

Tarea-04

AUTHOR

Thomas M. Rudolf

Cargamos los paquetes que necesitaremos:

```
library(tidyverse)
```

— Attaching core tidyverse packages — tidyverse 2.0.0 —

```
✓ dplyr      1.1.2    ✓ readr      2.1.4
✓ forcats    1.0.0    ✓ stringr    1.5.0
✓ ggplot2    3.4.2    ✓ tibble     3.2.1
✓ lubridate  1.9.2    ✓ tidyr      1.3.0
✓ purrr      1.0.2
```

— Conflicts — tidyverse_conflicts() —

```
* dplyr::filter() masks stats::filter()
```

```
* dplyr::lag()     masks stats::lag()
```

```
i Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to become errors
```

```
library(nullabor)
library(patchwork)
```

Pruebas visuales y Permutación

Pruebas de permutación

1. La tabla de datos *Phillies2009* contiene información de la temporada de baseball 2009 para el equipo de Philadelphia *Phillies*, en este ejercicio queremos comparar los strikes (*StrikeOuts*) entre los juegos en casa y de visitante:

```
# Lectura de datos
phillies <- read_csv("Phillies2009.csv")
```

Rows: 162 Columns: 7

— Column specification —

Delimiter: ","

chr (3): Date, Location, Outcome

dbl (4): Hits, Doubles, HomeRuns, StrikeOuts

```
i Use `spec()` to retrieve the full column specification for this data.
```

```
i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

```
glimpse(phillies)
```

Rows: 162

Columns: 7

```

$ Date      <chr> "5-Apr", "7-Apr", "8-Apr", "10-Apr", "11-Apr", "12-Apr", "1...
$ Location  <chr> "Home", "Home", "Home", "Away", "Away", "Away", "Away", "Aw...
$ Outcome   <chr> "Lose", "Lose", "Win", "Lose", "Win", "Win", "Win", "Lose",...
$ Hits      <dbl> 4, 6, 11, 7, 15, 13, 10, 5, 14, 8, 9, 13, 8, 2, 8, 9, 12, 1...
$ Doubles   <dbl> 2, 1, 3, 2, 3, 3, 3, 1, 3, 2, 1, 4, 0, 0, 2, 0, 2, 5, 0, 0,...
$ HomeRuns  <dbl> 0, 0, 1, 1, 1, 2, 3, 0, 1, 3, 3, 1, 1, 1, 2, 2, 0, 2, 4, 1,...
$ StrikeOuts <dbl> 6, 3, 6, 3, 6, 4, 7, 3, 5, 7, 5, 8, 4, 4, 12, 8, 7, 7, 5, 8...

```

- a. Realiza un *lineup* donde cada panel muestre un diagrama de caja y brazos para la variable de interés separando los juegos jugados en casa (home) y los juegos jugados fuera (away). ¿Puedes localizar los datos verdaderos entre los nulos?

```

# Line up data
set.seed(123)
perms_Location <- lineup(null_permute("Location"), phillies, n = 12)

```

```
decrypt("nsW7 Ykjk l3 gCPljLC3 4b")
```

```
glimpse(perms_Location)
```

Rows: 1,944

Columns: 8

```

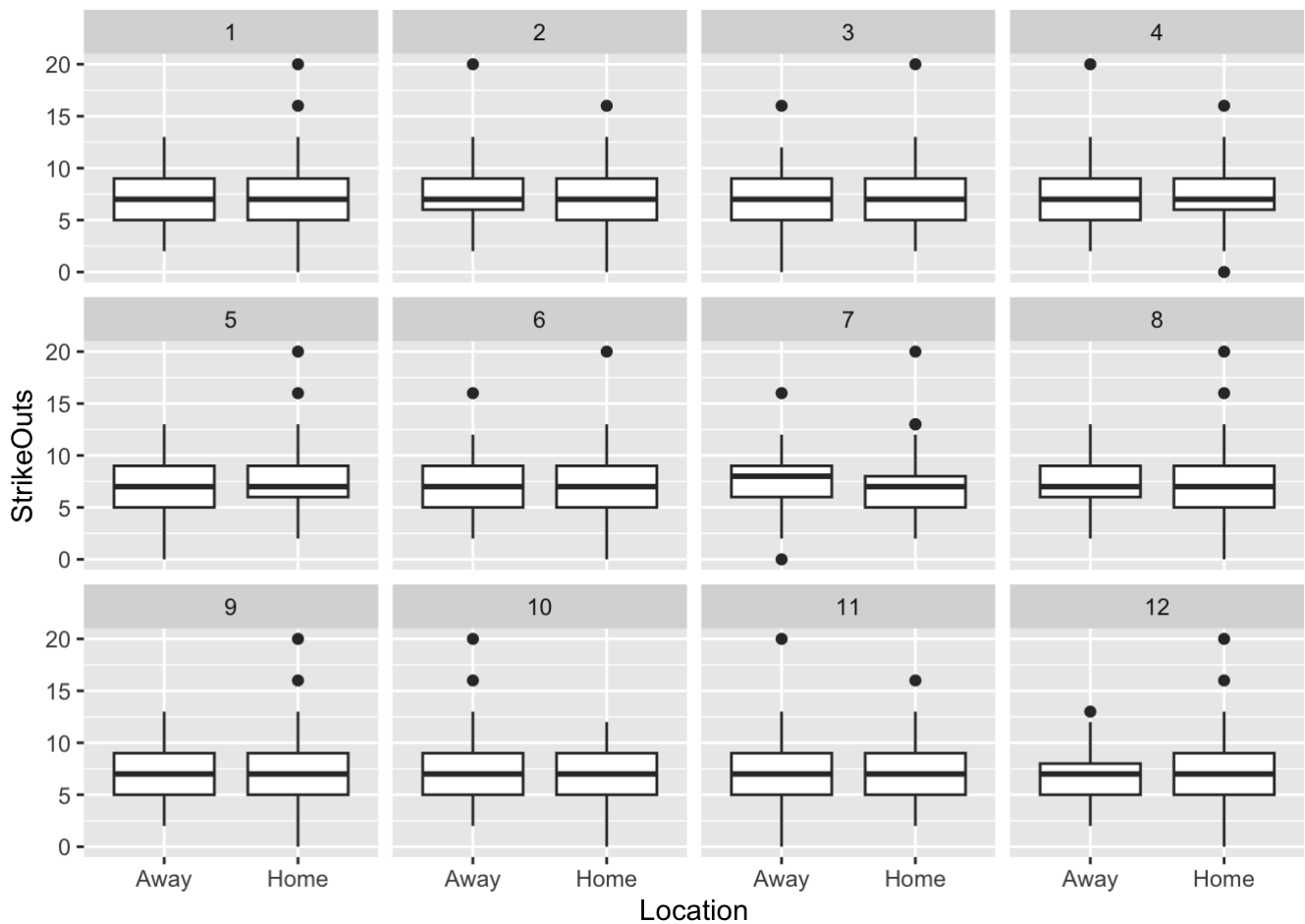
$ Date      <chr> "5-Apr", "7-Apr", "8-Apr", "10-Apr", "11-Apr", "12-Apr", "1...
$ Location  <chr> "Home", "Home", "Away", "Home", "Home", "Home", "Home", "Aw...
$ Outcome   <chr> "Lose", "Lose", "Win", "Lose", "Win", "Win", "Win", "Lose",...
$ Hits      <dbl> 4, 6, 11, 7, 15, 13, 10, 5, 14, 8, 9, 13, 8, 2, 8, 9, 12, 1...
$ Doubles   <dbl> 2, 1, 3, 2, 3, 3, 3, 1, 3, 2, 1, 4, 0, 0, 2, 0, 2, 5, 0, 0,...
$ HomeRuns  <dbl> 0, 0, 1, 1, 1, 2, 3, 0, 1, 3, 3, 1, 1, 1, 2, 2, 0, 2, 4, 1,...
$ StrikeOuts <dbl> 6, 3, 6, 3, 6, 4, 7, 3, 5, 7, 5, 8, 4, 4, 12, 8, 7, 7, 5, 8...
$ .sample   <dbl> 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,...

```

```

ggplot(perms_Location, aes(x = Location, y = StrikeOuts)) +
  geom_boxplot() +
  facet_wrap(~.sample)

```



```
#"nsW7 Ykjk l3 gCPljLC3 4b"
decrypt("nsW7 Ykjk l3 gCPljLC3 4b")
```

[1] "True data in position 7"

b. Calcula el promedio de strikes por juego en cada grupo (donde grupo se define por Location).

```
resumen_Strikes <- phillies |> group_by(Location) |>
  summarise(media_Strikes = mean(StrikeOuts)) |>
  pivot_wider(names_from = Location, values_from = media_Strikes) |>
  mutate(difference = Home - Away)

glimpse(resumen_Strikes)
```

Rows: 1

Columns: 3

```
$ Away      <dbl> 7.308642
$ Home      <dbl> 6.950617
$ difference <dbl> -0.3580247
```

c. Realiza una prueba de permutación para la diferencia de las medias. Grafica la distribución de referencia y calcula el valor p de dos colas.

```
set.seed(123)
Location_StrikeOut_tbl <- phillies |> select(Location, StrikeOuts)
# cinco mil permutaciones
perms_StrikeOuts <- lineup(null_permute("StrikeOuts"),
                           Location_StrikeOut_tbl, n = 5000)
```

```
decrypt("nsW7 Ykjk l3 gCPljLC3 4bFb")
```

```
#glimpse(perms_StrikeOuts)

perm_test <- perms_StrikeOuts |>
  group_by(.sample, Location) |>
  summarise(mean_values_per_sample = mean(StrikeOuts)) |>
  pivot_wider(names_from = Location, values_from = mean_values_per_sample) |>
  mutate(difference_per_sample = Home - Away)
```

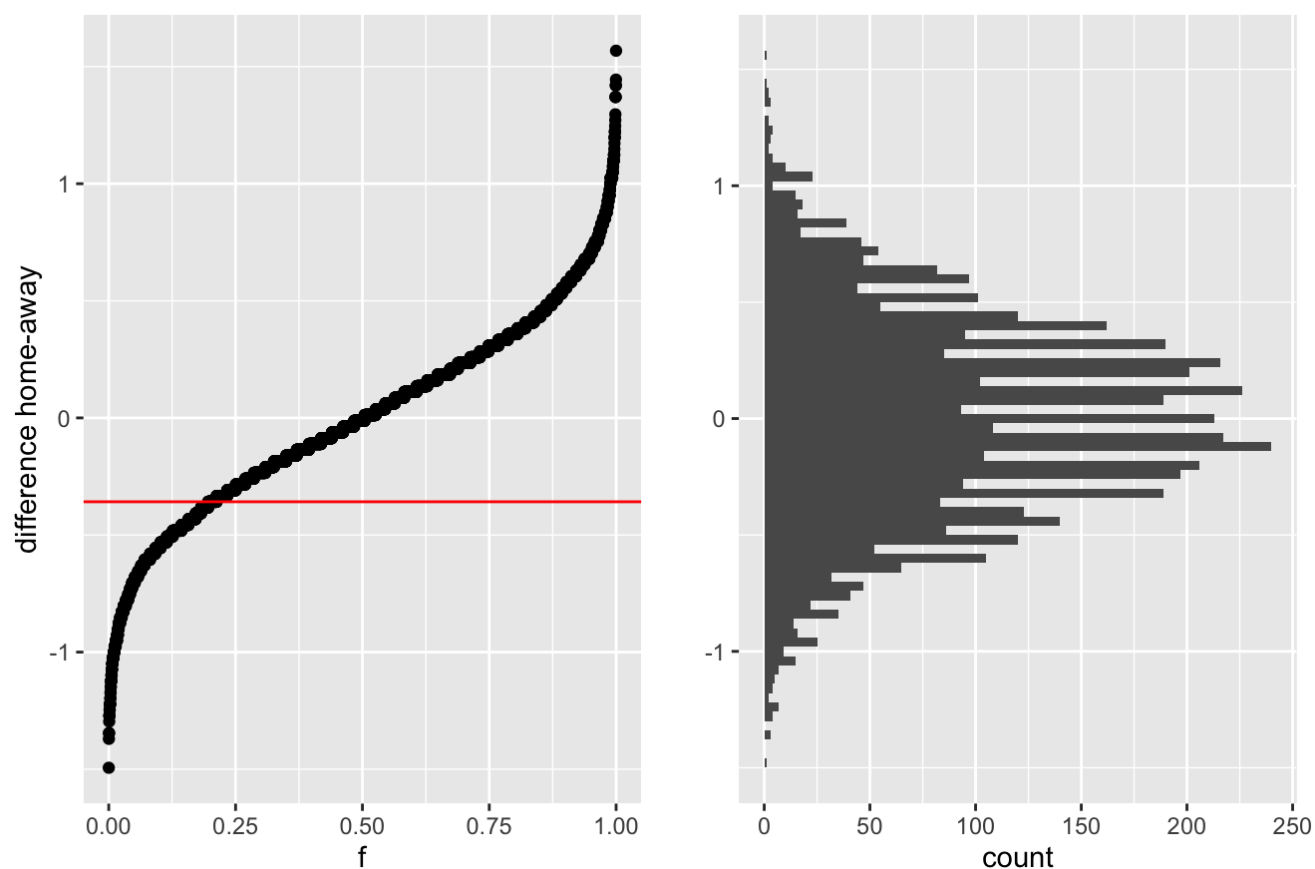
`summarise()` has grouped output by 'sample'. You can override using the
`.groups` argument.

```
g1 <- ggplot(perm_test, aes(sample=difference_per_sample)) +
  geom_qq(distribution=stats::qunif) +
  geom_hline(yintercept = resumen_Strikes$difference, color = "red") +
  xlab("f") + ylab("difference home-away") + labs(subtitle = "Distribucion nula o de r

g2 <- ggplot(perm_test, aes(x=difference_per_sample)) +
  geom_histogram(binwidth = 0.04) +
  coord_flip() +
  xlab("") + labs(subtitle = "")

g1 + g2
```

Distribucion nula o de referencia



```
dist_ref <- ecdf(resumen_Strikes$difference)
valor_p <- 2 * min(dist_ref(resumen_Strikes$difference), (1 - dist_ref(resumen_Strike
valor_p
```

```
[1] 0
```

Pruebas pareadas

En este ejemplo buscamos comparar la diferencia entre dos medicinas para dormir.

- ID es el identificador de paciente, group el identificador de la medicina, y extra son las horas extras de sueño.
- Examina los datos.

```
sleep
```

	extra	group	ID
1	0.7	1	1
2	-1.6	1	2

3	-0.2	1	3
4	-1.2	1	4
5	-0.1	1	5
6	3.4	1	6
7	3.7	1	7
8	0.8	1	8
9	0.0	1	9
10	2.0	1	10
11	1.9	2	1
12	0.8	2	2
13	1.1	2	3
14	0.1	2	4
15	-0.1	2	5
16	4.4	2	6
17	5.5	2	7
18	1.6	2	8
19	4.6	2	9
20	3.4	2	10

La pregunta de interés es si una medicina es mejor que otra para prolongar el sueño. Nótese que en este caso, no tenemos grupos, sino mediciones repetidas.

- Escribe la hipótesis nula. La hipótesis nula es que ambas medicinas tienen el mismo efecto con respecto a la prolongación de sueño. Prolongación medicina 1 (pm1) = Prolongación medicina 2 (pm2)
- Nuestra estadística de interés es media de las diferencias entre las medicinas. Calcula la diferencia observada.

```
sleep <- sleep |>
  group_by(group) |>
  pivot_wider(names_from = group, values_from = extra) |>
  rename(pm1 = "1", pm2 = "2") |>
  mutate(difference_sleep = pm1 - pm2)

glimpse(sleep)
```

Rows: 10

Columns: 4

```
$ ID          <fct> 1, 2, 3, 4, 5, 6, 7, 8, 9, 10
$ pm1         <dbl> 0.7, -1.6, -0.2, -1.2, -0.1, 3.4, 3.7, 0.8, 0.0, 2.0
$ pm2         <dbl> 1.9, 0.8, 1.1, 0.1, -0.1, 4.4, 5.5, 1.6, 4.6, 3.4
$ difference_sleep <dbl> -1.2, -2.4, -1.3, -1.3, 0.0, -1.0, -1.8, -0.8, -4.6, ...
```

- Hay variación entre los pacientes. ¿Tenemos evidencia para rechazar que son iguales? ¿Cómo hacemos nuestra distribución de referencia?

```
perm_sleep <- sleep |>
  group_by(.sample, ID) |>
  summarise()
```

```
Error in `group_by()` :
! Must group by variables found in `.data`.
* Column `.sample` is not found.
```

```
pivot_wider(names_from = group, values_from = extra) |>
mutate(difference_sleep = pm1 - pm2)
```

```
Error in eval(expr, envir, enclos): object 'group' not found
```

```
diff_patient
```

```
Error in eval(expr, envir, enclos): object 'diff_patient' not found
```

Parece que si hay una diferencia entre el tiempo de horas extras por paciente. Solo con el paciente Id 5 se queda igual, los demas tiene entre 0.8 horas y 2.4 horas mas.

- Haz una gráfica de la distribución de referencia y grafica encima el valor observado en los datos originales.

```
set.seed(2)
reps_group <- lineup(null_permute("1"), sleep_pivot, n = 200)
```

```
Error in eval(expr, envir, enclos): object 'sleep_pivot' not found
```

```
valores_ref <- reps_group |>
  group_by(.sample) |>
  mutate(difference = X1-X2)
```

```
Error in eval(expr, envir, enclos): object 'reps_group' not found
```

```
g_3 <- ggplot(valores_ref, aes(sample=difference)) +
  geom_qq(distribution=stats::qunif) +
  xlab("f") + ylab("difference sleeping time") + labs(subtitle = "Distribucion nula o
```

```
Error in eval(expr, envir, enclos): object 'valores_ref' not found
```

```
g_3
```

```
Error in eval(expr, envir, enclos): object 'g_3' not found
```

```
g_4 <- ggplot(valores_ref, aes(x = difference)) +
  geom_histogram() +
  coord_flip()
```

```
Error in eval(expr, envir, enclos): object 'valores_ref' not found
```

g_4

Error in eval(expr, envir, enclos): object 'g_4' not found

- Calcula el valor p (justifica porque a una o dos colas).
 - Se puede justificar el uso del valor p de dos colas, porque nos interesa si hay una diferencia o no. Al final, la medicina "2" puede ser no "peor" que la medicina "1" o visa versa. Como se ve en los datos, la medicina "2" parece de dar mejores resultados y como es son los datos que permuta, la hipotesis alterna puede ser que $\mu_1 < \mu_2$. Por lo tanto, por estos datos uno una cola.

```
# Función de distribución acumulada (inverso de función de cuantiles)
dist_perm <- ecdf(valores_ref$difference)
```

Error in eval(expr, envir, enclos): object 'valores_ref' not found

```
# Calculamos el percentil del valor observado
percentil_obs <- dist_perm(is.na(valores_ref))
```

Error in dist_perm(is.na(valores_ref)): could not find function "dist_perm"

```
valor_p_medicina <- 1-mean(percentil_obs)
```

Error in eval(expr, envir, enclos): object 'percentil_obs' not found

Valores-p (opcional)

Cuando usamos simulación para pruebas de permutación, el valor-p de una cola se puede calcular como

$$\hat{P} = (X + 1)/(N + 1)$$

donde X es el número de estadísticas de prueba remuestreadas que son al menos tan extremas como la observada. Supongamos que el verdadero valor p (si pudiéramos hacer todas las permutaciones de manera exhaustiva) es p

- ¿Cuál es la varianza de \hat{P} ?

- ¿Cuál es la varianza de \hat{P}_2 para la prueba de dos lados? (suponiendo que p no es cercano a 0.5, donde p es el verdadero valor p para la prueba de una cola).

Pruebas de hipótesis (opcional)

Ve el video [Is Most Published Research Wrong?](#)

