TallerMuestreoAleatorio

Sebastian Chaparo J

2022-10-07

Contexto.

Se dispone de la base de datos de Cierres de TESLA para el año 2010 a 2022, la cual cuenta con las siguientes variables.

```
colnames(TSLA)
```

```
## [1] "Date"      "Open"      "High"      "Low"       "Close"     "Adj.Close"
## [7] "Volume"
```

- Date : Fecha
- Open : Precio cuando abre el mercado
- High : Precio más alto registrado para el día
- Low : Precio más bajo registrado para el día
- Close : Precio cuando el mercado cierra
- Adj.Close : Precio de cierre modificado basado en acciones corporativas
- volume : Cantidad de acciones vendidas en un día.

Con la información disponible se busca realizar un muestreo aleatorio simple para el Adj.Close

Analisis descriptivo Adj.Close

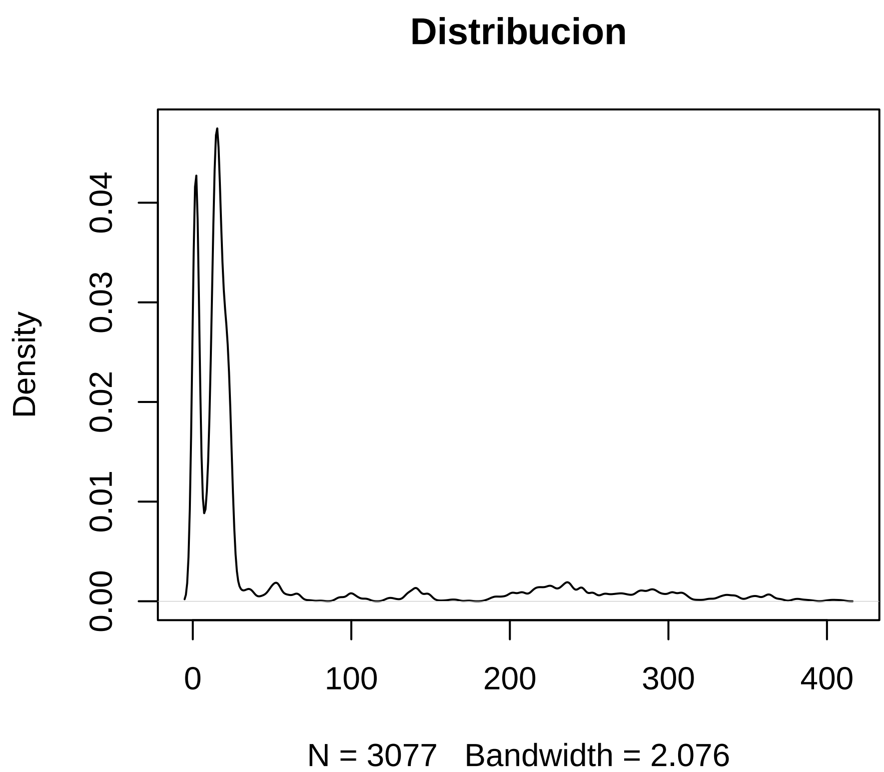
Resumen descriptivo

```
resumen <-  as.array(summary(TSLA$Adj.Close))
kable(resumen , col.names = c("Descriptivo", "Valor"), digits = 2)
```

Descriptivo	Valor
Min.	1.05
1st Qu.	8.11
Median	16.00
Mean	55.50
3rd Qu.	23.52
Max.	409.97

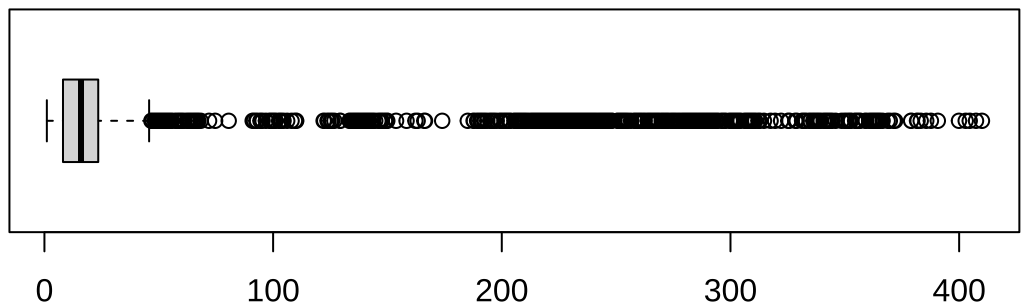
Distribucion de la variable

```
plot(density(TSLA$Adj.Close), main = "Distribucion" )
```



La variable en estudio no es normal, por lo tanto, los estadísticos paramétricos no tendrán mucha efectividad. Por fines académicos, asumimos una distribución normal.

```
boxplot(x = TSLA$Adj.Close ,horizontal = T)
```




```
atipicos <- boxplot.stats(TSLA$Adj.Close)$out
N.atipico <- length(atipicos)
Faltantes <- sum(is.na(TSLA$Adj.Close))
```

Valores atípicos, en total : 635.

Valores Faltantes : 0.

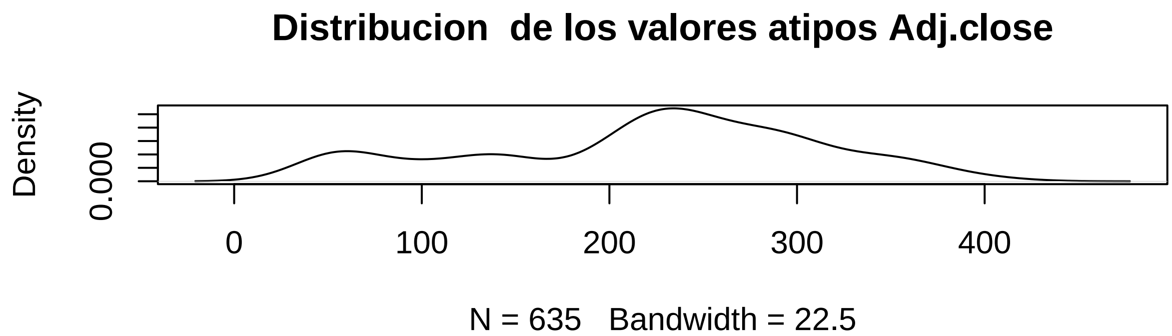
Supongamos que la empresa quiere realizar un estudio de los días en que el cierre ajustado consideraron atípicos, para ello, requiere de personal y tiempo. Ya se dispone de la base datos y se pueden filtrar estos valores, sin embargo, Realizar un estudio de las 635 observaciones, requiere de tiempo y personal, para ello se realiza un muestreo aleatorio simple.

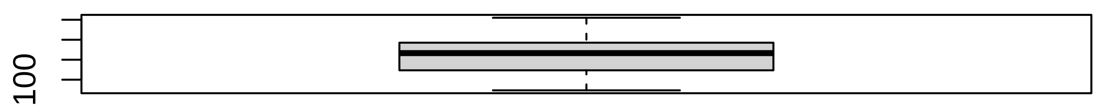
```
TSLA.1 <- TSLA %>% filter(Adj.Close %in% atipicos)

N <- nrow(TSLA.1)
```

Distribucion de los valores atipicos

```
par(mfrow = c(2,1))
plot(density(TSLA.1$Adj.Close),main ="Distribucion  de los valores atipos Adj.close")
boxplot(TSLA.1$Adj.Close)
```





Diseño del muestreo aleatorio simple.

Eleccion de la prueba piloto

```
n.0 <- round(0.1*nrow(TSLA.1),0)
n <- round(1.1*n.0,0)
```

Para seleccionar la prueba piloto teóricamente se recomienda seleccionar una muestra del 0.1% lo cual equivale a 64 + un 10% de esta muestra en caso de perder o dañar la información. El tamaño de la muestra piloto es de 70.

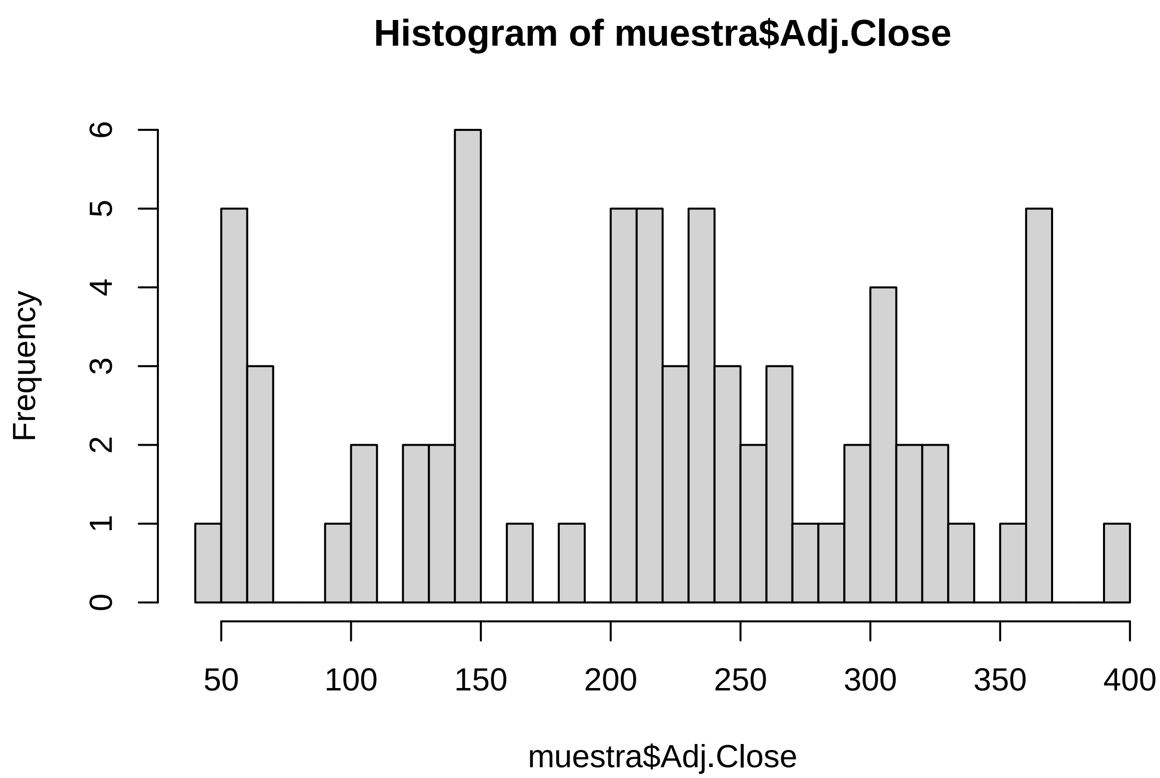
```
set.seed(13)
id.muestra <- sample(1:635,size = n,  replace = F)
muestra <- TSLA.1[id.muestra,]
kable(muestra[1:5,],caption = "Primeras 5 observaciones de la muestra piloto seleccionada",digits = 3)
```

Table 2: Primeras 5 observaciones de la muestra piloto seleccionada

	Date	Open	High	Low	Close	Adj.Close	Volume
472	2022-01-25	304.733	317.087	301.070	306.133	306.133	86595900
586	2022-07-11	252.103	253.063	233.627	234.343	234.343	99241200

	Date	Open	High	Low	Close	Adj.Close	Volume
320	2021-06-18	204.457	209.450	203.933	207.770	207.770	73682700
221	2021-01-27	290.117	297.167	286.220	288.053	288.053	82002000
248	2021-03-08	200.183	206.710	186.263	187.667	187.667	155361000

```
hist(muestra$Adj.Close,30)
```

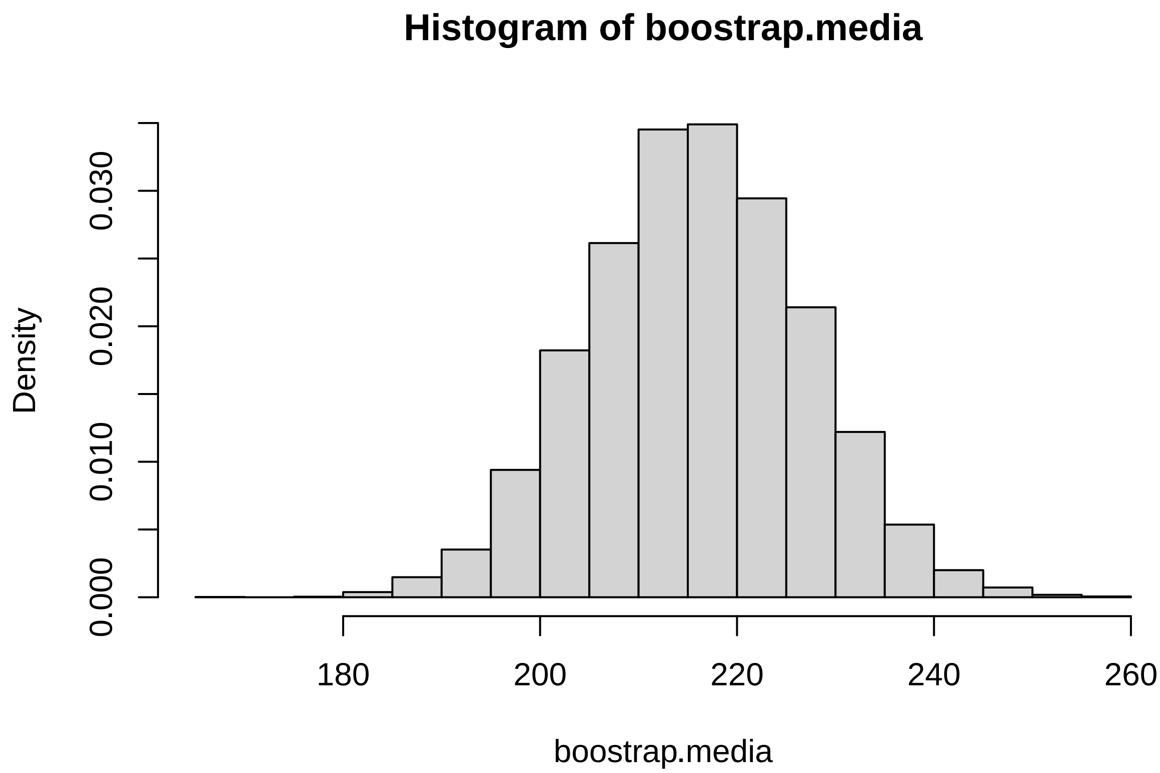


La eleccion de la muestra piloto claramento no tiene una distribucion normal, Por lo tanto, sería aconsejable utilizar estadísticas no paramétricas que nos permitan inferir sobre el posible estimador poblacional, como es necesario tener conocimiento de la media y la sd estándar, sé realizar una primera estimación de estos por un método no paramétrico denominado boostrap.

```
boostrap.media <- rep(NA,9999)
boostrap.sd <- rep(NA,9999)
set.seed(123)
for(i in 1:9999){
  muestras <- sample(muestra$Adj.Close, replace = T)
  boostrap.sd[i] <- sd(muestras)
  boostrap.media[i] <- mean(muestras)

}

Media.boostrap <- mean(boostrap.media)
sd.boostrap <- mean(boostrap.sd)
hist(boostrap.media,breaks = 30,probability = T)
```



De acuerdo a la gráfica obtenida el boostrap se distribuye de manera normal. Recordando el teorema del límite central, podemos asumir que las medias de esta distribución serán un estimador del parámetro poblacional de la media aritmética de la población, esto como un primer acercamiento al parámetro.

Media: 215.9215373.
Sd: 93.7354074

Eleccion del tamaño muestral

Una vez seleccionada la muestra piloto podemos determinar el error con el cual trabajar y el tamaño de la muestra para ser representativo de la población. Para el error se recomienda que no sobrepase $0.1(\mu)$ de la muestra piloto

```
Error <- Media.boostrap*0.1
Error.seq <- seq(0 ,Error, Error/20 )
Ns <- 0.05
z <- qnorm(1 - 0.05/2)
posibles.n <- ((z^2)*sd.boostrap^2)/Error.seq^2  # Para poblacion infinita
posibles.n.corre <-  posibles.n/(1+(posibles.n/N))
Tabla2 <- cbind(Error.seq, posibles.n, posibles.n.corre)
kable(Tabla2 ,col.names = c("Error", "n (N inf)", "n"),digits = 2)
```

Error	n (N inf)	n
0.00	Inf	NaN

Error	n (N inf)	n
1.08	28958.20	621.37
2.16	7239.55	583.79
3.24	3217.58	530.34
4.32	1809.89	470.07
5.40	1158.33	410.15
6.48	804.39	354.86
7.56	590.98	306.10
8.64	452.47	264.21
9.72	357.51	228.73
10.80	289.58	198.88
11.88	239.32	173.82
12.96	201.10	152.73
14.03	171.35	134.94
15.11	147.75	119.86
16.19	128.70	107.01
17.27	113.12	96.01
18.35	100.20	86.54
19.43	89.38	78.35
20.51	80.22	71.22
21.59	72.40	64.99

El cálculo anterior se puede realizar de manera más sencilla con la librería ´samplingbook´

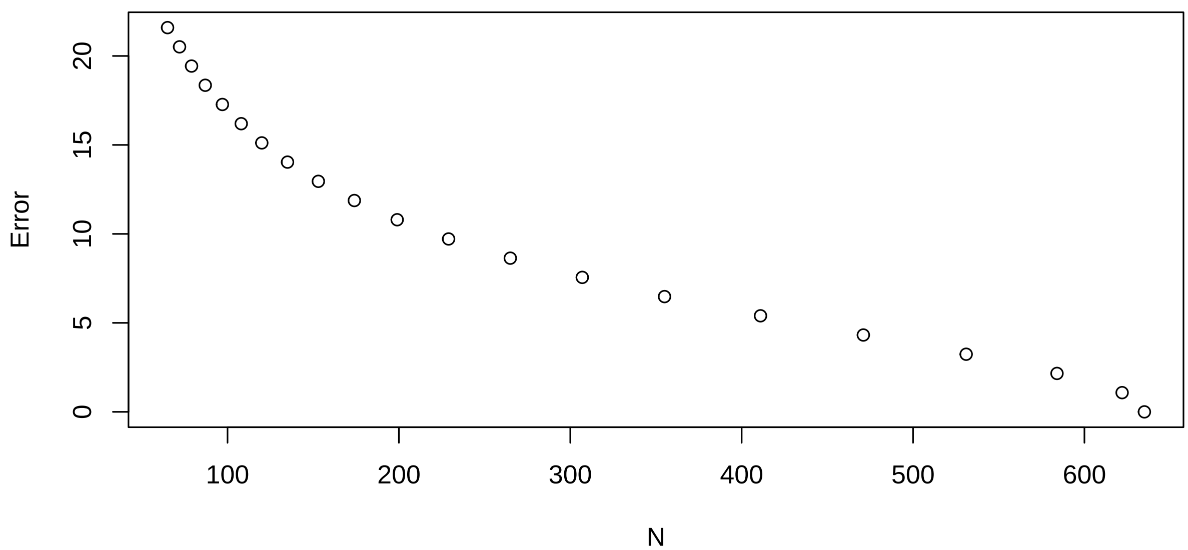
```
library(samplingbook)
resulatado <- rep(NA,length(Error.seq))
err.1 <-  rep(NA,length(Error.seq))
for( i in 1:length(Error.seq) ){
  resulatado[i] <-  sample.size.mean(Error.seq[i], sd.boostrap, 635)$n
  err.1[i] <- Error.seq[i]
}

kable(cbind(resulatado,err.1) ,digits = 3 , col.names = c("N", "Error") )
```

N	Error
635	0.000
622	1.080
584	2.159
531	3.239
471	4.318
411	5.398
355	6.478
307	7.557
265	8.637
229	9.716
199	10.796
174	11.876
153	12.955
135	14.035
120	15.115
108	16.194

N	Error
97	17.274
87	18.353
79	19.433
72	20.513
65	21.592

```r
plot(y=err.1, x= resulatado ,ylab = "Error" , xlab = "N")
```




```r
sample.size.mean(e = 4.3184,S =    sd.boostrap ,N = 635,    level = 0.95)
```

```
##
## sample.size.mean object: Sample size for mean estimate
## With finite population correction: N=635, precision e=4.3184 and standard deviation S=93.7354
##
## Sample size needed: 471
```

Para un error de 4.3184(2%)con un nivel de confianza del 95% se requiere de una muestra de 471 fechas registradas de valores atipicos de adj.close

```r
id.sample2 <- sample(1:635,size = 471, replace = F)
muestra.final <- TSLA.1[id.sample2,]
```

Al determinar el intervalo de confianza del 95% se puede evidenciar que si se cumple el criterio del error, es decir, se estima que la media es 219.6699 ± 4.1283 valor que se aproxima muy bien al propuesto de 4.3184

```
Smean(muestra.final$Adj.Close, N = 635 , level = 0.95)
```

```
##
## Smean object: Sample mean estimate
## With finite population correction: N=635
##
## Mean estimate: 222.3127
## Standard error: 2.1607
## 95% confidence interval: [218.0778,226.5477]
```

Se puede realizar una comparación con la verdadera media poblacional

```
mean(TSLA.1$Adj.Close)
```

```
## [1] 220.1503
```

La cual se encuentra en el intervalo de la muestra seleccionada [215.5417,223.7982]

Se puede comprobar lo mismo para diferentes tamaño muestrales, supongamos un error del 5% equivalente a 10.79 segun la prueba piloto

```
sample.size.mean(e = 10.79,S =    sd.boostrap ,N = 635,    level = 0.95)
```

```
##
## sample.size.mean object: Sample size for mean estimate
## With finite population correction: N=635, precision e=10.79 and standard deviation S=93.7354
##
## Sample size needed: 200
```

```
id.sample2 <- sample(1:635,size = 200, replace = F)
muestra.final <- TSLA.1[id.sample2,]
Smean(muestra.final$Adj.Close, N = 635 , level = 0.95)
```

```
##
## Smean object: Sample mean estimate
## With finite population correction: N=635
##
## Mean estimate: 202.688
## Standard error: 5.6461
## 95% confidence interval: [191.6219,213.754]
```