Problema2 - Validación de resultados

Carlos Sierra Guzman, Camilo Vega Rámirez

2023-02-27

```
options(scipen=999)
library(tidyverse)
library(knitr)
library(ggdist)
library(tidyquant)
library(nortest)
library(qqplotr)
```

Validación de resultados

La comparación de tratamientos es una práctica fundamental en las ciencias agropecuarias y para ello a nivel estadístico se cuenta con herramientas para apoyar el proceso de toma de decisiones y así poder lograr concluir con un alto grado de confianza sobre los resultados observados en una muestra. A través de la simulación es posible evaluar estimadores y sus propiedades, que nos permitan usarlos con toda tranquilidad.

Suponga un escenario en el cual se aplicó tratamientos diferentes a dos lotes de una misma plantas y se desea analizar si alguno de los dos tratamientos presenta un mejor desempeño en el control de una plaga presente en ambos al momento inicial. Para ello utilizará como criterio de desempeño el tratamiento, el menor porcentaje de plantas enfermas presente después de un tiempo de aplicación (es decir, si se presentan o no diferencias en las proporciones de enfermos p1 y p2 - proporciones poblacionales).

a. Realice una simulación en la cual genere dos poblaciones de $N_1 = 1000$ (Lote 1) y $N_2 = 1500$ (Lote 2), para los cuales se asume que el porcentaje de individuos (plantas) enfermas en ambos lotes es del 10% (es decir, sin diferencias entre los tratamientos).

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

plantas_enfermas_10_1 <- sim_plantas_enfermas(1000, 0.1)
plantas_enfermas_10_2 <- sim_plantas_enfermas(1500, 0.1)



grupo1 <- table(plantas_enfermas_10_1) %>%
```

```
    as_tibble() %>%
    mutate(grupo = "grupo_1") %>%
    rename(enferma = 1)

grupo2 <- table(plantas_enfermas_10_2) %>%
    as_tibble() %>%
    mutate(grupo = "grupo_2") %>%
    rename(enferma = 1)

bind_rows(grupo1, grupo2) %>%
    pivot_wider(names_from = enferma, values_from = n) %>%
    set_names(c("grupo", "no_enferma", "eneferma")) %>%
    kable()
```

grupo	no_enferma	eneferma
grupo_1	900	100
grupo_2	1350	150

b. Genere una función que permita obtener una muestra aleatoria de los lotes y calcule el estimador de la proporción muestral para cada lote ($\hat{p1}$ y $\hat{p2}$) para un tamaño de muestra dado $n_1 = n_2$. Calcule la diferencia entre los estimadores ($\hat{p1}$ - $\hat{p2}$).

```
sample_prop_multi <- function(x, y,n){
    x_sample <- sample(x, n) %>%
        sum()/n
    y_sample <- sample(y, n) %>%
        sum()/n
    x_sample-y_sample
}

set.seed(4321)

sample_prop_multi(plantas_enfermas_10_1, plantas_enfermas_10_2,500)
```

```
## [1] 0.016
```

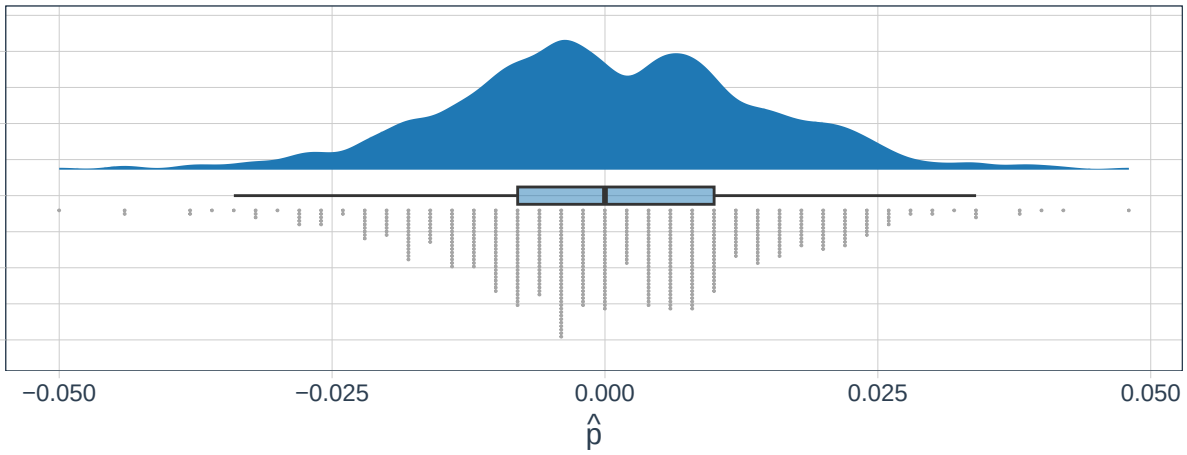
c. Repita el escenario anterior (b) 500 veces y analice los resultados en cuanto al comportamiento de los 500 estimadores (diferencias $\hat{p1}$ - $\hat{p2}$). ¿Qué tan simétricos son los resultados?, ¿Son siempre cero las diferencias?

```
rep_sample_prop_multi <- function(x, y, n, rep){
    map_dbl(1:rep, ~ sample_prop_multi(x, y, n))
}

set.seed(4321)

muesta_repetida_10_multi <-  rep_sample_prop_multi(plantas_enfermas_10_1,
                                                   plantas_enfermas_10_2,500, 500)
```

```
muesta_repetida_10_multi %>%
    as_tibble() %>%
    summarise(mean = mean(value), median = median(value), sd = sd(value),
              min = min(value), max = max(value),
              skewness = skewness(value))  %>%
    mutate(across(where(is.numeric), ~ round(.,4))) %>%
    kable()
```

mean	median	sd	min	max	skewness
0.0002	0	0.0144	-0.05	0.048	-0.0348

```
ggplot(muesta_repetida_10_multi %>% 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")))
```



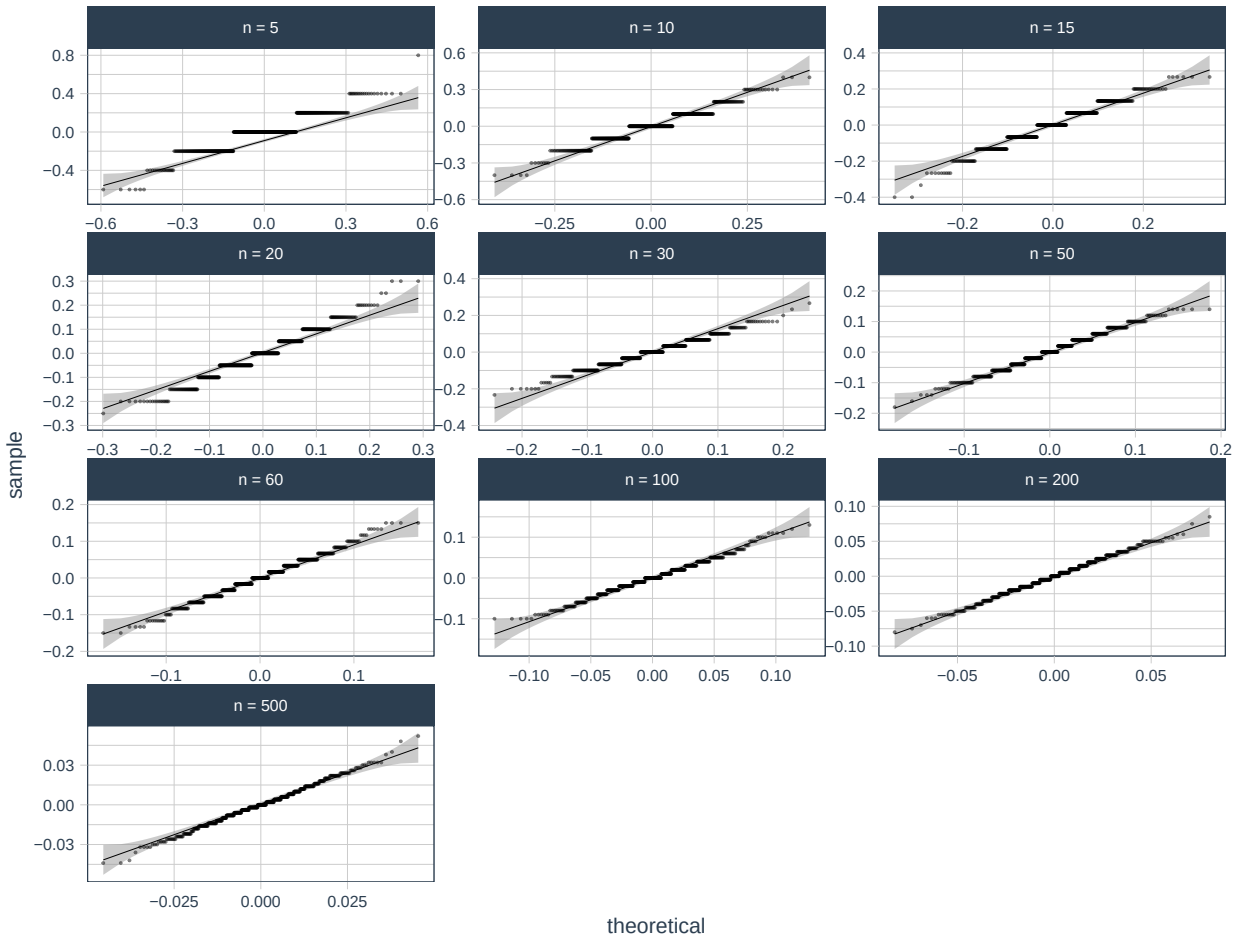
d. Realice los puntos b y c para tamaños de muestra $n_1 = n_2 =$ 5, 10, 15, 20, 30, 50, 60, 100, 200, 500. Compare los resultados de los estimadores ($\hat{p1}$ - $\hat{p2}$) en cuanto a la normalidad. También analice el comportamiento de las diferencias y evalúe. ¿Considera que es más probable concluir que existen diferencias entre los tratamientos con muestras grandes que pequeñas, es decir, cuál considera usted que es el efecto del tamaño de muestra en el caso de la comparación de proporciones?

```
set.seed(4321)

muesta_repetida_multiple_10 <-  map_df(c(5, 10, 15, 20, 30, 50, 60, 100, 200, 500),
```

```
                                    ~ tibble(
                                        p_hat = rep_sample_prop_multi(
                                            plantas_enfermas_10_1,
                                            plantas_enfermas_10_2,
                                            ., 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	-0.0128	0	0.1874	0.0000	0.0000
10	0.0024	0	0.1325	0.0000	0.0000
15	-0.0017	0	0.1129	0.0000	0.0000
20	-0.0040	0	0.0955	0.0000	0.0000
30	-0.0009	0	0.0781	0.0000	0.0000
50	0.0027	0	0.0596	0.0001	0.0000
60	0.0008	0	0.0544	0.0012	0.0000
100	-0.0004	0	0.0414	0.0024	0.0004
200	-0.0011	0	0.0263	0.1494	0.0199
500	0.0000	0	0.0147	0.1694	0.0405

```
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 =  4, scales = "free") +
    theme_tq() +
    theme(panel.spacing = unit(0, "lines"),
          text = element_text(size = 8))
```



e. Ahora realice nuevamente los puntos a-d bajo un escenario con dos lotes, pero de proporciones de enfermos diferentes ($\hat{p}1 = 0.1$ y $\hat{p}2 = 0.15$), es decir, el tratamiento del lote 1 si presentó un mejor desempeño reduciendo en un 5% el porcentaje de enfermos. Bajo este nuevo escenario compare la distribución de estas diferencias ($\hat{p}1$ - $\hat{p}2$) con las observadas bajo igualdad de condiciones en los lotes. ¿Qué puede concluir? ¿Existen puntos en los cuales es posible que se observen diferencias de $\hat{p}1$ - $\hat{p}2$ bajo ambos escenarios (escenario 1: sin diferencias entre $\hat{p}1$ y $\hat{p}2$, escenario 2: diferencia de 5%)?

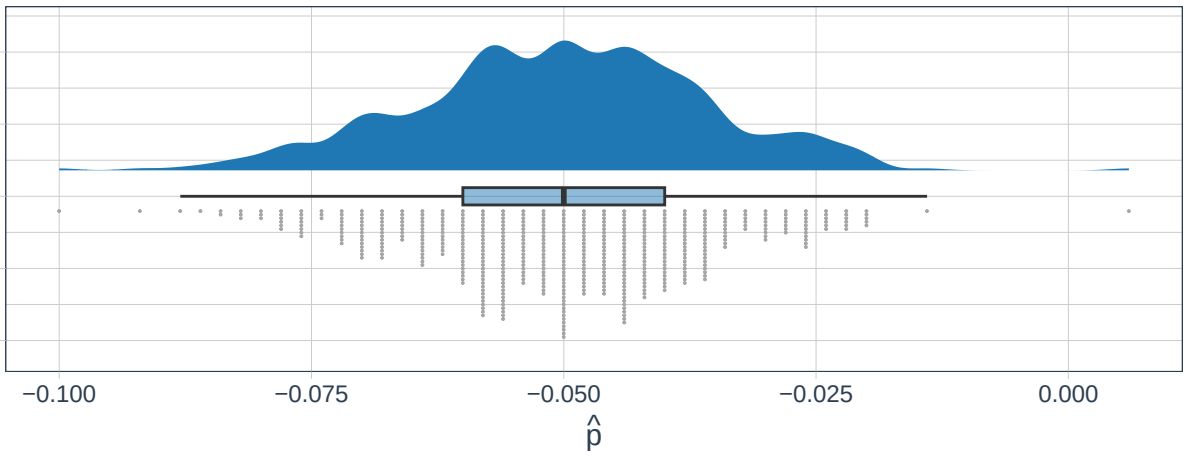
```
plantas_enfermas_15_1 <- sim_plantas_enfermas(1000, 0.15)

grupo3 <- table(plantas_enfermas_15_1) %>%
    as_tibble() %>%
    mutate(grupo = "grupo_3") %>%
    rename(enferma = 1)

bind_rows(grupo1, grupo3) %>%
    pivot_wider(names_from = enferma, values_from = n) %>%
    set_names(c("grupo", "no_enferma", "eneferma")) %>%
    kable()
```

grupo	no_enferma	eneferma
grupo_1	900	100
grupo_3	850	150

```
# set.seed(1234)
#
# sample_prop_multi(plantas_enfermas_10_1, plantas_enfermas_15_1,500)

set.seed(4321)

muesta_repetida_10_15_multi <-  rep_sample_prop_multi(plantas_enfermas_10_1,
                                                      plantas_enfermas_15_1,500, 500)

muesta_repetida_10_15_multi %>%
    as_tibble() %>%
    summarise(mean = mean(value), median = median(value), sd = sd(value),
              min = min(value), max = max(value),
              skewness = skewness(value))  %>%
    mutate(across(where(is.numeric), ~ round(.,4))) %>%
    kable()
```

mean	median	sd	min	max	skewness
-0.0503	-0.05	0.0144	-0.1	0.006	-0.0737

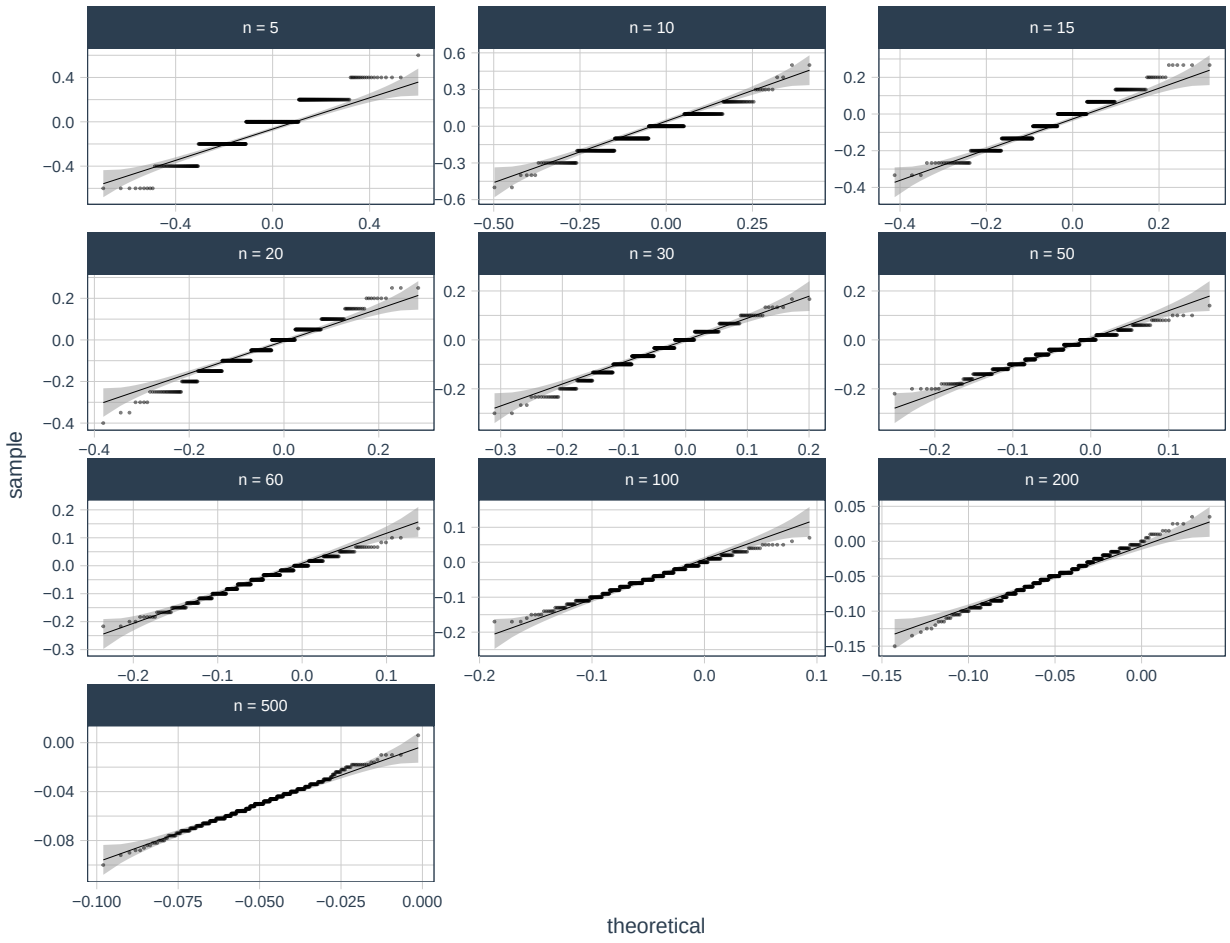
```
ggplot(muesta_repetida_10_15_multi %>% as_tibble() , aes( y = value)) +
    stat_halfeye(adjust = 0.5, justification = -0.2, .width = 0, point_colour = NA,
                 fill = "#1F78B4") +
    geom_boxplot(width = 0.12, outlier.color = NA, alpha = 0.5, fill = "#1F78B4") +
    stat_dots(side = "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(4321)

muesta_repetida_multiple_10_15 <-  map_df(c(5, 10, 15, 20, 30, 50, 60, 100, 200, 500),
                                    ~ tibble(
                                        p_hat = rep_sample_prop_multi(
                                            plantas_enfermas_10_1,
                                            plantas_enfermas_15_1,
                                            ., 500),
                                        n = as_factor(.)))

muesta_repetida_multiple_10_15 %>%
    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	-0.0488	0.0000	0.2104	0.0000	0.0000
10	-0.0412	0.0000	0.1479	0.0000	0.0000
15	-0.0476	-0.0667	0.1182	0.0000	0.0000
20	-0.0489	-0.0500	0.1076	0.0000	0.0000
30	-0.0546	-0.0667	0.0827	0.0000	0.0000
50	-0.0496	-0.0400	0.0654	0.0001	0.0000
60	-0.0490	-0.0500	0.0604	0.0030	0.0001
100	-0.0466	-0.0500	0.0454	0.0088	0.0021
200	-0.0517	-0.0500	0.0295	0.2128	0.0250
500	-0.0496	-0.0500	0.0157	0.2803	0.0843

```
ggplot(muesta_repetida_multiple_10_15,  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 =  4, scales = "free") +
    theme_tq() +
    theme(panel.spacing = unit(0, "lines"),
          text = element_text(size = 8))
```


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