

bootstrap_uriel.R

ASUS Vivobook

2025-09-16

```
#
=====
===
# BOOTSTRAP CON ARCHIVO "puno.csv" - VERSION FUNCIONAL
#
=====
===

# Paso 1: Cargar Librerías
library(readr)

## Warning: package 'readr' was built under R version 4.4.3

library(dplyr)

## Warning: package 'dplyr' was built under R version 4.4.3

##
## Adjuntando el paquete: 'dplyr'

## The following objects are masked from 'package:stats':
##
##   filter, lag

## The following objects are masked from 'package:base':
##
##   intersect, setdiff, setequal, union

library(boot)
library(ggplot2)

## Warning: package 'ggplot2' was built under R version 4.4.3

# Paso 2: Leer CSV correctamente (manejo de comillas)
ruta <- "C:/Users/ASUS Vivobook/Desktop/DECIMO SEMESTRE/ESTADISTICA
ESPACIAL/puno.csv"

# Leemos como delimitado por comas, con comillas
datos_raw <- read_delim(ruta, delim = ",", quote = "\"", col_names =
TRUE, show_col_types = FALSE)

#
=====
===
# LIMPIAR DATOS: separar columnas si están pegadas en una sola
```

```

#
=====
===
# Si al imprimir colnames(datos_raw) ves todo pegado, hacemos:
datos <- datos_raw %>%
  mutate(
    # Eliminamos posibles comillas
    fila = gsub('\\"', '', datos_raw[[1]])
  ) %>%
  tidyr::separate(fila, into =
c("NOMBREDI", "NOMBREPV", "superficie_total_ha",
"productores_total", "parcelas_total", "superficie_promedio"),
    sep = ",", convert = TRUE)

# Ahora las columnas numéricas son tipo numérico
datos <- datos %>%
  mutate(
    superficie_total_ha = as.numeric(superficie_total_ha),
    productores_total   = as.numeric(productores_total),
    parcelas_total      = as.numeric(parcelas_total),
    superficie_promedio = as.numeric(superficie_promedio)
  )

#
=====
===
# ESTADÍSTICAS ORIGINALES
#
=====
===
cat("\n--- Estadísticas originales ---\n")

##
## --- Estadísticas originales ---

resumen <- datos %>%
  summarise(
    media_superficie_total   = mean(superficie_total_ha, na.rm = TRUE),
    media_productores_total  = mean(productores_total, na.rm = TRUE),
    media_parcelas_total     = mean(parcelas_total, na.rm = TRUE),
    media_superficie_prom    = mean(superficie_promedio, na.rm = TRUE)
  )
print(resumen)

## # A tibble: 1 × 4
##   media_superficie_total media_productores_total media_parcelas_total
##               <dbl>               <dbl>               <dbl>
## 1                27.7                1880.                27.7
## # [i] 1 more variable: media_superficie_prom <dbl>

```

```

#
=====
===
# FUNCIÓN BOOTSTRAP
#
=====
===
estadistico <- function(data, indices) {
  muestra <- data[indices, ]
  return(mean(muestra$superficie_total_ha, na.rm = TRUE))
}

#
=====
===
# BOOTSTRAP
#
=====
===
set.seed(123)
boot_superficie <- boot(data = datos, statistic = estadistico, R = 1000)

cat("\n--- Resultados Bootstrap ---\n")

##
## --- Resultados Bootstrap ---

print(boot_superficie)

##
## ORDINARY NONPARAMETRIC BOOTSTRAP
##
##
## Call:
## boot(data = datos, statistic = estadistico, R = 1000)
##
##
## Bootstrap Statistics :
##      original      bias    std. error
## t1*  27.67778  0.07003333    3.184579

cat("\n--- Intervalo de confianza (95%) ---\n")

##
## --- Intervalo de confianza (95%) ---

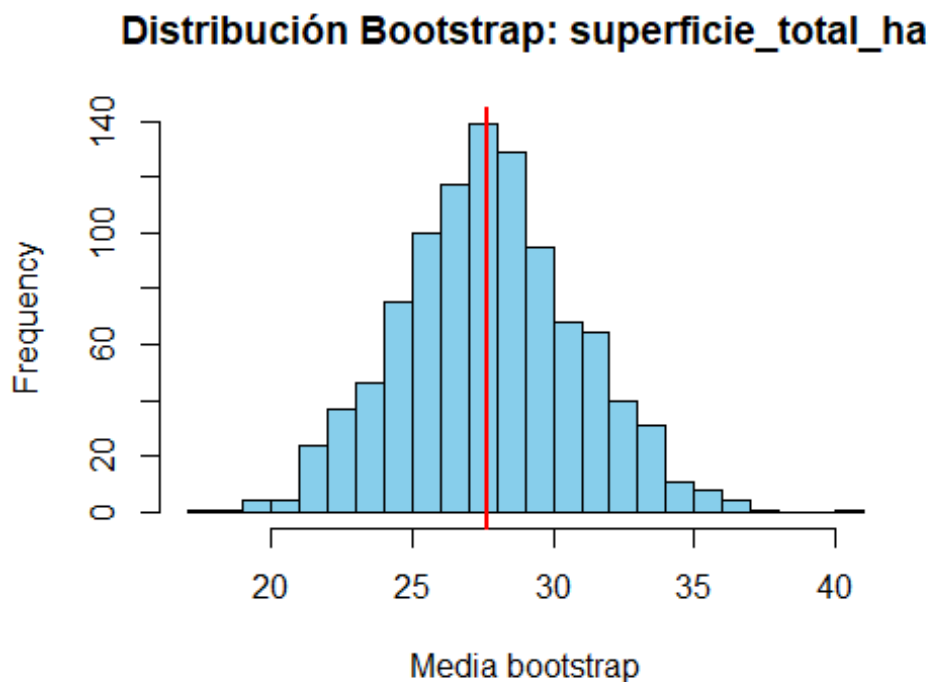
print(boot.ci(boot_superficie, type = c("perc", "bca")))

## BOOTSTRAP CONFIDENCE INTERVAL CALCULATIONS
## Based on 1000 bootstrap replicates
##

```

```
## CALL :
## boot.ci(boot.out = boot_superficie, type = c("perc", "bca"))
##
## Intervals :
## Level      Percentile      BCa
## 95%   (21.70, 34.06 )   (22.04, 34.95 )
## Calculations and Intervals on Original Scale

#
=====
===
# GRAFICO HISTOGRAMA
#
=====
===
hist(boot_superficie$t, breaks = 30, col = "skyblue",
     main = "Distribución Bootstrap: superficie_total_ha",
     xlab = "Media bootstrap")
abline(v = mean(datos$superficie_total_ha, na.rm = TRUE), col = "red",
       lwd = 2)
```



```
#
=====
===
# COMPARACIÓN ANTES VS DESPUÉS
#
=====
```

```

===
df_plot <- data.frame(
  Metodo = c("Original", "Bootstrap"),
  Media = c(mean(datos$superficie_total_ha, na.rm = TRUE),
             mean(boot_superficie$t, na.rm = TRUE))
)

ggplot(df_plot, aes(x = Metodo, y = Media, fill = Metodo)) +
  geom_bar(stat = "identity", width = 0.5) +
  geom_text(aes(label = round(Media, 2)), vjust = -0.5) +
  labs(title = "Comparación de medias: Original vs Bootstrap",
       y = "Media de superficie_total_ha") +
  theme_minimal()

```

