Problema1 - Teorema del Límite Central

Carlos Sierra Guzman, Camilo Vega Rámirez

2023-02-27

```
library(tidyverse)
library(knitr)
library(ggdist)
library(tidyquant)
library(nortest)
library(qqplotr)
```

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:

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%.

```
sim_plantas_enfermas <- function(n, prop){
    p <- round(n*prop)
    q <- n-p
    c(rep(TRUE,p), rep(FALSE,q))
}

plantas_enfermas_50 <- sim_plantas_enfermas(1000, 0.5)

table(plantas_enfermas_50) %>%
    as_tibble() %>%
    kable()
```

plantas_enfermas_50	n
FALSE	500
TRUE	500

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.

```
sample_prop <- function(x, n){
    sample(x, n) %>%
        sum()/n
}

set.seed(4321)

sample_prop(plantas_enfermas_50,500)
```

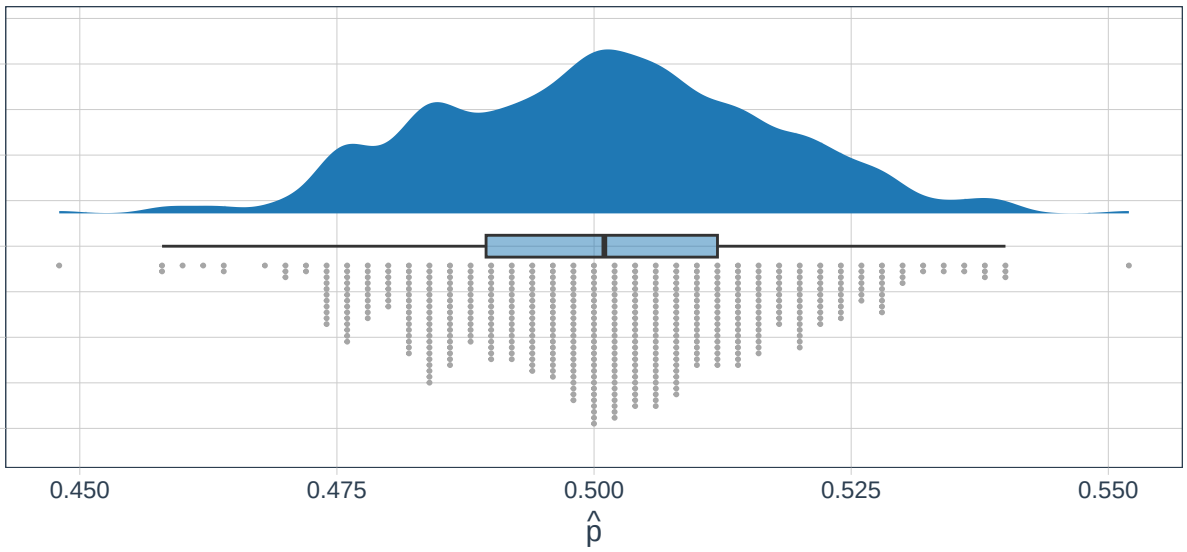
```
## [1] 0.482
```

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.

```
rep_sample_prop <- function(x, n, rep){
    map_dbl(1:rep, ~ sample_prop(x,n))
}

set.seed(4321)

muesta_repetida_50 <-  rep_sample_prop(plantas_enfermas_50,500, 500)

muesta_repetida_50 %>%
    as_tibble() %>%
    summarise(mean = mean(value), median = median(value), sd = sd(value)) %>%
    mutate(across(where(is.numeric), ~ round(.,4))) %>%
    kable()
```

mean	median	sd
0.5007	0.501	0.0162

```
ggplot(muesta_repetida_50 %>% 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()) +
    xlab("") +
    ylab(expression(hat("p")))
```



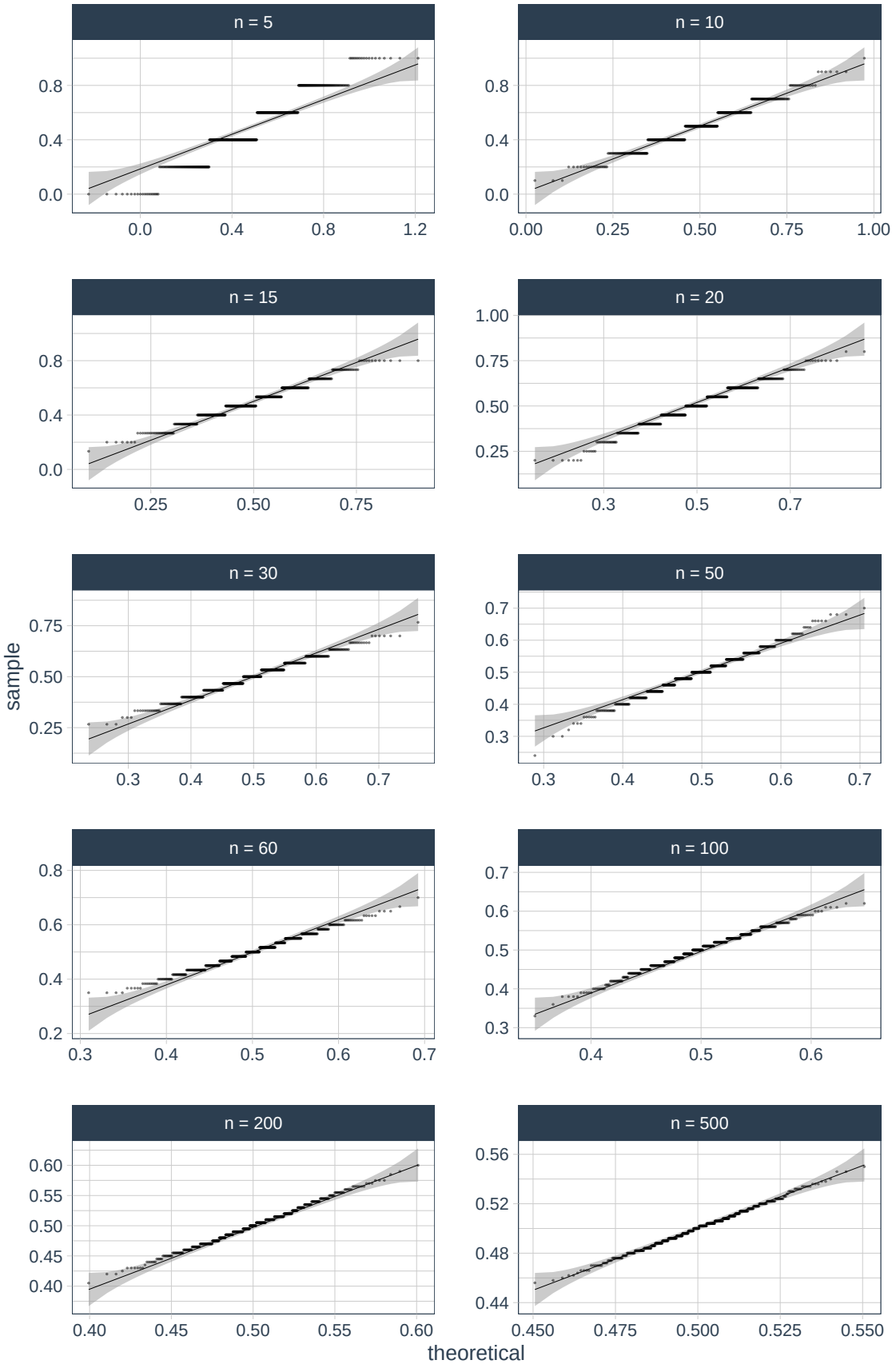
d. Repita los puntos b y c para tamaños de muestra n=5, 10, 15, 20, 30, 50, 60, 100, 200, 500. Compare los resultados obtenidos para los diferentes tamaños de muestra en cuanto a la normalidad. Utilice pruebas de bondad y ajuste (shapiro wilks :shspiro.test()) y métodos gráficos (grafico de normalidad: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

```
set.seed(4321)

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(.)))

muesta_repetida_multiple_50 %>%
    group_by(n) %>%
    summarise(mean = mean(p_hat), median = median(p_hat), sd = sd(p_hat),
              `Shapiro test P-Value` = shapiro.test(p_hat)$p.value,
              `Anderson-Darling test P-Value` = ad.test(p_hat)$p.value) %>%
    ungroup() %>%
    mutate(across(where(is.numeric), ~ round(.,4))) %>%
    kable()
```

n	mean	median	sd	Shapiro test P-Value	Anderson-Darling test P-Value
5	0.4936	0.4000	0.2329	0.0000	0.0000
10	0.4992	0.5000	0.1534	0.0000	0.0000
15	0.4995	0.4667	0.1298	0.0000	0.0000
20	0.5057	0.5000	0.1144	0.0000	0.0000
30	0.4997	0.5000	0.0852	0.0001	0.0000
50	0.4972	0.5000	0.0675	0.0017	0.0000
60	0.5011	0.5000	0.0620	0.0045	0.0002
100	0.4987	0.5000	0.0485	0.0090	0.0011
200	0.5002	0.5000	0.0326	0.2231	0.0434

n	mean	median	sd	Shapiro test P-Value	Anderson-Darling test P-Value
500	0.5006	0.5020	0.0162	0.4944	0.2147

```
ggplot(muesta_repetida_multiple_50,  aes(sample = p_hat)) +
    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 = ",n),
                            levels = c(str_c("n = ",c(5, 10, 15, 20, 30, 50, 60,
                                                      100, 200, 500))))),
               nrow =  5, scales = "free") +
    theme_tq()
```


n = 5
n = 10
n = 15
n = 20
n = 30
n = 50
n = 60
n = 100
n = 200
n = 500
sample
theoretical

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.

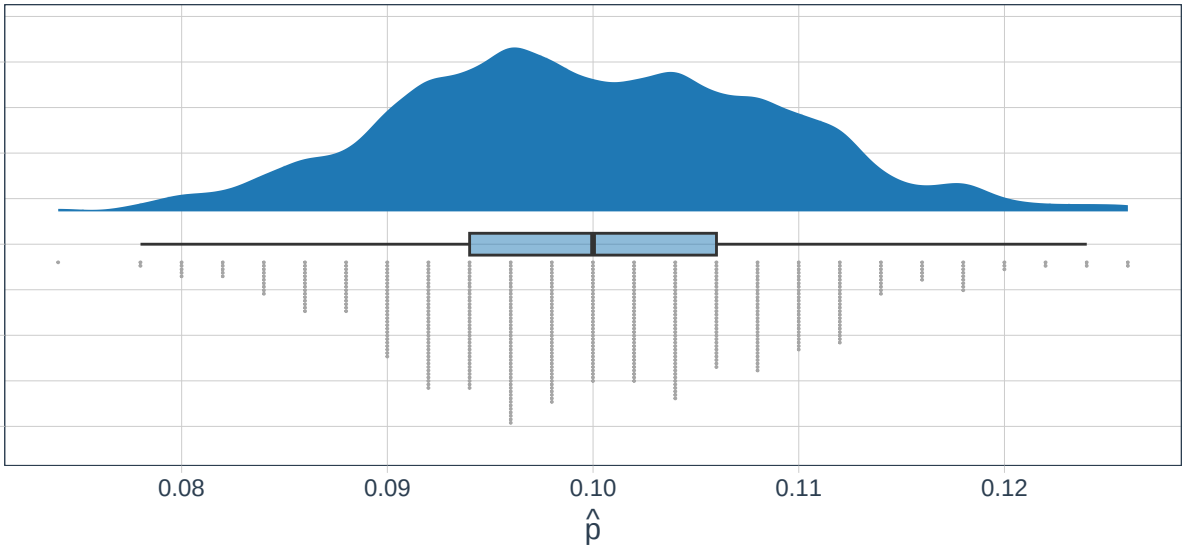
```
plantas_enfermas_10 <- sim_plantas_enfermas(1000, 0.1)

table(plantas_enfermas_10) %>%
    as_tibble() %>%
    kable()
```

plantas_enfermas_10	n
FALSE	900
TRUE	100

```
# set.seed(1234)
#
# sample_prop(plantas_enfermas_10,500)

set.seed(1234)

muesta_repetida_10 <-  rep_sample_prop(plantas_enfermas_10,500, 500)

muesta_repetida_10 %>%
    as_tibble() %>%
    summarise(mean = mean(value), median = median(value), sd = sd(value)) %>%
    mutate(across(where(is.numeric), ~ round(.,4))) %>%
    kable()
```

mean	median	sd
0.0999	0.1	0.0091

```
ggplot(muesta_repetida_10 %>% 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()) +
    xlab("") +
    ylab(expression(hat("p")))
```

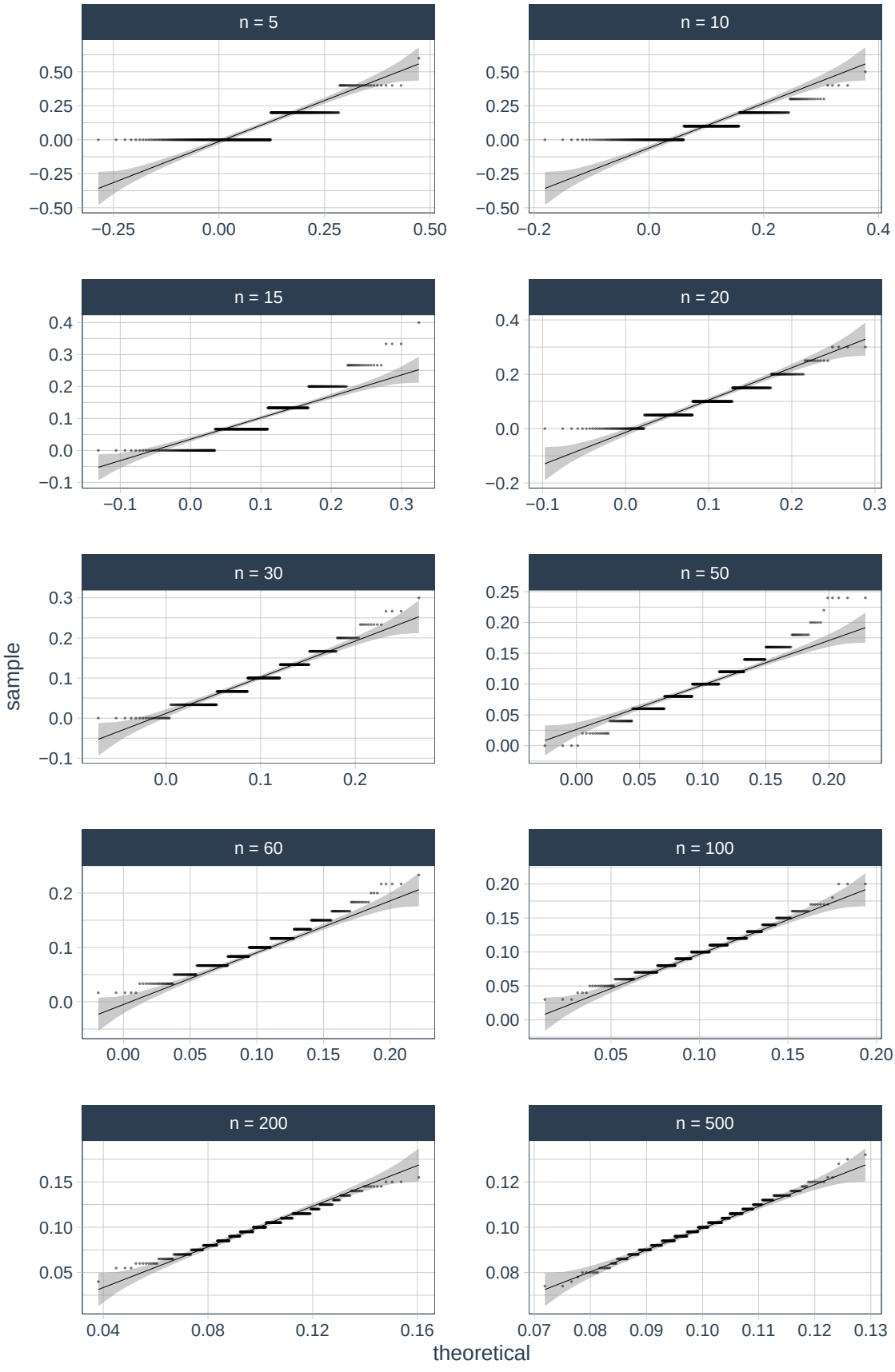



```
set.seed(1234)

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(.)))

muesta_repetida_multiple_10 %>%
    group_by(n) %>%
    summarise(mean = mean(p_hat), median = median(p_hat), sd = sd(p_hat),
              `Shapiro test P-Value` = shapiro.test(p_hat)$p.value,
              `Anderson-Darling test P-Value` = ad.test(p_hat)$p.value) %>%
    ungroup() %>%
    mutate(across(where(is.numeric), ~ round(.,4))) %>%
    kable()
```

n	mean	median	sd	Shapiro test P-Value	Anderson-Darling test P-Value
5	0.0940	0.0000	0.1229	0.0000	0.0000
10	0.0980	0.1000	0.0904	0.0000	0.0000
15	0.0968	0.0667	0.0738	0.0000	0.0000
20	0.0957	0.1000	0.0625	0.0000	0.0000
30	0.0979	0.1000	0.0548	0.0000	0.0000
50	0.1019	0.1000	0.0411	0.0000	0.0000
60	0.1013	0.1000	0.0389	0.0000	0.0000
100	0.1032	0.1000	0.0293	0.0000	0.0000
200	0.0994	0.1000	0.0198	0.0027	0.0003
500	0.1005	0.1000	0.0093	0.0466	0.0055

```
ggplot(muesta_repetida_multiple_10,  aes(sample = p_hat)) +
    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 = ",n),
                            levels = c(str_c("n = ",c(5, 10, 15, 20, 30, 50, 60,
                                                      100, 200, 500)))),
               nrow =  5, scales = "free") +
    theme_tq()
```


n = 5
n = 10
n = 15
n = 20
n = 30
n = 50
n = 60
n = 100
n = 200
n = 500
sample
theoretical

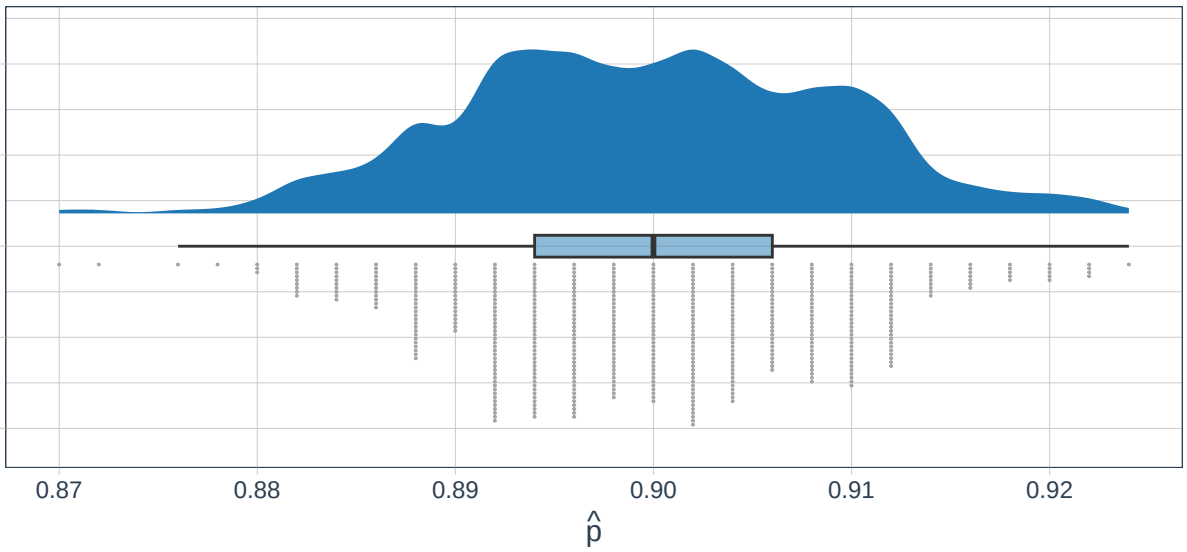
```r
plantas_enfermas_90 <- sim_plantas_enfermas(1000, 0.9)

table(plantas_enfermas_90) %>%
    as_tibble() %>%
    kable()
```

plantas_enfermas_90	n
FALSE	100
TRUE	900

```r
# set.seed(1234)
#
# sample_prop(plantas_enfermas_90,500)

set.seed(1234)

muesta_repetida_90 <-  rep_sample_prop(plantas_enfermas_90, 500, 500)

muesta_repetida_90 %>%
    as_tibble() %>%
    summarise(mean = mean(value), median = median(value), sd = sd(value)) %>%
    mutate(across(where(is.numeric), ~ round(.,4))) %>%
    kable()
```

mean	median	sd
0.8998	0.9	0.0092

```r
ggplot(muesta_repetida_90 %>% 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()) +
    xlab("") +
    ylab(expression(hat("p")))
```

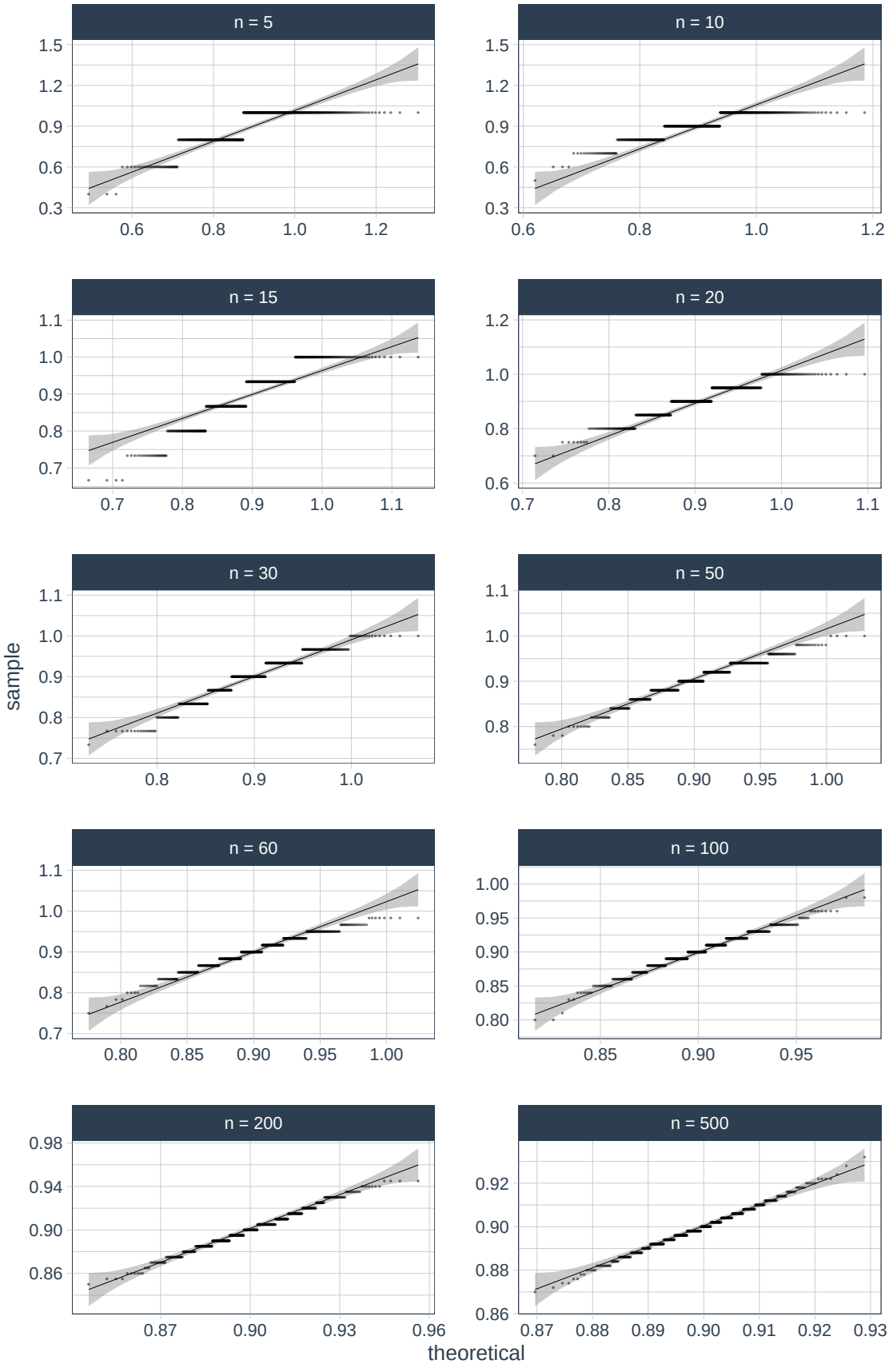



```
set.seed(1234)

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(.)))

muesta_repetida_multiple_90 %>%
    group_by(n) %>%
    summarise(mean = mean(p_hat), median = median(p_hat), sd = sd(p_hat),
              `Shapiro test P-Value` = shapiro.test(p_hat)$p.value,
              `Anderson-Darling test P-Value` = ad.test(p_hat)$p.value) %>%
    ungroup() %>%
    mutate(across(where(is.numeric), ~ round(.,4))) %>%
    kable()
```

n	mean	median	sd	Shapiro test P-Value	Anderson-Darling test P-Value
5	0.8984	1.0000	0.1313	0.0000	0.0000
10	0.9030	0.9000	0.0916	0.0000	0.0000
15	0.9017	0.9333	0.0765	0.0000	0.0000
20	0.9053	0.9000	0.0618	0.0000	0.0000
30	0.8991	0.9000	0.0550	0.0000	0.0000
50	0.9042	0.9000	0.0403	0.0000	0.0000
60	0.8998	0.9000	0.0401	0.0000	0.0000
100	0.9007	0.9000	0.0272	0.0000	0.0000
200	0.9011	0.9000	0.0179	0.0015	0.0001
500	0.8993	0.8980	0.0096	0.1224	0.0075

```
ggplot(muesta_repetida_multiple_90,  aes(sample = p_hat)) +
    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 = ",n),
                            levels = c(str_c("n = ",c(5, 10, 15, 20, 30, 50, 60,
                                                      100, 200, 500))))),
               nrow =  5, scales = "free") +
    theme_tq()
```


n = 5
n = 10
n = 15
n = 20
n = 30
n = 50
n = 60
n = 100
n = 200
n = 500
sample
theoretical