Universidad de Costa Rica

Facultad de Ingeniería

Escuela de Ciencias de la Computación e Informática

Diseño de Experimentos

Laboratorio 3

Experimentos comparativos simples. Prueba T para comparar dos medias independientes

2025

Autor

Brandon Trigueros Lara C17899

David González Villanueva C13388

Ejercicio de laboratorio A: Tamaño del paquete

Se quiere determinar si existen evidencias significativas de que el tamaño del paquete envíado afecta el tiempo de transferencia entre nodos.

NOTA: Para todo el ejercicio supondremos un nivel de significación de 0.05 (el Alpha!)

Carga de datos y exploración de variable

Cargamos los datos desde el csv y hacemos la primera exploración con la función head():

```
library(tidyverse)
```

```
redes = (read.csv("red2025aleatorio.csv", header=TRUE, encoding = "UTF-8"))
```

```
attach(redes)
```

```
head(redes, 10)
```

Description: df [10 × 10]

	Random1 <int>	Registro <int>	uso_cpu <int>	uso_memoria <int>	distancia <chr>	hora <int>	minuto <int>	tiempo <dbl>	paquetes <chr>
1	1	126	NA	18	largo	7	40	1.69	pequenos
2	1	73	29	30	corto	15	44	6.38	grandes
3	3	50	NA	32	corto	8	30	6.81	pequenos
4	3	13	25	19	corto	13	47	8.38	grandes
5	4	99	34	31	corto	14	44	7.50	pequenos
6	4	90	32	25	corto	13	50	7.75	pequenos
7	5	72	27	21	corto	8	26	5.94	pequenos
8	6	2	34	36	corto	5	35	7.69	grandes
9	8	43	35	28	corto	10	25	6.88	grandes
10	9	80	39	37	corto	5	45	5.13	grandes

1-10 of 10 rows | 1-10 of 10 columns

Utilizamos además la función table sobre la variable paquetes para explorar sus valores y frecuencias:

```
table(paquetes)
```

```
paquetes
 grandes pequenos 
     136      114 
```

Vemos que la variable paquetes tiene dos valores posibles: grandes (136) y pequenos (114).

1. Establecer la Hipótesis

H0: no hay diferencia entre las medias poblacionales: $\mu(grandes) = \mu(pequenos)$

HA: si hay diferencia entre las medias poblacionales: $\mu(grandes) \neq \mu(pequenos)$

2. Parámetro estimado (estadístico)

Se calcula la diferencia entre las medias muestrales. Primero se crea time_grandes con los datos de tiempos filtrados por el tamaño de paquete grande, y luego time_pequenos para los paquetes pequeños. Luego se calcula la diferencia de las medias de ambas:

```
time_grandes <- redes %>% filter(paquetes == "grandes") %>% pull(tiempo)

time_pequenos <- redes %>% filter(paquetes == "pequenos") %>% pull(tiempo)

mean(time_grandes) - mean(time_pequenos)
```

```
[1] 0.4987049
```

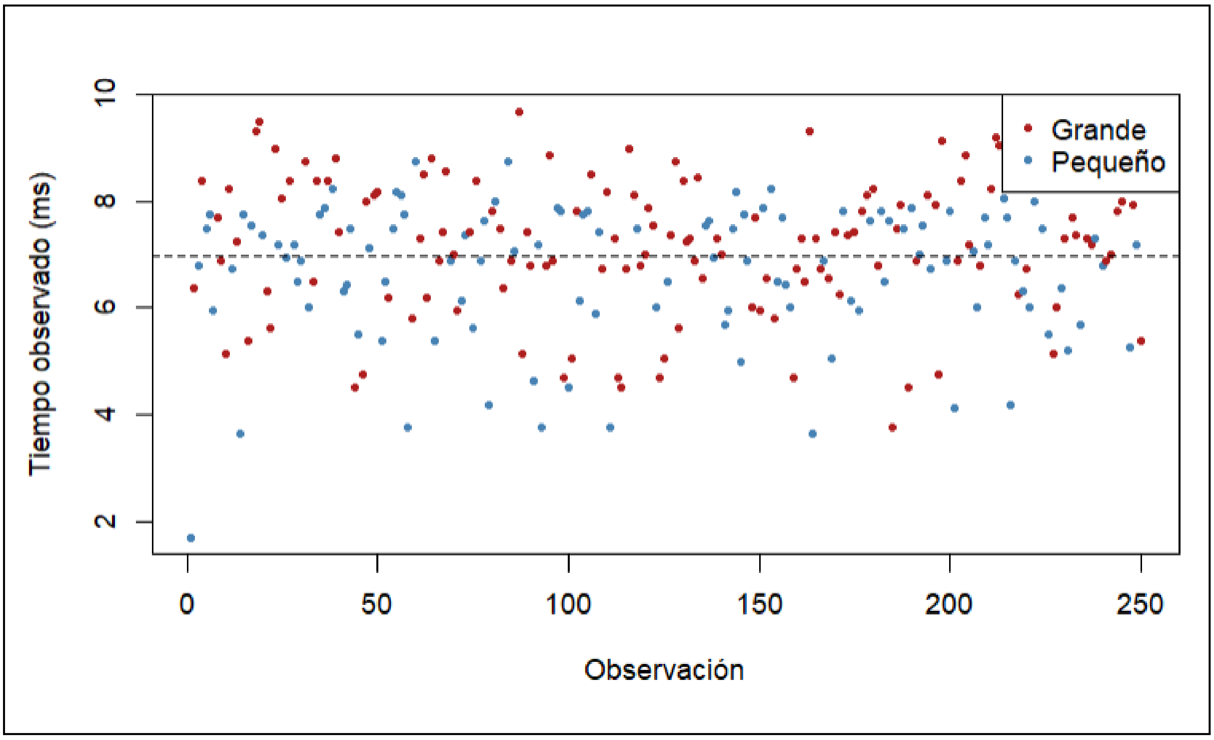
Hay una diferencia de 0.4987049 milisegundos entre ambas medias, pero todavía no sabemos si esta diferencia es significativa o no.

3. Validar las condiciones para aplicar una prueba T

3.1. Independencia:

Podemos realizar un gráfico de las observaciones respecto del tiempo en que se tomaron coloreado por tamaño de paquete:

```
if (length(dev.list()) > 1) dev.off()

plot(tiempo,

     ylab = "Tiempo observado (ms)",

     xlab = "Observación",

     col = ifelse(paquetes=="grandes","firebrick","steelblue"),

     pch = 20)

legend("topright",

       legend = c("Grande","Pequeño"),

       col = c("firebrick","steelblue"),

       pch = 20)

abline(h = mean(redes$tiempo), lty = 2)
```

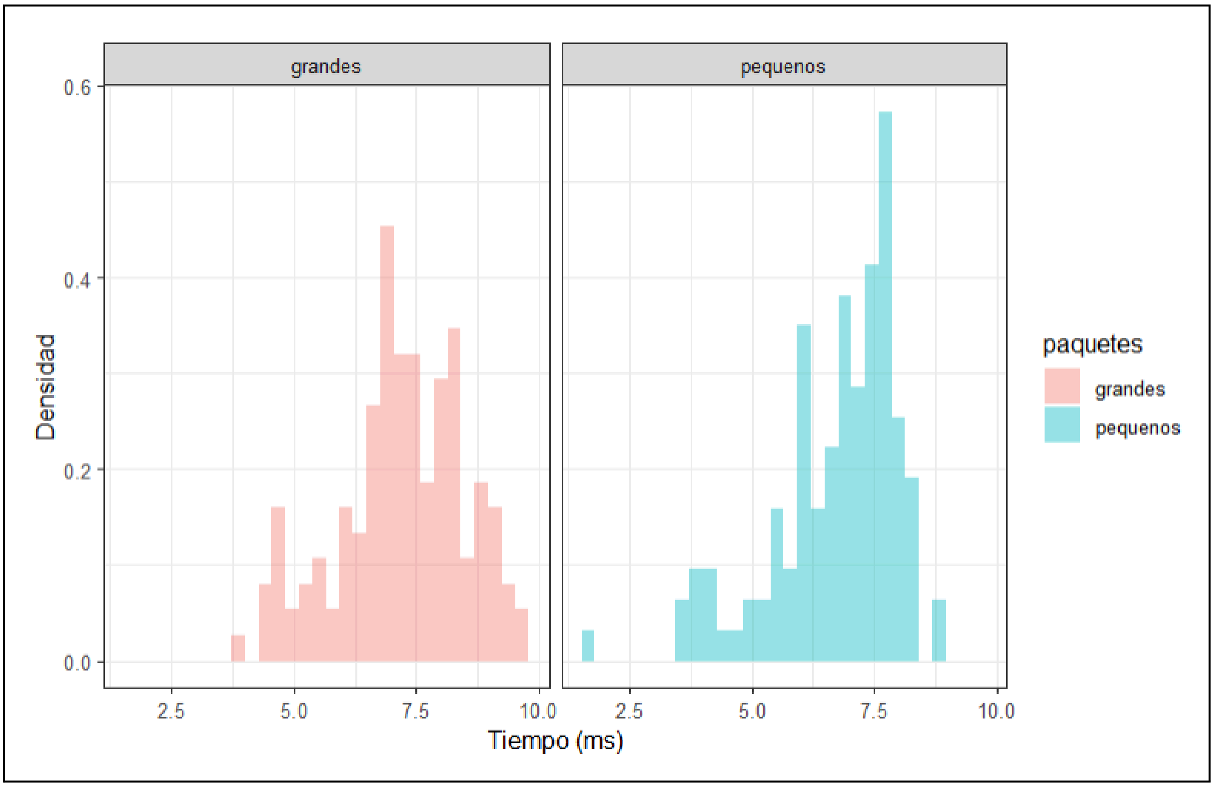


Puede verse que no se observan patrones aparentes que comprometan la independencia de las observaciones.

3.2. Normalidad:

Se crean histogramas por nivel del tiempo, uno por cada valor de la variable paquete, esto para ver la forma de las observaciones:

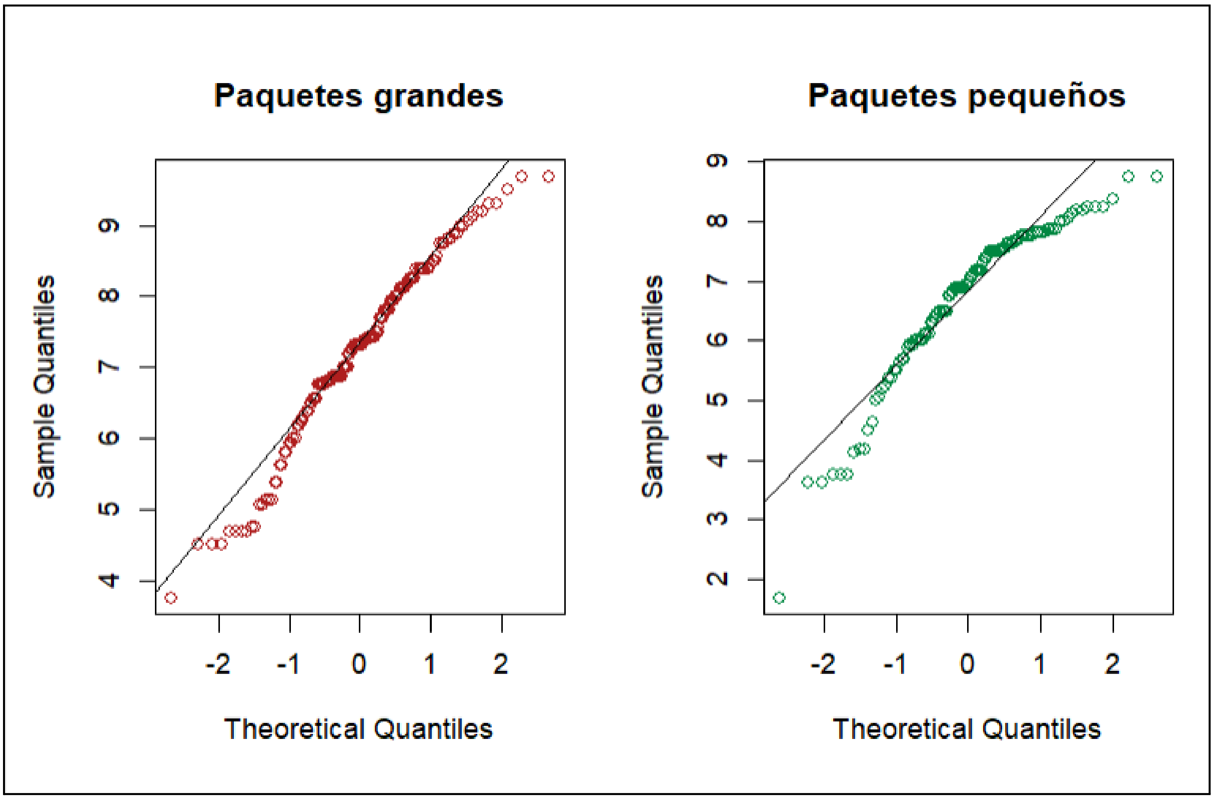
```
ggplot(redes,

       aes(x = tiempo, fill = paquetes)) +

geom_histogram(aes(y = after_stat(density)),

               alpha = 0.4,

               position = "identity",

               bins = 30) +

labs(x = "Tiempo (ms)",

     y = "Densidad") +

facet_wrap(~ paquetes) +

theme_bw()
```



Las formas de los histogramas no muestran claramente que las observaciones siguen una distribución normal. En ambos casos parece que hay una asimetría hacia la izquierda, siendo esta asimetría más pronunciada para el caso de paquetes pequeños.

Se muestra también la gráfica de probabilidad normal. Este gráfico compara los cuantiles de los datos observados con los cuantiles teóricos de la distribución normal. Si los puntos del gráfico caen sobre la línea recta, los datos se distribuyen aproximadamente de forma normal:

```
par(mfrow = c(1, 2))

qqnorm(time_grandes, main = "Paquetes grandes", col = "firebrick");
qqline(time_grandes)

qqnorm(time_pequenos, main = "Paquetes pequeños", col = "springgreen4");
qqline(time_pequenos)
```



En ambos casos las observaciones más pequeñas y más grandes caen por debajo de la línea teórica por lo que no parece que se siga una distribución normal para ninguno de los dos tipos de paquetes.

Se ejecuta el estadístico de prueba Shapiro-Wilks, primero para los datos de paquetes grandes y luego para los pequenos.

```
shapiro.test(time_grandes)
```

```
shapiro.test(time_pequenos)
```

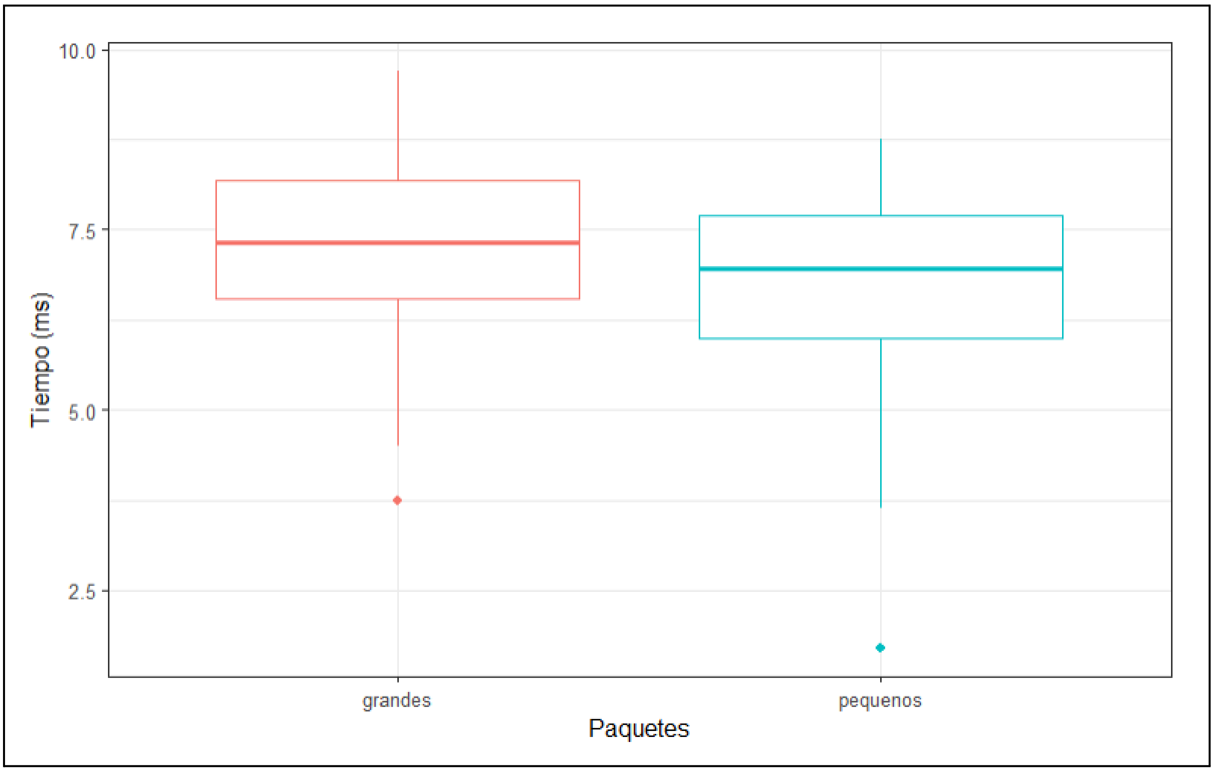
```
        Shapiro-Wilk normality test

data:  time_grandes
W = 0.97431, p-value = 0.01129


        Shapiro-Wilk normality test

data:  time_pequenos
W = 0.90942, p-value = 1.064e-06
```

Ambas pruebas encuentran evidencias significativas de no normalidad para ambos tipos tamaños de paquetes, ya que 1.064e-06 < 0.01129 < 0.05.

3.3. Igualdad de varianzas:

Podemos graficar boxplots con los datos para observar visualmente las distribuciones:

```
ggplot(redes,
       aes(x = paquetes, y = tiempo, colour = paquetes)) +
geom_boxplot() +
labs(x = "Paquetes",
     y = "Tiempo (ms)") +
theme_bw() +
theme(legend.position = "none")
```



Los boxplots nos dicen que los paquetes pequeños tienen ligeramente mayor dispersión en el rango intercuartil. Los valores máximos y mínimos son similares comparados con la mediana, que es menor para paquetes pequeños.

Viendo estos boxplots parecería que los datos para ambos tamaños de paquete sí presentan varianzas similares. Vamos a revisarlo con estadísticos para estar seguros. Dado que no se cumple el criterio de normalidad, uno estadístico recomendados es el test Levene o el test no paramétrico de Fligner-Killeen.Si se cumpliera el criterio de normalidad, se podría usar la prueba de Bartlett (en R bartlett.test):

```
library(car)

fligner.test(tiempo ~ paquetes, data = redes)

leveneTest(tiempo ~ as.factor(paquetes), data = redes, center = "median")
```

```
        Fligner-Killeen test of homogeneity of variances

data:  tiempo by paquetes
Fligner-Killeen:med chi-squared = 0.1345, df = 1, p-value = 0.7138

Levene's Test for Homogeneity of Variance (center = "median")
       Df F value Pr(>F)
group   1  0.0986 0.7538
      248
```

Ambas pruebas generan un p-value mayor que el Alpha establecido de 0.05, por lo que no hay razones para rechazar la hipótesis nula de que las varianzas son iguales. $0.05 < 0.7138 < 0.7538$.

4. Determinar el tipo de test

Se trata de una prueba de dos colas, dado que la hipótesis nula es con igualdad. Además, concluimos que las varianzas son iguales. También sabemos que las observaciones no siguen una distribución normal.

5. Determinar el nivel de significancia

Como se indicó al inicio del ejercicio, se usará $\alpha = 0.05$.

6. Cálculo de p-value

1) Realizamos entonces la prueba T para los datos de los dos paquetes, indicando mu = 0, dado que estamos usando la prueba para comparar ambas medias, o lo que es igual, que la diferencia de las medias es 0. También indicamos que las varianzas son iguales y que el nivel de confianza es 0.95, ó 1 – 0.005 (el nivel de significación):

```
t.test(
  x = time_grandes,
  y = time_pequenos,
  alternative = "two.sided",
  mu = 0,
  var.equal = TRUE,
  conf.level = 0.95 )
```

```
        Two Sample t-test

data:  time_grandes and time_pequenos
t = 3.0355, df = 248, p-value = 0.002657
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 0.1751243 0.8222854
sample estimates:
mean of x mean of y 
 7.213529  6.714825 
```

Esta prueba muestra un p-value de 0.002657, menor que el nivel de significancia de 0.05, por lo que sí habría bastante evidencia para rechazar la hipótesis nula.

2) Dado que no es cierto el supuesto de normalidad, vamos a ejecutar la prueba no paramétrica de Mann-Whitney.

```
wilcox.test(
  x = time_grandes,
  y = time_pequenos,
  alternative = "two.sided",
  mu = 0,
  paired = FALSE,
  conf.int = 0.95)
```

```
	Wilcoxon rank sum test with continuity correction

data:  time_grandes and time_pequenos
W = 9289, p-value = 0.006954
alternative hypothesis: true location shift is not equal to 0
95 percent confidence interval:
 0.1200314 0.7500419
sample estimates:
difference in location 
             0.4399555 
```

Puede verse que el p-value de 0.006954 es menor que el Alpha 0.05. Sí se dispone de evidencia para rechazar la hipótesis nula, la cuál dice que las medias de los tiempos de transferencia son iguales para los dos tamaños de paquetes.

7. Cálculo tamaño de efecto

Utilizamos la biblioteca effsize, con la función cohen.d:

```
library(effsize)
```

```
cohen.d(formula = tiempo ~ paquetes, data = redes, paired = FALSE)
```

```
Cohen's d

d estimate: 0.3854624 (small)
95 percent confidence interval:
    lower     upper 
0.1330643 0.6378605 
```

Puede verse que el d estimado es de 0.3854624 (en magnitud), por lo que se considera que el efecto del tamaño de paquete en los tiempos de transmisión es pequeño.

8. Potencia de la prueba

En este caso particular, sabemos que las varianzas muestrales son similares. No contamos con información sobre las varianzas poblacionales. Como las varianzas son similares, calculamos la media de ambas y usamos esta media como se muestra a continuación:

```
sd_grupo1 <- sd(time_grandes)

sd_grupo2 <- sd(time_pequenos)

sd_comun <- mean(c(sd_grupo1, sd_grupo2))

sd_comun
```

```
[1] 1.294015
```

La media de las varianzas es de 1.294015

También se necesita el delta que es la diferencia entre las medias de las poblaciones. Como no tenemos información, usaremos la diferencia entre las medias muestrales, que ya habíamos calculado en el punto 2:

```
delta_observado <- mean(time_grandes) - mean(time_pequenos)

delta_observado
```

```
[1] 0.4987049
```

El delta observado es 0.4987049

Además, debemos revisar si las muestras son de igual tamaño:

```
length(time_grandes)

length(time_pequenos)
```

```
[1] 136
[1] 114
```

En este caso hay 136 observaciones para paquetes grandes y 114 para paquetes pequeños. Para el cálculo de la potencia utilizaremos el valor más pequeño de los dos.

Seguidamente ejecutamos el comando power.t.test() indicando:

```
power.t.test(
  n = 114,
  delta = delta_observado,
  sd = sd_comun,
  sig.level = 0.05,
  type = "two.sample")
```

```
     Two-sample t test power calculation 

              n = 114
          delta = 0.4987049
             sd = 1.294015
      sig.level = 0.05
          power = 0.8256994
    alternative = two.sided

NOTE: n is number in *each* group
```

De este resultado se puede ver que con este conjunto de datos se obtuvo una potencia de prueba de 0.8256994, la cual es apropiada.

Se puede también utilizar esta misma función para saber de cuánto es la muestra mínima para llegar a una potencia de 0.80. Se modifica la invocación de la siguiente forma:

```
power.t.test(
  power = 0.8,
  delta = delta_observado,
  sd = sd_comun,
  sig.level = 0.05,
  type = "two.sample")
```

```
     Two-sample t test power calculation

              n = 106.6578
          delta = 0.4987049
             sd = 1.294015
      sig.level = 0.05
          power = 0.8
    alternative = two.sided

NOTE: n is number in *each* group
```

El resultado muestra que para obtener una potencia de prueba de 0.80 que esté relacionada con el delta encontrado (0.4987049) se necesitan al menos 106 observaciones para cada tamaño de paquete. Por lo que con las 114 que tenemos nos basta.

9. Análisis de resultados y conclusiones

En el análisis del efecto del tamaño de paquete sobre el tiempo de transferencia se ha encontrado evidencia estadística consistente de que los paquetes “grandes” tardan en promedio más que los “pequeños”. La diferencia muestral de medias es de 0.4987 ms, la prueba t de Student arrojó un $t = 3.0355$ con $df = 248$ y $p = 0.002657$ (< 0.05), confirmando la significación estadística de la diferencia.

La prueba no paramétrica de Mann–Whitney concordó con un $W = 9289$ y $p = 0.006954$, reforzando la decisión de rechazar H_0 pese a la no normalidad de las muestras.

El tamaño del efecto estimado por Cohen’s $d = 0.3855$ se considera “pequeño” según los umbrales de 0.2, 0.5 y 0.8.

Además, la potencia calculada con power.t.test() = 0.8257 supera el nivel recomendado de 0.8, mostrando que la muestra actual es suficiente para detectar el efecto observado.

Hay evidencia significativa de que el tamaño del paquete afecta el tiempo de transferencia entre nodos. Aunque el efecto estadístico es pequeño ($d \approx 0.39$), la prueba cuenta con suficiente potencia y confirma que el fenómeno es real, no atribuible al azar. Por tanto, al diseñar redes, conviene considerar la influencia del tamaño de los paquetes para optimizar el rendimiento.

Ejercicio de laboratorio B: Distancia

Se quiere determinar si existen evidencias significativas de que la distancia entre nodos afecta el tiempo de transferencia.

NOTA: Para todo el ejercicio supondremos un nivel de significación de 0.05 (el Alpha!)

Carga de datos y exploración de variable

Cargamos los datos desde el csv, hacemos la primera exploración con la función head().

```
library(tidyverse)
```

```
redes = (read.csv("red2025aleatorio.csv", header=TRUE, encoding = "UTF-8"))
```

```
attach(redes)
```

```
head(redes, 10)
```

Description: df [10 × 10]

	Random1 <int>	Registro <int>	uso_cpu <int>	uso_memoria <int>	distancia <chr>	hora <int>	minuto <int>	tiempo <dbl>	paquetes <chr>
1	1	126	NA	18	largo	7	40	1.69	pequenos
2	1	73	29	30	corto	15	44	6.38	grandes
3	3	50	NA	32	corto	8	30	6.81	pequenos
4	3	13	25	19	corto	13	47	8.38	grandes
5	4	99	34	31	corto	14	44	7.50	pequenos
6	4	90	32	25	corto	13	50	7.75	pequenos
7	5	72	27	21	corto	8	26	5.94	pequenos
8	6	2	34	36	corto	5	35	7.69	grandes
9	8	43	35	28	corto	10	25	6.88	grandes
10	9	80	39	37	corto	5	45	5.13	grandes

1-10 of 10 rows | 1-10 of 10 columns

Utilizamos además la función table sobre la variable distancia para explorar sus valores y frecuencias

```
table(distancia)
```

```
distancia
corto largo
  167    83
```

Vemos que la variable distancia tiene dos valores posibles: corto(167) y largo(83).

1. Establecer la Hipótesis

H0: no hay diferencia entre las medias poblacionales: $\mu(\text{corto}) = \mu(\text{largo})$

HA: si hay diferencia entre las medias poblacionales: $\mu(\text{corto}) \neq \mu(\text{largo})$

2. Parámetro estimado (estadístico)

Se calcula la diferencia entre las medias muestrales. Primero se crea time_corto con los datos de tiempos filtrados por las distancias cortas, y luego time_largo para los distancias largas. Luego se calcula la diferencia de las medias de ambas:

```
time_corto <- redes %>% filter(distancia == "corto") %>% pull(tiempo)

time_largo <- redes %>% filter(distancia == "largo") %>% pull(tiempo)

mean(time_corto) - mean(time_largo)
```

```
[1] 1.083399
```

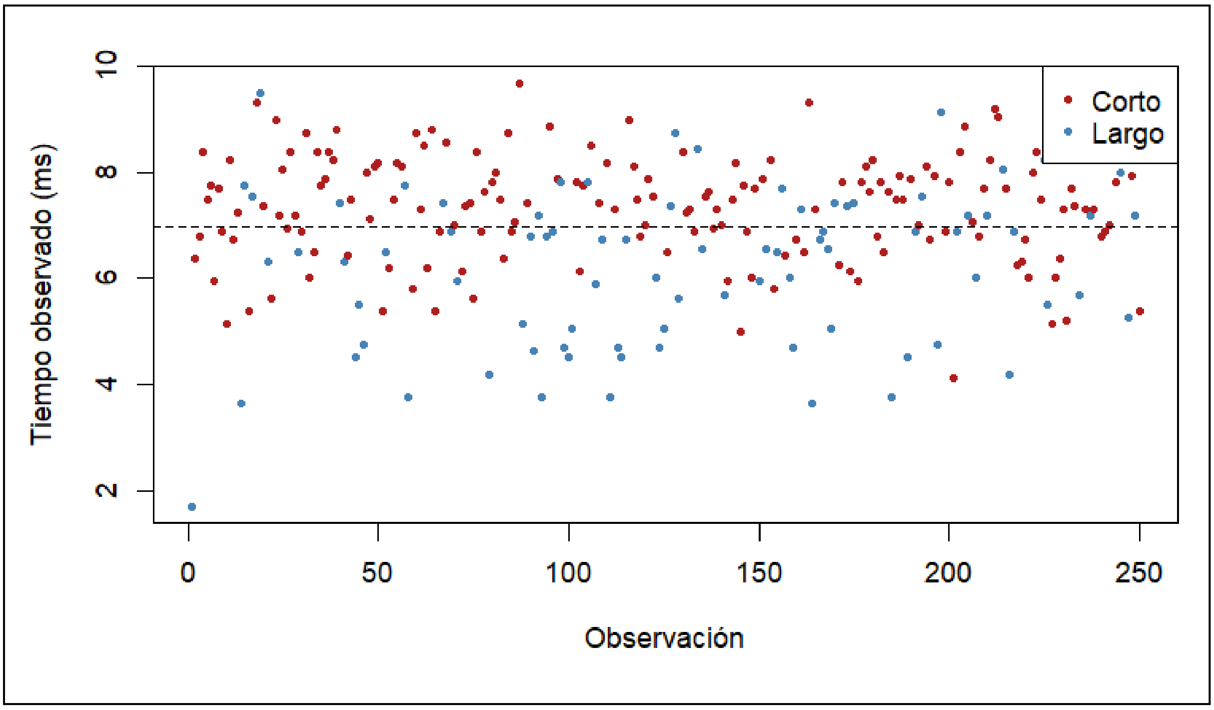
Hay una diferencia de 1.083399 milisegundos entre ambas medias, pero todavía no sabemos si esta diferencia es significativa o no.

3. Validar las condiciones para aplicar una prueba T

3.1. Independencia:

Podemos realizar un gráfico de las observaciones respecto del tiempo en que se tomaron coloreado por distancia:

```
if (length(dev.list()) > 1) dev.off()

plot(tiempo,

     ylab = "Tiempo observado (ms)",

     xlab = "Observación",

     col = ifelse(distancia=="corto","firebrick","steelblue"),

     pch = 20)

legend("topright",

       legend = c("Corto","Largo"),

       col = c("firebrick","steelblue"),

       pch = 20)

abline(h = mean(tiempo), lty = 2)
```

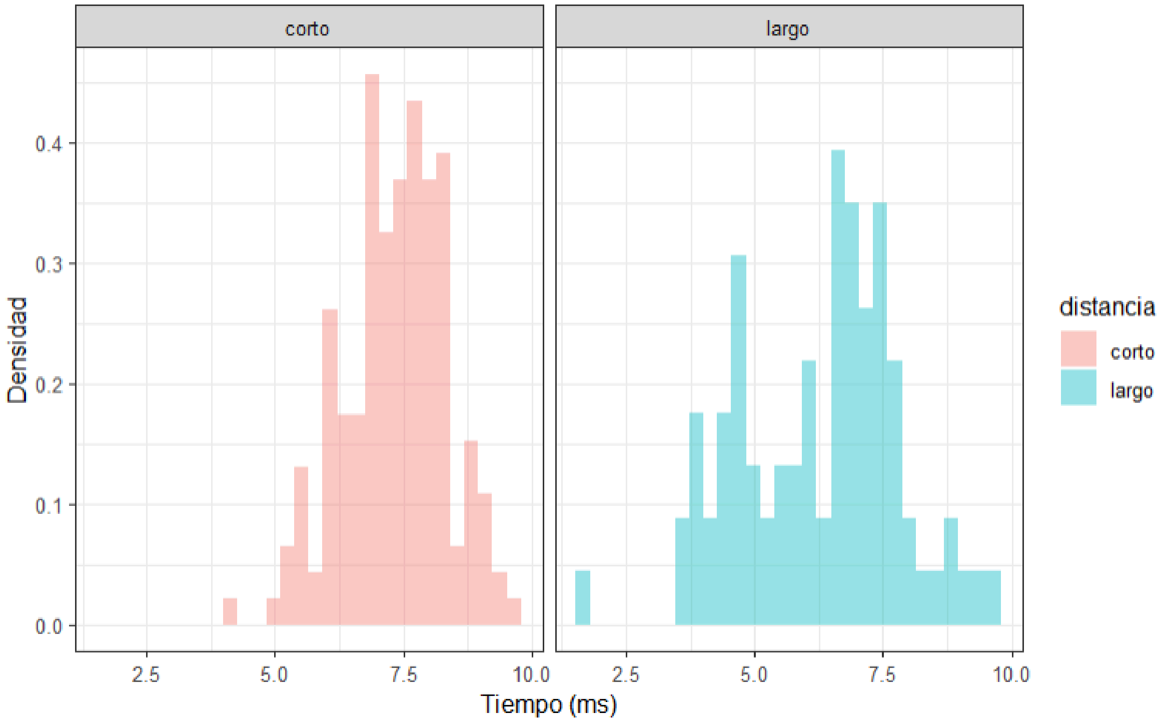


Puede verse que no se observan patrones aparentes que comprometan la independencia de las observaciones.

3.2. Normalidad:

Se crean histogramas por nivel para ver la forma de las observaciones:

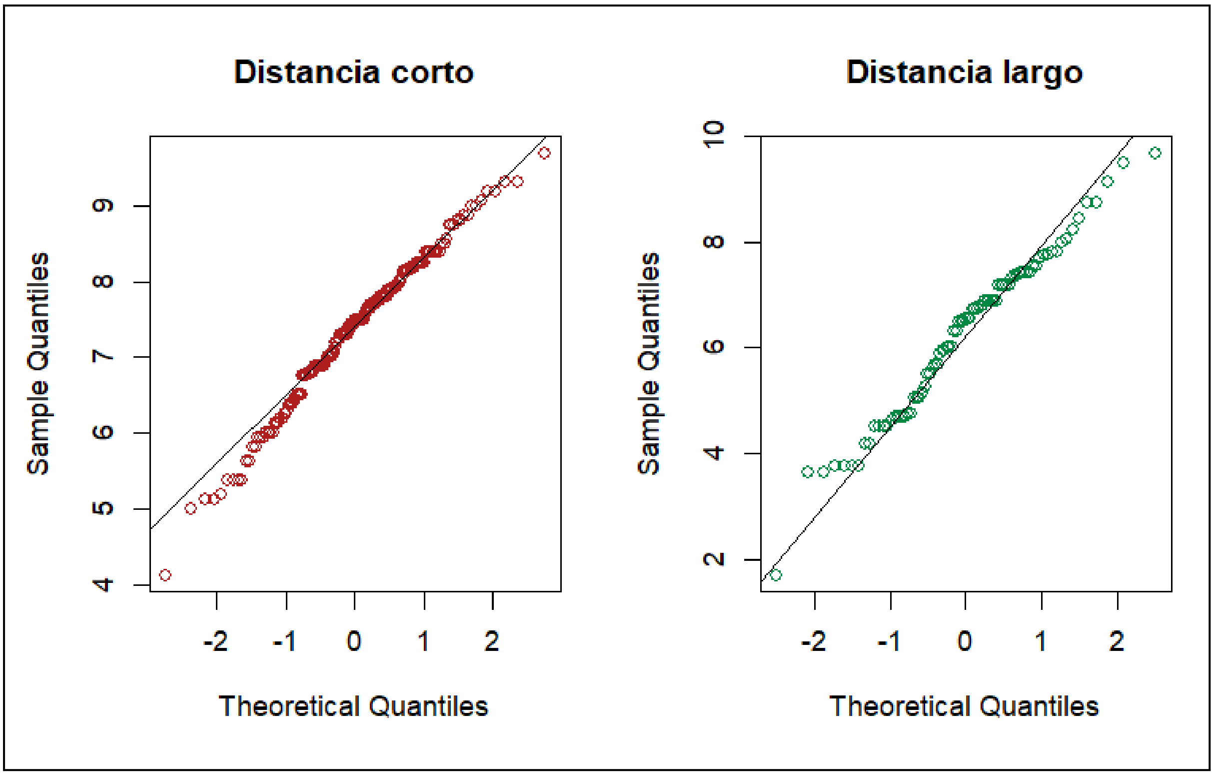
```
ggplot(redes,
       aes(x = tiempo, fill = distancia)) +
geom_histogram(aes(y = after_stat(density)),
               alpha = 0.4,
               position = "identity",
               bins = 30) +
labs(x = "Tiempo (ms)",
     y = "Densidad") +
facet_wrap(~ distancia) +
theme_bw()
```



Los histogramas podrían estar siguiendo una distribución normal pero no es del todo claro. En ambos casos parece que hay una asimetría hacia la izquierda.

Se muestra también la gráfica de probabilidad normal:

```
par(mfrow = c(1, 2))

qqnorm(time_corto, main = "Distancia corto", col = "firebrick");
qqline(time_corto)

qqnorm(time_largo, main = "Distancia largo", col = "springgreen4");
qqline(time_largo)
```



El gráfico QQ compara los cuantiles de los datos observados con los cuantiles teóricos de la distribución normal. El gráfico de distancia corto parece seguir más la distribución normal, aunque las observaciones menores caen bajo la línea. Mientras que en el caso de la distancia largo los valores oscilan más sobre la línea teórica.

Se ejecuta el estadístico de prueba Shapiro-Wilks, primero para los datos de distancia corto y luego para los largos:

```
shapiro.test(time_corto)
```

```
shapiro.test(time_largo)
```

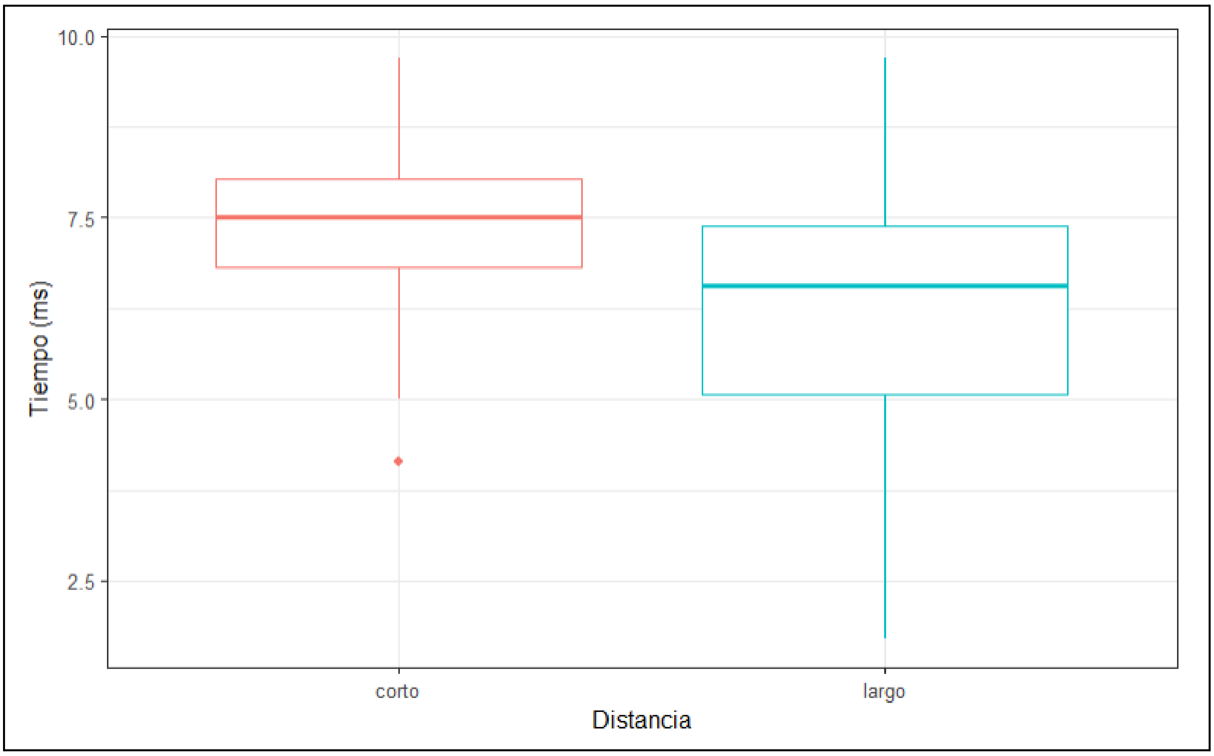
```
        Shapiro-Wilk normality test

data:  time_corto
W = 0.98748, p-value = 0.1427


        Shapiro-Wilk normality test

data:  time_largo
W = 0.98007, p-value = 0.2251
```

Los valores de ambas pruebas arrojan que 0.1427 > 0.2251 > 0.05, por lo que no hay evidencia para rechazar la hipótesis nula (los datos provienen de una distribución normal). Concluímos normalidad.

3.3. Igualdad de varianzas:

Podemos graficar boxplots con los datos para observar visualmente las distribuciones:

```
ggplot(redes,
       aes(x = distancia, y = tiempo, colour = distancia)) +
geom_boxplot() +
labs(x = "Distancia",
     y = "Tiempo (ms)") +
theme_bw() +
theme(legend.position = "none")
```



A primera vista las distribuciones en las muestras parecen no ser iguales, la diferencia en el tamaño de los rangos intercuartil es pronunciada, así como los máximos y mínimos. Vamos a comprobarlo con un estadístico.

Dado que se cumple el criterio de normalidad, se puede usar la prueba de Bartlett (en R bartlett.test):

```
bartlett.test(tiempo ~ as.factor(distancia), data = redes)
```

```
	Bartlett test of homogeneity of variances

data:  tiempo by as.factor(distancia)
Bartlett's K-squared = 22, df = 1, p-value = 2.727e-06
```

Obtenemos el resultado para el p value de 2.727e-06 < 0.05 por lo tanto hay evidencia para rechazar la hipótesis de que las varianzas son iguales. Concluímos que no hay igualdad de varianzas.

4. Determinar el tipo de test

Se trata de una prueba de dos colas, dado que la hipótesis nula es con igualdad. Además, concluimos que las varianzas no son iguales. También sabemos que las observaciones siguen una distribución normal.

Podemos usar una prueba T con la corrección de Welch

5. Determinar el nivel de significancia

Como se indicó al inicio del ejercicio, se usará $\alpha = 0.05$.

6. Cálculo de p-value

Realizamos entonces la prueba T con la corrección de Welch para los datos de las dos distancias, indicando mu = 0, dado que estamos usando la prueba para comparar ambas medias, o lo que es igual, que la diferencia de las medias es 0.

También indicamos que las varianzas no son iguales y que el nivel de confianza es 0.95, ó 1 – 0.005 (el nivel de significación):

```
t.test(
  x = time_corto,
  y = time_largo,
  alternative = "two.sided",
  mu = 0,
  var.equal = FALSE,
  conf.level = 0.95 )
```

```
        Welch Two Sample t-test

data:  time_corto and time_largo
t = 5.785, df = 117.31, p-value = 6.139e-08
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 0.7125139 1.4542836
sample estimates:
mean of x mean of y
 7.345808  6.262410
```

Esta prueba muestra un p-value de 6.139e-08, menor que el nivel de significancia de 0.05, por lo que sí hay bastante evidencia para rechazar la hipótesis nula.

7. Cálculo tamaño de efecto

Utilizamos la biblioteca effsize, con la función cohen.d:

```
library(effsize)
cohen.d(formula = tiempo ~ distancia, data = redes, paired = FALSE)
```

```
Cohen's d

d estimate: 0.8924623 (large)
95 percent confidence interval:
    lower     upper
0.6165164 1.1684081
```

Puede verse que el d estimado es de 0.8924623 (en magnitud), por lo que se considera que el efecto del tamaño de paquete en los tiempos de transmisión es alto.

8. Potencia de la prueba

8.1 Prueba PWR para Welch

Para calcular la potencia cuando no se cumplen las varianzas iguales, vamos a emplear la función pwr.t2n.test() del paquete pwr, que está pensada para dos muestras de distinto tamaño. Para ello mismo debemos obtener la varianza en común:

```
sd_corto = sd(time_corto)

sd_largo = sd(time_largo)

sd_comun = mean(time_corto) - mean(time_largo)

print (sd_comun)
```

```
[1] 1.083399
```

El valor arrojado es 1.083399

Debemos revisar si las muestras son de igual tamaño:

```
n1 = length(time_corto)

n2 = length(time_largo)

print (n1)

print (n2)
```

```
[1] 167
[1] 83
```

En este caso hay 167 observaciones para la distancia corto y 83 para la distancia larga.

Para usar esta prueba de potencia necesitamos calcular un efecto d no-pooled de la siguiente manera:

```
d = sd_comun/sqrt(((sd_corto^2) + (sd_largo^2))/2)

print (d)
```

```
[1] 0.8284778
```

El valor d observado es de 0.8284778

Ejecutamos entonces la prueba de potencia para varianzas diferentes de la siguiente manera:

```
library(pwr)

pwr.t2n.test(

  n1        = n1,

  n2        = n2,

  d         = d,

  sig.level = 0.05,

  power     = NULL,

  alternative = "two.sided")
```

```
     t test power calculation

              n1 = 167
              n2 = 83
               d = 0.8284778
       sig.level = 0.05
           power = 0.9999857
     alternative = two.sided
```

Esta prueba nos arroja una potencia de 0.9999857, lo cual es muy alto y nos dice que la cantidad de muestras es suficiente.

Se puede utilizar esta misma función para saber de cuánto es la muestra mínima para llegar a una potencia de 0.80. Para ello necesitamos calcular el delta, que es la diferencia entre las medias de las poblaciones. Como no tenemos información, usaremos la diferencia entre las medias muestrales.

```
delta_observado <- mean(time_corto) - mean(time_largo)

delta_observado
```

```
[1] 1.083399
```

El delta observado es 1.083399

Ahora modificamos la función de la siguiente manera

```
power.t.test(

  n           = NULL,             # deja que la función calcule n

  power       = 0.80,             # potencia deseada

  delta       = delta_observado,

  sd          = sd_comun,

  sig.level = 0.05,

  type        = "two.sample",     # prueba de dos muestras

  alternative = "two.sided"        # dos colas)
```

```

     Two-sample t test power calculation 

              n = 16.71477
          delta = 1.083399
             sd = 1.083399
      sig.level = 0.05
          power = 0.8
    alternative = two.sided

NOTE: n is number in *each* group
```

El resultado muestra que para obtener una potencia de prueba de 0.80 que esté relacionada con el delta encontrado (1.083399) se necesitan al menos 16 observaciones para cada tamaño de paquete.

8.1 Prueba de potencia para prueba T

Tambíen podemos ejecutar la prueba de potencia para la prueba T que hemos trabajo en el ejercicio anterior, calculamos la media de las varianzas:

```
sd_grupo1 <- sd(time_corto)

sd_grupo2 <- sd(time_largo)

sd_comun2 <- mean(c(sd_grupo1, sd_grupo2))

sd_comun2
```

```
[1] 1.278909
```

La media de las varianzas es de 1.278909

Seguidamente ejecutamos el comando power.t.test() indicando:

```
power.t.test(

  n = n2,

  delta = delta_observado,

  sd = sd_comun2,

  sig.level = 0.05,

  type = "two.sample")
```

```
     Two-sample t test power calculation 

              n = 83
          delta = 1.083399
             sd = 1.278909
      sig.level = 0.05
          power = 0.999735
    alternative = two.sided

NOTE: n is number in *each* group
```

De este resultado se puede ver que con este conjunto de datos se obtuvo una potencia de prueba de 0.999735, la cual es apropiada y similar a la de la prueba de potencia para varianzas diferentes.

Se puede utilizar esta misma función para saber de cuánto es la muestra mínima para llegar a una potencia de 0.80. Se modifica la invocación de la siguiente forma:

```
power.t.test(
  power = 0.8,
  delta = delta_observado,
  sd = sd_comun2,
  sig.level = 0.05,
  type = "two.sample")
```

```
     Two-sample t test power calculation

              n = 22.87586
          delta = 1.083399
             sd = 1.278909
      sig.level = 0.05
          power = 0.8
    alternative = two.sided

NOTE: n is number in *each* group
```

El resultado muestra que para obtener una potencia de prueba de 0.80 que esté relacionada con el delta encontrado (1.083399) se necesitan al menos 22 observaciones para cada tamaño de paquete.

9. Análisis de resultados y conclusiones

En el análisis de la influencia de la distancia (corto vs. largo) sobre el tiempo de transferencia, se observa una diferencia media de 1.0834 ms a favor de los paquetes en distancias cortas. La prueba t de Welch (para varianzas desiguales) arrojó un $t \approx -6.910$ con $df \approx 182$ y $p \approx 6.14 \times 10^{-8}$ (< 0.05), confirmando la significación estadística de la diferencia. El tamaño del efecto calculado por Cohen's $d = 0.8925$ se clasifica como "grande", indicando una diferencia práctica relevante. La potencia estimada con pwr.t2n.test() fue prácticamente 1.00, lo que garantiza alta sensibilidad del estudio; según power.t.test(), bastarían ≈ 17–23 observaciones por grupo para alcanzar potencia = 0.8, muy por debajo de los 167 y 83 muestras disponibles. En conjunto, estos resultados demuestran sólidamente que la distancia entre nodos sí incide en el tiempo de transferencia, con un efecto notable y perfectamente detectable con el tamaño de muestra actual.