Análisis de encuestas de hogares con R

Módulo 2: Análisis de variables categóricas

CEPAL - Unidad de Estadísticas Sociales

Tabla de contenidos I

Introducción

Motivación

- En el mundo de la estadística y el análisis de datos, nos encontramos con una variedad de variables que pueden ser clasificadas en dos categorías principales: cualitativas y cuantitativas.
- Las variables cualitativas, también conocidas como categóricas, representan características o cualidades que no se pueden medir con números, como el género, el estado civil o el tipo de vivienda.
- Algunas variables cuantitativas se transforman en categóricas al dividir su rango en categorías, y viceversa, algunas variables categóricas se convierten en cuantitativas mediante análisis especializados.
- En esta presentación, exploraremos esta distinción y cómo abordar variables cualitativas en el contexto de encuestas y análisis de datos.

Definición del diseño y creación de variables categóricas

Lectura de la base

Iniciemos con la lectura de la encuesta.

```
encuesta <- readRDS("Imagenes/02_variable_continua/ENIGH_HND_Pers.rds")
```

El paso siguiente es realizar declaración del objeto tipo diseño.

```
options(survey.lonely.psu = "adjust")
library(srvyr)
diseno <- encuesta %>% # Base de datos.
  mutate(estrato = haven::as_factor(F1_A0_ESTRATO),
         Sexo = haven::as_factor(F2_A6_P3_SEXO),
         Area = haven::as_factor(F1_A0_AREA)) %>%
  as_survey_design(
    strata = estrato,  # Id de los estratos.
    ids = F1_A0_UPM,          # Id para las observaciones.
    weights = Factor,       # Factores de expansión.
    nest = TRUE            # Valida el anidado dentro del estrato
  )
```

Creación de nuevas variables

Durante los análisis de encuesta surge la necesidad de crear nuevas variables a partir de las existentes, aquí mostramos la definición de algunas de ellas.

```
diseno <- diseno %>% mutate(
 Edad_cat = case_when(F2_A6_P4_EDAD < 16 ~ "0 - 15",
                      F2_A6_P4_EDAD < 31 ~ "16 - 30",
                      F2_A6_P4_EDAD < 46 ~ "31 - 45",
                      F2_A6_P4_EDAD < 61 ~ "46 - 60",
                      F2_A6_P4_EDAD > 60 ~ "60 +",
                      TRUE ~ NA_character_
  )
)
```

Se ha introducido la función `case_when` la cual es una extensión del a función `ifelse` que permite crear múltiples categorías a partir de una o varias condiciones.

Dividiendo la muestra en Sub-grupos

En ocasiones se desea realizar estimaciones por sub-grupos de la población, en este caso se extraer 4 sub-grupos de la encuesta.

```
sub_Urbano <- diseno %>%  filter(Area == "1. Urbana") #
sub_Rural  <- diseno %>%  filter(Area == "2. Rural") #
sub_Mujer  <- diseno %>%  filter(Sexo == "2. Mujer") #
sub_Hombre <- diseno %>%  filter(Sexo == "1. Hombre") #
```

Estimación del tamaño.

El primer parámetro estimado serán los tamaños de la población y subpoblaciones.

```
(tamano_zona <- diseno %>% group_by(Area) %>%
   summarise(
     n = unweighted(n()), # Observaciones en la muestra.
     Nd = survey_total(vartype = c("se","ci"))))
```

Area	n	Nd	Nd_se	Nd_low	Nd_upp
1. Urbana	26923	5445857	78340	5292056	5599658
2. Rural	5106	4340256	162236	4021748	4658764

En la tabla *n* denota el número de observaciones en la muestra por Área y *Nd* denota la estimación del total de observaciones en la población.

Estimación de tamaño

Empleando una sintaxis similar es posible estimar el número de personas en condición en el un decil dado el ingreso disponible percápita del hogar

```
(tamano_decil <- diseno %>%
     mutate(DECIL = haven::as_factor(DECIL_YDISPO_PER) ) %>%
   group_by(DECIL) %>%
   summarise(Nd = survey_total(vartype = c("se", "ci"))))
```

DECIL	Nd	Nd_se	Nd_low	Nd_upp
1	1209670	105509	1002530	1416810
2	1146871	76995	995711	1298030
3	1121981	67205	990041	1253921
4	1110289	63055	986496	1234081
5	1014202	60200	896014	1132390
6	966416	50825	866634	1066197
7	914053	51030	813868	1014239
8	851633	52534	748496	954771
9	764664	37502	691039	838290
10	686333	44447	599072	773594

Estimación de tamaño

En forma similar es posible estimar el número de personas por etnia

```
(tamano_etnia<- diseno %>%
     mutate(etnia = haven::as_factor(F2_A6_P5_ETNIA) ) %>%
   group_by(etnia) %>%
   summarise(
       Nd = survey_total(vartype = c("se","ci"))))
```

etnia	Nd	Nd_se	Nd_low	Nd_upp
1. Indigena	663165	123041	421605.7	904724
2. Afrohondureño(a)	29540	5907	17943.6	41136
3. Negro(a)	34682	6142	22623.8	46740
4. Mestizo(a)	8324693	199547	7932934.3	8716451
5. Blanco(a)	731372	50477	632272.6	830472
6. Otro (especifique)	2661	1470	-224.5	5547

Estimación de tamaño

Otra variable de interés es conocer el estado de ocupación de la personas.

```
(
  tamano_ocupacion <- diseno %>%
    mutate(ocupacion = haven::as_factor(F2_A9_P3_TIPOEMPLEADO)) %>%
    group_by(ocupacion) %>%
    summarise(Nd = survey_total(vartype = c("se", "ci")))
)
```

Estimación de tamaño

ocupacion	Nd	Nd_se	Nd_low	Nd_upp
1. Empleado(a) u obrero en el sector público	285209.5	15448.0	254881.28	315538
2. Empleado(a) u obrero en el sector privado	1967228.5	53262.6	1862661.23	2071796
3. Empleado(a) doméstico(a)	159951.9	12102.5	136191.69	183712
4. Miembro de cooperativa, asentamiento o grupo	2476.5	1297.2	-70.15	5023
5. Cuenta propia que no contrata mano de obra temporal	1261268.5	50839.3	1161458.79	1361078
6. Cuenta propia que contrata mano de obra temporal	386324.8	23524.5	340140.45	432509
7. Empleador, patrón o socio activo	95915.2	8573.6	79083.05	112747
8. Trabajador familiar auxiliar	191943.2	17254.8	158067.76	225819
9. Practicante o pasante de carrera remunerado en el sector público	785.2	317.4	162.00	1408
10. Practicante o pasante de carrera remunerado en el sector privado	1074.1	835.4	-566.06	2714
12. Aprendiz remunerado en el sector privado	2504.8	907.5	723.24	4286
NA	5431430.8	119104.0	5197600.89	5665261

Estimación de tamaño

Utilizando la función group_by es posible obtener resultados por más de un nivel de agregación.

```
(tamano_etnia_sexo <- diseno %>%
     mutate(etnia = haven::as_factor(F2_A6_P5_ETNIA)) %>%
   group_by(etnia, Sexo) %>%
   cascade(
       Nd = survey_total(vartype = c("se","ci")),
       .fill = "Total") %>%
   data.frame()
   )
```

Estimación de tamaño

etnia	Sexo	Nd	Nd_se	Nd_low	Nd_upp
1. Indigena	1. Hombre	330443	60571.8	211525.7	449360
1. Indigena	2. Mujer	332722	63593.6	207872.4	457572
1. Indigena	Total	663165	123041.0	421605.7	904724
2. Afrohondureño(a)	1. Hombre	13536	3001.7	7643.3	19429
2. Afrohondureño(a)	2. Mujer	16004	3369.5	9388.6	22619
2. Afrohondureño(a)	Total	29540	5906.8	17943.6	41136
3. Negro(a)	1. Hombre	17841	3454.6	11059.2	24624
3. Negro(a)	2. Mujer	16841	3380.2	10204.4	23477
3. Negro(a)	Total	34682	6142.0	22623.8	46740
4. Mestizo(a)	1. Hombre	3941279	106130.9	3732918.6	4149640
4. Mestizo(a)	2. Mujer	4383414	102802.7	4181587.2	4585240
4. Mestizo(a)	Total	8324693	199546.8	7932934.3	8716451
5. Blanco(a)	1. Hombre	330192	23598.8	283861.6	376522
5. Blanco(a)	2. Mujer	401180	30228.8	341834.0	460527
5. Blanco(a)	Total	731372	50477.5	632272.6	830472
6. Otro (especifique)	1. Hombre	1326	788.0	-221.5	2873
6. Otro (especifique)	2. Mujer	1336	732.1	-101.7	2773
6. Otro (especifique)	Total	2661	1469.9	-224.5	5547
Total	Total	9786113	180160.3	9432414.8	10139811

Estimación de Proporciones Poblacionales

En encuestas de hogares, a menudo es importante estimar la proporción de una característica particular en una población, como la proporción de personas que tienen un cierto nivel de educación, la proporción de hogares con acceso a servicios básicos, entre otros.

La estimación de una proporción poblacional se puede hacer utilizando la siguiente ecuación:

$$\hat{\pi} = p = \frac{\sum_{i=1}^{n} \omega_i y_i}{\sum_{i=1}^{n} \omega_i}$$

Donde:

- $\hat{\pi}$ es la estimación de la proporción poblacional.
- n es el tamaño de la muestra.
- ω_i son los pesos de muestreo para cada unidad de la muestra.
- y_i es la variable binaria que indica si la unidad de muestreo tiene la característica de interés (1 si la tiene, 0 si no la tiene).

Estimación de proporción de urbano y rural

El procedimiento estándar para el calculo de proporciones es crear una *variable dummy* y sobre está realizar las operaciones. Sin embargo, la librería `srvy` nos simplifica el calculo, mediante la sintaxis.

```
(prop_Area <- diseno %>% group_by(Area) %>%
   summarise(
     prop = survey_mean(vartype = c("se","ci"),
                        proportion = TRUE )))
```

Area	prop	prop_se	prop_low	prop_upp
1. Urbana	0.5565	0.0099	0.5370	0.5758
2. Rural	0.4435	0.0099	0.4242	0.4630

Note que, se utilizo la función `survey_mean` para la estimación.

Estimación de proporción de urbano y rural

La función idónea para realizar la estimación de las proporciones es `survey_prop` y la sintaxis es como sigue:

```
(prop_area2 <- diseno %>% group_by(Area) %>%
   summarise(
     prop = survey_prop(vartype = c("se","ci") )))
```

Area	prop	prop_se	prop_low	prop_upp
1. Urbana	0.5565	0.0099	0.5370	0.5758
2. Rural	0.4435	0.0099	0.4242	0.4630

Proporción de hombres y mujeres en la área urbana

Si el interés es obtener la estimación para una subpoblación, procedemos así:

```
(prop_sexoU <- sub_Urbano %>% group_by(Sexo) %>%
   summarise(
       prop = survey_prop(vartype = c("se","ci"))))
```

Sexo	prop	prop_se	prop_low	prop_upp
1. Hombre	0.4616	0.0031	0.4555	0.4678
2. Mujer	0.5384	0.0031	0.5322	0.5445

¿Cómo estimar el Proporción de hombres dado que están en zona rural?

Proporción de hombres y mujeres en la zona rural

```
(prop_sexoR <- sub_Rural %>% group_by(Sexo) %>%
   summarise(
     n = unweighted(n()),
     prop = survey_prop(vartype = c("se","ci"))))
```

Sexo	n	prop	prop_se	prop_low	prop_upp
1. Hombre	2490	0.4886	0.006	0.4766	0.5006
2. Mujer	2616	0.5114	0.006	0.4994	0.5234

¿Cómo estimar el Proporción de hombres en la área rural dado que es hombre?

Proporción de hombres en la área urbana y rural

```
(prop_AreaH <- sub_Hombre %>% group_by(Area) %>%
   summarise(
     prop = survey_prop(vartype = c("se","ci"))))
```

Area	prop	prop_se	prop_low	prop_upp
1. Urbana	0.5424	0.011	0.5208	0.5639
2. Rural	0.4576	0.011	0.4361	0.4792

¿Cómo estimar el Proporción de mujeres en la área rural dado que es mujer?

Proporción de mujeres en la área urbana y rural

```
(prop_AreaM <- sub_Mujer %>% group_by(Area) %>%
   summarise(
    prop = survey_prop(vartype = c("se","ci"))))
```

Area	prop	prop_se	prop_low	prop_upp
1. Urbana	0.5691	0.0099	0.5496	0.5884
2. Rural	0.4309	0.0099	0.4116	0.4504

Proporción de hombres en la área urbana y rural

Con el uso de la función `group_by` es posible estimar un mayor numero de niveles de agregación al combinar dos o más variables.

```
(prop_AreaH_edad <- sub_Hombre %>%
   group_by(Area, Edad_cat ) %>%
   summarise(
     prop = survey_prop(vartype = c("se","ci")))%>%
   data.frame())
```

Proporción de hombres en la área urbana y rural

Area	Edad_cat	prop	prop_se	prop_low	prop_upp
1. Urbana	0 - 15	0.3302	0.0052	0.3201	0.3405
1. Urbana	16 - 30	0.2573	0.0049	0.2478	0.2671
1. Urbana	31 - 45	0.1823	0.0042	0.1743	0.1907
1. Urbana	46 - 60	0.1254	0.0037	0.1184	0.1329
1. Urbana	60 +	0.1047	0.0034	0.0982	0.1115
2. Rural	0 - 15	0.3592	0.0123	0.3353	0.3837
2. Rural	16 - 30	0.2484	0.0115	0.2265	0.2716
2. Rural	31 - 45	0.1762	0.0076	0.1617	0.1917
2. Rural	46 - 60	0.1070	0.0068	0.0944	0.1210
2. Rural	60 +	0.1092	0.0086	0.0934	0.1274

Proporción de mujeres en la área urbana y rural

```r
(prop_AreaM_edad <- sub_Mujer %>%
   group_by(Area, Edad_cat) %>%
   summarise(
     prop = survey_prop(vartype = c("se","ci"))) %>%
   data.frame())
```

Area	Edad_cat	prop	prop_se	prop_low	prop_upp
1. Urbana	0 - 15	0.2838	0.0047	0.2746	0.2932
1. Urbana	16 - 30	0.2489	0.0042	0.2407	0.2573
1. Urbana	31 - 45	0.2057	0.0034	0.1992	0.2124
1. Urbana	46 - 60	0.1404	0.0034	0.1338	0.1473
1. Urbana	60 +	0.1212	0.0037	0.1141	0.1287
2. Rural	0 - 15	0.3278	0.0105	0.3075	0.3487
2. Rural	16 - 30	0.2539	0.0095	0.2358	0.2729
2. Rural	31 - 45	0.1888	0.0090	0.1718	0.2070
2. Rural	46 - 60	0.1214	0.0079	0.1068	0.1378
2. Rural	60 +	0.1081	0.0075	0.0942	0.1238

Proporción de hombres en la area disponible para trabajar

```
#F2_A8_P13_DISPONIBLETRABAJAR:  Estaba disponible para trabajar

(prop_AreaH_disponible <- sub_Hombre %>%
    mutate(disponible = haven::as_factor(F2_A8_P13_DISPONIBLETRABAJAR)) %>
   group_by(Area, disponible) %>%
   summarise(
     prop = survey_prop(vartype = c("se","ci"))) %>%
   data.frame())
```

Proporción de hombres en la área disponible para trabajar

Area	disponible	prop	prop_se	prop_low	prop_upp
1. Urbana	1. Sí	0.0502	0.0027	0.0452	0.0558
1. Urbana	2. No, pero lo estará en 15 días o menos	0.0001	0.0001	0.0000	0.0007
1. Urbana	3. No, pero lo estará en más de 15 días pero menos de 12 meses	0.0004	0.0002	0.0002	0.0013
1. Urbana	4. No	0.2024	0.0045	0.1937	0.2113
1. Urbana	5. No sabe	0.0046	0.0008	0.0032	0.0064
1. Urbana	NA	0.7423	0.0050	0.7324	0.7520
2. Rural	1. Sí	0.0361	0.0050	0.0274	0.0473
2. Rural	3. No, pero lo estará en más de 15 días pero menos de 12 meses	0.0014	0.0014	0.0002	0.0096
2. Rural	4. No	0.1647	0.0099	0.1461	0.1851
2. Rural	5. No sabe	0.0038	0.0016	0.0017	0.0085
2. Rural	NA	0.7941	0.0107	0.7723	0.8143

Proporción de mujeres en la área urbana y rural

```
(prop_AreaM_disponible <- sub_Mujer %>%
 mutate(
  disponible = haven::as_factor(F2_A8_P13_DISPONIBLETRABAJAR)) %>%
   group_by(Area, disponible) %>%
   summarise( prop = survey_prop(vartype = c("se","ci"))) %>%
   data.frame())
```

Proporción de mujeres en la área urbana y rural

Area	disponible	prop	prop_se	prop_low	prop_upp
1. Urbana	1. Sí	0.0778	0.0032	0.0718	0.0844
1. Urbana	2. No, pero lo estará en 15 días o menos	0.0004	0.0002	0.0002	0.0010
1. Urbana	3. No, pero lo estará en más de 15 días pero menos de 12 meses	0.0010	0.0003	0.0005	0.0017
1. Urbana	4. No	0.3574	0.0051	0.3475	0.3675
1. Urbana	5. No sabe	0.0084	0.0009	0.0067	0.0105
1. Urbana	NA	0.5549	0.0054	0.5444	0.5654
2. Rural	1. Sí	0.0763	0.0089	0.0605	0.0958
2. Rural	3. No, pero lo estará en más de 15 días pero menos de 12 meses	0.0005	0.0005	0.0001	0.0036
2. Rural	4. No	0.4062	0.0191	0.3694	0.4441
2. Rural	5. No sabe	0.0171	0.0044	0.0103	0.0283
2. Rural	NA	0.4999	0.0174	0.4657	0.5341

Estimación de la proporción de personas por rango de edad

```r
diseno <-  diseno %>%
  mutate(
    disponible = case_when(
      F2_A8_P13_DISPONIBLETRABAJAR == 1 ~ "Sí",
      F2_A8_P13_DISPONIBLETRABAJAR %in% c(2:5) ~
        "No",
      TRUE ~ NA_character_
    )
  )
diseno %>%  group_by(disponible, Edad_cat) %>%
  summarise(Prop = survey_prop(vartype =  c("se", "ci"))) %>%
  data.frame()
```

Estimación de la proporción de personas por rango de edad

disponible	Edad_cat	Prop	Prop_se	Prop_low	Prop_upp
No	0 - 15	0.3504	0.0070	0.3369	0.3642
No	16 - 30	0.2467	0.0070	0.2332	0.2606
No	31 - 45	0.1021	0.0058	0.0912	0.1141
No	46 - 60	0.0944	0.0050	0.0850	0.1048
No	60 +	0.2064	0.0082	0.1908	0.2229
Sí	0 - 15	0.0852	0.0102	0.0671	0.1075
Sí	16 - 30	0.4487	0.0173	0.4149	0.4829
Sí	31 - 45	0.2494	0.0137	0.2235	0.2774
Sí	46 - 60	0.1366	0.0121	0.1145	0.1622
Sí	60 +	0.0801	0.0096	0.0632	0.1010
NA	0 - 15	0.3316	0.0062	0.3196	0.3438
NA	16 - 30	0.2358	0.0054	0.2254	0.2465
NA	31 - 45	0.2240	0.0046	0.2151	0.2333
NA	46 - 60	0.1380	0.0040	0.1302	0.1461
NA	60 +	0.0706	0.0030	0.0649	0.0767

Estimación de la proporción de personas por rango de edad

```
sub_Rural %>%
  group_by(Edad_cat) %>%
  summarise(
    Prop = survey_prop(
      vartype =  c("se", "ci"))) %>%
  data.frame()
```

Edad_cat	Prop	Prop_se	Prop_low	Prop_upp
0 - 15	0.3431	0.0099	0.3239	0.3630
16 - 30	0.2512	0.0086	0.2346	0.2686
31 - 45	0.1826	0.0066	0.1698	0.1961
46 - 60	0.1144	0.0064	0.1023	0.1276
60 +	0.1087	0.0074	0.0949	0.1242

Estimación de la proporción de mujeres rango de edad

```
sub_Mujer %>%
  mutate(
    disponible = case_when(
      F2_A8_P13_DISPONIBLETRABAJAR == 1 ~ "Sí",
      F2_A8_P13_DISPONIBLETRABAJAR %in% c(2:5) ~
        "No",
      TRUE ~ NA_character_
    )
  )  %>% group_by(disponible, Edad_cat) %>%
  summarise(Prop = survey_prop(
      vartype =  c("se", "ci"))) %>%   data.frame()
```

Estimación de la proporción de mujeres rango de edad

disponible	Edad_cat	Prop	Prop_se	Prop_low	Prop_upp
No	0 - 15	0.2645	0.0074	0.2502	0.2793
No	16 - 30	0.2732	0.0092	0.2556	0.2916
No	31 - 45	0.1351	0.0082	0.1197	0.1521
No	46 - 60	0.1228	0.0065	0.1106	0.1362
No	60 +	0.2043	0.0090	0.1871	0.2226
Sí	0 - 15	0.0672	0.0112	0.0482	0.0929
Sí	16 - 30	0.4430	0.0227	0.3990	0.4879
Sí	31 - 45	0.2923	0.0181	0.2582	0.3290
Sí	46 - 60	0.1447	0.0148	0.1180	0.1762
Sí	60 +	0.0528	0.0098	0.0366	0.0755
NA	0 - 15	0.3652	0.0096	0.3466	0.3841
NA	16 - 30	0.2068	0.0066	0.1942	0.2200
NA	31 - 45	0.2314	0.0076	0.2168	0.2467
NA	46 - 60	0.1373	0.0055	0.1268	0.1485
NA	60 +	0.0593	0.0033	0.0531	0.0662

Estimación de la proporción de hombres rango de edad

```
sub_Hombre %>%
  mutate(
    disponible = case_when(
      F2_A8_P13_DISPONIBLETRABAJAR == 1 ~ "Sí",
      F2_A8_P13_DISPONIBLETRABAJAR %in% c(2:5) ~
        "No",
      TRUE ~ NA_character_
    )
  )  %>% group_by(disponible, Edad_cat) %>%
  summarise(Prop = survey_prop(
      vartype =  c("se", "ci"))) %>%   data.frame()
```

Estimación de la proporción de hombres rango de edad

disponible	Edad_cat	Prop	Prop_se	Prop_low	Prop_upp
No	0 - 15	0.5470	0.0137	0.5199	0.5737
No	16 - 30	0.1859	0.0106	0.1660	0.2075
No	31 - 45	0.0265	0.0044	0.0191	0.0366
No	46 - 60	0.0294	0.0037	0.0229	0.0377
No	60 +	0.2112	0.0103	0.1917	0.2322
Sí	0 - 15	0.1203	0.0198	0.0865	0.1650
Sí	16 - 30	0.4598	0.0234	0.4143	0.5061
Sí	31 - 45	0.1653	0.0198	0.1300	0.2079
Sí	46 - 60	0.1208	0.0165	0.0919	0.1571
Sí	60 +	0.1337	0.0185	0.1014	0.1744
NA	0 - 15	0.3057	0.0068	0.2925	0.3191
NA	16 - 30	0.2582	0.0071	0.2445	0.2723
NA	31 - 45	0.2184	0.0055	0.2078	0.2293
NA	46 - 60	0.1385	0.0047	0.1295	0.1481
NA	60 +	0.0793	0.0043	0.0712	0.0882

Tablas cruzadas.

Tablas Cruzadas en el Análisis de Encuestas de Hogares

- Las tablas cruzadas son una herramienta esencial.
- Se utilizan para resumir información de variables categóricas.
- Pueden tener dos o más filas y columnas.
- En esta sección, nos enfocaremos principalmente en tablas 2×2.

Estructura de una Tabla de Contingencia:

- Se asume como un arreglo bidimensional de filas y columnas.
- Marginales de fila y columna se calculan sumando las frecuencias.

Ejemplo Gráfico de una Tabla 2×2.

Representa la relación entre dos variables categóricas.

Variable 2	Variable 1	Marginal fila	
	0	1	
0	n_{00}	n_{01}	n_{0+}
1	n_{10}	n_{11}	n_{1+}
Marginal columna	n_{+0}	n_{+1}	n_{++}

Las frecuencias en la tabla pueden ser estimadas o ponderadas. Por ejemplo, $\hat{N}_{01}$ se calcula como la suma ponderada.

Cálculo de Proporciones Estimadas

Las proporciones se obtienen dividiendo las frecuencias ponderadas por el total. Por ejemplo:

$$p_{rc} = \frac{\hat{N}_{rc}}{\hat{N}_{++}}$$

.

Estas tablas cruzadas son fundamentales para explorar la relación entre diferentes variables categóricas en encuestas de hogares y extraer información valiosa para la toma de decisiones.

Estimación de Proporciones para Variables Binarias

- La estimación de una sola proporción, π, para una variable binaria se relaciona con el estimador de razón.
- Al recodificar las respuestas en 0 y 1, podemos estimar la proporción π.
- El estimador de proporción es:

$$p = \frac{\sum_{h=1}^{H}\sum_{\alpha=1}^{\alpha_h}\sum_{i\in(0,1)}^{n_{h\alpha}} \omega_{h\alpha i} I\left(y_i = 1\right)}{\sum_{h=1}^{H}\sum_{\alpha=1}^{\alpha_h}\sum_{i\in(0,1)}^{n_{h\alpha}} \omega_{h\alpha i}} = \frac{\hat{N}_1}{\hat{N}}$$

Estimación de la varianza $\hat{v}(p)$

- La varianza del estimador se calcula con Linealización de Taylor:

$$\hat{v}(p) \approx \frac{V(\hat{N}_1) + p^2 V(\hat{N}) - 2p\, cov(\hat{N}_1, \hat{N})}{\hat{N}^2}$$

- Para evitar límites no interpretables en el intervalo de confianza cuando p está cerca de 0 o 1, podemos utilizar el método de *Wilson modificado*.
- El intervalo de confianza se calcula a través de la transformación Logit:

$$IC[logit(p)] = \left\{ ln\left(\frac{p}{1-p}\right) \pm \frac{t_{1-\alpha/2,gl}\, se(p)}{p(1-p)} \right\}$$

Estimación de la varianza $IC(p)$

El intervalo de confianza para p es:

$$IC(p) = \left\{ \frac{exp\left[ln\left(\frac{p}{1-p}\right) \pm \frac{t_{1-\alpha/2,gl}se(p)}{p(1-p)}\right]}{1 + exp\left[ln\left(\frac{p}{1-p}\right) \pm \frac{t_{1-\alpha/2,gl}se(p)}{p(1-p)}\right]} \right\}$$

Estimación de Proporciones para Variables Multinomiales

- Cuando se trabaja con variables multinomiales, el estimador de proporción se adapta.
- El estimador para la categoría k es:

$$p_k = \frac{\sum_{h=1}^{H}\sum_{\alpha=1}^{\alpha_h}\sum_{i=1}^{n_{h\alpha}}\omega_{h\alpha i}I\left(y_i = k\right)}{\sum_{h=1}^{H}\sum_{\alpha=1}^{\alpha_h}\sum_{i=1}^{n_{h\alpha}}\omega_{h\alpha i}} = \frac{\hat{N}_k}{\hat{N}}$$

Estos métodos permiten estimar proporciones para variables binarias y multinomiales en el contexto de encuestas de hogares.

Tabla Zona Vs Sexo

Haciendo uso de la función `group_by` organizada en forma de `data.frame`.

```
diseno <- diseno %>%
   mutate(etnia = haven::as_factor(F2_A6_P5_ETNIA))
(
  prop_sexo_etnia <- diseno %>%
    group_by(etnia, Sexo) %>%
    summarise(
      prop = survey_prop(vartype = c("se", "ci"))) %>%
    data.frame()
)
```

Esta forma de organizar la información es recomendable cuando el realizar el análisis sobre las estimaciones puntuales.

Tabla Etnia Vs Sexo

etnia	Sexo	prop	prop_se	prop_low	prop_upp
1. Indigena	1. Hombre	0.4983	0.0127	0.4733	0.5232
1. Indigena	2. Mujer	0.5017	0.0127	0.4768	0.5267
2. Afrohondureño(a)	1. Hombre	0.4582	0.0404	0.3807	0.5379
2. Afrohondureño(a)	2. Mujer	0.5418	0.0404	0.4621	0.6193
3. Negro(a)	1. Hombre	0.5144	0.0432	0.4299	0.5981
3. Negro(a)	2. Mujer	0.4856	0.0432	0.4019	0.5701
4. Mestizo(a)	1. Hombre	0.4734	0.0038	0.4660	0.4809
4. Mestizo(a)	2. Mujer	0.5266	0.0038	0.5191	0.5340
5. Blanco(a)	1. Hombre	0.4515	0.0128	0.4266	0.4766
5. Blanco(a)	2. Mujer	0.5485	0.0128	0.5234	0.5734
6. Otro (especifique)	1. Hombre	0.4981	0.0738	0.3574	0.6392
6. Otro (especifique)	2. Mujer	0.5019	0.0738	0.3608	0.6426

Tablas de doble entrada.

Una alternativa es utilizar la función `svyby` con la siguiente sintaxis.

```
tab_Sex_etnia <- svyby(~Sexo,  ~etnia, diseno, svymean)
tab_Sex_etnia %>% select(-"se.Sexo1. Hombre", -"se.Sexo2. Mujer")
```

	etnia	Sexo1. Hombre	Sexo2. Mujer
1. Indigena	1. Indigena	0.4983	0.5017
2. Afrohondureño(a)	2. Afrohondureño(a)	0.4582	0.5418
3. Negro(a)	3. Negro(a)	0.5144	0.4856
4. Mestizo(a)	4. Mestizo(a)	0.4734	0.5266
5. Blanco(a)	5. Blanco(a)	0.4515	0.5485
6. Otro (especifique)	6. Otro (especifique)	0.4981	0.5019

Tablas de doble entrada.

```
tab_Sex_etnia %>% select(-"Sexo1. Hombre", -"Sexo2. Mujer")
```

	etnia	se.Sexo1. Hombre	se.Sexo2. Mujer
1. Indigena	1. Indigena	0.0127	0.0127
2. Afrohondureño(a)	2. Afrohondureño(a)	0.0404	0.0404
3. Negro(a)	3. Negro(a)	0.0432	0.0432
4. Mestizo(a)	4. Mestizo(a)	0.0038	0.0038
5. Blanco(a)	5. Blanco(a)	0.0128	0.0128
6. Otro (especifique)	6. Otro (especifique)	0.0738	0.0738

Tablas de doble entrada.

Para la estimación de los intervalos de confianza utilizar la función `confint`.

```
confint(tab_Sex_etnia) %>% as.data.frame()
```

	2.5 %	97.5 %
1. Indigena:Sexo1. Hombre	0.4733	0.5232
2. Afrohondureño(a):Sexo1. Hombre	0.3791	0.5374
3. Negro(a):Sexo1. Hombre	0.4297	0.5992
4. Mestizo(a):Sexo1. Hombre	0.4660	0.4809
5. Blanco(a):Sexo1. Hombre	0.4265	0.4765
6. Otro (especifique):Sexo1. Hombre	0.3536	0.6427
1. Indigena:Sexo2. Mujer	0.4768	0.5267
2. Afrohondureño(a):Sexo2. Mujer	0.4626	0.6209
3. Negro(a):Sexo2. Mujer	0.4008	0.5703
4. Mestizo(a):Sexo2. Mujer	0.5191	0.5340
5. Blanco(a):Sexo2. Mujer	0.5235	0.5735
6. Otro (especifique):Sexo2. Mujer	0.3573	0.6464

Prueba de independencia χ^2

- La prueba de independencia χ^2 se utiliza para determinar si dos variables cualitativas son independientes o si hay una asociación entre ellas.
- La prueba se aplica comúnmente a tablas de contingencia, especialmente las 2×2.
- La fórmula para el estadístico χ^2 de Pearson es:

$$\chi^2 = n_{++} \sum_r \sum_c \frac{(p_{rc} - \hat{\pi}_{rc})^2}{\hat{\pi}_{rc}}$$

- Donde $\hat{\pi}_{rc}$ se calcula como:

$$\hat{\pi}_{rc} = \frac{n_{r+}}{n_{++}} \cdot \frac{n_{+c}}{n_{++}} \cdot p_{r+} \cdot p_{+c}$$

Prueba de independencia.

Para realizar la prueba de independencia χ^2 puede ejecuta la siguiente linea de código.

```
svychisq(~Sexo + etnia, diseno, statistic="F")
```

```
    Pearson's X^2: Rao & Scott adjustment

data:  NextMethod()
F = 2.2, ndf = 3.7, ddf = 2661.4, p-value = 0.07
```

Más adelante se profundiza en la metodología de esta prueba.

Tablas de doble entrada.

```
(tab_Sex_Ocupa <- svyby(~Sexo,  ~disponible,
                        diseno, svymean))
```

	disponible	Sexo1. Hombre	Sexo2. Mujer	se.Sexo1. Hombre	se.Sexo2. Mujer
No	No	0.3042	0.6958	0.0067	0.0067
Sí	Sí	0.3377	0.6623	0.0171	0.0171

Tablas de doble entrada

```
confint(tab_Sex_Ocupa) %>% as.data.frame()
```

	2.5 %	97.5 %
No:Sexo1. Hombre	0.2910	0.3173
Sí:Sexo1. Hombre	0.3041	0.3712
No:Sexo2. Mujer	0.6827	0.7090
Sí:Sexo2. Mujer	0.6288	0.6959

Prueba de independencia.

La prueba de independencia χ^2 se obtiene con la siguiente linea de código.

```
svychisq(~Sexo + disponible,
         design = diseno,  statistic="F")
```

```
    Pearson's X^2: Rao & Scott adjustment

data:  NextMethod()
F = 3.3, ndf = 1, ddf = 725, p-value = 0.07
```

Tablas de doble entrada.

```
#F2_A6_P7_SEGUROMEDIC__1    Cobertura seguro social (IHSS)

diseno <- diseno %>%
  mutate(SEGUROMEDIC = haven::as_factor(F2_A6_P7_SEGUROMEDIC__1),
         DEPARTAMENTO  = haven::as_factor(F1_A0_DEPARTAMENTO ))

tab_dam_IHSS <-
  svyby( ~SEGUROMEDIC, ~DEPARTAMENTO, diseno, svymean)
```

Tablas de doble entrada.

```
tab_dam_IHSS %>% select(-"se.SEGUROMEDICNo",
       -"se.SEGUROMEDICSí")
```

	DEPARTAMENTO	SEGUROMEDICNo	SEGUROMEDICSí
1. Atlántida	1. Atlántida	0.8987	0.1013
2. Colon	2. Colon	0.9533	0.0467
3. Comayagua	3. Comayagua	0.9056	0.0944
4. Copan	4. Copan	0.9608	0.0392
5. Cortes	5. Cortes	0.7759	0.2241
6. Choluteca	6. Choluteca	0.9000	0.1000
7. El Paraíso	7. El Paraíso	0.9242	0.0758
8. Francisco Morazán	8. Francisco Morazán	0.7881	0.2119
9. Gracias A Dios	9. Gracias A Dios	0.9877	0.0123
10. Intibuca	10. Intibuca	0.9798	0.0202
11. Islas De La Bahía	11. Islas De La Bahía	0.8716	0.1284
12. La Paz	12. La Paz	0.9517	0.0483
13. Lempira	13. Lempira	0.9910	0.0090
14. Ocotepeque	14. Ocotepeque	0.9910	0.0090
15. Olancho	15. Olancho	0.9566	0.0434
16. Santa Bárbara	16. Santa Bárbara	0.9728	0.0272
17. Valle	17. Valle	0.8759	0.1241
18. Yoro	18. Yoro	0.9171	0.0829

Tablas de doble entrada.

```
tab_dam_IHSS %>%
select("se.SEGUROMEDICNo", "se.SEGUROMEDICSí")
```

	se.SEGUROMEDICNo	se.SEGUROMEDICSí
1. Atlántida	0.0156	0.0156
2. Colon	0.0126	0.0126
3. Comayagua	0.0158	0.0158
4. Copan	0.0143	0.0143
5. Cortes	0.0123	0.0123
6. Choluteca	0.0200	0.0200
7. El Paraíso	0.0167	0.0167
8. Francisco Morazán	0.0134	0.0134
9. Gracias A Dios	0.0052	0.0052
10. Intibuca	0.0103	0.0103
11. Islas De La Bahía	0.0276	0.0276
12. La Paz	0.0139	0.0139
13. Lempira	0.0059	0.0059
14. Ocotepeque	0.0078	0.0078
15. Olancho	0.0156	0.0156
16. Santa Bárbara	0.0078	0.0078
17. Valle	0.0454	0.0454
18. Yoro	0.0178	0.0178

Prueba de independencia.

Una vez más la prueba de independencia es:

```
svychisq( ~SEGUROMEDIC + DEPARTAMENTO,,
          design = diseno,  statistic="F")
```

```

    Pearson's X^2: Rao & Scott adjustment

data:  NextMethod()
F = 20, ndf = 13, ddf = 9630, p-value <2e-16
```

Razón de odds

- La razón de odds es una medida que expresa la probabilidad de que un evento ocurra en comparación con la probabilidad de que no ocurra.
- Es una forma de cuantificar la asociación entre los niveles de una variable y un factor categórico.
- La razón de odds se calcula como la proporción de la probabilidad de éxito (ocurrencia del evento) sobre la probabilidad de fracaso (no ocurrencia del evento).
- La fórmula general para la razón de odds es:

$$Odds = \frac{P(\text{Éxito})}{P(\text{Fracaso})}$$

- Se utiliza comúnmente en estadística y análisis de datos para evaluar la asociación entre variables y en modelos de regresión logística.

Razón de obbs

```
diseno <-
  diseno %>%
  mutate(SEGUROMEDIC2 = case_when(SEGUROMEDIC == "Sí" ~ 1,
                          SEGUROMEDIC == "No" ~ 0,
                                TRUE ~ NA_real_))
(tab_Sex <- svyby(~SEGUROMEDIC2,  ~Sexo, diseno,
                 svymean, vartype = c("se", "ci")))
```

	Sexo	SEGUROMEDIC2	se	ci_l	ci_u
1. Hombre	1. Hombre	0.1353	0.0055	0.1246	0.1460
2. Mujer	2. Mujer	0.1073	0.0046	0.0983	0.1163

```
svycontrast(tab_Sex, quote(`1. Hombre`/`2. Mujer`)  )
```

```
         nlcon   SE
contrast  1.26 0.07
```

Razón de obbs

```
tab_Sex_IHSS <-
   svymean(~interaction (Sexo, SEGUROMEDIC), diseno,
             se=T, na.rm=T, ci=T, keep.vars=T)
  tab_Sex_IHSS %>%  as.data.frame()
```

	mean	SE
interaction(Sexo, SEGUROMEDIC)1. Hombre.No	0.4095	0.0040
interaction(Sexo, SEGUROMEDIC)2. Mujer.No	0.4699	0.0037
interaction(Sexo, SEGUROMEDIC)1. Hombre.Sí	0.0641	0.0026
interaction(Sexo, SEGUROMEDIC)2. Mujer.Sí	0.0565	0.0025

Razón de obbs

Suponga que se desea calcular la siguiente razón de obbs.

$$\text{Razón de odds} = \frac{\frac{P(\text{SEGUROMEDIC} = \text{No}|\text{Hombre})}{P(\text{SEGUROMEDIC} = \text{Sí}|\text{Hombre})}}{\frac{P(\text{SEGUROMEDIC} = \text{No}|\text{Mujer})}{P(\text{SEGUROMEDIC} = \text{Sí}|\text{Mujer})}}$$

La forma de cálculo en sería:

```
 svycontrast(tab_Sex_IHSS,
quote(
  (`interaction(Sexo, SEGUROMEDIC)1. Hombre.No`/`interaction(Sexo, SEGUROMEDIC)1. Hombre.Sí`)/
(`interaction(Sexo, SEGUROMEDIC)2. Mujer.No`/`interaction(Sexo, SEGUROMEDIC)2. Mujer.Sí`)))
```

```
         nlcon   SE
contrast 0.768 0.03
```

Diferencia de proporciones en tablas de contingencias

Como lo menciona *Heeringa, S. G. (2017)* las estimaciones de las proporciones de las filas en las tablas de doble entrada son, de hecho, estimaciones de subpoblaciones en las que la subpoblación se define por los niveles de la variable factorial. Ahora bien, si el interés se centra en estimar diferencias de las proporciones de las categorías entre dos niveles de una variable factorial, se pueden utilizando contrastes.

Contrastes

El interés ahora es realizar en contraste de proporciones, por ejemplo: $\hat{p}_H - \hat{p}_M$

```
(tab_sex_IHSS <- svyby(~SEGUROMEDIC2, ~Sexo,
                        diseno ,
                    svymean, na.rm=T,
                    covmat = TRUE,
                    vartype = c("se", "ci")))
```

	Sexo	SEGUROMEDIC2	se	ci_l	ci_u
1. Hombre	1. Hombre	0.1353	0.0055	0.1246	0.1460
2. Mujer	2. Mujer	0.1073	0.0046	0.0983	0.1163

- *Paso 1:* Calcular la diferencia de estimaciones

```
0.1353 - 0.1073
```

```
[1] 0.028
```

Contrastes de la diferencia de proporciones

Con la función `vcov` obtiene la matriz de covarianzas

```
library(kableExtra)
vcov(tab_sex_IHSS)%>% as.data.frame() %>%
  kable(digits = 10,
        format.args = list(scientific = FALSE))
```

	1. Hombre	2. Mujer
1. Hombre	0.00002978	0.00001523
2. Mujer	0.00001523	0.00002129

- *Paso 2:* El cálculo del error estándar es:

```
sqrt(0.00002978 + 0.00002129 - 2*0.00001523)
```

```
[1] 0.00454
```

Contrastes de la diferencia de proporciones en R

Para realizar la diferencia de proporciones se hace uso de la función `svycontrast`.

```
svycontrast(tab_sex_IHSS,
            list(diff_Sex = c(1, -1))) %>%
  data.frame()
```

	contrast	diff_Sex
diff_Sex	0.028	0.0045

Contrastes de la diferencia de proporciones

Diferencia en disponibilidad para el empleo por sexo.

```
diseno <-
  diseno %>%
  mutate(disponible2 = case_when(disponible == "Sí" ~ 1,
                                 disponible == "No" ~ 0,
                                 TRUE ~ NA_real_))
(
  tab_sex_desempleo <- svyby(
    ~ disponible2,
    ~ Sexo,
    diseno ,
    svymean,
    na.rm = T,
    covmat = TRUE,
    vartype = c("se", "ci")
  )
)
```

Contrastes de la diferencia de proporciones

	Sexo	disponible2	se	ci_l	ci_u
1. Hombre	1. Hombre	0.1869	0.0107	0.1659	0.2079
2. Mujer	2. Mujer	0.1646	0.0092	0.1467	0.1826

- *Paso 1*: Diferencia de las estimaciones

```
0.1869 - 0.1646
```

```
[1] 0.0223
```

Contrastes de la diferencia de proporciones

Estimación de la matriz de covarianza:

```
vcov(tab_sex_desempleo) %>% as.data.frame() %>%
  kable(digits = 10,
        format.args = list(scientific = FALSE))
```

	1. Hombre	2. Mujer
1. Hombre	0.00011491	0.00002256
2. Mujer	0.00002256	0.00008382

- *Paso 2*: Estimación del error estándar.

```
sqrt(0.00011491  + 0.00008382 - 2*0.00002256)
```

```
[1] 0.01239
```

Contrastes de la diferencia de proporciones en R

Siguiendo el ejemplo anterior se tiene que:

```
svycontrast(tab_sex_desempleo,
            list(diff_Sex = c(1, -1))) %>%
  data.frame()
```

	contrast	diff_Sex
diff_Sex	0.0223	0.0124

¡Gracias!

Email: andres.gutierrez@cepal.org