



Muestras complejas y precisión de inferencia estadística

Sesión N° 7

04 octubre 2021

Análisis de datos estadísticos en R

Profesora Valentina Andrade de la Horra
Ayudantes Dafne Jaime y Nicolás Godoy

Contenidos Sesión 7



Diseño muestral complejo

El diseño muestral de CASEN 2020

Creación de objetos encuesta con srvyr

Estimación e inferencia

Creación de tabulado con las estimaciones



Diseño muestral complejo

¿Qué es?



- Cuando trabajamos con encuestas, obtenemos información de sólo un grupo de la población objetivo: es decir, escogemos una muestra.
- La mayoría de las veces, esperamos que la muestra pueda dar cuenta de determinadas características de la **población**.
 - Es decir, esperamos que sea **representativa**.
- Para ello, al diseñar la encuesta, establecemos ciertos criterios de selección de casos.
- El ideal es que la selección de casos se realice a partir de un **muestreo aleatorio simple**, en que *cada sujeto tiene la misma posibilidad de ser elegido*.

¿Qué es?



- Sin embargo, llevar a cabo un muestreo aleatorio simple suele no ser posible, dado lo costoso de su producción.
- Por ello, se han elaborado métodos más complejos de selección de casos en que, si bien no todos los sujetos tienen la misma probabilidad de ser elegidos, es posible **conocer** su probabilidad de ser escogidos para la muestra.

Es decir, se crean muestras probabilísticas

¿Qué es?



- Estos métodos de selección de casos suelen ser más complejos que el muestreo aleatorio simple, pues van más allá de una selección aleatoria de casos entre todos los individuos que componen a la población. Hay distintos tipos
 - Estratificado
 - Por conglomerados
 - Bietápico, multietápico
 - Entre otros...

¿Por qué emplear diseños muestrales?



Permite estimaciones a nivel poblacional

Posibilita mejorar la precisión de nuestras estimaciones

Permite trabajar con un nivel de error conocido



El diseño muestral de CASEN 2020

El diseño muestral de CASEN 2020



Según el **manual metodológico de la encuesta**, su diseño muestral es

Probabilístico

Estratificado

Multietápico

¿Qué significa esto?



Probabilístico

- Conocemos la probabilidad de selección de cada sujeto (aunque sea $\neq 1$)

¿Qué significa esto?



Estratificado

- Se establece un criterio para definir estratos (en este caso, comuna/zona), y se escogen aleatoriamente unidades más pequeñas (conglomerados), a partir de las cuales se levanta el muestreo.
- En el caso de CASEN, los conglomerados están constituidos por las **manzanas**, *unidades primarias de muestreo (UPM)*.

¿Qué significa esto?



Multietápico

- La estratificación se realiza en distintos niveles
 - Primero, se seleccionan aleatoriamente manzanas de cada estrato comuna/zona
 - Luego, se escogen al azar un número de viviendas de cada manzana
 - De cada vivienda, se escoge al azar un hogar
 - De cada hogar, responde (idealmente) el o la jefa de hogar, u otro adulto/a presente

¿Qué significa esto?



De este modo, CASEN 2020

- Es representativa a nivel nacional
- Presenta un error muestral de
 - A nivel nacional, 0.4 puntos porcentuales (pp.) de error absoluto y 3.9% de error relativo
 - A nivel regional, un error absoluto promedio de 1.6 pp. (con un máximo de 2.1 pp. para Coquimbo) y un error relativo promedio de 15.4% (con un máximo de 30.4% para Magallanes)
- Para más información, revisar el **manual metodológico de la encuesta.**

Recursos de la práctica

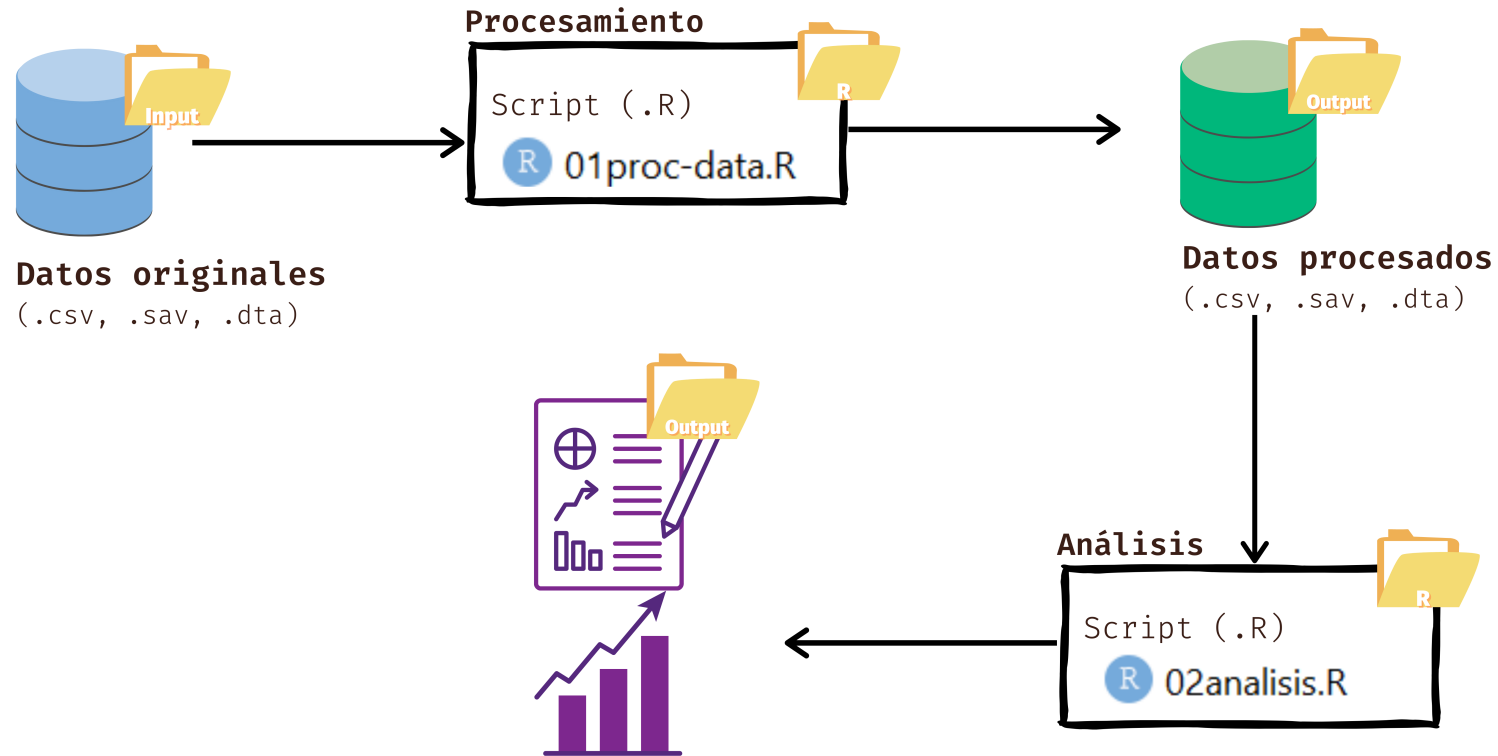


- Este práctico fue trabajado con datos de **CASEN 2020**.
- Los datos ya fueron procesados anteriormente, para centrar el trabajo en el cálculo de parámetros.
- Al trabajar con muestras complejas, es **fundamental** asegurarnos de:
 - Recodificar correctamente los valores de cada variable
 - Eliminar los valores nulos de los datos
 - Transformar cada variable a su datatype correspondiente
- Pueden revisar el *script de procesamiento* en la carpeta **R**



1: Flujo del RProject

Etapas del flujo

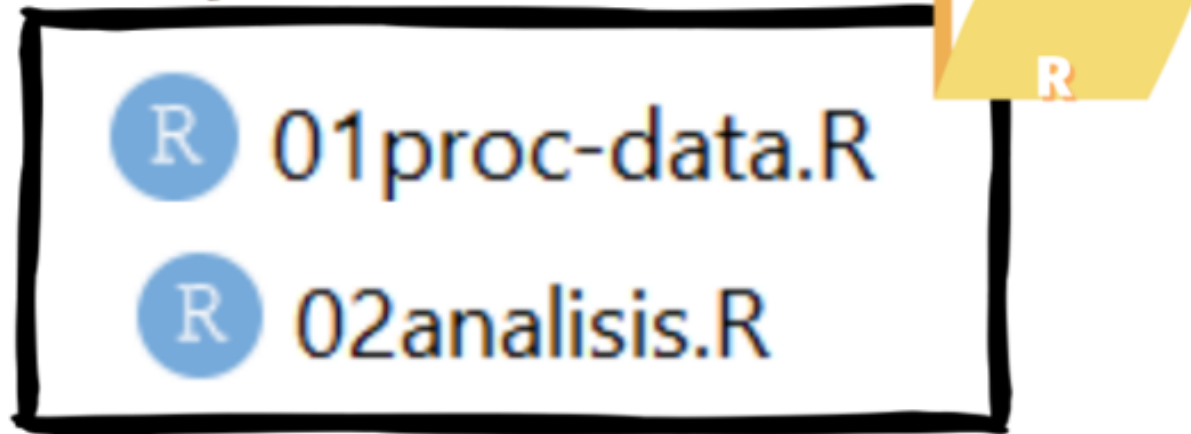


- Hoy nosotras/os nos centraremos en la parte de **análisis**.

Orden de un script de análisis



Script



- # 1. Cargar librerías -----
- # 2. Cargar datos -----
- # 3. Explorar los datos -----
- # 4. Calcular lo necesario según el análisis -----
- # 5. Crear de tablas, gráficos y/o modelos (según sea necesario) -----
- # 6. Exportar gráficos, tablas y modelos -----



Figura 1: Estudiantes de Análisis de datos en R haciendo los **pasos 4 y 5.**



Paso 1: Cargar paquetes

Paso 1: Cargar paquetes



```
pacman::p_load(tidyverse, #Universo de paquetes  
               sjmisc, #Para explorar datos  
               srvyr, #Para trabajar con muestras complejas  
               dplyr, #Para manipular datos  
               tidyr) #Para transformar la estructura de los d
```



Paso 2: Importar datos

Consideraciones antes de importar datos



Para **importar** los datos en R debemos tener en consideración tres cosas:

1. 2. 3.

Consideraciones antes de importar datos



Para **importar** los datos en R debemos tener en consideración tres cosas:

1. Cómo se llaman los datos (en nuestro caso `casen_proc`);
2. El formato de nuestros datos (en nuestro caso `.rds`); y
3. El lugar de donde están alojados nuestros datos (en este caso, desde GitHub).

Paso 2: Importar datos



```
data <- readRDS(url("https://github.com/learn-R/07-class/blob/"))
```




Como resultado

Nuevo objeto en el Enviroment



Paso 3: Explorar datos

Explorar datos



¡Recordemos!

Para variables categóricas: `frq()`

Para variables continuas: `descr()`

En la tarea de explorar los datos, la librería `sjmisc` será nuestra mejor aliada

Explorar datos para procesar



```
## Variables categóricas
```

```
frq(data$region) #Examinamos la columna región
```

```
frq(data$pobreza) #Examinamos la columna pobreza
```

```
frq(data$sexo) #Examinamos la columna sexo
```

Explorar datos para procesar



```
## Variables continuas
```


```
descr(data$exp) #Ponderador regional
```

```
sum(data$exp) #Total de la población
```

```
descr(data$varstrat) #Estrato de varianza
```

```
descr(data$varunit) #Conglomerado de varianza
```

```
descr(data$ing_tot_hog) #Ingreso total del hogar
```

¡Continuemos con la  creación del objeto encuesta!



Pero antes...

¿Qué es un objeto encuesta?



Es una lista creada con la función `as_survey_design` de `srvyr`

En este caso, la lista contiene 9 elementos diferentes

Si bien su contenido es algo críptico, crear el objeto encuesta es crucial, pues nos permitirá trabajar con los datos como si fuese un dataframe

Creando el objeto encuesta



```
obj_encuesta <- data %>% #Creamos un nuevo objeto encuesta con  
  as_survey_design(ids = conglomerado, #Aplicamos diseño muestr  
    strata = estrato, #strat con los estratos de  
    fpc = nestrato, #especificando que la estim  
    weights = ponderador) #y los ponderadores c
```

Objeto encuesta



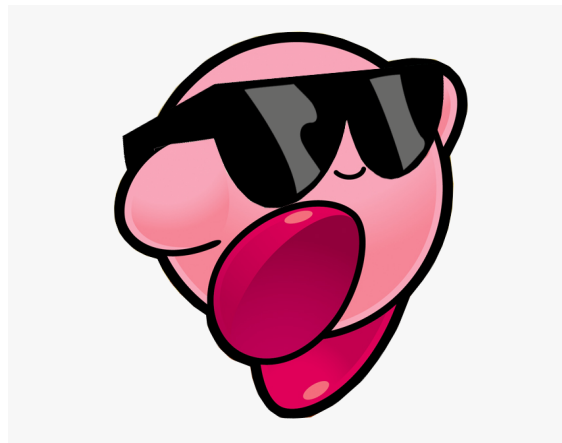
¿Por qué trabajar con objetos encuesta?



Los distintos argumentos especifican elementos del diseño muestral

Así, podremos realizar estimaciones mucho más precisas

Además de conocer el nivel de error de estas



Creando el objeto encuesta



En CASEN 2020 no viene incluida la variable fpc, por lo que debemos crearla

```
data <- data %>%  
  group_by(varstrat) %>% #Agrupando por varstrat  
  mutate(stratn = sum(exp)) %>% #Calculamos el total de person  
  ungroup() #¡No olvidemos desagrupar!
```

Creando el objeto encuesta (¡por fin!)



En CASEN 2020 no viene incluida la variable fpc, por lo que debemos crearla

```
casen_regional <- data %>% #Creamos un nuevo objeto llamado ca  
  as_survey_design(ids = varunit, #Aplicamos diseño muestral,  
    strata = varstrat, #los estratos a partir de  
    fpc = stratn, #especificando que la estimac  
    weights = exp) #y los ponderadores con exp
```

Las ventajas de `srvyr`



Existen otras librerías que nos permiten crear objetos encuesta

Sin embargo, nos quedamos con `srvyr` ¿por qué?

Nos permite dialogar con librerías conocidas, como `dplyr`

¡Sin considerar la simpleza de los cálculos!

¡A calcular!



¿Qué calcularemos?



- `srvyr` provee de muchas funciones para cálculos de diferentes estadísticos
- No obstante, aquí calcularemos **medias, proporciones y totales**

Son los estadísticos más usuales de reportar

además de ser **insesgados**

- Entonces, emplearemos `survey_mean()`, `survey_prop()` y `survey_total()`

Cálculo de medias `survey_mean()`



```
## Cálculo simple
casen_regional %>% #Con casen_regional
  summarize(ing_medio = srvyr::survey_mean(ing_tot_hog, na.rm=
```

Comparamos con el cálculo a nivel muestral



```
data %>% #Con data
  summarise(ing_medio = mean(ing_tot_hog, na.rm=T)) #Calculamo
```

Incorporamos Intervalos de Confianza al 95%

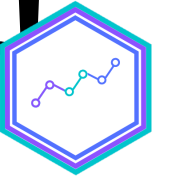


```
casen_regional %>%#Con casen_regional  
  summarise(ing_medio = survey_mean(ing_tot_hog, vartype = "ci
```

Agrupamos por sexo (¡como con dplyr!)

```
casen_regional %>% #Con casen_regional
  group_by(sexo) %>% #Agrupamos por sexo
  summarise(ing_medio = survey_mean(ing_tot_hog, vartype = "ci
```

¡Transformemos en wide con tidyrr!



```
ing_region <- casen_regional %>%  
  group_by(sexo) %>% #Agrupamos por region  
  summarise(ing_medio = survey_mean(ing_tot_hog, vartype = "ci"  
select(sexo, ing_medio) %>% #Seleccionamos region e ing_medio  
pivot_wider(names_from = "sexo", #Pivoteamos, extrayendo los  
              values_from = "ing_medio") #Y los valores desde
```

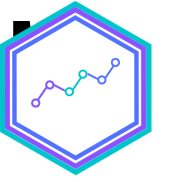
Cálculo de proporciones con `survey_prop()`



Una diferencia con `survey_mean()`: ¡Debemos agrupar por la(s) variable(s) categórica(s) de interés!

```
## Cálculo simple
casen_regional %>% #Con casen_regional
  group_by(pobreza) %>% #Agrupamos por pobreza
  summarise(prop = survey_prop(na.rm = T)) #Y calculamos las p
```

Transformando a porcentaje (%) con `mutate()`



```
## Transformando a porcentaje
casen_regional %>% #Con casen_regional
  group_by(pobreza) %>% #Agrupamos por pobreza
  summarise(prop = survey_prop(na.rm = T))%>% #Calculamos las
  mutate(per = prop*100) #Creamos una nueva columna multiplica
```

Incorporamos los totales con `survey_total()`



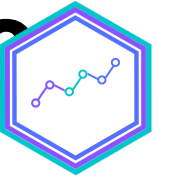
```
## Incorporamos cálculo de frecuencias
casen_regional %>% #Con casen_regional
  group_by(pobreza) %>% #Agrupamos por pobreza
  summarise(prop = survey_prop(na.rm = T), #Calculamos las pro
            total = survey_total(na.rm=T))%>% #Y el total por
  mutate(per = prop*100) #Creamos una nueva columna multiplica
```


Y los Intervalos de Confianza al 95%



```
## Con Intervalos de confianza al 95%
casen_regional %>% #Con casen_regional
  group_by(pobreza) %>% #Agrupamos por pobreza
  summarise(prop = survey_prop(vartype = "ci", na.rm = T)) #In
```

¡También podemos transformarlos en porcentajes!



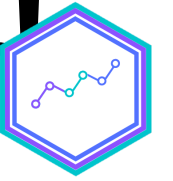
```
## Transformamos el estimador puntual y los límites del interv  
## Incorporamos el total  
casen_regional %>% #Con casen_regional  
  group_by(pobreza) %>% #Agrupamos por pobreza  
  summarise(prop = survey_prop(vartype = "ci", na.rm = T), #Ca  
            total = survey_total(vartype = "ci", na.rm=T)) %>%  
  mutate(prop = prop*100, #Multiplicamos las proporciones *100  
         prop_low = prop_low*100, #así como el límite inferior  
         prop_upp = prop_upp*100) #y superior, para obtener po
```

Cruzamos variables



```
## Cruzar dos variables
casen_regional %>% #Creamos un objeto llamado pobreza_reg con
  group_by(pobreza, sexo) %>% #Agrupamos por pobreza y sexo
  summarise(prop = survey_prop(vartype = "ci", na.rm = T), #Ca
            total = survey_total(vartype = "ci", na.rm=T)) %>%
  mutate(prop = prop*100)
```

¡Transformemos en wide con tidyrr!



```
## Crear objeto wide
pobreza_reg <- casen_regional %>% #Creamos un objeto llamado p
  group_by(region, pobreza) %>% #Agrupamos por region y pobrez
  summarise(prop = survey_prop(vartype = "ci", na.rm = T), #Ca
            total = survey_total(vartype = "ci", na.rm=T)) %>%
  mutate(per = prop*100) %>% #Multiplicamos las proporciones *
  select(region, pobreza, per, total) %>% #Seleccionamos regio
  pivot_wider(names_from = "pobreza", #Pivoteamos a lo ancho,
              values_from = c("per", "total")) #y los valores
```

En síntesis



Diseño muestral complejo

El diseño muestral de CASEN 2020

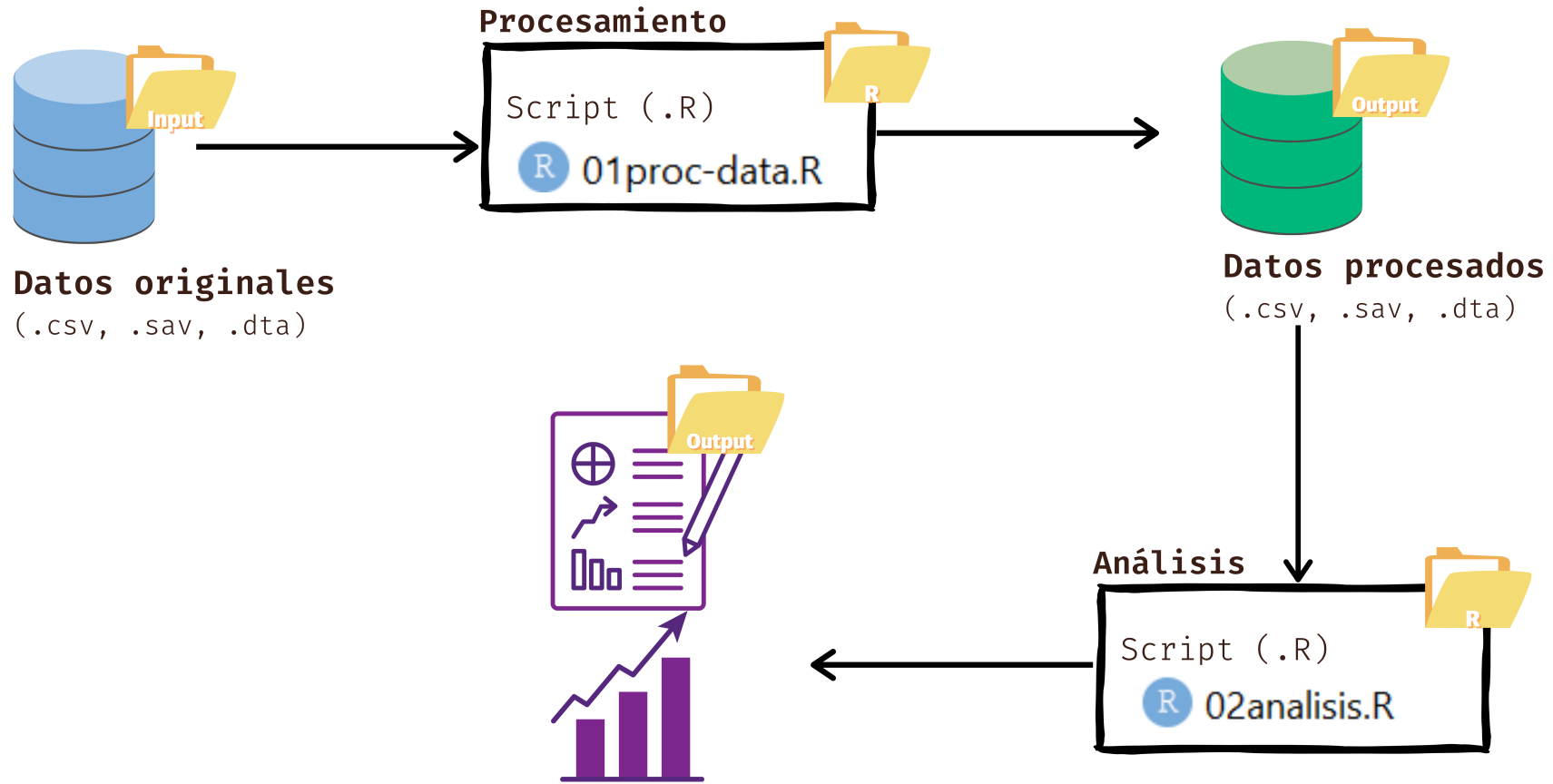
Creación de objetos encuesta con srvyr

Estimación, inferencia y tabulado

¡Y a no olvidar el flujo para el análisis!



Nos permite hacernos amigas/os más rápido del programa



¿Y eso era?



¡Ahora si que si! Nos vemos el próximo lunes





Muestras complejas y precisión de inferencia estadística

Sesión N° 7

04 octubre 2021

Análisis de datos estadísticos en R

Profesora Valentina Andrade de la Horra
Ayudantes Dafne Jaime y Nicolás Godoy