

Tarea 10: Bootstrap Paramétrico

Roman Alberto Velez Jimenez

```
## -- Attaching packages ----- tidyverse 1.3.1 --
## v ggplot2 3.4.0      v purrr 0.3.5
## v tibble 3.1.8       v dplyr 1.0.10
## v tidyr 1.2.1        v stringr 1.4.1
## v readr 2.1.2        v forcats 0.5.1

## -- Conflicts ----- tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag() masks stats::lag()
```

Problema

Encontrar cual es la probabilidad que un examen se haya hecho o no al azar bajo el modelo teórico

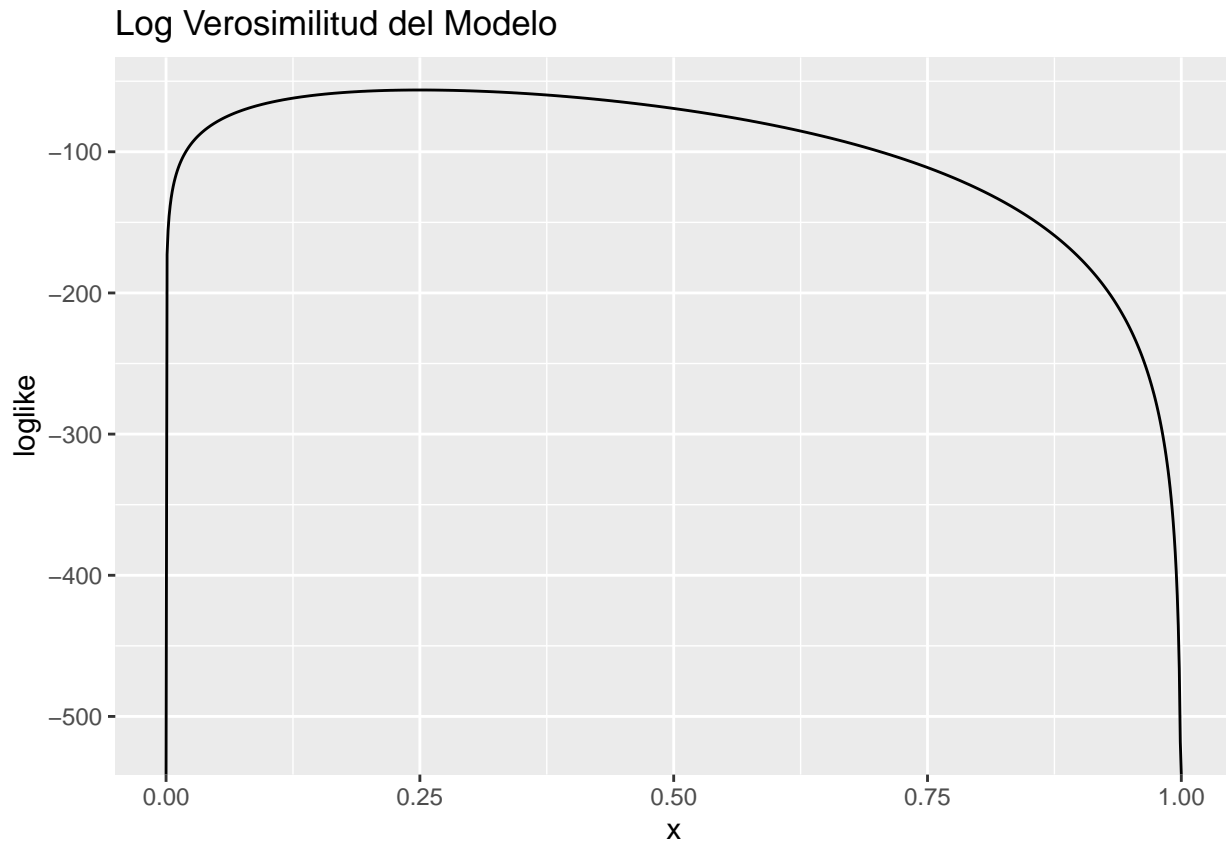
$$\mathcal{L}(p; x) = \prod_i^n p^n (1 - p)^{n-x},$$

donde p es desconocida.

La log verosimilitud para $n = 500$ y $x = 121$ se ve da la siguiente manera

```
# create log likelihood
bin_loglike <- function(n, n_err){
  n_corr <- n - n_err
  log_verosim <- function(p){
    n_err * log(p) + n_corr * log(1-p)
  }
}

# create some sim
tibble(x = seq(0, 1, 0.001)) |>
  mutate(loglike = map_dbl(x, bin_loglike(n = 100, n_err = 25))) |>
  ggplot(aes(x, loglike)) +
  geom_line() +
  ggtitle("Log Verosimilitud del Modelo")
```



Encontrando el máximo de este modelo, se encuentra que es cuando $p = 0.242$.

```
# optimize
p_mlike <- optimize(
  f = bin_loglike(n = 500, n_err = 121),
  interval = c(0, 1), # support of p
  maximum = TRUE # maximization
)

# print
print(str_glue("Maxima verosimilitud, p: {round(p_mlike[[1]], 4)}"))
```

```
## Maxima verosimilitud, p: 0.242
```

Bootstrap Paramétrico

Simulación del Modelo Teórico

Suponiendo cierto el proceso generador de datos, se generan B simulaciones con el parámetro máximo verosímil poblacional con la función `rbinom(n, size, p)`, la cual genera una muestra de tamaño n de una distribución binomial con $size$ ensayos bernoulli con una probabilidad de p .

```
# simular modelo
sim_modelo <- function(n, p, B=250){
  rbinom(B, n, p)
}

# una simulación
```

```
xsim_model <- sim_modelo(p = 0.5, n = 500, B=1)
cat("La simulación de estos datos es:", xsim_model, "\n")
```

```
## La simulación de estos datos es: 261
```

Simulación del Modelo Empírico

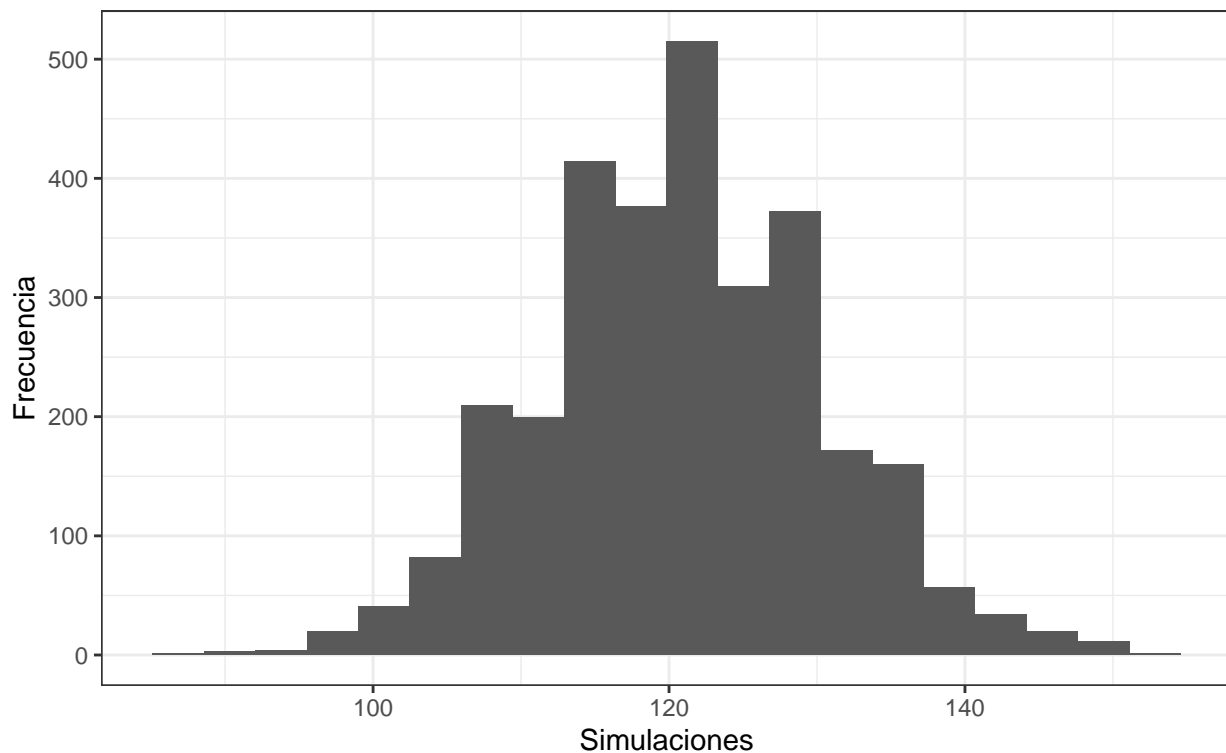
Con el estimador maximo verosimil, se generan $B = 3000$ simulaciones con el parámetro maximo verosimil poblacional.

```
B <- 3000
xsim_3000 <- sim_modelo(p = p_mlike[[1]], n = 500, B=B)

# grafica histograma
ggplot(tibble(xsim_3000), aes(x = xsim_3000)) +
  geom_histogram(bins = 20) +
  labs(x = "Simulaciones", y = "Frecuencia") +
  ggtitle("Histograma de simulaciones del Proceso Generador de Datos", "Bajo el estimador MV") +
  theme_bw()
```

Histograma de simulaciones del Proceso Generador de Datos

Bajo el estimador MV



Cálculo del estimador MV para cada simulación

Para cada simulación, se encuentra el estimador máximo verosimil.

```
rep_boot <- function(logv, est_mle, n, B=250){
  # create parametric bootstrap
  xsim <- sim_modelo(n = n, p = est_mle, B = B)
```

```

# find max verosimilitud for each simulation
mle_sim <- c()
for (i in 1:B){
  log_v_sim <- logv(n = n, n_err = xsim[i])
  optim_sim <- optimize(log_v_sim, interval = c(0, 1), maximum = TRUE)
  mle_sim[i] <- optim_sim$maximum
}
return(mle_sim)
}

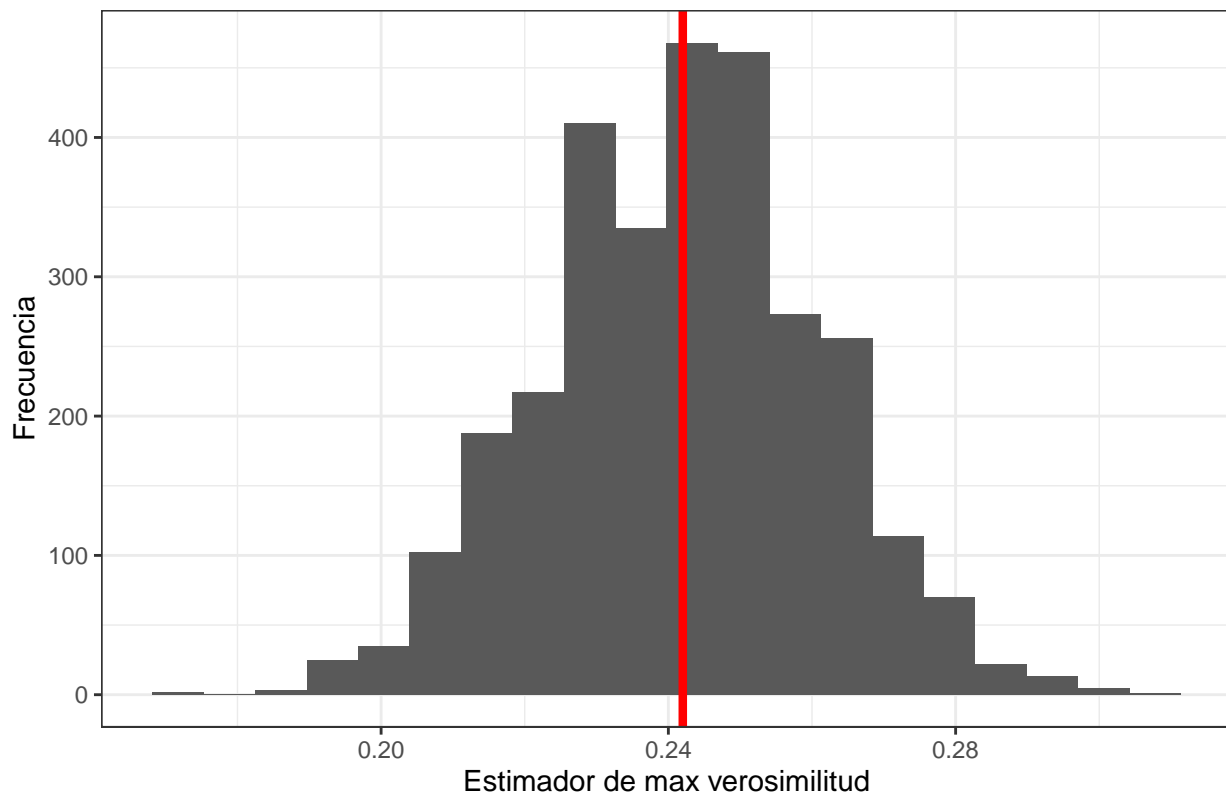
# calcular mv para cada sim
reps_boot <- rep_boot(logv = bin_loglike, est_mle = p_mlike[[1]], n = 500, B = 3000)

# grafica histograma
ggplot(tibble(reps_boot), aes(x = reps_boot)) +
  geom_histogram(bins = 20) +
  geom_vline(xintercept = p_mlike$maximum, color = "red", size = 1.5) +
  labs(x = "Estimador de max verosimilitud", y = "Frecuencia") +
  ggtitle("Histograma de estimadores de max verosimilitud") +
  theme_bw()

```

Warning: Using `size` aesthetic for lines was deprecated in ggplot2 3.4.0.
 ## i Please use `linewidth` instead.

Histograma de estimadores de max verosimilitud



Observamos como se distribuyen las simulaciones

```

# descripcion de las simulaciones
cat("descripcion de las simulaciones: \n")

```

```
## descripcion de las simulaciones:
```

```
summary(reps_boot)
```

```
##      Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
## 0.1720 0.2280 0.2420 0.2418 0.2560 0.3080
```

Desviación Estándar

```
cat("error estándar del estimador MV muestral \n")
```

```
## error estándar del estimador MV muestral
```

```
cat(sd(reps_boot))
```

```
## 0.01915407
```