Problema 1 - Teorema del Límite Central

Carlos Sierra Guzman, Camilo Vega Rámirez

Contenido

- Introducción . . . 2
  - Problema 1 . . . 2
- Punto A . . . 2
  - Metodología Punto A . . . 2
  - Resultado A . . . 2
- Punto B . . . 2
  - Metodología Punto B . . . 3
  - Resultado B . . . 3
- Punto C . . . 3
  - Metodología Punto C . . . 3
  - Resultado C . . . 3
- Punto D . . . 4
  - Metodología Punto D . . . 4
  - Resultado D . . . 4
- Punto E . . . 5
  - Metodología Punto E (10%) . . . 5
  - Resultado E (10%) . . . 6
  - Metodología Punto E (90%) . . . 8
  - Resultado E (90%) . . . 8
- Conclusión . . . 10
- Anexos . . . 10
  - Código Librerias . . . 10
  - Código Punto A . . . 10
  - Código Punto B . . . 11
  - Código Punto C . . . 11
  - Código Punto D . . . 12
  - Código Punto E (10%) . . . 13
  - Código Punto E (90%) . . . 14

Introducción

El presente documento es la respuesta al problema 1 de la Unidad 2 del curso Métodos y Simulación Estidística.

Cada seccion esta compuesta por el puto a resolver, metodologia y resultado.

Al final del documuento se encuentra como anexos los códigos usados para la creación de las metodologias de las secciones, y se cuentan con links en el cuerpo del documento para navegar a travez del mismo.

Problema 1

> *Teorema del Límite Central*
>
> *El Teorema del Límite Central es uno de los más importantes en la inferencia estadística y habla sobre la convergencia de los estimadores como la proporción muestral a la distribución normal. Algunos autores afirman que esta aproximación es bastante buena a partir del umbral n>30.*
>
> *A continuación se describen los siguientes pasos para su verificación:*

Punto A

> *a. Realice una simulación en la cual genere una población de N=1000 (Lote), donde el porcentaje de individuos (supongamos plantas) enfermas sea del 50%.*

Metodología Punto A

Se crea la función `sim_plantas_enfermas` para simular la una proporción de plantas enfermas dada una población.

Se genera la simulación de N = 1000 con 50% de plantas enfermas y se genera tabla para comprobar que las cantidades sean las correctas.

Ir a código sección a

Resultado A

plantas_enfermas_50	n
FALSE	500
TRUE	500

Punto B

> *b. Genere una función que permita: Obtener una muestra aleatoria de la población y Calcule el estimador de la proporción muestral $\hat{p}$* para un tamaño de muestra dado n.

Metodología Punto B

Se crea la función `sample_prop` para extraer `n` muestras de un vector `x` y calcular el estimador de la proporción muestral.

Se verifica el funcionamiento de la función para un `n = 500` sobre la población simulada.

Ir a código sección b

Resultado B

```
## [1] "Estimador de prueba = 0.482"
```

Punto C

> *c. Repita el escenario anterior (b) n=500 veces y analice los resultados en cuanto al comportamiento de los 500 resultados del estimador $\hat{p}$. ¿Qué tan simétricos o sesgados son los resultados obtenidos? y ¿qué se puede observar en cuanto a la variabilidad?. Realice en su informe un comentario sobre los resultados obtenidos.*

Metodología Punto C

Se crea la función `rep_sample_prop` que nos permite repetir la función `sample_prop` un numero `rep` de veces.

Se realiza la simulación de 500 veces el calculo del estimador $\hat{p}$ con una muestra de `n = 500` sobre la población simulada.

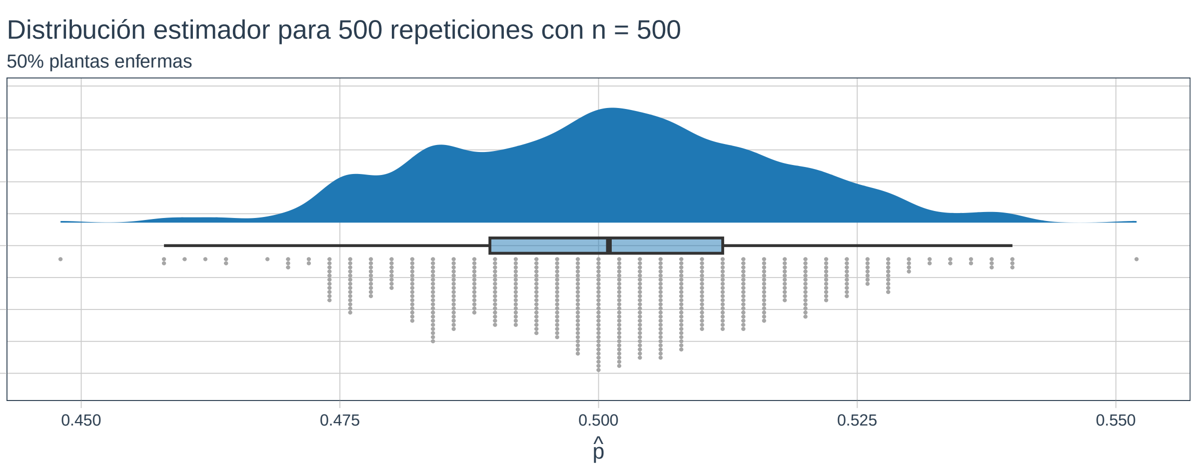
Se crea la función `gg_rain_cloud`, que toma un vector y genera un grafico de rain cloud.

Se crea la función `medidas_resumen`, que toma un vector y muestra en forma de tabla medidas de resumen respecto a simetria, sesgo y variabilidad.

Se usan `gg_rain_cloud` y `medidas_resumen` sobre las 500 simulaciones para su analisis.

Ir a código sección c

Resultado C



mean	median	sd	min	max	skewness	kurtosis
0.5007	0.501	0.0162	0.448	0.552	0.0028	-0.1586

Con un tamaño de muestra n = 500 y 500 repeticiones, se observa que los estimadores presentan indicadores de skewness y kurtosis, bajos, que sumados a la grafica nos muestran que los datos pueden considerarse simetricos, igualmente tanto la mediana como el promedio del estimador se aproximan al valor real de la proporcion de la población que es de 50% lo que nos indica que la distribución de los estimadores es in-sesgadas, por último la desviación estandar de los estimadores es del 1.60%, con un rango que oscila aproximadamente dentro del ± 5%.

Todo lo anterior nos muestra que con un tamaño de muestra de n = 500 la distribución de los estimadores se asemeja a una distribución normal y muestran una buena aproximación a la proporción real de la población.

Punto D

> *d. Repita los puntos b y c para tamaños de muestra n=5, 10, 15, 20, 30, 50, 60, 100, 200, 500. Compare los resultados obtenidos para los diferentes tamaños de muestra en cuanto a la normalidad. Utilice pruebas de bondad y ajuste (shapiro wilks :shspiro.test()) y métodos gráficos (grafico de normalidad: qqnorm()). Comente ensu informe los resultados obtenidos.*

Metodología Punto D

Se realiza la simulación de 500 veces el calculo del estimador $\hat{p}$ con multiles tamaños de muiestra `n` (5, 10, 15, 20, 30, 50, 60, 100, 200 y 500) sobre la población simulada, y se colocan en un data frame.

Se crea la función `medidas_resumen_multiple`, que toma un data frame y muestra en forma de tabla medidas de resumen y tests de normalidad de una columna seleccionada agupados por otra columna seleccionada.

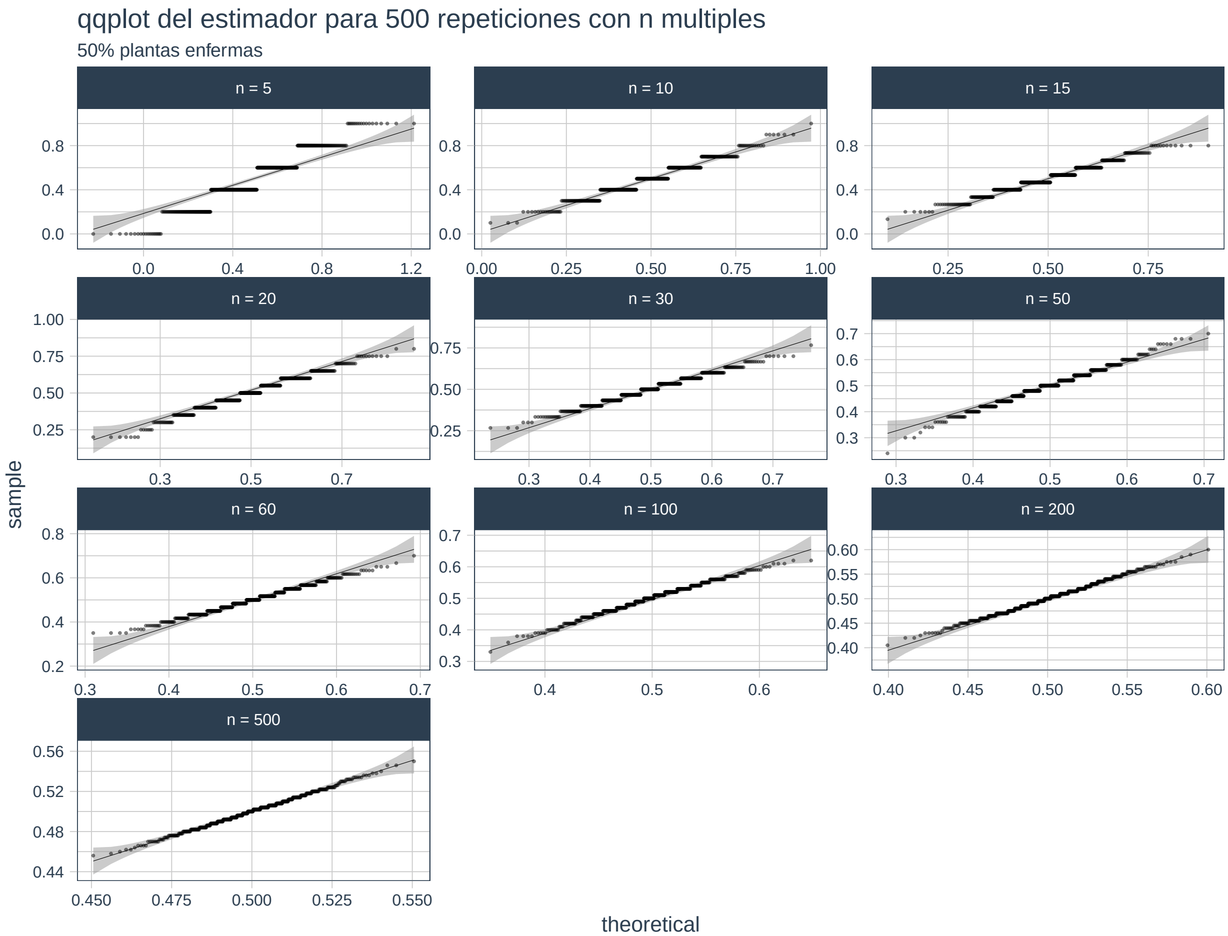
Se crea la función `gg_qq_plot`, que toma un data frame y realiza graficos de normalidad tipo `qqnor` de una columna seleccionada, agrupados por otra columna seleccionada.

Se usan `medidas_resumen_multiple` y `gg_qq_plot` sobre el data frame con las 500 simulaciones para distintos tamaños de nuestra `n` para su analisis.

Ir a código sección d

Resultado D

n	mean	median	sd	Shapiro-Wilk test P-Value
5	0.4936	0.4000	0.2329	0.0000
10	0.4992	0.5000	0.1534	0.0000
15	0.4995	0.4667	0.1298	0.0000
20	0.5057	0.5000	0.1144	0.0000
30	0.4997	0.5000	0.0852	0.0001
50	0.4972	0.5000	0.0675	0.0017
60	0.5011	0.5000	0.0620	0.0045
100	0.4987	0.5000	0.0485	0.0090
200	0.5002	0.5000	0.0326	0.2231
500	0.5006	0.5020	0.0162	0.4944



Podemos ver que a medida que aumentan los tamaños de muestas el promedio y mediana de los estimadores se aproximan cada vez más a la proporción real de la población, igualmente la desviación estardar disminuye. Así mismo se observa en las graficas de qqnorm que a mayor n, la distribución de los estimadores se parece más a una distrivución normal lo cual se comprueba con el test de Shapiro-Wilk el cual es positivo para normalidad a partir de n = 200.

Podemos entonces decir que estas simulaciones demuestran el teorema del limite central, ya que la distribución de la media de nuestras muestras aleatorias de plantas enfermas se aproxima a una distribución normal cuando el tamaño de la muestra es suficientemente grande.

Punto E

> *e. Repita toda la simulación (puntos a – d), pero ahora para lotes con 10% de plantas enfermas y de nuevo para lotes con un 90% de plantas enfermas. Concluya sobre los resultados del ejercicio.*

Metodología Punto E (10%)

Se genera la simulación de N = 1000 con 10% de plantas enfermas y se genera tabla para comprobar que las cantidades sean las correctas.

Se realiza la simulación de 500 veces el calculo del estimador $\hat{p}$ con una muestra de `n = 500` sobre la población simulada para 10% de plantas enfermas.

Se usan `gg_rain_cloud` y `medidas_resumen` sobre las 500 simulaciones de plantas enfermas al 10% para su analisis.

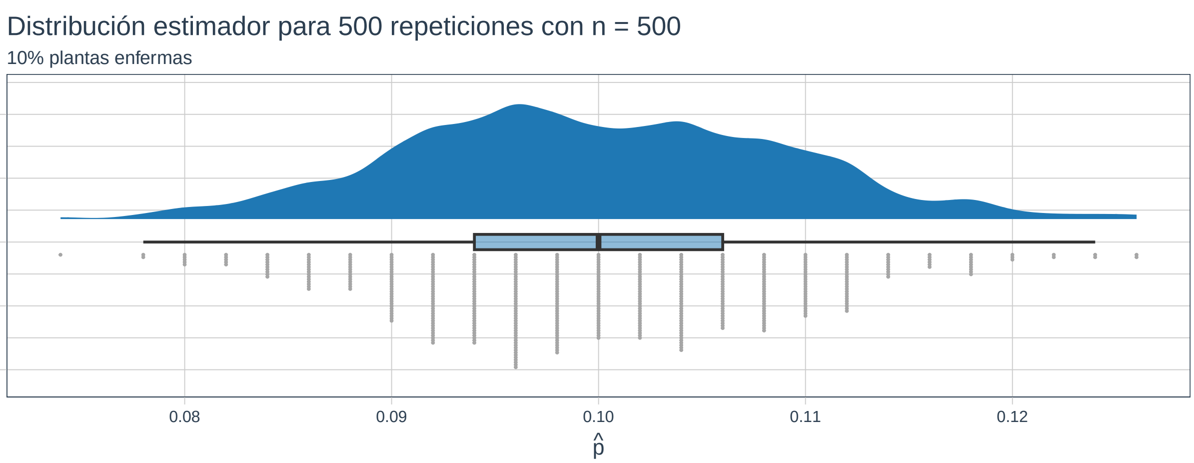
Se realiza la simulación de 500 veces el calculo del estimador $\hat{p}$ con multiles tamaños de muiestra `n` (5, 10, 15, 20, 30, 50, 60, 100, 200 y 500) sobre la población simulada con plantas enfermas al 10%, y se colocan en un data frame.

Se usan `medidas_resumen_multiple` y `gg_qq_plot` sobre el data frame con las 500 simulaciones para distintos tamaños de nuestra `n` sobre la población simulada con plantas enfermas al 10% para su analisis,

Ir a código sección e

Resultado E (10%)

plantas_enfermas_10	n
FALSE	900
TRUE	100



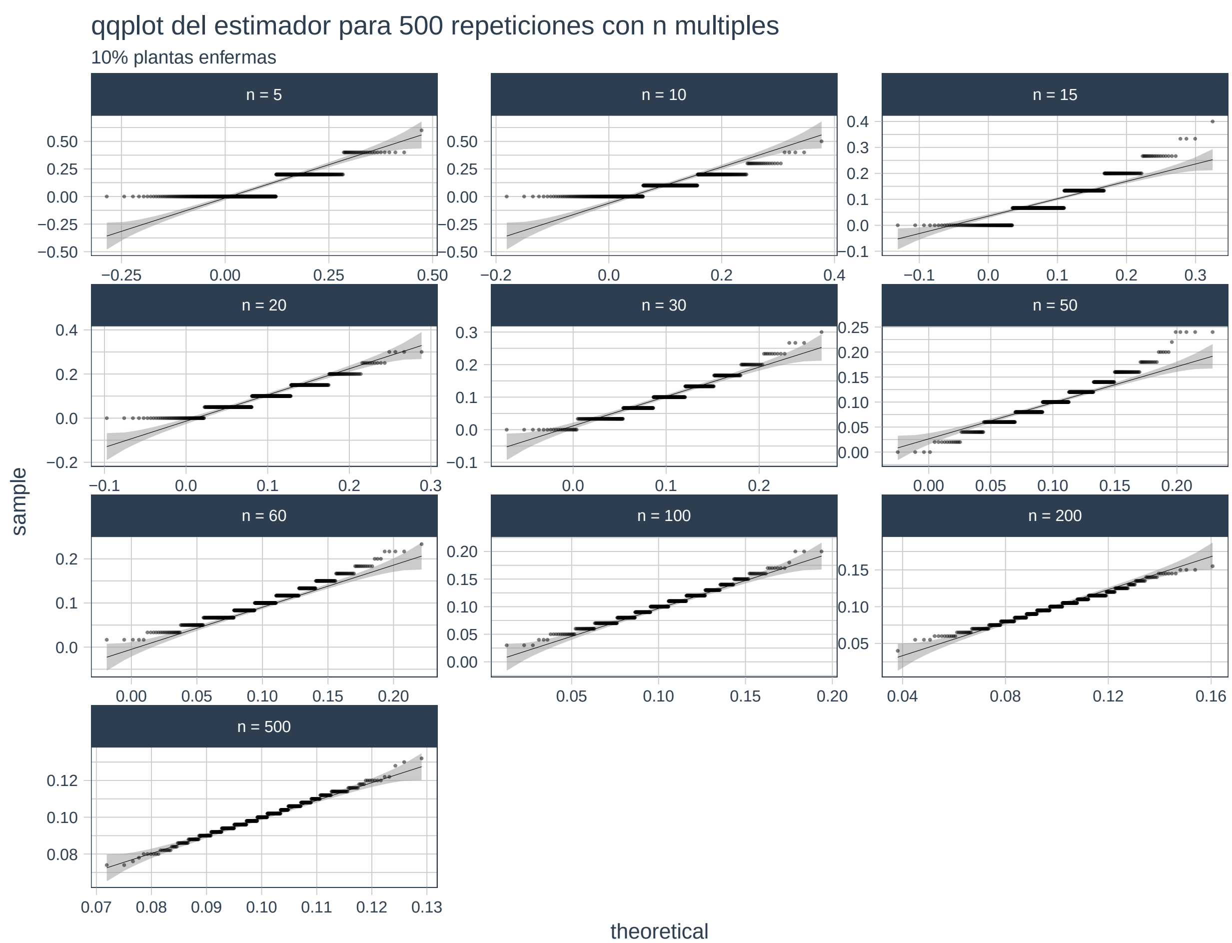
mean	median	sd	min	max	skewness	kurtosis
0.0999	0.1	0.0091	0.074	0.126	0.1205	-0.2449

Con una proporción de plantas enfermas del 10%, se observa que los estimadores presentan indicadores de skewness y kurtosis, bajos, que sumados a la grafica nos muestran que los datos pueden considerarse simetricos, igualmente tanto la mediana como el promedio del estimador se aproximan al valor real de la proporcion de la población que es de 10% lo que nos indica que la distribución de los estimadores es in-sesgadas, así mismo la desviación estandar de los estimadores es del 0.91%, con un rango que oscila aproximadamente dentro del $\pm$ 2%.

Todo lo anterior nos muestra que con un tamaño de muestra de n = 500 la distribución de los estimadores, de la simulación con plantas enfermas al 10% se asemeja a una distribución normal y muestran una buena aproximación a la proporción real de la población.

Comparado con la simulación de plantas enfermas al 50% se nota que la simulación con plantas enfermas al 10% presenta una distibución más aplanada, pero con una variabilidad menor.

n	mean	median	sd	Shapiro-Wilk test P-Value
5	0.0940	0.0000	0.1229	0.0000
10	0.0980	0.1000	0.0904	0.0000
15	0.0968	0.0667	0.0738	0.0000
20	0.0957	0.1000	0.0625	0.0000
30	0.0979	0.1000	0.0548	0.0000
50	0.1019	0.1000	0.0411	0.0000
60	0.1013	0.1000	0.0389	0.0000
100	0.1032	0.1000	0.0293	0.0000
200	0.0994	0.1000	0.0198	0.0027
500	0.1005	0.1000	0.0093	0.0466



Para la simulación con plantas enfermas al 10%, podemos ver que a medida que aumentan los tamaños de muestas el promedio y mediana de los estimadores se aproximan cada vez más a la proporción real de la población, igualmente la desviación estardar disminuye. Así mismo se observa en las graficas de qqnorm que para las muestras de tamaño n = a 50 a n = 60, las distribuaciones suelen tener mayor peso hacia los porcentajes menores a 10% por lo cual no se alcanza la normalidad, este fenomeno desaparece al llegar a n = 100 donde la distribución de los estimadores se parece más a una distribución normal, se ve que solo n = 500 logra pasar el test de Shapiro-Wilk para normalidad.

Podemos entonces decir que estas simulaciones demuestran el teorema del limite central, ya que la distribución de la media de las muestras aleatorias de plantas enfermas al 10% se aproxima a una distribución normal cuando el tamaño de la muestra es grande, con la anitación que para esta simulación en particular fue necesario un 50% de muestras de la población para alcanzar la normalidad mediante un test estadistico, graficamente podemos decir que se aproxima a partir de n = 100.

Metodología Punto E (90%)

Se genera la simulación de N = 1000 con 90% de plantas enfermas y se genera tabla para comprobar que las cantidades sean las correctas.

Se realiza la simulación de 500 veces el calculo del estimador $\hat{p}$ con una muestra de `n = 500` sobre la población simulada para 90% de plantas enfermas.

Se usan `gg_rain_cloud` y `medidas_resumen` sobre las 500 simulaciones de plantas enfermas al 90% para su analisis.

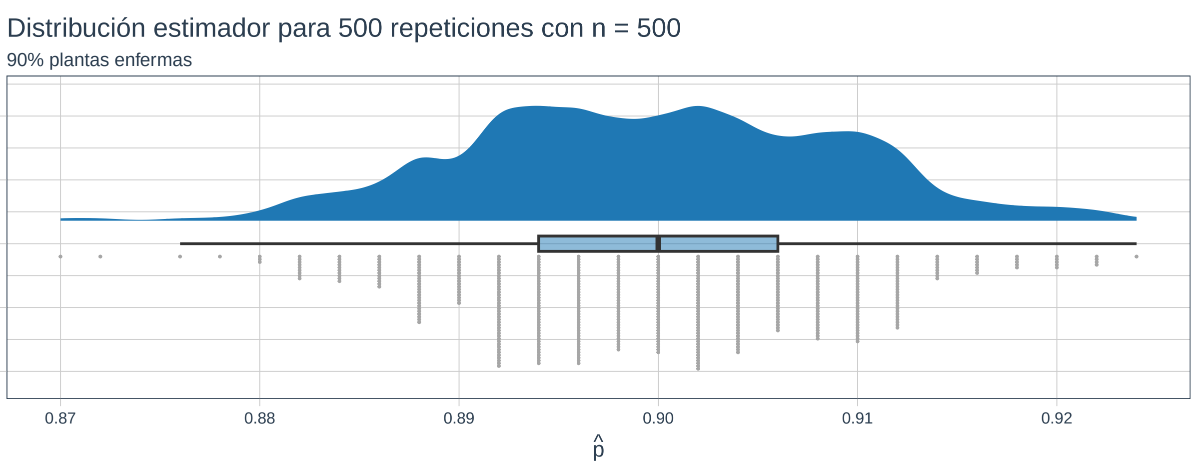
Se realiza la simulación de 500 veces el calculo del estimador $\hat{p}$ con multiles tamaños de muiestra `n` (5, 10, 15, 20, 30, 50, 60, 100, 200 y 500) sobre la población simulada con plantas enfermas al 90%, y se colocan en un data frame.

Se usan `medidas_resumen_multiple` y `gg_qq_plot` sobre el data frame con las 500 simulaciones para distintos tamaños de nuestra `n` sobre la población simulada con plantas enfermas al 90% para su analisis,

Ir a código sección e

Resultado E (90%)

plantas_enfermas_90	n
FALSE	100
TRUE	900

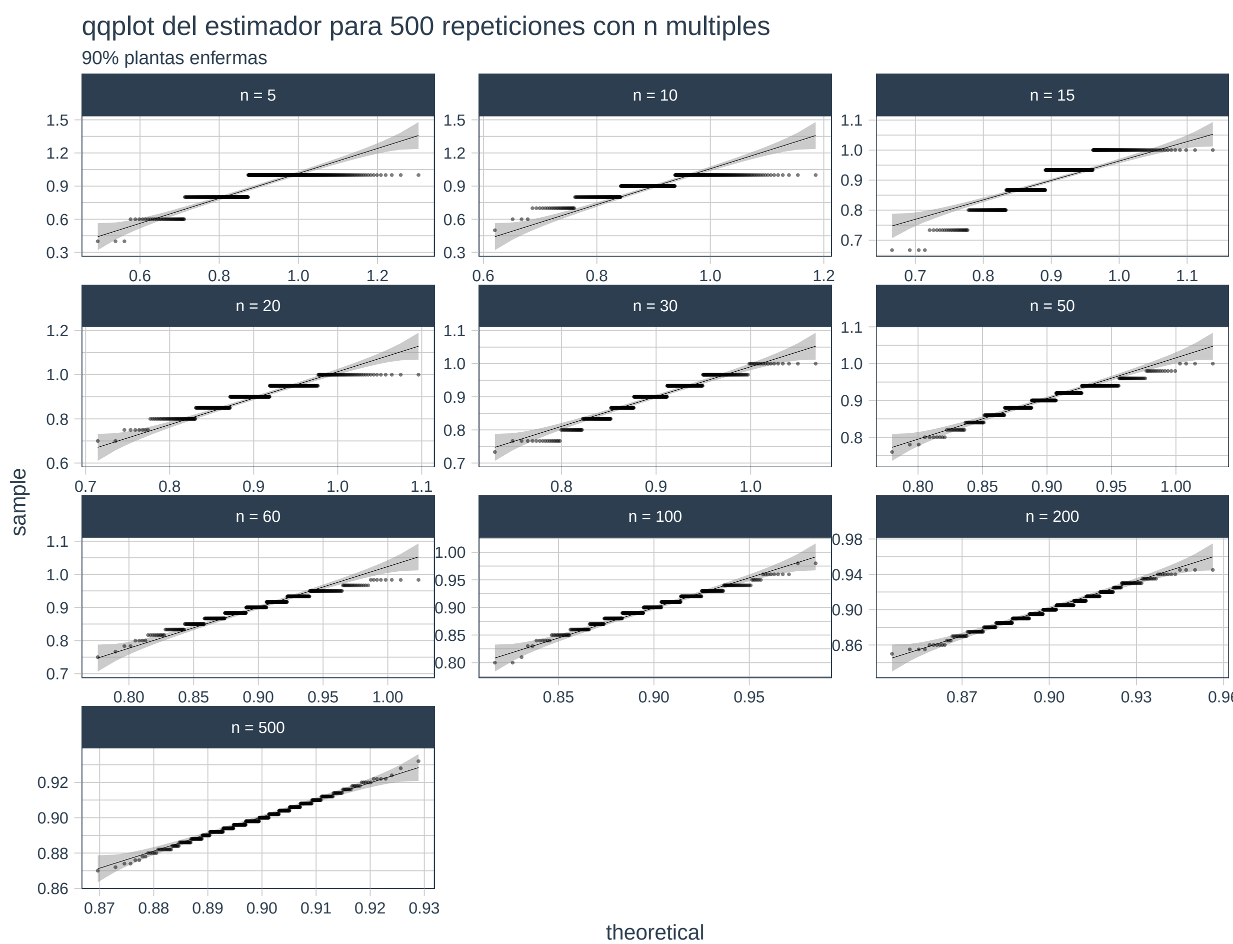


mean	median	sd	min	max	skewness	kurtosis
0.8998	0.9	0.0092	0.87	0.924	-0.0223	-0.273

Con una proporción de plantas enfermas del 90% se nota un comportamiento similar al de plantas enfermas al 10% con kewness y kurtosis bajos, una grafica algo aplanada, mediana y media aproximadas a la proporción real de la poblacion (90%), baja desviación 0.92% y rango de aproximadamente ± 2%, en general datos que muestran una distribución in-sesgadas y baja variabilidad.

Vemos que tambien con un tamaño de muestra de n = 500 la distribución de los estimadores, de la simulación con plantas enfermas, en esta ocación al 90%, se asemeja a una distribución normal y muestran una buena aproximación a la proporción real de la población.

n	mean	median	sd	Shapiro-Wilk test P-Value
5	0.8984	1.0000	0.1313	0.0000
10	0.9030	0.9000	0.0916	0.0000
15	0.9017	0.9333	0.0765	0.0000
20	0.9053	0.9000	0.0618	0.0000
30	0.8991	0.9000	0.0550	0.0000
50	0.9042	0.9000	0.0403	0.0000
60	0.8998	0.9000	0.0401	0.0000
100	0.9007	0.9000	0.0272	0.0000
200	0.9011	0.9000	0.0179	0.0015
500	0.8993	0.8980	0.0096	0.1224



Para la simulación con plantas enfermas al 90%, vemos igualmente que a medida que aumentan los tamaños de muestas el promedio y mediana de los estimadores se aproximan cada vez más a la proporción real de la población (para este caso 90%) e igualment su desviación estardar disminuye. Así mismo obsevamos en las graficas de qqnorm para las muestras de tamaño n = a 50 a n = 60, las distribuaciones suelen tener mayor peso hacia los porcentajes mayores a 90% no encajando dentro de la normalidad, al llegar a n = 100 las graficas se asemejan más a la normalidad pero con algunos puntos por fuera en los 2 extremos, solo al llegar a n = 200 se ve claramente la normalidad y solo en n = 500 se logra pasar el test de Shapiro-Wilk con un p-value de 0.12.

Podemos entonces decir que estas simulaciones demuestran una vez más el teorema del limite central, ya que la distribución de la media de las muestras aleatorias de plantas enfermas al 90% se aproxima a una distribución normal cuando el tamaño de la muestra es grande, sin embargo de los 3 porcentajes de plantas enfermas, se nota que que la simulación de plantas enfermas al 90% fue el que necesitó mayor tamño de muestra para acercarce a la normalidad.

Conclusión

Mediante los ejercicios realizados con simulaciones de proporciones de poblacones de plantas enfermas, se pudo demostrar el teorema de limite central el cual dice que; *"bajo ciertas condiciones, la distribución de la media de una muestra aleatoria de una población se aproxima a una distribución normal cuando el tamaño de la muestra es suficientemente grande"*, viendo como a medida que aumentabamos el tamaño de muestra efectivamente las estimaciones de nuestras simulaciones tomaban formas cada vez más normales.

Obserbamos tambien que para las proporciones simuladas el umbral de normalidad se encontro muy por arriba de n>30 llegando a ubicarse por arriba de n>60 cuando la proporcion era del 50% y por arriba de n>200 cuando la proporción se acerca la los extremos (10% y 90%), lo cual nos indica que, en el caso de las proporciones, se necesitan tamoños de muestras mayores para alcanzar niveles de normalidad.

Anexos

Código Librerias

```
library(tidyverse)      # Transformación de datos
library(normtest)       # Pruebas de normalidad
library(knitr)          # Renderizar tablas
library(ggdist)         # Expanción de graficas de ggplot
library(tidyquant)      # Tema de graficas de ggplot
library(nortest)        # Pruebas de normalidad
library(rapportools)    # Pruebas de normalidad
library(qqplotr)        # QQplot usando ggplot
```

Código Punto A

Volver a metodología sección a

```
# Función para generar población n, con una propoporción prop de plantas enfermas
sim_plantas_enfermas <- function(n, prop){
    p <- round(n*prop)
    q <- n-p
    c(rep(TRUE,p), rep(FALSE,q))
```

```
}

# Simulando para N = 1000 y 0.5 de plantas enfermas
plantas_enfermas_50 <- sim_plantas_enfermas(1000, 0.5)

# Tabla para visualizar simulación
table(plantas_enfermas_50) %>%
    as_tibble() %>%
    kable()
```

Código Punto B

Volver a metodología sección b

```
# Función para tomar una muestra de tamaño n de un vector x y calcular la
# proporcion del estimador.
sample_prop <- function(x, n){
    sample(x, n) %>%
        sum()/n
}

# Reproducibilidad
set.seed(4321)

# Test de función con n = 500, sobre el vector plantas_enfermas_50
str_c("Estimador de prueba = ",
sample_prop(plantas_enfermas_50,500))
```

Código Punto C

Volver a metodología sección c

```
# Función para repetir la función sample_prop un rep número de veces
rep_sample_prop <- function(x, n, rep){
    map_dbl(1:rep, ~ sample_prop(x,n))
}

# Reproducibilidad
set.seed(4321)

# Creación de 500 estimadores, para un n = 500 de muestras de plantas_enfermas_50
muesta_repetida_50 <-  rep_sample_prop(plantas_enfermas_50,500, 500)

# Función para crear grafico de rain cloud sobre un vector
gg_rain_cloud <- function(x, title, subtitle){
    ggplot(x %>% as_tibble() , aes( y = value)) +
    stat_halfeye(adjust = 0.5, justification = -0.2, .width = 0, point_colour = NA,
                 fill = "#1F78B4") +
    geom_boxplot(width = 0.12, outlier.color = NA, alpha = 0.5, fill = "#1F78B4") +
    stat_dots(side = "left", justification = 1.1, fill = "#1F78B4") +
    coord_flip() +
    theme_tq() +
```

```
    scale_fill_tq(theme = "light") +
    theme(axis.text.y = element_blank(),
          text = element_text(size = 8)) +
    xlab("") +
    ylab(expression(hat("p"))) +
    ggtitle(label = title,
            subtitle = subtitle)
}

# Función para crear tabla con medidas de resumen sobre un vector
medidas_resumen <-  function(x){
    x %>%
    as_tibble() %>%
    summarise(mean = mean(value), median = median(value), sd = sd(value),
              min = min(value), max = max(value),
              skewness = skewness(value), kurtosis= kurtosis(value))  %>%
    mutate(across(where(is.numeric), ~ round(.,4))) %>%
    kable()
}

# Creación rain cloud y tabla de resumen sobre las 500 rerpeticiones con muestra
# n = 500, sobre la población simulada
gg_rain_cloud(
    muesta_repetida_50,
    "Distribución estimador para 500 repeticiones con n = 500",
    "50% plantas enfermas")
medidas_resumen(muesta_repetida_50)
```

Código Punto D

Volver a metodología sección d

```
# Reproducibilidad
set.seed(4321)

# Creación de 500 estimadores, para multiples n de muestras de plantas_enfermas_50
muesta_repetida_multiple_50 <-  map_df(c(5, 10, 15, 20, 30, 50, 60, 100, 200, 500),
                                    ~ tibble(
                                        p_hat = rep_sample_prop(plantas_enfermas_50,
                                                                ., 500),
                                        n = as_factor(.)))

# Función para crear tabla con medidas de resumen sobre un data frame, para una
# columna agrupada por otra columna
medidas_resumen_multiple <- function(df, value, group){
  value <- enquo(value)
  group <- enquo(group)
  df %>%
    group_by(!!group) %>%
    summarise(mean = mean(!!value), median = median(!!value),
              sd = sd(!!value),
              `Shapiro-Wilk test P-Value` = shapiro.test(!!value)$p.value) %>%
    ungroup() %>%
```

```r
    mutate(across(where(is.numeric), ~ round(.,4))) %>%
    kable()
}

# Función para crear grafico de qqplot sobre un data frame, para una columna
# agrupada por otra columna
gg_qq_plot <- function(df, value, group, title, subtitle){
  ggplot(df,  aes(sample = {{value}})) +
    stat_qq_band(alpha = 0.5) +
    stat_qq_line(linewidth = 0.1) +
    stat_qq_point(alpha = 0.5, size = 0) +
    facet_wrap(vars(factor(str_c("n = ",{{group}}),
                           levels = c(str_c("n = ",c(5, 10, 15, 20, 30, 50, 60,
                                                     100, 200, 500))))),
               nrow =  4, scales = "free") +
    theme_tq() +
    theme(panel.spacing = unit(0, "lines"),
          text = element_text(size = 8)) +
    ggtitle(label = title,
            subtitle = subtitle)
}


# Creación de tabla de resumen y qqplots sobre las 500 rerpeticiones con muestra
# n multiples, sobre la población simulada
medidas_resumen_multiple(muesta_repetida_multiple_50, p_hat, n)
gg_qq_plot(muesta_repetida_multiple_50, p_hat, n,
           "qqplot del estimador para 500 repeticiones con n multiples",
           "50% plantas enfermas")
```

Código Punto E (10%)

Volver a metodología sección e

```r
# Simulando para N = 100 y 0.1 de plantas enfermas
plantas_enfermas_10 <- sim_plantas_enfermas(1000, 0.1)

# Tabla para visualizar simulación
table(plantas_enfermas_10) %>%
    as_tibble() %>%
    kable()

# Reproducibilidad
set.seed(1234)

# Creación de 500 estimadores, para un n = 500 de muestras de plantas_enfermas_10
muesta_repetida_10 <-  rep_sample_prop(plantas_enfermas_10,500, 500)

# Creación rain cloud y tabla de resumen sobre las 500 rerpeticiones con muestra
# n = 500, sobre la población simulada al 10% de plantas enfermas
gg_rain_cloud(
muesta_repetida_10,
"Distribución estimador para 500 repeticiones con n = 500",
```

```
"10% plantas enfermas")
medidas_resumen(muesta_repetida_10)
```

```
# Reproducibilidad
set.seed(1234)

# Creación de 500 estimadores, para multiples n de muestras de plantas_enfermas_10
muesta_repetida_multiple_10 <-  map_df(c(5, 10, 15, 20, 30, 50, 60, 100, 200, 500),
                                    ~ tibble(
                                        p_hat = rep_sample_prop(plantas_enfermas_10,
                                                                ., 500),
                                        n = as_factor(.)))

# Creación de tabla de resumen y qqplots sobre las 500 rerpeticiones con muestra
# n multiples, sobre la población simulada al 10% de plantas enfermas
medidas_resumen_multiple(muesta_repetida_multiple_10, p_hat, n)
gg_qq_plot(muesta_repetida_multiple_10, p_hat, n,
"qqplot del estimador para 500 repeticiones con n multiples",
"10% plantas enfermas")
```

Código Punto E (90%)

Volver a metodología sección e

```
# Simulando para N = 100 y 0.9 de plantas enfermas
plantas_enfermas_90 <- sim_plantas_enfermas(1000, 0.9)

# Tabla para visualizar simulación
table(plantas_enfermas_90) %>%
    as_tibble() %>%
    kable()

# Reproducibilidad
set.seed(1234)

# Creación de 500 estimadores, para un n = 500 de muestras de plantas_enfermas_90
muesta_repetida_90 <-  rep_sample_prop(plantas_enfermas_90,500, 500)

# Creación rain cloud y tabla de resumen sobre las 500 rerpeticiones con muestra
# n = 500, sobre la población simulada al 90% de plantas enfermas
gg_rain_cloud(
muesta_repetida_90,
"Distribución estimador para 500 repeticiones con n = 500",
"90% plantas enfermas")
medidas_resumen(muesta_repetida_90)
```

```
# Reproducibilidad
set.seed(1234)

# Creación de 500 estimadores, para multiples n de muestras de plantas_enfermas_90
muesta_repetida_multiple_90 <-  map_df(c(5, 10, 15, 20, 30, 50, 60, 100, 200, 500),
                                    ~ tibble(
```

```
                                        p_hat = rep_sample_prop(plantas_enfermas_90,
                                                                ., 500),
                                        n = as_factor(.)))

# Creación de tabla de resumen y qqplots sobre las 500 rerpeticiones con muestra
# n multiples, sobre la población simulada al 90% de plantas enfermas
medidas_resumen_multiple(muesta_repetida_multiple_90, p_hat, n)
gg_qq_plot(muesta_repetida_multiple_90, p_hat, n,
"qqplot del estimador para 500 repeticiones con n multiples",
"90% plantas enfermas")
```