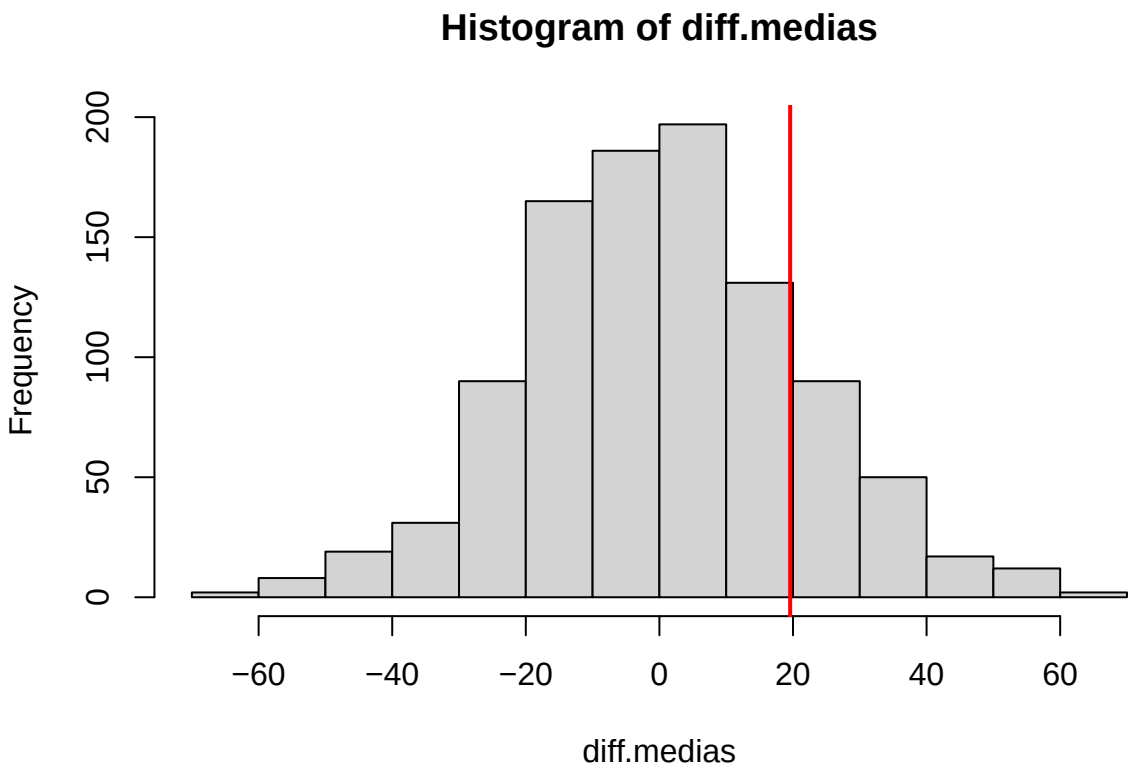
Veamos la proporción de las permutaciones más alla de la diferencia. Más 1 para considerar una subestimación del valor p, es decir, ponernos exigentes.

```
(sum(abs(diff.medias)>=abs(diferencia)+1))/(length(diff.medias)+1)
```

```
## [1] 0.04995005
```

Con una muestra más pequeña

```
N <- 3
diff.medias <- replicate(1000, {
     todo <- sample(c(conteos.ctrl, conteos.exp))
     nuevos.ctrl <- todo[1:N]
     nuevos.exp <- todo[(N+1):(2*N)]
     return(mean(nuevos.ctrl)-mean(nuevos.exp))
})
hist(diff.medias)
##Ahí podría estar el valor p para rechazar H0
abline(v=diferencia, col="red", lwd=2)
```




```
##Probabilidad
(sum(abs(diff.medias)>=abs(diferencia)+1))/(length(diff.medias)+1)
```

```
## [1] 0.3086913
```

La p ya no funciona, la diferencia observada ya no es signficativa utilizando esta aproximación

Consideraciones de las permutaciones:

```
No hay garantía teórica de que la distribución
estimada de permutaciones se aproxime a la distribución
nula real. Las colas pueden ser más grandes.

Con poca muestra, estas permutaciones no las podemos
hacer.

Se asumen muestras como independientes y no intercambiables

Las permutaciones pueden resultar en distribuciiones nulas
estimadas que subestimen el tamaño de las colas.
```