Análisis de encuestas de hogares con R

Módulo 5: Modelos lineales generalizados

CEPAL - Unidad de Estadísticas Sociales

Tabla de contenidos I

Tabla de contenidos II

Introducción

Introducción

- Los Modelos Lineales Generalizados (MLGs) son una aproximación unificada a la mayoría de los procedimientos utilizados en estadística aplicada.
- Generalizan los modelos lineales clásicos que se basan en la suposición de una distribución normal para la variable respuesta.
- Los MLGs son ampliamente utilizados en diversas disciplinas y presentan un marco teórico unificado para estimar parámetros.
- La genialidad de Nelder & Wedderburn (1972) radica en demostrar que muchos métodos estadísticos aparentemente no relacionados se pueden abordar con un mismo marco teórico.

Introducción

- Los MLGs son especialmente útiles cuando la suposición de normalidad en la variable respuesta no es razonable, como en el caso de respuestas categóricas, proporciones o conteos.
- Estos modelos son adecuados para datos con no normalidad y varianza no constante, lo que es común en encuestas de hogares.
- Las variables en las encuestas de hogares a menudo son de tipo conteo, binarias, etc., lo que hace que el análisis mediante MLGs sea relevante y útil.

Lectura de las bases de datos de hogares.

```
encuesta_hog <- readRDS("Imagenes/06_MLG1/ENIGH_HND_Hogar.rds")
encuesta_hog <- encuesta_hog %>% # Base de datos.
  transmute(
    LLAVE_HOGAR,
    estrato = haven::as_factor(F1_A0_ESTRATO),
    Area = haven::as_factor(F1_A0_AREA),
    ingreso_per  = ifelse(YDISPONIBLE_PER < 0 , 0 , YDISPONIBLE_PER) ,
    gasto_per = GASTO_CORRIENTE_HOGAR / CANTIDAD_PERSONAS,
    pobreza_LP = case_when(
      ingreso_per < 3046 & Area == "1. Urbana" ~ "1",
      ingreso_per < 1688  &
        Area == "1. Rural" ~ "1",
      TRUE ~ "0"
    ),
     TIPOVIVIENDA = haven::as_factor(F1_A1_P1_TIPOVIVIENDA),
     TIENEVEHICULOS = haven::as_factor(F2_A2_P1_TIENEVEHICULOS)
  )
```

Lectura de las bases de datos de personas.

```
encuesta_per <- readRDS("Imagenes/06_MLG1/ENIGH_HND_Pers.rds")
dim(encuesta_per)
```

```
[1] 32029   345
```

```
dim(encuesta_hog)
```

```
[1] 8746    8
```

```
encuesta <- inner_join(encuesta_hog, encuesta_per,
                       by = join_by(LLAVE_HOGAR))
dim(encuesta)
```

```
[1] 32029   352
```

```
rm(encuesta_per, encuesta_hog)
```

Definición del diseño muestral.

Definiendo el diseño muestral, esto se hace de forma análoga a la anterior.

```
library(srvyr)
library(survey)
diseno <-  encuesta %>% 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.

Las nuevas variables son definidas de la siguiente forma.

```
diseno <- diseno %>% mutate(
   Sexo = haven::as_factor(F2_A6_P3_SEXO),
   etnia = haven::as_factor(F2_A6_P5_ETNIA ),
  log_ingreso_per = log(ingreso_per  + 500),
  log_gasto_per = log(gasto_per  + 500)
)
```

Tablas de doble entrada para el tamaño El cálculo de tablas de doble entrada las obtenemos con así:

```
(tab_pobreza_sexo <- svyby(~pobreza_LP, ~Sexo,
      FUN = svytotal, design = as.svrepdesign(diseno),
      se=F, na.rm=T, ci=T, keep.var=TRUE))
```

	Sexo	pobreza_LP0	pobreza_LP1	se1	se2
1. Hombre	1. Hombre	4107143	527474	94184	26244
2. Mujer	2. Mujer	4498829	652667	91239	27859

Tablas de doble entrada para el tamaño

Sin embargo para la estimación de tamaños más simples podemos emplear la función.

```
tab <- svytable(~pobreza_LP + Sexo, design = diseno)
data.frame(tab)
```

pobreza_LP	Sexo	Freq
0	1. Hombre	4107143
1	1. Hombre	527474
0	2. Mujer	4498829
1	2. Mujer	652667

Tablas de doble entrada para el proporción

Al hacer uso de la función `svymean` es posible estimar al proporciones.

```
(tab_pobreza_sexo <- svyby(~pobreza_LP, ~Sexo,
      FUN = svymean, design = as.svrepdesign(diseno),
      se=F, na.rm=T, ci=T, keep.var=TRUE))
```

	Sexo	pobreza_LP0	pobreza_LP1	se1	se2
1. Hombre	1. Hombre	0.8862	0.1138	0.0056	0.0056
2. Mujer	2. Mujer	0.8733	0.1267	0.0053	0.0053

Tablas de doble entrada para el proporción

En forma alternativa es posible usar la función `prop.table` del paquete base.

```
prop.table(tab, margin = 2) %>% data.frame()
```

pobreza_LP	Sexo	Freq
0	1. Hombre	0.8862
1	1. Hombre	0.1138
0	2. Mujer	0.8733
1	2. Mujer	0.1267

Estas diferentes formas de proceder son de mucha importancia al momento de hacer uso de pruebas de independencia en tablas cruzadas.

Prueba de independencia F

Prueba de independencia F

La prueba de independencia F de Fisher permite analizar si dos variables dicotómicas están asociadas cuando la muestra a estudiar es demasiado pequeña y no se cumplen las condiciones para aplicar la prueba χ^2. Para utilizar esta técnica, tengamos en cuenta que la probabilidad estimada se escribe como:

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

Prueba de independencia F

Teniendo en cuenta esta expresión, la estadística $\chi 2$ de Pearson se define de la siguiente manera:

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

y la estadística de razón de verosimilitud se define como:

$$G^2 = 2 \times n_{++} \times \sum_r \sum_c p_{cr} \times \ln \left(\frac{p_{rc}}{\hat{\pi}_{rc}} \right)$$

donde, r es el número de filas y c representa el número de columnas, la prueba tiene $(R-1) \times (C-1)$ grados de libertad.

Correcciones del Estadístico Chi-Cuadrado en Encuestas

- La corrección del estadístico chi-cuadrado de Pearson se utiliza en análisis de datos de encuestas para ajustar el efecto de diseño.
- *Fay (1979, 1985)* y *Fellegi (1980)* fueron pioneros en proponer correcciones basadas en un efecto de diseño generalizado (GDEFF).
- *Rao y Scott (1984)*, junto con *Thomas y Rao (1987)*, ampliaron la teoría de las correcciones del efecto de diseño generalizado.
- El método de Rao-Scott es un estándar para el análisis de datos de encuestas categóricas en software como Stata y SAS.

Estadísticos de Prueba

- Los estadísticos de prueba Rao-Scott Pearson y razón de verosimilitud chi-cuadrado se utilizan para analizar la asociación en datos de encuestas categóricas.
- Estos estadísticos se calculan mediante correcciones basadas en efectos de diseño generalizados.
- Las correcciones de Rao-Scott son analíticamente más complicadas que el enfoque de Fellegi, pero se consideran más precisas.
- Son ampliamente utilizados en el análisis de datos de encuestas, especialmente en software estadístico como Stata, SAS y R.

Estadísticos de Prueba χ^2 y G^2

Los estadísticos de prueba Rao-Scott Pearson ajustados por diseño y razón de verosimilitud chi-cuadrado se calculan de la siguiente manera:

$$\chi^2_{(R-S)} = \chi^2_{(Pearson)} / GDEFF$$

y, para la estadística basada en la razón de verosimilitud se calcula como:

$$G^2_{(R-S)} = G^2 / GDEFF$$

donde el efecto generalizado del diseño ($GDEFF$) de Rao–Scott, está dado por

$$GDEFF = \frac{\sum_r \sum_c (1 - p_{rc})\, d^2\,(p_{rc}) - \sum_r (1 - p_{r+})\, d^2\,(p_{r+}) - \sum_c (1 - p_{+c})\, d^2\,(p_{+c})}{(R-1)\,(C-1)}$$

Prueba de independencia F

La estadística F para independencia basada en la chi-cuadrado de Pearson se calcula como sigue:

$$F_{R-S,Pearson} = \chi^2_{R-S} \Big/ \left[(R-1)\,(C-1)\right] \sim F_{(R-1)(C-1),(R-1)(C-1)df}$$

y, la estadística F para independencia basada en la razón de verosimilitudes se calcula como sigue:

$$F_{R-S,LRT} = G^2_{R-S} \Big/ \left(C-1\right) \sim F_{(C-1),df}$$

donde C es el número de columnas de la tabla cruzada

Prueba de independencia ChiSq

En R, el cálculo de las estadísticas chi-cuadrado y F se calculan usando la función summary como se muestra a continuación:

```
summary(tab, statistic = "Chisq")
```

```
          Sexo
pobreza_LP 1. Hombre 2. Mujer
         0   4107143  4498829
         1    527474   652667

    Pearson's X^2: Rao & Scott adjustment

data:  NextMethod()
X-squared = 12, df = 1, p-value = 2e-04
```

Se puede concluir que el estado de pobreza y el sexo no están relacionados con una confianza del 95%.

Prueba de independencia F

```r
summary(tab, statistic = "F")
```

```
           Sexo
pobreza_LP 1. Hombre 2. Mujer
         0   4107143  4498829
         1    527474   652667

    Pearson's X^2: Rao & Scott adjustment

data:  NextMethod()
F = 14, ndf = 1, ddf = 725, p-value = 2e-04
```

Estadístico de Wald

Estadístico de Wald

Este estadístico se aplica cuando ya se ha elegido un modelo estadístico (regresión lineal simple, regresión logística, entre otros).

El estadístico de prueba de Wald χ^2 para la hipótesis nula de independencia de filas y columnas en una tabla de doble entrada se define de la siguiente manera:

$$Q_{wald} = \hat{Y}^t \left(H\hat{V}\left(\hat{N}\right) H^t \right)^{-1} \hat{Y}$$

donde,

$$\hat{Y} = \left(\hat{N} - E\right)$$

es un vector de $R \times C$ de diferencias entre los recuentos de celdas observadas y esperadas, esto es, $\hat{N}_{rc} - E_{rc}$

La matriz $H\hat{V}\left(\hat{N}\right) H^t$, representa la matriz de varianza-covarianza estimada para el vector de diferencias.

Estadístico de Wald

La matriz H es la inversa de la matriz J dada por:

$$J = -\left[\frac{\delta^2 \ln PL\left(B\right)}{\delta^2 B}\right] \mid B = \hat{B}$$

$$\sum_h \sum_a \sum_i x_{hai}^t x_{hai} w_{hai} \hat{\pi}_{hai}\left(B\right)\left(1 - \hat{\pi}_{hai}\left(B\right)\right)$$

Bajo la hipótesis nula de independencia, el estadístico de wald se distribuye chi cuadrado con $(R-1) \times (C-1)$ grados de libertad,

$$Q_{wald} \sim \chi^2_{(R-1)\times(C-1)}$$

Estadístico de Wald

La transformación F del estadístico de Wald es:

$$F_{wald} = Q_{wald} \times \frac{df - (R-1)(C-1) + 1}{(R-1)(C-1)\,df} \sim F_{(R-1)(C-1),df-(R-1)(C-1)+1}$$

Prueba de independencia Wald

En R, para calcular el estadístico de Wald se hace similarmente al cálculo de los estadísticos anteriores usando la función summary como sigue:

```
summary(tab, statistic = "Wald")
```

```
           Sexo
pobreza_LP 1. Hombre 2. Mujer
         0   4107143  4498829
         1    527474   652667

    Design-based Wald test of association

data:  NextMethod()
F = 15, ndf = 1, ddf = 725, p-value = 1e-04
```

Se puede concluir que, con una confianza del 95% y basado en la muestra no hay relación entre el estado de pobreza y el sexo.

Prueba de independencia adjWald

El estadístco de Wald ajustado en R se se calcula similarmente al anterior y los resultados fueron similares:

```
summary(tab, statistic = "adjWald")
```

```
          Sexo
pobreza_LP 1. Hombre 2. Mujer
         0   4107143  4498829
         1    527474   652667

    Design-based Wald test of association

data:  NextMethod()
F = 15, ndf = 1, ddf = 725, p-value = 1e-04
```

Modelo log lineal

Modelo log lineal para tablas de contingencia

El término modelo log-lineal, que básicamente describe el papel de la función de enlace que se utiliza en los modelos lineales generalizados. Iniciaremos esta sección con los modelos log-lineales en tablas de contingencia. El modelo estadístico es el siguiente:

$$\log(p_{ijk}) = \mu + \lambda_i^X + \lambda_j^Y + \lambda_k^Z + \lambda_{ij}^{XY},$$

donde:

- p_{ijk} = la proporción esperada en la celda bajo el modelo.
- $\mu = \log(p_0) = \frac{1}{\#\ de\ celdas}$

Modelo log lineal para tablas de contingencia

El modelo log-lineal en R se ajusta utilizando la función `svyloglin` como sigue:

```
mod1 <- svyloglin(~pobreza_LP + Sexo + pobreza_LP:Sexo , diseno)
(s1 <- summary(mod1))
```

```
Loglinear model: svyloglin(~pobreza_LP + Sexo + pobreza_LP:Sexo, diseno)
                       coef       se         p
pobreza_LP1         0.99572 0.024751 0.000e+00
Sexo1              -0.07601 0.008243 2.932e-20
pobreza_LP1:Sexo1   0.03047 0.008141 1.820e-04
```

Los resultados muestran que, con una confianza del 95% el estado de pobreza es independiente del sexo, como se ha mostrado con las pruebas anteriores.

Modelo log lineal para tablas de contingencia

En la salida anterior se pudo observar que la interacción es no significativa, entonces, ajustemos ahora el modelo sin interacción:

```
  mod2 <- svyloglin(~pobreza_LP + Sexo, diseno)
(s2 <- summary(mod2))
```

```
Loglinear model: svyloglin(~pobreza_LP + Sexo, diseno)
                coef       se         p
pobreza_LP1  0.99341 0.024565 0.000e+00
Sexo1       -0.05287 0.006498 4.102e-16
```

Modelo log lineal para tablas de contingencia

Mediante un análisis de varianza es posible comparar los dos modelos.

```
anova(mod1, mod2)
```

```
Analysis of Deviance Table
 Model 1: y ~ pobreza_LP + Sexo
Model 2: y ~ pobreza_LP + Sexo + pobreza_LP:Sexo
Deviance= 12.52 p= 0.0003458
Score= 12.5 p= 0.0003505
```

De la anterior salida se puede concluir que, con una confianza del 95%, la interacción no es significativa en el modelo log-lineal ajustado.

Modelo de regresión logistica

Modelo de regresión logistica

Un modelo de regresión logística es un modelo matemático que puede ser utilizado para describir la relación entre un conjunto de variables independientes y una variable dicotomica Y. El modelo logístico se describe a continuación:

$$g(\pi(x)) = logit(\pi(x))$$

De aquí,

$$z = \ln\left(\frac{\pi(x)}{1-\pi(x)}\right) = B_0 + B_1 x_1 + \cdots + B_p x_p$$

Modelo de regresión logistica

La probabilidad estimada utilizando el modelo logístico es la siguiente:

$$\hat{\pi}(x) = \frac{\exp\left(X\hat{B}\right)}{1 - \exp\left(X\hat{B}\right)} = \frac{\exp\left(\hat{B}_0 + \hat{B}_1 x_1 + \cdots + \hat{B} x_p\right)}{1 - \exp\left(\hat{B}_0 + \hat{B}_1 x_1 + \cdots + \hat{B} x_p\right)}$$

$$\pi(x_i) = \frac{\exp(x_i B)}{1 - \exp(x_i B)}$$

La varianza del modelo de regresión logistica

La varianza de los parámetros estimados se calcula como sigue:

$$var\left(\hat{B}\right) = J^{-1} var\left(S\left(\hat{B}\right)\right) J^{-1}$$

con,

$$S\left(B\right) = \sum_{h}\sum_{a}\sum_{i} w_{hai} D_{hai}^{t}\left[\left(\pi_{hai}\left(B\right)\right)\left(1-\pi_{hai}\left(B\right)\right)\right]^{-1}\left(y_{hai}-\pi_{hai}\left(B\right)\right) = 0$$

$$D_{hai} = \frac{\delta\left(\pi_{hai}\left(B\right)\right)}{\delta B_{j}}$$

donde $j = 0, \ldots, p$

Prueba de Wald para los parámetros del modelo

El estadístico de Wald para la significancia de los parámetros del modelo se utiliza la razón de verosimilitud. En este caso se contrastan el modelo con todos los parámetros (modelo full) versus el modelo reducido, es decir, el modelo con menos parámetros (modelo reduced),

$$G = -2 \ln \left[\frac{L\left(\hat{\beta}_{MLE}\right)_{reduced}}{L\left(\hat{\beta}_{MLE}\right)_{full}} \right]$$

Intervalo de confianza

Para construir los intervalos de confianza se debe aplicar el función exponencial a cada parámetro,

$$\hat{\psi} = \exp\left(\hat{B}_1\right)$$

por ende, el intervalo de confianza es:

$$CI\left(\psi\right) = \exp\left(\hat{B}_j \pm t_{df,1-\frac{\alpha}{2}} se\left(\hat{B}_j\right)\right)$$

Modelo log lineal ajustado

En R se muestra el ajuste de un modelo logístico teniendo e cuenta el diseño muestral

```
mod_loglin <- svyglm(
  pobreza_LP ~ Area + TIPOVIVIENDA + TIENEVEHICULOS +
    etnia,
  family=quasibinomial, design=diseno %>%
    mutate(pobreza_LP = as.numeric(pobreza_LP)))
```

Modelo log lineal ajustado

La salida muestra que algunas covariables son significativas con una confianza del 95%.

term	estimate	std.error	statistic	p.value
(Intercept)	-1.3641	0.1708	-7.986	0.0000
Area2. Rural	-19.0585	0.0874	-217.940	0.0000
TIPOVIVIENDA2. Apartamento	-0.4459	0.1693	-2.634	0.0086
TIPOVIVIENDA3. Cuarto en mesón o cuartería	0.2542	0.1683	1.510	0.1315
TIPOVIVIENDA4. Local no construido para vivienda	0.9883	0.6069	1.628	0.1039
TIENEVEHICULOS2. No	1.3655	0.0890	15.349	0.0000
etnia2. Afrohondureño(a)	-1.5477	0.5986	-2.585	0.0099
etnia3. Negro(a)	-1.9177	0.4459	-4.301	0.0000
etnia4. Mestizo(a)	-0.6669	0.1637	-4.074	0.0001
etnia5. Blanco(a)	-1.0927	0.2192	-4.985	0.0000
etnia6. Otro (especifique)	-18.7503	0.7002	-26.780	0.0000

Intervalo de confianza para el modelo

Los intervalos de confianza en los cuales

	OR	2.5	97.5	exp(2.5)	exp(97.5)
(Intercept)	0.2556	-1.6994	-1.0287	0.1828	0.3575
Area2. Rural	0.0000	- 19.2301	- 18.8868	0.0000	0.0000
TIPOVIVIENDA2. Apartamento	0.6402	-0.7783	-0.1136	0.4592	0.8926
TIPOVIVIENDA3. Cuarto en mesón o cuartería	1.2894	-0.0763	0.5846	0.9265	1.7943
TIPOVIVIENDA4. Local no construido para vivienda	2.6866	-0.2032	2.1798	0.8161	8.8441
TIENEVEHICULOS2. No	3.9175	1.1908	1.5401	3.2897	4.6652
etnia2. Afrohondureño(a)	0.2127	-2.7231	-0.3724	0.0657	0.6891
etnia3. Negro(a)	0.1469	-2.7931	-1.0422	0.0612	0.3527
etnia4. Mestizo(a)	0.5133	-0.9882	-0.3455	0.3723	0.7078
etnia5. Blanco(a)	0.3353	-1.5230	-0.6623	0.2181	0.5156
etnia6. Otro (especifique)	0.0000	- 20.1249	- 17.3757	0.0000	0.0000

Plot de la distribución de los betas

```
g1 <- plot_summs(mod_loglin,
                 scale = TRUE,
                 plot.distributions = TRUE)
```

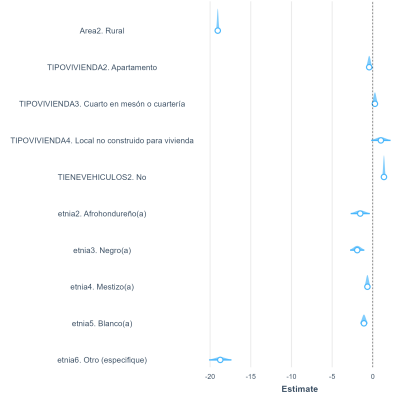


Figura 1: Intervalo de confianza para los coeficiente

Estadístico de Wald sobre los parámetros

El estadístico de Wald para el cada una de las variables del modelo se calcula a continuación con la función `regTermTest`, aquí se evalúa si el conjunto de categorías de la variable tipo de vivienda aporta significativamente al modelo.

```
regTermTest(model = mod_loglin, ~TIPOVIVIENDA)
```

```
Wald test for TIPOVIVIENDA
 in svyglm(formula = pobreza_LP ~ Area + TIPOVIVIENDA + TIENEVEHICULOS +
    etnia, design = diseno %>% mutate(pobreza_LP = as.numeric(pobreza_LP)),
    family = quasibinomial)
F =  4.43  on  3  and  715  df: p= 0.0043
```

$p < 0.0043$ indica que el resultado es altamente significativo.

Estadístico de Wald sobre los parámetros

```
regTermTest(model = mod_loglin, ~etnia)
```

```
Wald test for etnia
 in svyglm(formula = pobreza_LP ~ Area + TIPOVIVIENDA + TIENEVEHICULOS +
    etnia, design = diseno %>% mutate(pobreza_LP = as.numeric(pobreza_LP))
    family = quasibinomial)
F =  147.9  on  5  and  715  df: p= <2e-16
```

$p < 0.00000$ indica que el resultado es altamente significativo.

```
regTermTest(model = mod_loglin, ~Area)
```

```
Wald test for Area
 in svyglm(formula = pobreza_LP ~ Area + TIPOVIVIENDA + TIENEVEHICULOS +
    etnia, design = diseno %>% mutate(pobreza_LP = as.numeric(pobreza_LP))
    family = quasibinomial)
F =  47498  on  1  and  715  df: p= <2e-16
```

$p < 0.00000$ indica que el resultado es altamente significativo.

Efecto del modelo.

Para evaluar los efectos de la variable en el modelo:

```
effe_Area <- effect_plot(mod_loglin, pred = Area,
                         interval = TRUE)
effe_vehiculo <- effect_plot(mod_loglin, pred = TIENEVEHICULOS,
                          interval = TRUE)
effe_etnia <- effect_plot(mod_loglin, pred = etnia,
                          interval = TRUE)
effe_mod1 <-   (effe_Area | effe_vehiculo)/(effe_etnia)
```

Efecto del modelo.

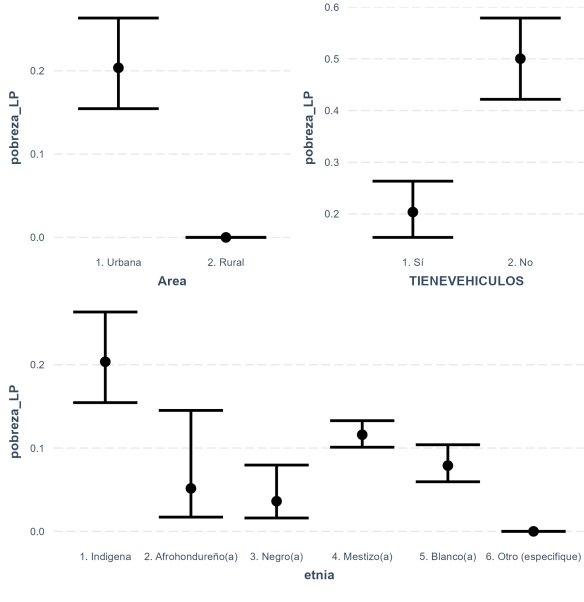


Figura 2: Efectos del modelo

Modelo log lineal ajustado con interacciones

```
mod_loglin_int <- svyglm(
  pobreza_LP ~ Area + etnia + TIPOVIVIENDA +
    TIPOVIVIENDA:TIENEVEHICULOS + TIPOVIVIENDA:etnia ,
  family = quasibinomial,
  design = diseno %>% mutate(pobreza_LP = as.numeric(pobreza_LP))
)
```

Modelo log lineal ajustado con interacciones

term	estimate	std.error	statistic	p.value
(Intercept)	-1.2373	0.1780	-6.9530	0.0000
Area2. Rural	-19.0753	0.0919	-207.6332	0.0000
etnia2. Afrohondureño(a)	-1.3528	0.5977	-2.2635	0.0239
etnia3. Negro(a)	-1.7751	0.4589	-3.8680	0.0001
etnia4. Mestizo(a)	-0.7659	0.1732	-4.4220	0.0000
etnia5. Blanco(a)	-1.2480	0.2413	-5.1721	0.0000
etnia6. Otro (especifique)	-18.8501	0.6958	-27.0917	0.0000
TIPOVIVIENDA2. Apartamento	-2.2227	0.8612	-2.5808	0.0101
TIPOVIVIENDA3. Cuarto en mesón o cuartería	-1.7915	0.7613	-2.3532	0.0189
TIPOVIVIENDA4. Local no construido para vivienda	-11.7490	1.0072	-11.6651	0.0000
TIPOVIVIENDA1. Casa:TIENEVEHICULOS2. No	1.3197	0.0929	14.2031	0.0000
TIPOVIVIENDA2. Apartamento:TIENEVEHICULOS2. No	2.4576	0.5602	4.3868	0.0000
TIPOVIVIENDA3. Cuarto en mesón o cuartería:TIENEVEHICULOS2. No	1.6842	0.4311	3.9066	0.0001
TIPOVIVIENDA4. Local no construido para vivienda:TIENEVEHICULOS2. No	18.1533	1.2783	14.2016	0.0000
etnia2. Afrohondureño(a):TIPOVIVIENDA2. Apartamento	-16.5554	0.9313	-17.7769	0.0000
etnia3. Negro(a):TIPOVIVIENDA2. Apartamento	-16.2910	0.9661	-16.8635	0.0000
etnia4. Mestizo(a):TIPOVIVIENDA2. Apartamento	0.8697	0.7113	1.2227	0.2218
etnia5. Blanco(a):TIPOVIVIENDA2. Apartamento	0.8202	0.8455	0.9701	0.3323
etnia2. Afrohondureño(a):TIPOVIVIENDA3. Cuarto en mesón o cuartería	0.9738	1.4689	0.6630	0.5076
etnia3. Negro(a):TIPOVIVIENDA3. Cuarto en mesón o cuartería	1.2118	1.3739	0.8820	0.3781
etnia4. Mestizo(a):TIPOVIVIENDA3. Cuarto en mesón o cuartería	1.7096	0.6391	2.6752	0.0076
etnia5. Blanco(a):TIPOVIVIENDA3. Cuarto en mesón o cuartería	2.7170	0.7201	3.7728	0.0002
etnia4. Mestizo(a):TIPOVIVIENDA4. Local no construido para vivienda	-2.2438	0.8797	-2.5508	0.0110

Plot de la distribución de los betas

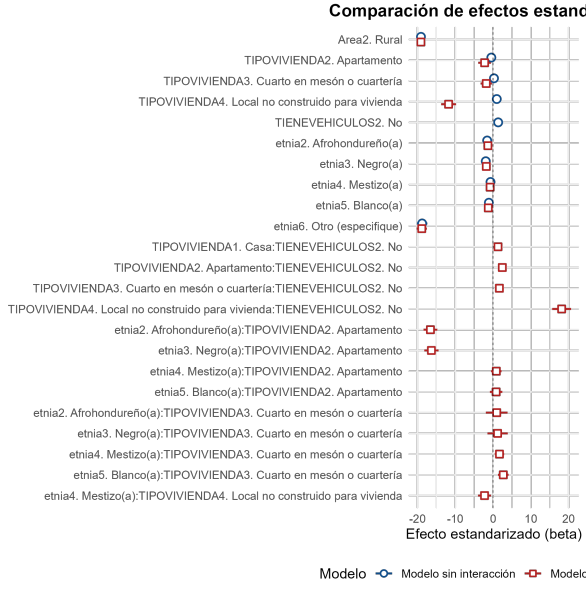


Figura 3: Comparando los modelos

Modelo log lineal ajustado

Observándose que con una confianza del 95% algunos de los parámetros del modelo son significativo.

	OR	2.5	97.5	exp(2.5)	exp(97.5)
(Intercept)	2.902e-01	-1.5867	-0.8880	2.046e-01	4.115e-01
Area2. Rural	0.000e+00	-19.2556	-18.8949	0.000e+00	0.000e+00
etnia2. Afrohondureño(a)	2.585e-01	-2.5262	-0.1794	8.000e-02	8.358e-01
etnia3. Negro(a)	1.695e-01	-2.6761	-0.8741	6.880e-02	4.172e-01
etnia4. Mestizo(a)	4.649e-01	-1.1060	-0.4259	3.309e-01	6.532e-01
etnia5. Blanco(a)	2.871e-01	-1.7218	-0.7743	1.787e-01	4.610e-01
etnia6. Otro (especifique)	0.000e+00	-20.2161	-17.4840	0.000e+00	0.000e+00
TIPOVIVIENDA2. Apartamento	1.083e-01	-3.9136	-0.5318	2.000e-02	5.875e-01
TIPOVIVIENDA3. Cuarto en mesón o cuartería	1.667e-01	-3.2862	-0.2968	3.740e-02	7.432e-01
TIPOVIVIENDA4. Local no construido para vivienda	0.000e+00	-13.7265	-9.7716	0.000e+00	1.000e-04
TIPOVIVIENDA1. Casa:TIENEVEHICULOS2. No	3.742e+00	1.1372	1.5021	3.118e+00	4.491e+00
TIPOVIVIENDA2. Apartamento:TIENEVEHICULOS2. No	1.168e+01	1.3577	3.5575	3.887e+00	3.508e+01
TIPOVIVIENDA3. Cuarto en mesón o cuartería:TIENEVEHICULOS2. No	5.388e+00	0.8377	2.5306	2.311e+00	1.256e+01
TIPOVIVIENDA4. Local no construido para vivienda:TIENEVEHICULOS2. No	7.654e+07	15.6436	20.6630	6.222e+06	9.415e+08
etnia2. Afrohondureño(a):TIPOVIVIENDA2. Apartamento	0.000e+00	-18.3839	-14.7270	0.000e+00	0.000e+00
etnia3. Negro(a):TIPOVIVIENDA2. Apartamento	0.000e+00	-18.1876	-14.3943	0.000e+00	0.000e+00
etnia4. Mestizo(a):TIPOVIVIENDA2. Apartamento	2.386e+00	-0.5268	2.2662	5.905e-01	9.643e+00
etnia5. Blanco(a):TIPOVIVIENDA2. Apartamento	2.271e+00	-0.8398	2.4802	4.318e-01	1.194e+01
etnia2. Afrohondureño(a):TIPOVIVIENDA3. Cuarto en mesón o cuartería	2.648e+00	-1.9101	3.8578	1.481e-01	4.736e+01
etnia3. Negro(a):TIPOVIVIENDA3. Cuarto en mesón o cuartería	3.359e+00	-1.4857	3.9093	2.263e-01	4.986e+01
etnia4. Mestizo(a):TIPOVIVIENDA3. Cuarto en mesón o cuartería	5.527e+00	0.4549	2.9643	1.576e+00	1.938e+01
etnia5. Blanco(a):TIPOVIVIENDA3. Cuarto en mesón o cuartería	1.513e+01	1.3031	4.1309	3.681e+00	6.223e+01
etnia4. Mestizo(a):TIPOVIVIENDA4. Local no construido para vivienda	1.061e-01	-3.9709	-0.5167	1.890e-02	5.965e-01

Estadístico de Wald sobre los parámetros

Evaluando las variables en el modelo

```
regTermTest(model = mod_loglin_int, ~TIPOVIVIENDA )
```

```
Wald test for TIPOVIVIENDA
 in svyglm(formula = pobreza_LP ~ Area + etnia + TIPOVIVIENDA + TIPOVIVIENI
    TIPOVIVIENDA:etnia, design = diseno %>% mutate(pobreza_LP = as.numeric
    family = quasibinomial)
F =  49.61  on  3  and  703  df: p= <2e-16
```

```
regTermTest(model = mod_loglin_int, ~Area)
```

```
Wald test for Area
 in svyglm(formula = pobreza_LP ~ Area + etnia + TIPOVIVIENDA + TIPOVIVIENI
    TIPOVIVIENDA:etnia, design = diseno %>% mutate(pobreza_LP = as.numeric
    family = quasibinomial)
F =  43112  on  1  and  703  df: p= <2e-16
```

Estadístico de Wald sobre los parámetros

Evaluando las variable región en el modelo

```
  regTermTest(model = mod_loglin_int, ~etnia )
```

```
Wald test for etnia
 in svyglm(formula = pobreza_LP ~ Area + etnia + TIPOVIVIENDA + TIPOVIVIENI
    TIPOVIVIENDA:etnia, design = diseno %>% mutate(pobreza_LP = as.numeric
    family = quasibinomial)
F =  150.2  on  5  and  703  df: p= <2e-16
```

Estadístico de Wald sobre los parámetros

Evaluando la interacción de los modelos.

```
regTermTest(model = mod_loglin_int, ~TIPOVIVIENDA:etnia)
```

```
Wald test for TIPOVIVIENDA:etnia
 in svyglm(formula = pobreza_LP ~ Area + etnia + TIPOVIVIENDA + TIPOVIVIENI
    TIPOVIVIENDA:etnia, design = diseno %>% mutate(pobreza_LP = as.numeric
    family = quasibinomial)
F =  126.7  on  9  and  703  df: p= <2e-16
```

```
regTermTest(model = mod_loglin_int, ~TIPOVIVIENDA:TIENEVEHICULOS)
```

```
Wald test for TIPOVIVIENDA:TIENEVEHICULOS
 in svyglm(formula = pobreza_LP ~ Area + etnia + TIPOVIVIENDA + TIPOVIVIENI
    TIPOVIVIENDA:etnia, design = diseno %>% mutate(pobreza_LP = as.numeric
    family = quasibinomial)
F =  110.9  on  4  and  703  df: p= <2e-16
```

Modelo log lineal ajustado con Q-Weighting

Realizando el modelo con los Q-Weighting

```
fit_wgt <- lm(Factor ~  Area + TIPOVIVIENDA +TIENEVEHICULOS,
              data = diseno$variables)
wgt_hat <- predict(fit_wgt)
summary(wgt_hat)
```

Min.	1st Qu.	Median	Mean	3rd Qu.	Max.
109.1	201	202.4	305.5	202.4	864.7

```
encuesta <- diseno$variables %>%  mutate(Factor2 = Factor/wgt_hat)

diseno_qwgt <- encuesta %>%
  as_survey_design(
    strata = estrato, ids = F1_A0_UPM ,
    weights = Factor2,
    nest = TRUE  ) %>%
  mutate(pobreza_LP = as.numeric(pobreza_LP))
```

Modelo log lineal ajustado con Q_Weighting

Defiendo la variable pobreza dentro de la base de datos.

```
mod_loglin_qwgt <-
  svyglm(
    pobreza_LP ~ Area + TIPOVIVIENDA + TIENEVEHICULOS +
    etnia,
    family = quasibinomial,
    design = diseno_qwgt
  )
(  tab_mod <- tidy(mod_loglin_qwgt) )
```

Modelo log lineal ajustado con Q_Weighting

term	estimate	std.error	statistic	p.value
(Intercept)	-1.3616	0.1708	-7.970	0.0000
Area2. Rural	-17.5326	0.0876	-200.120	0.0000
TIPOVIVIENDA2. Apartamento	-0.4452	0.1693	-2.629	0.0087
TIPOVIVIENDA3. Cuarto en mesón o cuartería	0.2541	0.1684	1.509	0.1318
TIPOVIVIENDA4. Local no construido para vivienda	0.9966	0.6068	1.642	0.1009
TIENEVEHICULOS2. No	1.3669	0.0890	15.358	0.0000
etnia2. Afrohondureño(a)	-1.5476	0.5989	-2.584	0.0100
etnia3. Negro(a)	-1.9181	0.4442	-4.318	0.0000
etnia4. Mestizo(a)	-0.6702	0.1638	-4.093	0.0000
etnia5. Blanco(a)	-1.1007	0.2196	-5.012	0.0000
etnia6. Otro (especifique)	-17.8557	0.7012	-25.464	0.0000

Plot de la distribución de los betas

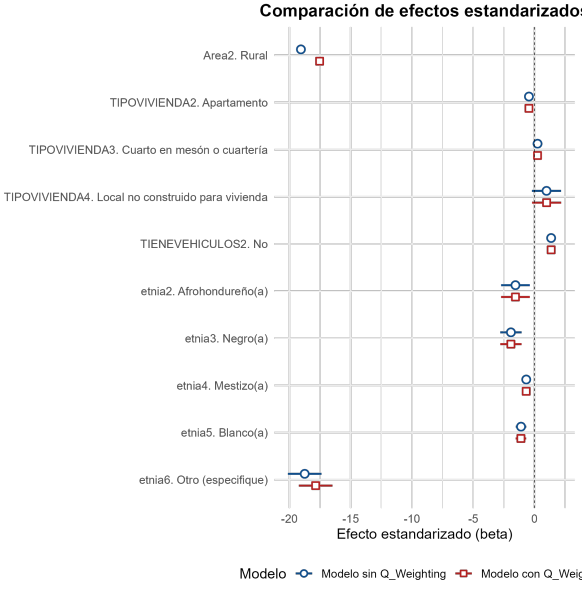


Figura 4: Comparando los modelos

Modelo log lineal ajustado

	OR	2.5	97.5	exp(2.5)	exp(97.5)
(Intercept)	0.2563	-1.6970	-1.0261	0.1832	0.3584
Area2. Rural	0.0000	-17.7046	-17.3606	0.0000	0.0000
TIPOVIVIENDA2. Apartamento	0.6407	-0.7777	-0.1128	0.4595	0.8934
TIPOVIVIENDA3. Cuarto en mesón o cuartería	1.2893	-0.0765	0.5847	0.9263	1.7945
TIPOVIVIENDA4. Local no construido para vivienda	2.7091	-0.1947	2.1879	0.8231	8.9166
TIENEVEHICULOS2. No	3.9233	1.1922	1.5417	3.2943	4.6724
etnia2. Afrohondureño(a)	0.2127	-2.7234	-0.3719	0.0656	0.6894
etnia3. Negro(a)	0.1469	-2.7902	-1.0460	0.0614	0.3514
etnia4. Mestizo(a)	0.5116	-0.9917	-0.3487	0.3709	0.7056
etnia5. Blanco(a)	0.3326	-1.5319	-0.6695	0.2161	0.5119
etnia6. Otro (especifique)	0.0000	-19.2324	-16.4790	0.0000	0.0000

Estadístico de Wald sobre los parámetros

```
  regTermTest(model = mod_loglin_qwgt, ~TIENEVEHICULOS)
```

```
Wald test for TIENEVEHICULOS
 in svyglm(formula = pobreza_LP ~ Area + TIPOVIVIENDA + TIENEVEHICULOS +
    etnia, design = diseno_qwgt, family = quasibinomial)
F =  235.9  on  1  and  715  df: p= <2e-16
```

```
  regTermTest(model = mod_loglin_qwgt, ~Area)
```

```
Wald test for Area
 in svyglm(formula = pobreza_LP ~ Area + TIPOVIVIENDA + TIENEVEHICULOS +
    etnia, design = diseno_qwgt, family = quasibinomial)
F =  40048  on  1  and  715  df: p= <2e-16
```

Estadístico de Wald sobre los parámetros

```
regTermTest(model = mod_loglin_qwgt, ~etnia)
```

```
Wald test for etnia
 in svyglm(formula = pobreza_LP ~ Area + TIPOVIVIENDA + TIENEVEHICULOS +
    etnia, design = diseno_qwgt, family = quasibinomial)
F =  134  on  5  and  715  df: p= <2e-16
```

Efecto del modelo.

```
effe_Area <- effect_plot(mod_loglin_qwgt, pred = Area,
                         interval = TRUE)
effe_vehiculo <- effect_plot(mod_loglin_qwgt, pred = TIENEVEHICULOS,
                          interval = TRUE)
effe_etnia <- effect_plot(mod_loglin_qwgt, pred = etnia,
                          interval = TRUE)
effe_mod3 <-   (effe_Area | effe_vehiculo)/(effe_etnia)
```

Efecto del modelo.

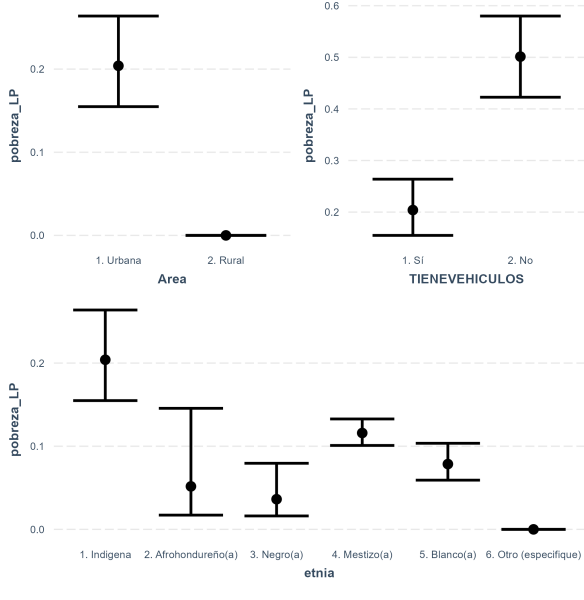


Figura 5: Efecto del modelo

Modelos lineales generalizados (Variable categóricas)

Introducción

- Leslie Kish destaca que en inferencia estadística, no se pueden asumir variables aleatorias independientes e idénticamente distribuidas en la mayoría de los casos prácticos. Las muestras no se dan, deben ser seleccionadas, asignadas o capturadas, y el tamaño de la muestra no es un número fijo, sino una variable aleatoria.
- En teoría de muestreo, las características de interés son parámetros, no realizaciones de variables aleatorias. Se requiere un experimento que defina todos los posibles resultados y una sigma-álgebra para hablar de una variable aleatoria.
- Al estimar la tasa de desempleo, el estado "desempleado" no es una realización de una variable aleatoria, sino una caracterización del estado de la naturaleza de un individuo en el momento de la medición.

Introducción

- La inferencia estadística es aplicable solo en el muestreo aleatorio simple con reemplazo, donde se cumplen las propiedades de independencia e idéntica distribución. En la selección de muestras, existen dos escenarios generales: selección con reemplazo y selección sin reemplazo.
- Selección sin reemplazo no permite construir muestras aleatorias independientes ni idénticamente distribuidas debido a la falta de independencia en el proceso de selección.
- En muestreo con reemplazo, las variables aleatorias X_i conforman una muestra aleatoria independiente e idénticamente distribuida, lo que es esencial para aplicar la teoría de inferencia estadística.

Introducción

- Para que las variables X_i tengan la misma esperanza y varianza que la población, se requiere que la probabilidad de selección sea igual para todos los individuos en la población.
- En muestreo aleatorio simple con reemplazo, las propiedades de estimadores clásicos, como la media muestral, coinciden con los resultados de inferencia clásica.
- En encuestas con selección no aleatoria, es necesario incluir los pesos de muestreo en análisis estadísticos para obtener resultados confiables en técnicas como regresiones y varianzas del promedio.

Modelos de superpoblación.

- Se asume que la estimación de máxima verosimilitud es apropiada para muestras aleatorias simples en modelos de regresión y otros.
- El modelo considera una función de densidad poblacional $f(y|\theta)$ con θ como el parámetro de interés.
- Se presenta un ejemplo con 100 realizaciones de variables Bernoulli independientes con $\theta = 0.3$.

```
1 1 0 1 0 0 0 0 0 0 0 1 1 0 0 0 0 0 0 0 1 0 0 1 0 0 0 0 0 0 0 1 0
0 1 0 0 0 1 0 0 0 0 1 0 0 0 0 0 0 0 1 1 0 1 0 0 0 1 1 0 0 0 1 1 0
0 1 0 0 0 1 0 0 1 1 1 0 0 0 0 0 1 0 1 0 1 0 0 0 0 0 0 0 0 0 1 0 1 0
```

- La población finita generada se basa en un modelo de superpoblación y contiene 28 éxitos.

Primer proceso inferencial: el modelo

- La inferencia se basa en la distribución binomial con parámetro 0.3.
- El estimador insesgado de mínima varianza es el promedio poblacional, utilizando todos los datos de la población.
- Se realiza una simulación de Monte Carlo con 1000 repeticiones para corroborar la propiedad del estimador insesgado.
- Se obtiene un valor estimado de θ (0.3) y se calcula el valor esperado (insesgado) de acuerdo a la simulación.

Primer proceso inferencial: el modelo

```
N = 100
theta = 0.3
nsim1 = 1000
Est0=rep(NA,nsim1)

for(i in 1:nsim1){
y=rbinom(N, 1, theta)
Est0[i]=mean(y)
}

Esp0 = mean(Est0)

cbind(theta, Esp0)
```

theta	Esp0
0.3	0.2988

Segundo proceso inferencial: el muestreo

- En el segundo proceso inferencial se considera que los valores de la medición son fijos pero desconocidos, y no siguen ningún modelo probabilístico.
- Se divide la población en conglomerados (hogares en este ejemplo) y se toma una muestra de estos hogares.

```
(1 1 0) (1 0) (0 0 0 0 0 0 1) (1 0) (0 0 0 0 0 0 1) (0 0
1) (0 0 0 0 0 0 0 1) (0 0 1) (0 0 0 1) (0 0 0 0 1) (0 0
0 0 0 0 0 1) (1 0) (1 0) (0 0 1) (1 0) (0 0 1) (1 0) (0 1)
(0 0 0 1) (0 0 1) (1 1 0) (0 0 0 0 1) (0 1) (0 1) (0 0 0 0
0 0 0 0 0 1) (0 1) (0)
```

- Se realiza un censo en cada hogar seleccionado, y la selección de hogares se hace aleatoriamente, sin reemplazo y con probabilidades de inclusión proporcionales al tamaño del hogar.

Segundo proceso inferencial: el muestreo

- Bajo el esquema anterior, el estimador insesgado para la proporción de desempleados es calculado como

$$\bar{y}_{\pi S} = \sum_{i \in S_I} \frac{t_{y_i}}{\pi_{Ii}} = \frac{\sum_{i \in S_I} \bar{y}_i}{n_I}$$

.

- También se presenta un estimador ingenuo que ignora el diseño de muestreo y se calcula como

$$\bar{y}_S = \frac{\sum_{i \in S_I} t_{y_i}}{\sum_{i \in S_I} N_i}$$

.

Simulación

1. Configuran los parámetros iniciales, el tamaño de la población (N) y el valor verdadero del parámetro de interés (θ), que es la proporción de éxitos en la población.

```
library(TeachingSampling)
N=100
theta=0.3
```

2. Genera una población de N elementos mediante la función `rbinom`, que simula variables aleatorias binomiales con parámetro θ.

```
set.seed(1234)
y=rbinom(N, 1, theta)
```

3. Calcular θ para la población.

```
theta_N=mean(y)
```

Simulación

4. Definir una estructura de conglomerados

```
clus=c(0,which((y[-N]-y[-1])!=0)+1)
NI=(length(clus)-1)
Ind=matrix(0, nrow=N, ncol=NI)
Tamaños=clus[-1]-clus[-(length(clus))]

for(l in 1:(length(clus)-1)){
a=(clus[l]+1):clus[l+1]
Ind[a,l]=a
}
```

5. Seleccionar una muestra de conglomerados 30% y realizar censo al interior

```
nI=floor(NI*0.3)
```

6. Estimar θ haciendo uso de los estimadores anteriores.

Simulación

7. Repetir el proceso 1000 veces y calcular la esperanza de los estimadores.

```
nsim2 = 1000
Est1 <- Est2 <- NA
for(j in 1:nsim2) {
  res <- S.piPS(nI, Tamaños)
  sam <- res[, 1]
  Ind.sam = Ind[, sam]
  Tamaños.sam = Tamaños[sam]
  #-------Espacio para las medias
  medias = matrix(NA)
  for (k in 1:ncol(Ind.sam)) {
    medias[k] = mean(y[Ind.sam[, k]])
  }
  Est1[j] = mean(medias)
  Est2[j] = sum(Tamaños.sam * medias) / sum(Tamaños)
}
```

Resultado de la simulación

- El primer estimador es insesgado (su esperanza equivale al parámetro de la población finita) dado que tiene encuenta el diseño muestral.
- El segundo estimador es sesgado porque no tiene en cuenta el diseño de muestreo

```
Esp1 = mean(Est1) ; Esp2 = mean(Est2)

cbind(theta_N, Esp1, Esp2)
```

theta_N	Esp1	Esp2
0.22	0.2216	0.0936

Inferencia doble: los modelos y el muestreo

Inferencia Doble:

Asuma que las variables de interés siguen un modelo probabilístico y se realiza un muestreo de una población finita. En este proceso, tanto el modelo como el diseño de muestreo y la medida de probabilidad que rige las superpoblaciones son factores fundamentales en la inferencia del parámetro de interés.

Máxima Pseudo-Verosimilitud (MPV):

Dado que el diseño de muestreo es complejo, no es apropiado utilizar técnicas clásicas como la máxima verosimilitud. En cambio, se recurre a la MPV, que considera las ponderaciones del diseño de muestreo. Para el ejemplo de las proporciones, el estimador $\bar{y}_{\pi S}$ cumple la siguiente relación:

$$E_{\xi p}\left(\bar{y}_{\pi S}\right) \quad = \quad E_{\xi} E_{p}\left(\bar{y}_{\pi S} \mid Y\right) = E_{\xi}\left(\bar{y}_{U}\right) = \theta = 0.3$$

Método de Pseudo máxima verosimilitud

Sea y_i el vector de observaciones los cuales provienen de los vectores aleatorios Y_i para $i \in U$. Suponga también que $Y_1, \ldots, Y_N$ son IID con función de densidad $f(y, \theta)$. Si todos los elementos de la población finita U fueran conocidos la función de log-verosimilitud estaría dada por:

$$l(\theta) = \sum_{i=1}^{n} \ln[w_i f(y_i, \theta)]$$

Calculando las derivadas parciales de $l(\theta)$ con respecto a θ e igualando a cero tenemos un sistema de ecuaciones como sigue:

$$\frac{\partial l(\theta)}{\partial \theta} = \sum_{i=1}^{n} w_i u_i(\theta) = 0$$

donde $ui = \partial \ln[f(y_i, \theta)]/\partial\theta$ es el vector de "score" de elementos $i, i \in n$ ponderado por w_i, ahora definiremos T como:

Método de Pseudo máxima verosimilitud

Si se cumplen las condiciones de regularidad (Ver Pag 281 de Cox and Hinkley 1974[1]), es posible considerar a

$$T = \sum_{i \in U} u_i(\theta)$$

como un vector de totales. La estimación T se puede hacer mediante

$$\hat{T} = \sum_{i \in S} w_i u_i(\theta),$$

donde w_i son los pesos previamente definidos.

[1] Cox, D. R., & Hinkley, D. V. (1974). Theoretical Statistics Chapman and Hall, London. See Also.

Método de Pseudo máxima verosimilitud (Definición)

Un estimador de Máxima Pseudo Verosimilitud (MVP) $\hat{\theta}_{MPV}$ de θ_U será la solución de las ecuaciones de Pseudo-Verosimilitud dadas por

$$\hat{T} = \sum_{i \in S} w_i u_i \left(\theta\right) = 0,$$

Mediante la linealización de Taylor y considerando los resultados de *Binder(1983)*, podemos obtener una varianza asintóticamente insesgada de la siguiente forma:

$$V_p\left(\hat{\theta}_{MPV}\right) \approx \left[J\left(\theta_U\right)\right]^{-1} V_p\left(\hat{T}\right) \left[J\left(\theta_U\right)\right]^{-1}$$

Donde

$$J\left(\theta_U\right) = \left.\frac{\partial T\left(\theta\right)}{\partial \theta}\right|_{\theta=\theta_U} = \sum_{i \in U} \left.\frac{\partial u_i\left(\theta\right)}{\partial \theta}\right|_{\theta=\theta_U}$$

Método de Pseudo máxima verosimilitud (Definición)

El estimador de la varianza

$$\hat{V}_p\left(\hat{\theta}_{MPV}\right) = \left[\hat{J}\left(\hat{\theta}_{MPV}\right)\right]^{-1} \hat{V}_p\left(\hat{T}\right) \left[\hat{J}\left(\hat{\theta}_{MPV}\right)\right]^{-1}$$

con

$$\hat{J}\left(\hat{\theta}_{MPV}\right) = \left.\frac{\partial \hat{T}\left(\theta\right)}{\partial \theta}\right|_{\theta=\hat{\theta}_{MPV}} = \left.\sum_{i \in s} w_i \frac{\partial u_i\left(\theta\right)}{\partial \theta}\right|_{\theta=\hat{\theta}_{MPV}}$$

$\hat{V}_p\left(T\right)$ es la matriz de varianza estimada y $\hat{V}_p\left(\hat{T}\right)$ es un estimador consistente para la varianza.

Introducción al GLM

Introducción al GLM

Un modelo lineal generalizado tiene tres componentes básicos:

- **Componente aleatoria**: Identifica la variable respuesta $(y_1, \dots, y_N)$ y su distribución de probabilidad.
- **Componente sistemática**: Especifica las variables explicativas (independientes o predictoras) utilizadas en la función predictora lineal.

  Las covariables $x_1, \dots, x_k$ producen un predictor lineal η_i que resulta de la combinación lineal $\eta_i = \sum_{j=1}^{k} x_{ij}\beta_j$ donde x_{ij} es el valor del j-ésimo predictor en el i-ésimo individuo, e $i = 1, \dots, N$.

Introducción al GLM

- **Función link**: Es una función del valor esperado de Y , $E(Y)$, como una combinación lineal de las variables predictoras.

  Se denota el valor esperado Y como $\mu = E(Y)$, entonces la función *link* especifica una función

$$g(\mu) = \sum_{j=1}^{k} x_{ij}\beta_j.$$

  Así, la función $g(\cdot)$ realciona las componentes aleatoria y sistemática. De este modo, para $i = 1, \ldots, N$

$$\mu_i = E(Y_i)$$
$$\eta_i = g(\mu_i) = \sum_j \beta_j x_{ij}$$

Introducción al GLM

- Todos los modelos se pueden incluir dentro de la llamada familia exponencial de distribuciones

$$f\left(y_i \mid \theta_i\right) = a\left(\theta_i\right) b\left(\theta_i\right) \exp\left[y_i Q\left(\theta_i\right)\right]$$

de modo que $Q\left(\theta\right)$ recibe el nombre de *parámetro natural*. Además, $a(\cdot)$ y $b(\cdot)$ son funciones conocidas.

- Los modelos de regresión lineal típicos para respuestas continuas son un caso particular de los GLM.

Modelo Gamma

Modelo Gamma para Variable Continua

- La función de enlace $g(\cdot)$ para el GLM con una variable dependiente distribuida por un modelo Gamma es el recíproco, $\frac{1}{\mu_i}$.
- El valor esperado de y_i observado ($E(y_i) = \mu_i$) se relaciona con las variables de entrada mediante la ecuación:

$$\frac{1}{\mu_i} = B_0 + B_1 x_1$$

- De manera equivalente, se puede expresar como:

$$\mu_i = \frac{1}{B_0 + B_1 x_1}$$

Estimador de momentos de la distribución gamma**

```
library(ggplot2)
encuesta_temp <- filter(encuesta, ingreso_per < 75000)
nrow(encuesta) -nrow(encuesta_temp)
```

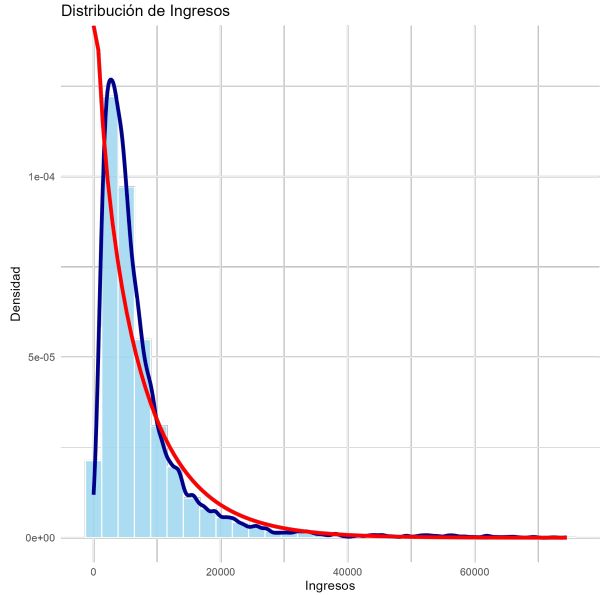
```
[1] 90
```

```
x <- encuesta_temp$ingreso_per
n = length(x)
shape1 = (n*mean(x)^2)/sum((x-mean(x))^2)
rate1 = (n*mean(x))/sum((x-mean(x))^2)
c(shape1 = shape1, rate1 = rate1)
```

```
   shape1     rate1
0.8759262 0.0001189
```

La densidad empírica para el ingreso.

La linea roja se obtiene con la estimación de los parámetros, la linea azul oscura es la densidad empírica.



Creación de nuevas variables.

Las nuevas variables son definidas de la siguiente forma.

```
diseno_qwgt2 <- diseno_qwgt %>%
  mutate(ingreso_per2  = ingreso_per + 1,
         edad = case_when(F2_A6_P4_EDAD < 16 ~ "0 - 15",
                          F2_A6_P4_EDAD < 21 ~ "16 - 20",
                          F2_A6_P4_EDAD < 26 ~ "21 - 25",
                          F2_A6_P4_EDAD < 31 ~ "26 - 30",
                          F2_A6_P4_EDAD < 36 ~ "31 - 35",
                          F2_A6_P4_EDAD < 41 ~ "36 - 40",
                          F2_A6_P4_EDAD < 46 ~ "41 - 45",
                          F2_A6_P4_EDAD < 51 ~ "46 - 50",
                          TRUE ~ "51 o más"))
```

Modelo gamma

El modelo ajustado es el siguiente:

```
modelo <- svyglm(formula = ingreso_per2  ~ edad + Area+   Area*Sexo +
                   TIPOVIVIENDA + TIENEVEHICULOS  ,
                   design = diseno_qwgt2,
                  family = Gamma(link = "inverse"))
```

Coeficientes del modelo

```
broom::tidy(modelo) %>% mutate(across(
  c("estimate", "std.error"),
  ~ format(., scientific = FALSE, digits = 10)
))
```

term	estimate	std.error	statistic	p.value
(Intercept)	0.000134095797413	0.000004879101479	27.484	0.0000
edad16 - 20	-0.000023457973014	0.000004058354936	-5.780	0.0000
edad21 - 25	-0.000037270729329	0.000004119436312	-9.047	0.0000
edad26 - 30	-0.000040814507290	0.000003871784162	-10.541	0.0000
edad31 - 35	-0.000042373247456	0.000004132405187	-10.254	0.0000
edad36 - 40	-0.000031885798792	0.000003230370724	-9.871	0.0000
edad41 - 45	-0.000046663865073	0.000003646574940	-12.797	0.0000
edad46 - 50	-0.000051947783074	0.000005046249329	-10.294	0.0000
edad51 o más	-0.000070681232531	0.000004720230395	-14.974	0.0000
Area2. Rural	0.000068030883610	0.000013591468393	5.005	0.0000
Sexo2. Mujer	0.000003079288368	0.000001363479002	2.258	0.0242
TIPOVIVIENDA2. Apartamento	-0.000013940689790	0.000006864153181	-2.031	0.0426
TIPOVIVIENDA3. Cuarto en mesón o cuartería	0.000039901919187	0.000007829384313	5.096	0.0000
TIPOVIVIENDA4. Local no construido para vivienda	0.000102670082103	0.000028756240487	3.570	0.0004
TIENEVEHICULOS2. No	0.000086563645724	0.000004443112265	19.483	0.0000
Area2. Rural:Sexo2. Mujer	-0.000012266830622	0.000005828844299	-2.104	0.0357

Modelo gamma

Es útil la estimación de la dispersión que ofrece *svyglm* de forma predeterminada dado que no tiene en cuenta la información especial sobre la dispersión que se puede calcular utilizando la distribución Gamma. **No todos los GLM tienen una forma mejorada y específica del modelo para estimar**.

```
(alpha = MASS::gamma.dispersion(modelo))
```

```
[1] 0.5617
```

```
mod_s <- summary(modelo, dispersion = alpha)
mod_s$dispersion
```

```
     variance   SE
[1,]     1.43 0.21
```

Predicción e intervalos de conficanza.

Una vez estimado los coeficientes, se estiman los intervalos de confianza para la predicción como sigue:

```
pred <- predict(modelo, type = "response", se = T)

pred_IC <- data.frame(confint(pred))

colnames(pred_IC) <- c("Lim_Inf", "Lim_Sup")

pred <- bind_cols(data.frame(pred), pred_IC)

pred$ingreso_per2 <- encuesta$ingreso_per + 1

pred %>% slice(1:15L)
```

Utilizando la función predict

response	SE	Lim_Inf	Lim_Sup	ingreso_per2
4789	302.6	4196	5382	1.00
3906	195.5	3522	4289	1.00
3464	163.3	3144	3784	1.00
3578	171.9	3241	3915	1.00
5514	144.9	5230	5798	1.00
4587	278.7	4041	5133	1.00
8179	917.4	6381	9978	1.00
4947	351.4	4259	5636	1.00
15769	784.7	14231	17307	1.00
6668	214.1	6248	7087	1.00
6533	192.1	6157	6910	1.00
6668	214.1	6248	7087	1.00
4478	160.9	4163	4794	19.75
4488	160.1	4174	4801	19.75
3838	111.0	3620	4055	19.75

Scaterplot de la predicción

Intervalos de confianza para la predicción en cada punto.

```
pd <- position_dodge(width = 0.2)
plot_pred <- ggplot(pred %>% slice(1:1000L),
       aes(x = ingreso_per2 , y = response)) +
  geom_errorbar(aes(ymin = Lim_Inf,
                    ymax = Lim_Sup),
                width = .1,
                linetype = 1) +
  geom_point(size = 2, position = pd) +
  theme_bw()
```

Scaterplot de la predicción

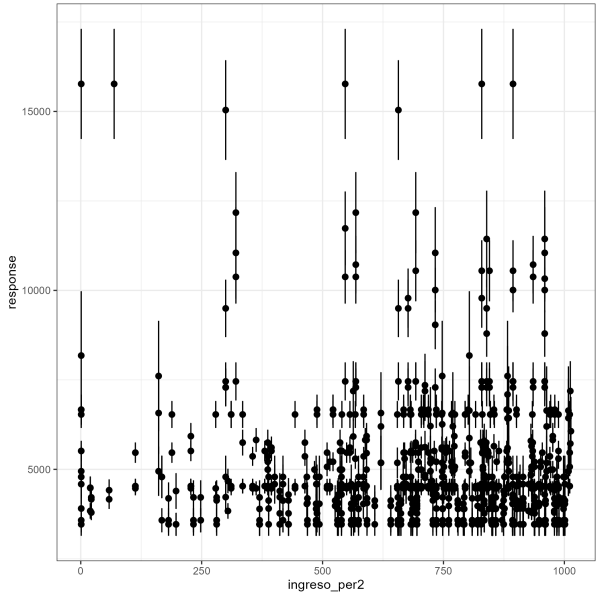


Figura 7: Intervalo de confizan para la predicción

¡Gracias!

Email: andres.gutierrez@cepal.org