

1. Muestreo bietápico

Un estudiante de posgrado tiene una colección de 400 libros en su librero. Para estimar el número total de palabras en su colección de libros, selecciona una muestra s_I de dos libros usando un muestreo aleatorio simple sin reposición. Para los dos libros seleccionados tiene el número de páginas de cada uno. Posteriormente, en cada libro seleccionado selecciona dos páginas usando un muestreo aleatorio simple sin reposición. Una vez seleccionada la página registra de cada una el número de palabras que contiene. La información muestral es como sigue.

Tabla 1: Muestras seleccionadas en cada etapa y valores observados

Libro seleccionado (s_I)	Número de páginas (N_i)	Páginas seleccionadas (s_i)	Número de Palabras en la página (y_k)
195	300	61	23
		212	25
288	200	99	20
		111	20

Aquí las páginas conforman la población de interés y los libros son las upm.

- i) Calcule las probabilidades de inclusión de primer y segundo orden correspondientes a las páginas seleccionadas. Por el **Resultado 5.4** de las notas, tenemos que las probabilidades de primer orden, considerando el muestreo bietápico están dadas por

$$\pi_k = \pi_{k|i} \pi_{Ii},$$

donde π_{Ii} representa la probabilidad de inclusión de la upm i (en este caso, serían los libros) y $\pi_{k|i}$ la probabilidad de inclusión del elemento k cuando se realiza la selección en la upm i (sería la probabilidad de escoger una página del libro i) y dado que en la primera y segunda 'etapa' se usó un muestreo aleatorio simple sin reposición, se tiene que las probabilidades de inclusión correspondientes a las páginas seleccionadas son:

$$\pi_{61} = \frac{2}{400} \cdot \frac{2}{300} = \frac{1}{30000} = \pi_{212}$$

y

$$\pi_{99} = \frac{2}{400} \cdot \frac{2}{200} = \frac{1}{20000} = \pi_{111},$$

pues

$$\pi_{I195} = \frac{2}{400} = \pi_{I288},$$

además

$$\pi_{61|195} = \frac{2}{300} = \pi_{212|195} \quad y \quad \pi_{99|288} = \frac{2}{200} = \pi_{111|288}.$$

Por otra parte, sabemos que las probabilidades de inclusión de segundo orden están dadas por

$$\pi_{kl} = \begin{cases} \pi_{Ii}\pi_{kl|i} & si \quad i = j \\ \pi_{Iij}\pi_{k|i}\pi_{l|j} & si \quad i \neq j \end{cases}$$

donde π_{Iij} representa la probabilidad de inclusión de segundo orden para las *upm* i y j , $i \neq j$ y $\pi_{kl|i}$ la probabilidad de inclusión de segundo orden de los elementos k y l cuando se realiza la selección en la *upm* i , entonces, se tiene que

$$\pi_{61\&212} = \pi_{I_{195}}\pi_{61\&212|195} = \frac{2}{400} \cdot \frac{2}{300(299)} = \frac{1}{8970000}$$

$$\pi_{99\&111} = \pi_{I_{288}}\pi_{99\&111|288} = \frac{2}{400} \cdot \frac{2}{200(199)} = \frac{1}{3980000}$$

$$\pi_{61\&99} = \pi_{I_{195\&288}}\pi_{61|195}\pi_{99|288} = \frac{2}{400(399)} \cdot \frac{2}{300} \cdot \frac{2}{200} = 8.354218881 \times 10^{-10}$$

$$\pi_{61\&111} = \pi_{I_{195\&288}}\pi_{61|195}\pi_{111|288} = \frac{2}{400(399)} \cdot \frac{2}{300} \cdot \frac{2}{200} = 8.354218881 \times 10^{-10}$$

$$\pi_{212\&99} = \pi_{I_{195\&288}}\pi_{212|195}\pi_{99|288} = \frac{2}{400(399)} \cdot \frac{2}{300} \cdot \frac{2}{200} = 8.354218881 \times 10^{-10}$$

$$\pi_{212\&111} = \pi_{I_{195\&288}}\pi_{212|195}\pi_{111|288} = \frac{2}{400(399)} \cdot \frac{2}{300} \cdot \frac{2}{200} = 8.354218881 \times 10^{-10}$$

- ii) Calcule el valor estimado del número total de palabras en la colección de libros. Por el **Resultado 5.5** el estimador HT para el total se ve como

$$\hat{t}_{y\pi} = \sum_{i \in S_I} \frac{\hat{t}_{i\pi}}{\pi_{Ii}}$$

con $\hat{t}_{i\pi} = \sum_{k \in s_i} \frac{y_k}{\pi_{k|i}}$, por ende, tenemos que la estimación quedaría de la siguiente manera:

- a) Para $S_I = 195$, se tiene que

$$\hat{t}_{195\pi} = \sum_{k \in \{61, 212\}} \frac{300}{2} y_k = 150(23 + 25) = 7200$$

- b) Para $S_I = 288$, se tiene

$$\hat{t}_{288\pi} = \sum_{k \in \{99, 111\}} \frac{200}{2} y_k = 100(40) = 4000$$

Por último, nuestra estimación del total quedaría como:

$$\hat{t}_{y\pi} = \sum_{i \in \{195, 288\}} 200\hat{t}_{i\pi} = 200(7200 + 4000) = 2240000$$

- iii) Dé una estimación insesgada de la varianza del estimador usado en ii). Sabemos, por el **Resultado 5.6**, que un estimador insesgado de la varianza del estimador del inciso anterior está dado por

$$\widehat{V}(\widehat{t}_{y\pi}) = \widehat{V}_{PSU} + \widehat{V}_{SSU} = \sum_{i \in s_I} \sum_{j \in s_I} \widehat{\Delta}_{Iij} \frac{\widehat{t}_{i\pi}}{\pi_{Ii}} \frac{\widehat{t}_{j\pi}}{\pi_{Ij}} + \sum_{i \in s_I} \frac{\widehat{V}_i}{\pi_{Ii}},$$

donde $\widehat{\Delta}_{Iij} = \frac{\pi_{Iij} - \pi_{Ii}\pi_{Ij}}{\pi_{Iij}}$ y $\widehat{V}_i = \sum_{k \in s_i} \sum_{l \in s_i} \widehat{\Delta}_{kl|i} \frac{y_k}{\pi_{k|i}} \frac{y_l}{\pi_{l|i}}$, con $\widehat{\Delta}_{kl|i} = \frac{\pi_{kl|i} - \pi_{k|i}\pi_{l|i}}{\pi_{kl|i}}$, dado que en la primera y segunda etapa se usó un muestreo aleatorio simple sin reemplazo, tenemos que

$$\sum_{i \in s_I} \sum_{j \in s_I} \widehat{\Delta}_{Iij} \frac{\widehat{t}_{i\pi}}{\pi_{Ii}} \frac{\widehat{t}_{j\pi}}{\pi_{Ij}}$$

es la expresión de la varianza del estimador HT para un muestreo aleatorio simple sin reemplazo para la primera etapa, por lo que se tiene que

$$\sum_{i \in s_I} \sum_{j \in s_I} \widehat{\Delta}_{Iij} \frac{\widehat{t}_{i\pi}}{\pi_{Ii}} \frac{\widehat{t}_{j\pi}}{\pi_{Ij}} = \frac{N^2}{n} \left(1 - \frac{n}{N}\right) S_{\widehat{t}_{i\pi} s_I}^2$$

con $S_{\widehat{t}_{i\pi} s_I}^2 = \frac{\sum_{k \in s_I} (\widehat{t}_{i\pi} - \overline{\widehat{t}_{i\pi}})^2}{n-1}$, $\overline{\widehat{t}_{i\pi}} = \frac{\sum_{i \in s_I} \widehat{t}_{i\pi}}{n}$, $N = 400$ y $n = 2$.

```
s_I <-c(195,195, 288, 288)
N_i <-c(300,300, 200, 200)
s_i <-c(61, 212, 99, 111)
y_k <-c(23, 25, 20,20)

Datos <- data.frame(s_I, N_i, s_i, y_k)

t_i = c(7200, 4000)

PSU = var(t_i)*(400^2 / 2) * (1 - 2/400)
PSU

## [1] 4.07552e+11
```

De manera similar, tenemos que el segundo término del sumando, dado que a cada V_i la podemos pensar como una suma de varianzas del estimador HT aplicado a las páginas de los libros seleccionados, se tiene que

$$\widehat{V}_i = \sum_{k \in s_i} \sum_{l \in s_i} \widehat{\Delta}_{kl|i} \frac{y_k}{\pi_{k|i}} \frac{y_l}{\pi_{l|i}} = \frac{N_i}{2} \left(1 - \frac{2}{N_i}\right) S_{y_{s_i}}^2$$

con $S_{y_{s_i}}^2 = \frac{\sum_{k \in s_i} (y_k - \bar{y}_{s_i})^2}{n-1}$ y $\bar{y}_{s_i} = \frac{\sum_{k \in s_i} y_k}{n}$

```
library(dplyr)

##
## Attaching package: 'dplyr'
## The following objects are masked from 'package:stats':
##
##   filter, lag
## The following objects are masked from 'package:base':
##
##   intersect, setdiff, setequal, union

V_i = Datos %>% group_by(s_I)%>%summarise(V_i =unique(N_i)^2/unique(2)*(1
    -unique(2)/unique(N_i))*var(y_k))%>%ungroup()

SSU = sum(V_i) * (400 / 2)

SSU

## [1] 17976600
```

Finalmente, la estimación de la varianza estaría dada por

```
PSU + SSU

## [1] 4.0757e+11
```

- iv) Dé una aproximación de la varianza del estimador usado en ii) que sólo use las probabilidades de inclusión de primer orden o los factores de expansión. Sabemos, por la ecuación (168) de las notas, que bajo los supuestos de $\pi_{Ii} = n_I p_i$ y $m_I = n_I$ usando un muestreo con reemplazo para seleccionar las upm, se tiene que una estimación de la varianza es la siguiente

$$\widehat{V}(\widehat{t}_{y\pi}) = \frac{n_I}{n_I - 1} \sum_{i=1}^{n_I} \left(\frac{\widehat{t}_{i\pi}}{\pi_{Ii}} - \frac{\widehat{t}_{y\pi}}{n_I} \right)^2$$

en nuestro caso, tenemos que $n_I = 2$, $\frac{\widehat{t}_{y\pi}}{2} = \frac{\frac{400}{2} \cdot (7200 + 4000)}{2} = \frac{2240000}{2} = 1120000$, por ende, tenemos lo siguiente

```
t_ypi2 = 1120000
estVar <- 2 * sum((400/2 * t_i - t_ypi2)^2)
estVar

## [1] 4.096e+11
```

2. Comparación de diseños de muestreo

Suponga que se realizará una nueva elección de diputaciones a nivel federal y le han encargado realizar el diseño de muestreo. Para esto cuenta con la información de los resultados a nivel acta (casilla) de los cómputos distritales de 2021 (<https://computos2021.ine.mx/base-de-datos>).

Por simplicidad suponga que se centrará en estudiar los diseños de manera que elegirá el que sirva para realizar de la mejor forma la estimación del porcentaje de votos a favor de una coalición integrada por Morena, PT y P. Verde a nivel nacional.

En el denominador considerará sólo el total de votos válidos, es decir, que no se considerarán los votos nulos para efectos del cálculo del porcentaje.

Consideré los siguientes tres diseños a comparar:

- i. Se seleccionan 1200 casillas de las 163,666 usando un muestreo aleatorio simple sin reemplazo.
- ii. Se considera una estratificación a partir de los 300 diferentes distritos electorales del país. El diseño de muestreo considerado en cada estrato corresponde a un muestreo aleatorio simple sin reemplazo de 4 casillas.
- iii. Se considera una estratificación considerando las cinco circunscripciones electorales de México. En cada estrato se usa un muestreo aleatorio simple sin reemplazo para seleccionar 8 distritos electorales y en cada distrito electoral seleccionado se usa un muestreo aleatorio simple para seleccionar 30 casillas.

Realice la comparación de los tres diseños a partir de simulaciones. Es decir, repita 1000 veces lo siguiente. Seleccione una muestra con cada diseño y realice la estimación, en este caso deberá usar un estimador de razón.

A partir de las 1000 estimaciones estime el ECM del estimador que se obtendría para cada diseño y con estos resultados indique cuál diseño parece ser el mejor.

```
options(digits=10)

library(tidyverse)

## Warning in system("timedatectl", intern = TRUE): running command 'timedatectl' had
## status 1
## -- Attaching packages ----- tidyverse 1.3.1 --
## v ggplot2 3.3.5      v purrr   0.3.4
## v tibble  3.1.3      v dplyr  1.0.7
## v tidyr   1.1.3      v stringr 1.4.0
## v readr   2.0.0      v forcats 0.5.1
## -- Conflicts ----- tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()

# leer el csv
df <- data.table::fread("../csv/diputaciones.csv", sep="|")
```

```
#solo nos interesa las casillas donde si hubo votaciones
df <- filter(df, TOTAL_VOTOS_CALCULADOS != 0)
df <- df %>%drop_na(TOTAL_VOTOS_CALCULADOS)
df <- df %>%drop_na(`VOTOS NULOS`)

df$totalvotos = df$TOTAL_VOTOS_CALCULADOS - df$`VOTOS NULOS`
df <- filter(df, totalvotos != 0)
#tamaño de población
N <- dim(df)[1]
#tamaño de muestra
n <- 1200

#datos que nos interesan
midf <- df[, c("MORENA", "PVE", "PT", "PT-MORENA",
               "PVE-PT", "PVE-MORENA", "PVE-PT-MORENA", "totalvotos")]
midf$estimacion = apply(midf,1, function(x) (sum(x)- x[length(x)])/x[length(x)])
#pesos para el mas
midf$wk.mas = N/n

diseno.mas <- function(){
  #seleccionamos muestra
  seleccion_muestra <- sample(1:N ,size = 1200)

  #estimamos con el estimador de razon de la media
  muestra <- midf[seleccion_muestra, c("estimacion", "wk.mas")]
  return(sum(muestra$estimacion*muestra$wk.mas)/sum(muestra$wk.mas))
}

simulacion.mas <- replicate(10, diseno.mas())
mean(simulacion.mas)

## [1] 0.4503768834

# #estratos
df$distritoelec <- paste(df$ID_ESTADO, df$ID_DISTRITO, df$NOMBRE_DISTRITO, sep="-")
#revisamos el tamaño deben ser 300 unicos
length(unique(df$distritoelec))

## [1] 300

midf2 <- df[, c("distritoelec")]
midf2$estimacion <- midf$estimacion

# otorgamos id a cada distrito
iddistrito = 1:300
```

```

distributoelec = unique(df$distributoelec)
idsdist <- data.frame(idistributo, distributoelec)
midf2 <- merge(midf2, idsdist)

#calculo de las probabilidades de inclusión para todos los elementos
prob.inc <- data.frame(midf2 %>% group_by(idistributo) %>% count())
prob.inc$wk <- 1/(4/prob.inc$n)

midf2 <- merge(midf2, prob.inc, by="idistributo")

#creamos un subid (id dentro del distrito electoral)
subid <- c()
for (i in 1:300){
  s0 = midf2$distributoelec[midf2$idistributo == i]
  subid = c(subid, 1:length(s0))
}
midf2$subid <- subid

# creamos un id unico en toda la ppoblacion
midf2$idun <- paste(midf2$idistributo, midf2$subid, sep = "-")

#prob.inc
disenost <- function(){
  #seleccionamos muestra con base en el idunico
  idun <- c()
  for (i in 1:300) {
    idun <- c(idun, paste(i, sample(1:prob.inc$n[i], size = 4), sep="-"))
  }
  seleccion <- data.frame(idun)
  #obtenemos los datos de la muestra
  muestra <- left_join(seleccion, midf2, by = "idun")

  #estimaciones
  return(sum(muestra$estimacion*muestra$wk)/sum(muestra$wk))
}
simulacion.estr <- replicate(10, disenost())
mean(simulacion.estr)

## [1] 0.4465471265

```

```

# trabajaremos con un dataframe con la siguiente
midf3 <- df[, c("NOMBRE_ESTADO", "NOMBRE_DISTRITO", "ID_DISTRITO", "ID_ESTADO", "distributoelec")]
# recuperamos las estimaciones hechas anteriormente
midf3$estimacion <- midf$estimacion

```

```

# asignamos circunscripciones con respecto a los estados
circ <- c(2, 1, 1, 3, 2, 5, 3, 1, 4, 1, 2, 4, 5, 1, 5, 5, 4, 1, 2, 3, 4,
         2, 3, 2, 1, 1, 3, 2, 4, 3, 3, 2)
# recuperamos los nombres de los estados y hacemos un dataframe con los estados
# y circunscripciones
NOMBRE_ESTADO <- unique(df$NOMBRE_ESTADO)
circdf <- data.frame(circ, NOMBRE_ESTADO)
# unimos los dataframes y ahora tenemos la circunscripción en el dataframe
midf3 <- merge(circdf, midf3, by="NOMBRE_ESTADO")

# otorgamos id unico a cada distrito
iddistrito = 1:300
distritoelec = unique(df$distritoelec)
idsdist <- data.frame(iddistrito, distritoelec)
midf3 <- merge(midf3, idsdist)

#agrupara cada estrato (hacemos un dataframe para cada circunscripción)
circ1 <- midf3 %>% filter(circ == 1)
circ2 <- midf3 %>% filter(circ == 2)
circ3 <- midf3 %>% filter(circ == 3)
circ4 <- midf3 %>% filter(circ == 4)
circ5 <- midf3 %>% filter(circ == 5)

# ahora a cada circunscripción le calcularemos cuantas casillas le
# pertenecen, tambien un sub-id (que sera un id dentro de cada distrito),
# vector que guardara los subids
subid <- c()
# vector que guardara cuantas casillas le pertenecen a cada distrito
npob <- c()
for (i in unique(circ1$iddistrito)){
  # seleccionamos distritos únicos por circunscripción
  s0 = circ1$distritoelec[circ1$iddistrito == i]
  # agregamos el subid
  subid = c(subid, 1:length(s0))
  # total de casillas que le pertenecen al distrito
  npob <- c(npob, rep(length(s0), length(s0)))
}
# agregamos los resultados anteriores al dataframe
circ1$subid <- subid
circ1$npob <- npob
# calculamos los pesos para cada casilla
circ1$wk <- 1/(8/length(unique(circ1$iddistrito))*30/circ1$npob)
# creamos un id unico para cada casilla con base en iddistrito y subid
circ1$idun <- paste(circ1$iddistrito, circ1$subid, sep="-")
# relaciona el numero de distrito con el numero de casillas

```



```

prob.c1 <- data.frame(circ1 %>% group_by(idistributo) %>% count())
#repetimos este proceso para cada circunscripción

#circunscripción 2
subid <- c()
npob <- c()
for (i in unique(circ2$idistributo)){
  s0 = circ2$distributoelec[circ2$idistributo == i]
  subid = c(subid, 1:length(s0))
  npob <- c(npob, rep(length(s0), length(s0)))
}
circ2$subid <- subid
circ2$npob <- npob
circ2$wk <- 1/(8/length(unique(circ2$idistributo))*30/circ2$npob)
prob.c2 <- data.frame(circ2 %>% group_by(idistributo) %>% count())
circ2$idun <- paste(circ2$idistributo, circ2$subid, sep="-")

# circunscripción 3
subid <- c()
npob <- c()
for (i in unique(circ3$idistributo)){
  s0 = circ3$distributoelec[circ3$idistributo == i]
  subid = c(subid, 1:length(s0))
  npob <- c(npob, rep(length(s0), length(s0)))
}
circ3$subid <- subid
circ3$npob <- npob
circ3$wk <- 1/(8/length(unique(circ3$idistributo))*30/circ3$npob)
prob.c3 <- data.frame(circ3 %>% group_by(idistributo) %>% count())
circ3$idun <- paste(circ3$idistributo, circ3$subid, sep="-")

# circunscripción 4
subid <- c()
npob <- c()
for (i in unique(circ4$idistributo)){
  s0 = circ4$distributoelec[circ4$idistributo == i]
  subid = c(subid, 1:length(s0))
  npob <- c(npob, rep(length(s0), length(s0)))
}
circ4$subid <- subid
circ4$npob <- npob
circ4$wk <- 1/(8/length(unique(circ4$idistributo))*30/circ4$npob)
prob.c4 <- data.frame(circ4 %>% group_by(idistributo) %>% count())
circ4$idun <- paste(circ4$idistributo, circ4$subid, sep="-")

```

```
# circunscripcion 5
subid <- c()
npob <- c()
for (i in unique(circ5$iddistrito)){
  s0 = circ5$distritoelec[circ5$iddistrito == i]
  subid = c(subid, 1:length(s0))
  npob <- c(npob, rep(length(s0), length(s0)))
}
circ5$subid <- subid
circ5$npob <- npob
circ5$wk <- 1/(8/length(unique(circ5$iddistrito))*30/circ5$npob)
prob.c5 <- data.frame(circ5 %>% group_by(iddistrito) %>% count())
circ5$idun <- paste(circ5$iddistrito, circ5$subid, sep="-")

diseno.est.c1 <- function(){
  # seleccionamos 8 distritos de la circunscripción
  seleccion <- sample(unique(circ1$iddistrito), size=8)
  # aquí guardaremos el idunico de las casillas selecciones en la muestra
  idun <- c()
  # para cada distrito de los 8 seleccionados
  for (i in seleccion) {
    # vemos cuantas casillas le pertenecen
    iddist <- prob.c1 %>% filter(iddistrito == i)
    # y seleccionamos 30 elemento con un m.a.s.
    idun <- c(idun, paste(i, sample(iddist$n, size = 30), sep="-"))
  }
  #obtenemos los datos de la muestra
  muestra.c1 <- left_join(data.frame(idun), circ1, by = "idun")
  # estimamos
  return(sum(muestra.c1$estimacion*muestra.c1$wk)/sum(muestra.c1$wk))
}
# de manera analoga para el resto de circunscripciones

#circunscripcion 2
diseno.est.c2 <- function(){
  seleccion <- sample(unique(circ2$iddistrito), size=8)
  idun <- c()
  for (i in seleccion) {
    iddist <- prob.c2 %>% filter(iddistrito == i)
    idun <- c(idun, paste(i, sample(iddist$n, size = 30), sep="-"))
  }
  muestra.c2 <- left_join(data.frame(idun), circ2, by = "idun")

  return(sum(muestra.c2$estimacion*muestra.c2$wk)/sum(muestra.c2$wk))
}
```

```

}

#circunscripción 3
diseno.est.c3 <- function(){
  seleccion <- sample(unique(circ3$iddistrito), size=8)
  idun <- c()
  for (i in seleccion) {
    iddist <- prob.c3 %>% filter(iddistrito == i)
    idun <- c(idun, paste(i, sample(iddist$n, size = 30), sep="-"))
  }
  muestra.c3 <- left_join(data.frame(idun), circ3, by = "idun")

  return(sum(muestra.c3$estimacion*muestra.c3$wk)/sum(muestra.c3$wk))
}

#circunscripción 4
diseno.est.c4 <- function(){
  seleccion <- sample(unique(circ4$iddistrito), size=8)
  idun <- c()
  for (i in seleccion) {
    iddist <- prob.c4 %>% filter(iddistrito == i)
    idun <- c(idun, paste(i, sample(iddist$n, size = 30), sep="-"))
  }
  muestra.c4 <- left_join(data.frame(idun), circ4, by = "idun")

  return(sum(muestra.c4$estimacion*muestra.c4$wk)/sum(muestra.c4$wk))
}

diseno.est.c5 <- function(){
  seleccion <- sample(unique(circ5$iddistrito), size=8)
  idun <- c()
  for (i in seleccion) {
    iddist <- prob.c5 %>% filter(iddistrito == i)
    idun <- c(idun, paste(i, sample(iddist$n, size = 30), sep="-"))
  }
  #obtenemos los datos de la muestra
  muestra.c5 <- left_join(data.frame(idun), circ5, by = "idun")
  return(sum(muestra.c5$estimacion*muestra.c5$wk)/sum(muestra.c5$wk))
}

c1 <- replicate(10, diseno.est.c1())
c2 <- replicate(10, diseno.est.c2())
c3 <- replicate(10, diseno.est.c3())
c4 <- replicate(10, diseno.est.c4())

```

```
c5 <- replicate(10, disenno.est.c5())

mean(c(c1, c2, c3, c4, c5))

## [1] 0.4598111733
```

3. Estimación básica de una encuesta con diseño multietápico

Considere la Encuesta Nacional de Vivienda (ENVI) 2020

<https://www.inegi.org.mx/programas/envi/2020/>

Suponga que será el encargado de generar los resultados básicos presentados en el tabulado llamado Cuadro 5.1 (https://www.inegi.org.mx/contenidos/programas/envi/2020/tabulados/envi2020_ema05xlsx.zip).

En particular realice lo siguiente

1. Describa brevemente el diseño de muestreo usado en la encuesta. Es decir, si es muestreo aleatorio simple, tiene estratificación, es por conglomerados, etc.

Se usó un diseño probabilístico, cuyo diseño muestral fue bietápico, donde en la primera etapa se usó un diseño de muestreo estratificado y en la segunda etapa un diseño de muestreo por conglomerados.

2. Identifique las variables asociadas al diseño de muestreo que están presentes en la base de datos a usar (THOGAR en <https://www.inegi.org.mx/programas/envi/2020/Microdatos>)

```
library(tidyverse)

## Warning in system("timedatectl", intern = TRUE): running command 'timedatectl'
## had status 1

## -- Attaching packages ----- tidyverse 1.3.1
##
## v ggplot2 3.3.5      v purrr 0.3.4
## v tibble 3.1.3      v dplyr 1.0.7
## v tidyr 1.1.3       v stringr 1.4.0
## v readr 2.0.0       v forcats 0.5.1

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

thogar <- data.table::fread("../csv/THOGAR.csv")

#P3A1_1 es la variable a la que nos interesa replicar la estimación
```

```
# primero observamos que no hay N.A. por lo cual no es necesario hacer correcciones
thogar2 <- thogar %>% drop_na(P3A1_1)
length(thogar2$P3A1_1) == length(thogar$P3A1_1)

## [1] TRUE

# summary de las variables asociada al diseño de muestreo
summary(thogar[, c("UPM_DIS", "EST_DIS", "FACTOR")])

##      UPM_DIS      EST_DIS      FACTOR
##  Min.   : 1    Min.   : 1.0    Min.   : 4
## 1st Qu.:2254  1st Qu.:135.0  1st Qu.: 254
##  Median :4292  Median :268.0  Median : 473
##   Mean  :4307   Mean  :267.7   Mean   : 640
## 3rd Qu.:6329  3rd Qu.:397.0  3rd Qu.: 788
##   Max.  :8301   Max.   :552.0   Max.   :8610
```

- Identifique la pregunta y variable asociada a la identificación de los Hogares con necesidad de rentar, comprar o construir una vivienda independiente de la que habitan.

```
#los que si tienen necesidad
# segun la estructura de archivo

# el número 1 corresponde a si tiene necesidad
a <- sum(thogar$P3A1_1 == 1)

# el número 2 corresponde a si no tiene necesidad
b <- sum(thogar$P3A1_1 == 2)

# el número 9 corresponde a que no sabe
c <- sum(thogar$P3A1_1 == 9)

sum(a + b + c) == length(thogar$P3A1_1)

## [1] TRUE

#los que tienen necesidad
thogar$si <- as.numeric(thogar$P3A1_1 == 1)
# los que no tienen necesidad
thogar$no <- as.numeric(thogar$P3A1_1 == 2)
# los que no especificaran
thogar$ne <- as.numeric(thogar$P3A1_1 == 9)
# para calcular el total
thogar$total <- 1
```

4. Con esta información, estime el número total de hogares y el porcentaje de hogares que tienen una necesidad de vivienda a nivel nacional y por entidad federativa.

```
library(survey)

## Loading required package: grid
## Loading required package: Matrix
##
## Attaching package: 'Matrix'
## The following objects are masked from 'package:tidyr':
##
##     expand, pack, unpack
## Loading required package: survival
##
## Attaching package: 'survey'
## The following object is masked from 'package:graphics':
##
##     dotchart

options(survey.lonely.psu="adjust")

#definimos el diseño
dsg.envi <- svydesign(id=~UPM_DIS, strat=~EST_DIS, weight=~FACTOR ,
                    data = thogar, nest=T)

## por nivel nacional

# porcentajes
svymean(~si + no + ne, dsg.envi)*100

##           mean      SE
## si 21.06728 0.0026
## no 78.78794 0.0026
## ne  0.14478 0.0002

# totales
svytotal(~si + no + ne, dsg.envi)

##           total      SE
## si  7628562 105958.2
## no 28529481 220719.5
## ne   52424   7310.7

# estimación del total de hogares
svytotal(~total, dsg.envi)
```

```
##          total      SE
## total 36210467 250261

rel.nac <- svymean(~si + no + ne, dsg.envi)*100
abs.nac <- svytotal(~si + no + ne, dsg.envi)
total.nac <- svytotal(~total, dsg.envi)

# iniciales de las entidades federativas
entidades=c("AGU", "BCN", "BCS", "CAM", "COA", "COL", "CHP", "CHH", "CMX", "DUR", "GUA", "GRO",
            "OAX", "PUE", "QUE", "ROO", "SLP", "SIN", "SON", "TAB", "TAM", "TLA", "VER", "YUC",

## por entidadad

# porcentajes
cbind(entidades, svyby(~si +no +ne,~ENT,design=dsg.envi, svymean)*100)

##      entidades  ENT      si      no      ne      se.si      se.no      se.ne
## 1      AGU  100 15.46245 84.26830 0.26925107 0.9173244 0.9131693 0.11946444
## 2      BCN  200 24.37978 75.40743 0.21279208 1.4942212 1.5010807 0.10776649
## 3      BCS  300 26.07039 73.92961 0.00000000 1.5627556 1.5627556 0.00000000
## 4      CAM  400 25.64717 74.28959 0.06324075 1.3411186 1.3407852 0.04930195
## 5      COA  500 17.94490 81.85348 0.20162681 1.0390496 1.0544508 0.10231157
## 6      COL  600 17.20948 82.64368 0.14683786 1.1218974 1.1228393 0.08601451
## 7      CHP  700 27.45335 72.41236 0.13428122 1.2105918 1.2190244 0.09621427
## 8      CHH  800 15.73845 84.15577 0.10577981 0.9798720 0.9821714 0.07688229
## 9      CMX  900 27.02795 72.85239 0.11966595 1.2233668 1.2256745 0.08471559
## 10     DUR 1000 19.37306 80.52658 0.10036320 1.1069716 1.1087669 0.07101579
## 11     GUA 1100 19.78255 79.94890 0.26855012 1.0433201 1.0429585 0.12197047
## 12     GRO 1200 31.28108 68.65755 0.06137266 1.4375165 1.4402630 0.06107272
## 13     HID 1300 16.02284 83.88643 0.09072945 0.9930539 0.9921498 0.06442120
## 14     JAL 1400 17.84720 82.03829 0.11450993 0.9626518 0.9609448 0.08085658
## 15     MEX 1500 18.40064 81.43798 0.16137683 1.0401217 1.0441224 0.08187207
## 16     MIC 1600 19.25450 80.56132 0.18417759 1.0366917 1.0617585 0.10077648
## 17     MOR 1700 18.36703 81.57169 0.06128348 1.1521323 1.1520267 0.06128005
## 18     NAY 1800 22.31348 77.63581 0.05071494 1.2116094 1.2183039 0.05081436
## 19     NLE 1900 11.59336 88.28131 0.12532852 1.0141366 1.0150155 0.07277455
## 20     OAX 2000 25.33217 74.66783 0.00000000 1.3738244 1.3738244 0.00000000
## 21     PUE 2100 23.92025 76.03424 0.04551170 1.4095638 1.4076676 0.04565093
## 22     QUE 2200 20.72767 79.07285 0.19948402 1.3677842 1.3558212 0.13031415
## 23     ROO 2300 24.49634 75.24793 0.25573361 1.5660673 1.5736470 0.11796421
## 24     SLP 2400 15.63750 84.25187 0.11063611 1.0016747 0.9989220 0.07890184
## 25     SIN 2500 25.57488 74.12816 0.29695910 1.0507984 1.0690384 0.13581520
## 26     SON 2600 25.20097 74.68874 0.11029266 1.0695561 1.0708116 0.07820734
## 27     TAB 2700 27.70826 72.16813 0.12360957 1.5293480 1.5338243 0.08754413
## 28     TAM 2800 15.68869 84.14082 0.17049428 1.0334478 1.0310602 0.09912332
## 29     TLA 2900 27.05889 72.76868 0.17243014 1.3022474 1.2957662 0.09959357
```

```
## 30      VER 3000 22.32074 77.44769 0.23156936 1.1029240 1.1054093 0.12233690
## 31      YUC 3100 23.72954 76.10457 0.16588357 1.2799885 1.2794174 0.09429167
## 32      ZAC 3200 13.84373 86.15627 0.00000000 0.8878610 0.8878610 0.00000000

# totales
cbind(entidades, svyby(~si + no + ne,~ENT,design=dsg.envi, svytotal))

##      entidades ENT      si      no      ne      se.si      se.no      se.ne
## 1          AGU   1  61505  335194 1071  3974.823  8564.767  475.7383
## 2          BCN   2 281959  872108 2461 19337.419 27919.457 1244.3163
## 3          BCS   3  64373  182547   0  5173.875  6449.383   0.0000
## 4          CAM   4  67321  195002  166  4445.643  8161.858  129.0969
## 5          COA   5 163939  747788 1842 10105.564 19628.362  933.9657
## 6          COL   6  40317  193611  344  3021.593  7590.486  201.1989
## 7          CHP   7 400920 1057487 1961 24523.241 51431.143 1401.0014
## 8          CHH   8 180625  965828 1214 11366.583 22561.522  882.9100
## 9          CMX   9 759121 2046170 3361 39095.924 48555.506 2378.1255
## 10         DUR  10  98252  408397  509  5787.977 12180.408  359.9236
## 11         GUA  11 329132 1330149 4468 18324.932 40831.520 2044.6736
## 12         GRO  12 303266  665626  595 16281.615 24941.543  595.0000
## 13         HID  13 140927  737813  798  9331.490 23459.519  566.4380
## 14         JAL  14 425646 1956569 2731 24713.322 55730.576 1931.9578
## 15         MEX  15 883449 3909988 7748 56765.649 136796.664 3909.3186
## 16         MIC  16 258117 1079968 2469 17019.088 41651.860 1348.1428
## 17         MOR  17 109093  484504  364  6113.354 20808.299  364.0000
## 18         NAY  18  81396  283203  185  5316.975  8445.250  185.0000
## 19         NLE  19 197403 1503188 2134 17956.984 45934.193 1237.1960
## 20         OAX  20 293325  864590   0 21077.694 42503.529   0.0000
## 21         PUE  21 425198 1351558  809 28040.784 54736.050  809.0000
## 22         QUE  22 141001  537897 1357 11145.868 28477.477  894.6390
## 23         ROO  23 138127  424299 1442  9283.650 20497.536  663.2074
## 24         SLP  24 123674  666332  875  8709.998 14608.143  623.5006
## 25         SIN  25 222713  645528 2586  9220.054 19008.975 1185.5168
## 26         SON  26 222551  659580  974 10768.714 22849.423  689.8594
## 27         TAB  27 192553  501518  859 12898.737 20890.775  613.4012
## 28         TAM  28 166002  890294 1804 12338.637 18993.141 1053.2303
## 29         TLA  29  94156  253211  600  5211.782  8215.155  346.0028
## 30         VER  30 536212 1860529 5563 31913.895 62997.584 2940.6362
## 31         YUC  31 162218  520260 1134  9319.293 21188.653  641.7367
## 32         ZAC  32  64071  398745   0  4393.677 15235.175   0.0000

# estimación del total de hogares
cbind(entidades, svyby(~total, ~ENT,design=dsg.envi, svytotal))

##      entidades ENT      total      se
```



```
## 1      AGU      1  397770    9362.560
## 2      BCN      2 1156528   30667.901
## 3      BCS      3  246920    9038.065
## 4      CAM      4  262489   10080.177
## 5      COA      5  913569   20581.754
## 6      COL      6  234272    8612.212
## 7      CHP      7 1460368   65259.754
## 8      CHH      8 1147667   21729.675
## 9      CMX      9 2808652   53774.155
## 10     DUR     10  507158   12451.339
## 11     GUA     11 1663749   43662.953
## 12     GRO     12  969487   29227.596
## 13     HID     13  879538   25147.754
## 14     JAL     14 2384946   60219.557
## 15     MEX     15 4801185  154102.192
## 16     MIC     16 1340554   48980.808
## 17     MOR     17  593961   20541.243
## 18     NAY     18  364784   10262.249
## 19     NLE     19 1702725   47509.691
## 20     OAX     20 1157915   53248.154
## 21     PUE     21 1777565   61405.772
## 22     QUE     22  680255   33271.499
## 23     ROO     23  563868   22050.122
## 24     SLP     24  790881   16199.544
## 25     SIN     25  870827   19436.071
## 26     SON     26  883105   26162.195
## 27     TAB     27  694930   25398.594
## 28     TAM     28 1058100   22381.890
## 29     TLA     29  347967    9438.489
## 30     VER     30 2402304   75073.973
## 31     YUC     31  683612   23033.343
## 32     ZAC     32  462816   16348.793
```

```
rel.ent <- svyby(~si +no +ne,~ENT,design=dsg.envi, svymean)
abs.ent <- svyby(~si + no + ne,~ENT,design=dsg.envi, svyttotal)
total.ent <- svyby(~total, ~ENT,design=dsg.envi, svyttotal)
```

5. Calcule intervalos de confianza para los parámetros estimados en el inciso anterior. Comente sobre los resultados obtenidos.

```
## a nivel nacional
```

```
# porcentajes
confint(rel.nac)
```

```
##          2.5 %          97.5 %
```

```
## si 21.0622346 21.0723338
## no 78.7828765 78.7930034
## ne 0.1443797 0.1451719

# totales
confint(abs.nac)

##          2.5 %          97.5 %
## si  7420887.68  7836236.32
## no 28096878.76 28962083.24
## ne   38095.29   66752.71

# estimación del total de hogares
confint(total.nac)

##          2.5 %          97.5 %
## total 35719965 36700969

## a nivel entidad

# porcentajes
cbind(rep(entidades, 3), confint(rel.ent)*100)

##          2.5 %          97.5 %
## 1:si  "AGU" "13.6645304687363" "17.2603758841812"
## 2:si  "BCN" "21.4511618313169" "27.3084012748076"
## 3:si  "BCS" "23.007442483402" "29.1333318564652"
## 4:si  "CAM" "23.0186257529112" "28.2757142003821"
## 5:si  "COA" "15.9083955092802" "19.9813949717891"
## 6:si  "COL" "15.0106045342917" "19.4083614539612"
## 7:si  "CHP" "25.0806379645452" "29.8260704815452"
## 8:si  "CHH" "13.8179363592096" "17.6589641702994"
## 9:si  "CMX" "24.6301930933702" "29.4257027598718"
## 10:si "DUR" "17.203430866432" "21.5426798130801"
## 11:si "GUA" "17.7376815535214" "21.8274211152105"
## 12:si "GRO" "28.4635989689857" "34.0985601935405"
## 13:si "HID" "14.0764939373289" "17.9691936850365"
## 14:si "JAL" "15.960433618833" "19.733959461765"
## 15:si "MEX" "16.3620438390808" "20.439245842529"
## 16:si "MIC" "17.2226237738248" "21.2863807123801"
## 17:si "MOR" "16.1088929925161" "20.6251686714652"
## 18:si "NAY" "19.938767795127" "24.6881895275626"
## 19:si "NLE" "9.60568639767547" "13.5810290143846"
## 20:si "OAX" "22.6395238844751" "28.0248167622909"
## 21:si "PUE" "21.1575562001109" "26.6829447098418"
```

## 22:si	"QUE"	"18.0468604896319"	"23.4084761120836"
## 23:si	"ROO"	"21.4269004554285"	"27.5657715883832"
## 24:si	"SLP"	"13.6742515940386"	"17.6007442650097"
## 25:si	"SIN"	"23.5153575584586"	"27.634411569049"
## 26:si	"SON"	"23.1046755332606"	"27.2972585515878"
## 27:si	"TAB"	"24.7107914525577"	"30.7057253189157"
## 28:si	"TAM"	"13.6631667525877"	"17.7142077866808"
## 29:si	"TLA"	"24.5065297648979"	"29.611245771288"
## 30:si	"VER"	"20.1590474464417"	"24.4824300684773"
## 31:si	"YUC"	"21.220811091825"	"26.2382738862035"
## 32:si	"ZAC"	"12.1035550404797"	"15.5839060671744"
## 1:no	"AGU"	"82.4785168116596"	"86.0580746859395"
## 2:no	"BCN"	"72.4653622203033"	"78.349490511321"
## 3:no	"BCS"	"70.8666681435348"	"76.992557516598"
## 4:no	"CAM"	"71.6616986717718"	"76.917479884282"
## 5:no	"COA"	"79.7867924268391"	"83.920163479064"
## 6:no	"COL"	"80.4429546397707"	"84.8444036446167"
## 7:no	"CHP"	"70.0231206445642"	"74.801608464811"
## 8:no	"CHH"	"82.2307493494221"	"86.0807904961951"
## 9:no	"CMX"	"70.4501082474268"	"75.2546640063092"
## 10:no	"DUR"	"78.3534383454412"	"82.6997245742012"
## 11:no	"GUA"	"77.9047375200843"	"81.9930595634902"
## 12:no	"GRO"	"65.8346842291058"	"71.4804112801687"
## 13:no	"HID"	"81.9418487715777"	"85.8310047037696"
## 14:no	"JAL"	"80.1548764154264"	"83.9217106435678"
## 15:no	"MEX"	"79.3915360012164"	"83.4844206636486"
## 16:no	"MIC"	"78.4803116884009"	"82.6423286528312"
## 17:no	"MOR"	"79.3137549020235"	"83.8296164641099"
## 18:no	"NAY"	"75.2479745788147"	"80.02363821122"
## 19:no	"NLE"	"86.2919200282269"	"90.2707075246663"
## 20:no	"OAX"	"71.9751832377091"	"77.3604761155249"
## 21:no	"PUE"	"73.2752601144671"	"78.7932155812177"
## 22:no	"QUE"	"76.4154870150386"	"81.7302083491999"
## 23:no	"ROO"	"72.1636388306108"	"78.3322219027523"
## 24:no	"SLP"	"82.2940147462205"	"86.209717167936"
## 25:no	"SIN"	"72.0328795344038"	"76.2234331430855"
## 26:no	"SON"	"72.5899880725554"	"76.787492521485"
## 27:no	"TAB"	"69.1618916606441"	"75.1743724235082"
## 28:no	"TAM"	"82.1199776021084"	"86.1616592942152"
## 29:no	"TLA"	"70.2290269490422"	"75.3083372262963"
## 30:no	"VER"	"75.2811295013883"	"79.6142542635307"
## 31:no	"YUC"	"73.5969619705195"	"78.6121859086868"
## 32:no	"ZAC"	"84.4160939328256"	"87.8964449595203"
## 1:ne	"AGU"	"0.0351050749785979"	"0.503397074504772"
## 2:ne	"BCN"	"0.00157363959715382"	"0.424010522653998"
## 3:ne	"BCS"	"0"	"0"

```
## 4:ne "CAM" "-0.0333893082706353" "0.159870798923577"
## 5:ne "COA" "0.00109981129865853" "0.402153801728929"
## 6:ne "COL" "-0.0217474845185062" "0.315423211878157"
## 7:ne "CHP" "-0.0542952904722583" "0.322857735006787"
## 8:ne "CHH" "-0.0449067039340671" "0.256466328807833"
## 9:ne "CMX" "-0.0463735604346287" "0.285705453456619"
## 10:ne "DUR" "-0.0388251942913052" "0.239551595136801"
## 11:ne "GUA" "0.0294923973348633" "0.507607850358749"
## 12:ne "GRO" "-0.0583276717889913" "0.181072999988338"
## 13:ne "HID" "-0.035533788561" "0.216992690848337"
## 14:ne "JAL" "-0.0439660637183478" "0.272985924126089"
## 15:ne "MEX" "0.000910525900165967" "0.321843127624953"
## 16:ne "MIC" "-0.0133406810741709" "0.381695853637156"
## 17:ne "MOR" "-0.0588232029018883" "0.181390172787116"
## 18:ne "NAY" "-0.0488793743964173" "0.150309261672175"
## 19:ne "NLE" "-0.0173069790507099" "0.267964014097473"
## 20:ne "OAX" "0" "0"
## 21:ne "PUE" "-0.0439624738904543" "0.134985868252967"
## 22:ne "QUE" "-0.05592703003762" "0.454895064083676"
## 23:ne "ROO" "0.0245280038728139" "0.486939218952386"
## 24:ne "SLP" "-0.0440086465626094" "0.265280873357791"
## 25:ne "SIN" "0.030766198219433" "0.563151996783708"
## 26:ne "SON" "-0.0429909014924246" "0.263576222603731"
## 27:ne "TAB" "-0.0479737646711182" "0.295192909045372"
## 28:ne "TAM" "-0.0237838546026372" "0.364772419010538"
## 29:ne "TLA" "-0.0227696606981732" "0.367629949173804"
## 30:ne "VER" "-0.00820655986797079" "0.471345280029949"
## 31:ne "YUC" "-0.0189246984601138" "0.350691841225308"
## 32:ne "ZAC" "0" "0"
```

totales

```
cbind(rep(entidades, 3),confint(abs.ent))
```

```
##          2.5 %          97.5 %
## 1:si "AGU" "53714.4893133055" "69295.5106866945"
## 2:si "BCN" "244058.354561328" "319859.645438672"
## 3:si "BCS" "54232.3918446551" "74513.6081553449"
## 4:si "CAM" "58607.6994862466" "76034.3005137534"
## 5:si "COA" "144132.457635813" "183745.542364187"
## 6:si "COL" "34394.7859584105" "46239.2140415895"
## 7:si "CHP" "352855.331248212" "448984.668751788"
## 8:si "CHH" "158346.906255903" "202903.093744097"
## 9:si "CMX" "682494.396170988" "835747.603829012"
## 10:si "DUR" "86907.773291305" "109596.226708695"
## 11:si "GUA" "293215.79380533" "365048.20619467"
## 12:si "GRO" "271354.621651472" "335177.378348528"
```

## 13:si	"HID"	"122637.615999675"	"159216.384000325"
## 14:si	"JAL"	"377208.778718661"	"474083.221281339"
## 15:si	"MEX"	"772190.371675435"	"994707.628324565"
## 16:si	"MIC"	"224760.200074885"	"291473.799925115"
## 17:si	"MOR"	"97111.047088583"	"121074.952911417"
## 18:si	"NAY"	"70974.9202183925"	"91817.0797816075"
## 19:si	"NLE"	"162207.957702892"	"232598.042297108"
## 20:si	"OAX"	"252013.478632569"	"334636.521367431"
## 21:si	"PUE"	"370239.072626055"	"480156.927373945"
## 22:si	"QUE"	"119155.4996067"	"162846.5003933"
## 23:si	"ROO"	"119931.379552437"	"156322.620447563"
## 24:si	"SLP"	"106602.716646523"	"140745.283353477"
## 25:si	"SIN"	"204642.02631857"	"240783.97368143"
## 26:si	"SON"	"201444.70925655"	"243657.29074345"
## 27:si	"TAB"	"167271.93927489"	"217834.06072511"
## 28:si	"TAM"	"141818.715755256"	"190185.284244744"
## 29:si	"TLA"	"83941.0945151171"	"104370.905484883"
## 30:si	"VER"	"473661.914512666"	"598762.085487334"
## 31:si	"YUC"	"143952.520664229"	"180483.479335771"
## 32:si	"ZAC"	"55459.5509611246"	"72682.4490388754"
## 1:no	"AGU"	"318407.366089602"	"351980.633910398"
## 2:no	"BCN"	"817386.868844379"	"926829.131155621"
## 3:no	"BCS"	"169906.44156183"	"195187.55843817"
## 4:no	"CAM"	"179005.051302798"	"210998.948697202"
## 5:no	"COA"	"709317.11758599"	"786258.88241401"
## 6:no	"COL"	"178733.920234704"	"208488.079765296"
## 7:no	"CHP"	"956683.811367403"	"1158290.1886326"
## 8:no	"CHH"	"921608.228647545"	"1010047.77135245"
## 9:no	"CMX"	"1951002.95627387"	"2141337.04372613"
## 10:no	"DUR"	"384523.838853772"	"432270.161146228"
## 11:no	"GUA"	"1250120.69195362"	"1410177.30804638"
## 12:no	"GRO"	"616741.473464492"	"714510.526535508"
## 13:no	"HID"	"691833.186995221"	"783792.813004779"
## 14:no	"JAL"	"1847339.07918145"	"2065798.92081855"
## 15:no	"MEX"	"3641871.46438956"	"4178104.53561044"
## 16:no	"MIC"	"998331.854830787"	"1161604.14516921"
## 17:no	"MOR"	"443720.4826963"	"525287.5173037"
## 18:no	"NAY"	"266650.614629701"	"299755.385370299"
## 19:no	"NLE"	"1413158.63614447"	"1593217.36385553"
## 20:no	"OAX"	"781284.613055556"	"947895.386944444"
## 21:no	"PUE"	"1244277.31248104"	"1458838.68751896"
## 22:no	"QUE"	"482082.17015211"	"593711.82984789"
## 23:no	"ROO"	"384124.567163538"	"464473.432836462"
## 24:no	"SLP"	"637700.565060547"	"694963.434939453"
## 25:no	"SIN"	"608271.093385722"	"682784.906614278"
## 26:no	"SON"	"614795.954108934"	"704364.045891066"

```
## 27:no "TAB" "460572.83436683" "542463.16563317"
## 28:no "TAM" "853068.128374293" "927519.871625707"
## 29:no "TLA" "237109.592778119" "269312.407221881"
## 30:no "VER" "1737056.00430001" "1984001.99569999"
## 31:no "YUC" "478731.002490147" "561788.997509853"
## 32:no "ZAC" "368884.605581362" "428605.394418638"
## 1:ne "AGU" "138.569979138354" "2003.43002086165"
## 2:ne "BCN" "22.1849095034713" "4899.81509049653"
## 3:ne "BCS" "0" "0"
## 4:ne "CAM" "-87.0252017204774" "419.025201720477"
## 5:ne "COA" "11.4607904715842" "3672.53920952842"
## 6:ne "COL" "-50.3426105818632" "738.342610581863"
## 7:ne "CHP" "-784.912340289206" "4706.91234028921"
## 8:ne "CHH" "-516.471726002968" "2944.47172600297"
## 9:ne "CMX" "-1300.04037449988" "8022.04037449988"
## 10:ne "DUR" "-196.437299075418" "1214.43729907542"
## 11:ne "GUA" "460.513446424591" "8475.48655357541"
## 12:ne "GRO" "-571.178570801331" "1761.17857080133"
## 13:ne "HID" "-312.198065904166" "1908.19806590417"
## 14:ne "JAL" "-1055.56773758859" "6517.56773758859"
## 15:ne "MEX" "85.8763228136422" "15410.1236771864"
## 16:ne "MIC" "-173.31132733532" "5111.31132733532"
## 17:ne "MOR" "-349.426890372579" "1077.42689037258"
## 18:ne "NAY" "-177.59333713991" "547.59333713991"
## 19:ne "NLE" "-290.859647429259" "4558.85964742926"
## 20:ne "OAX" "0" "0"
## 21:ne "PUE" "-776.610863492904" "2394.6108634929"
## 22:ne "QUE" "-396.460284536932" "3110.46028453693"
## 23:ne "ROO" "142.137463602637" "2741.86253639736"
## 24:ne "SLP" "-347.038723167684" "2097.03872316768"
## 25:ne "SIN" "262.429837610975" "4909.57016238903"
## 26:ne "SON" "-378.099590089893" "2326.09959008989"
## 27:ne "TAB" "-343.244208691891" "2061.24420869189"
## 28:ne "TAM" "-260.293395097477" "3868.29339509748"
## 29:ne "TLA" "-78.1530633965975" "1278.1530633966"
## 30:ne "VER" "-200.540956738309" "11326.5409567383"
## 31:ne "YUC" "-123.780831580438" "2391.78083158044"
## 32:ne "ZAC" "0" "0"
```

```
# estimación del total de hogares
cbind(entidades, confint(total.ent))
```

```
## entidades 2.5 % 97.5 %
## 1 "AGU" "379419.718756408" "416120.281243592"
## 2 "BCN" "1096420.01936802" "1216635.98063198"
## 3 "BCS" "229205.718782509" "264634.281217491"
```

## 4	"CAM"	"242732.215211245"	"282245.784788755"
## 5	"COA"	"873229.504324511"	"953908.495675489"
## 6	"COL"	"217392.37441018"	"251151.62558982"
## 7	"CHP"	"1332461.23163123"	"1588274.76836877"
## 8	"CHH"	"1105077.61981763"	"1190256.38018237"
## 9	"CMX"	"2703256.59244916"	"2914047.40755084"
## 10	"DUR"	"482753.823579244"	"531562.176420756"
## 11	"GUA"	"1578171.18460728"	"1749326.81539272"
## 12	"GRO"	"912201.965219499"	"1026772.0347805"
## 13	"HID"	"830249.308713981"	"928826.691286019"
## 14	"JAL"	"2266917.83666666"	"2502974.16333334"
## 15	"MEX"	"4499150.25416336"	"5103219.74583664"
## 16	"MIC"	"1244553.38086974"	"1436554.61913026"
## 17	"MOR"	"553700.902672852"	"634221.097327148"
## 18	"NAY"	"344670.362486969"	"384897.637513031"
## 19	"NLE"	"1609607.71646561"	"1795842.28353439"
## 20	"OAX"	"1053550.53500965"	"1262279.46499035"
## 21	"PUE"	"1657211.89759173"	"1897918.10240827"
## 22	"QUE"	"615044.059840002"	"745465.940159998"
## 23	"ROO"	"520650.555779677"	"607085.444220323"
## 24	"SLP"	"759130.476625839"	"822631.523374161"
## 25	"SIN"	"832733.000846209"	"908920.999153791"
## 26	"SON"	"831828.040677376"	"934381.959322624"
## 27	"TAB"	"645149.670200613"	"744710.329799387"
## 28	"TAM"	"1014232.30176307"	"1101967.69823693"
## 29	"TLA"	"329467.901020114"	"366466.098979886"
## 30	"VER"	"2255161.71615166"	"2549446.28384834"
## 31	"YUC"	"638467.477308857"	"728756.522691143"
## 32	"ZAC"	"430772.954668468"	"494859.045331532"

6. ****Punto extra opcional****. Considerando el porcentaje de hogares que tienen una necesidad de vivienda por entidad federativa, realice un mapa de calor (Geographic Heat Map) y comente los resultados.