Análisis de encuestas de hogares con R

Modulo 7: Modelos lineales generalizados (Variable categóricas)

CEPAL - Unidad de Estadísticas Sociales

Tabla de contenidos I

Introducción

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	0.3007

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

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

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

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

Donde

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

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 $i = 1, \dots, N$

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

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.

Lectura de la base y definición del diseño muestral

Para ilustrar el uso de los GLM en encuestas de hogares, se estimará un modelo para el ingreso empleando un modelo Gamma.

```
library(survey)
library(srvyr)
encuesta <- readRDS("../Data/encuesta.rds")
diseno <- encuesta %>%
  as_survey_design(
    strata = Stratum,
    ids = PSU,
    weights = wk,
    nest = T
  )
```

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:

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

Definir nueva variable

Creando nuevas variables en la base de datos.

```
diseno <- diseno %>%
  mutate(
  pobreza = ifelse(Poverty != "NotPoor", 1, 0),
  desempleo = ifelse(Employment == "Unemployed", 1, 0))
```

Estimador de momentos de la distribución gamma

```
library(ggplot2)

x <- encuesta$Income
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
1.442897 0.002494
```

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.

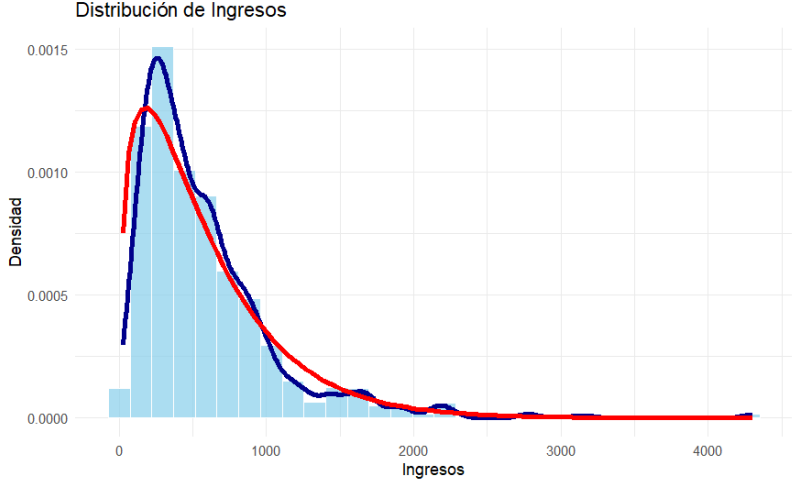

Figura 1: Modelo gamma para el ingreso

qweigth para modelo gamma ingreso

Ahora bien, para el ajuste del modelo Gamma, primero se definen los pesos qweigth como sigue:

```
mod_qw <- lm(wk ~ Age + Sex + Region + Zone,
             data = encuesta)

encuesta$wk2 <-   encuesta$wk/predict(mod_qw)
diseno <- encuesta %>%
  as_survey_design( strata = Stratum,
    ids = PSU, weights = wk2,
    nest = T)
```

Modelo gamma

El modelo ajustado es el siguiente:

```
modelo <- svyglm(formula = Income ~ Age + Sex +
                   Region + Zone,
                   design = diseno,
                  family = Gamma(link = "inverse"))
broom::tidy(modelo)
```

term	estimate	std.error	statistic	p.value
(Intercept)	0.0024	2e-04	10.9726	0.0000
Age	0.0000	0e+00	-1.2838	0.2019
SexMale	-0.0001	0e+00	-1.8423	0.0681
RegionSur	-0.0001	2e-04	-0.2316	0.8173
RegionCentro	0.0000	2e-04	0.1383	0.8903
RegionOccidente	0.0002	2e-04	1.0196	0.3101
RegionOriente	0.0000	3e-04	0.0319	0.9746
ZoneUrban	-0.0009	2e-04	-4.8762	0.0000

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] 0.4831
```

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

```
     variance   SE
[1,]    0.591 0.09
```

Modelo Gamma

Los coeficientes del modelo también se pueden obtener de la siguiente manera:

```
mod_s$coefficients
```

	Estimate	Std. Error	t value	Pr(>|t|)
(Intercept)	0.0024	2e-04	10.9726	0.0000
Age	0.0000	0e+00	-1.2838	0.2019
SexMale	-0.0001	0e+00	-1.8423	0.0681
RegionSur	-0.0001	2e-04	-0.2316	0.8173
RegionCentro	0.0000	2e-04	0.1383	0.8903
RegionOccidente	0.0002	2e-04	1.0196	0.3101
RegionOriente	0.0000	3e-04	0.0319	0.9746
ZoneUrban	-0.0009	2e-04	-4.8762	0.0000

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$Income <- encuesta$Income

pred$Age <- encuesta$Age

pred %>% slice(1:10L)
```

Utilizando la función predict

response	SE	Lim_Inf	Lim_Sup	Income	Age
456.8	41.80	374.9	538.8	409.87	68
434.4	38.07	359.8	509.0	409.87	56
423.5	37.33	350.3	496.7	409.87	24
441.2	39.35	364.1	518.3	409.87	26
416.7	37.78	342.6	490.7	409.87	3
436.1	38.36	360.9	511.3	823.75	61
423.2	37.34	350.0	496.4	823.75	23
433.8	39.46	356.5	511.1	823.75	5
416.0	37.85	341.9	490.2	823.75	1
453.4	40.94	373.1	533.6	90.92	59

Scaterplot de la predicción

Intervalos de confiza para la predicción en cada punto.

```
pd <- position_dodge(width = 0.2)
ggplot(pred %>% slice(1:100L),
       aes(x = Age , 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

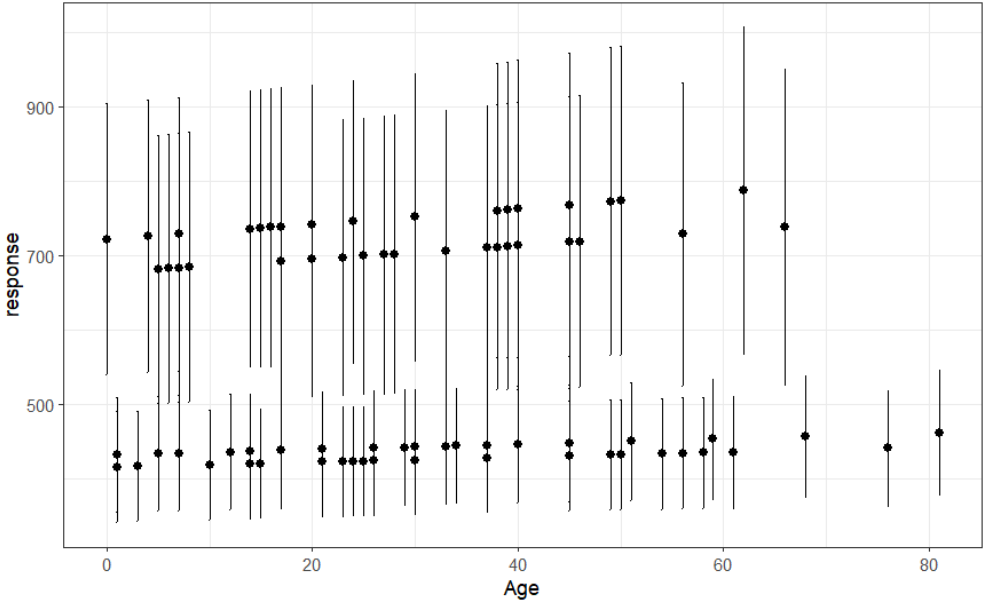

Figura 2: Intervalo de confizan para la predicción

Modelos multinomial

Modelos multinomial

- Extensión natural del modelo de regresión logística binomial.
- Utilizado para analizar variables con tres o más categorías distintas, especialmente apropiado para respuestas nominales en encuestas

Consideraciones para el Ajuste del Modelo Multinomial:

1. La variable dependiente debe ser nominal.
2. Se emplean una o más variables independientes, que pueden ser continuas, ordinales o nominales (incluyendo variables dicotómicas).
3. Requiere independencia de las observaciones y categorías mutuamente excluyentes y exhaustivas en la variable dependiente.

Modelo multinomial

4. Se evita la **multicolinealidad**, que surge de la alta correlación entre variables independientes.
5. Se busca una relación lineal entre variables independientes continuas y la transformación logit de la variable dependiente.
6. No deben existir valores atípicos, valores de apalancamiento elevados o puntos influyentes.

Modelo multinomial

El modelo múltinomial esta dado como:

$$Pr\left(Y_{ik}\right) = Pr\left(y_i = k \mid x_i : \beta_1, ... \beta_m\right) = \frac{\exp\left(\beta_{0k} + \beta_k x_i\right)}{\sum_{j=1}^{m} \exp\left(\beta_{0j} + \beta_j x_i\right)}$$

donde β_k es el vector de coeficiente de X para la k-ésima categoría de Y.

Estimación de Parámetros con Máxima Verosimilitud

Utilización de la máxima verosimilitud como técnica de estimación para el modelo logístico.

Función de Pseudoverosimilitud Multinomial (Heeringa):

- La función de pseudoverosimilitud multinomial se maximiza para estimar los parámetros $\widehat{\beta}$.
- Expresada como

$$PL_{Mult}\left(\widehat{\beta} \mid X\right) = \prod_{i=1}^{n} \left\{ \prod_{i=1}^{k} \widehat{\pi}_k \left(x_i\right)^{y_{i(k)}} \right\}^{w_i}$$

Estimación de Parámetros con Máxima Verosimilitud

Maximización mediante Newton-Raphson:

- Aplicación del algoritmo de Newton-Raphson para resolver un conjunto de ecuaciones de estimación $(K-1) \times (p+1)$.
- Supone un diseño de muestra complejo con estratos y conglomerados.

Ecuaciones de Estimación:

- Expresadas como

$$S(\beta)_{Mult} = \sum_{h} \sum_{\alpha} \sum_{i} \omega_{h\alpha i} \left(y_{h\alpha i}^{(k)} - \pi_k(\beta) \right) x'_{h\alpha i} = 0$$

Estimación de Parámetros con Máxima Verosimilitud

Probabilidades $\pi_k(\beta)$:

- $\pi_k(\beta) = \frac{exp(x'\beta_k)}{1+\sum_{k=2}^{K} exp(x'\beta_k)}$
- $\pi_{1(referencia)}(\beta) = 1 - \sum_{k=2}^{K} \pi_k(\beta)$

Matriz de Varianza-Covarianza:

- Calculada mediante la aplicación de Binder de la linealización de la serie de Taylor a las estimaciones derivadas usando la estimación de pseudomáxima verosimilitud.
- Expresada como

$$\widehat{Var}\left(\hat{\beta}\right) = (J^{-1})\, var\left[S\left(\hat{\beta}\right)\right](J^{-1})$$

Modelo multinomial

Para ejemplificar el uso de los modelos multinomiales en encuestas de hogares utilizando R se utilizan los siguientes códigos computacionales.

```
diseno %>% filter(Age >= 15)%>% group_by(Employment) %>%
  summarise(Prop = survey_mean(vartype = c("se", "ci")))
```

Employment	Prop	Prop_se	Prop_low	Prop_upp
Unemployed	0.0448	0.0078	0.0294	0.0602
Inactive	0.3798	0.0150	0.3501	0.4096
Employed	0.5754	0.0131	0.5495	0.6013

Modelo multinomial

La estimación usando el modelo multinomial con la función `svy_vglm` del paquete `svyVGAM` como se muestra a continuación:

```
library(svyVGAM)
diseno_15 <- diseno %>% filter(Age >= 15)
model_mul <- svy_vglm(
    formula = Employment ~ Age + Sex + Region + Zone,
                    design = diseno_15,
     crit = "coef",
    family = multinomial(refLevel = "Unemployed"))
```

La función `broom::tidy()`, que normalmente usamos para limpiar y estandarizar la salida del modelo, no puede ser empleada en este caso, sin embargo, en el link[2] encuentra la función que utilizamos a continuación.

[2] https://tech.popdata.org/pma-data-hub/posts/2021-08-15-covid-analysis/

Modelo multinomial

```
tab_model <- tidy.svyVGAM(model_mul,
                                exponentiate = FALSE,
                                conf.int = FALSE)
tab_model
```

Modelo multinomial

y.level	term	estimate	std.error	statistic	p.value
1	(Intercept)	2.2904	0.7846	2.9193	0.0035
1	Age	0.0247	0.0098	2.5132	0.0120
1	SexMale	-2.2195	0.3063	-7.2454	0.0000
1	RegionSur	-0.4362	0.7146	-0.6105	0.5415
1	RegionCentro	0.3713	0.6277	0.5916	0.5541
1	RegionOccidente	0.2536	0.6336	0.4002	0.6890
1	RegionOriente	0.6176	0.6730	0.9177	0.3588
1	ZoneUrban	-0.2346	0.4335	-0.5412	0.5884
2	(Intercept)	2.0931	0.6322	3.3108	0.0009
2	Age	0.0207	0.0084	2.4672	0.0136
2	SexMale	-0.5563	0.2718	-2.0470	0.0407
2	RegionSur	-0.2791	0.5746	-0.4857	0.6272
2	RegionCentro	0.2558	0.5373	0.4760	0.6341
2	RegionOccidente	0.0928	0.5143	0.1804	0.8568
2	RegionOriente	0.4706	0.6159	0.7640	0.4449
2	ZoneUrban	0.0567	0.3767	0.1506	0.8803

Plot del IC para los coeficientes.

```
tab_model %>%
  mutate(
    model = if_else(
      y.level == 1,
      "Inactive",
      "Employed",
    ),
    sig = gtools::stars.pval(p.value)
  )  %>%
  dotwhisker::dwplot(
    dodge_size = 0.3,
    vline = geom_vline(xintercept = 1, colour = "grey60",
                       linetype = 2)
  ) +
  guides(color = guide_legend(reverse = TRUE)) +
  theme_bw() + theme(legend.position = "top")
```

Plot del IC para los coeficientes.

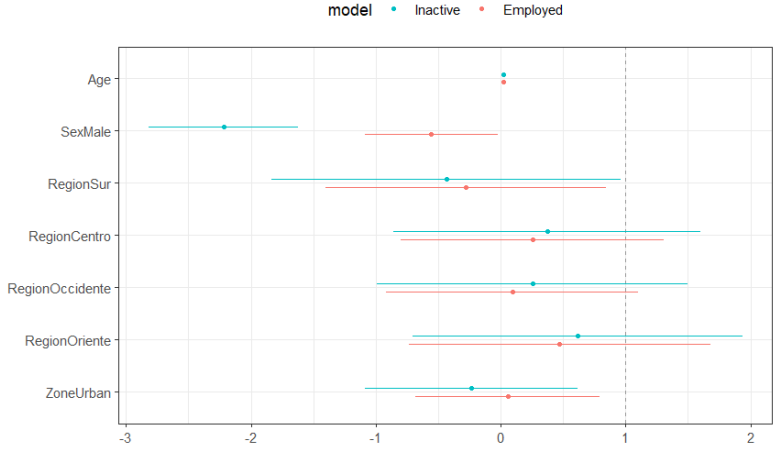

Figura 3: Intervalo de confianza

modelo multinomial función alternativa.

La función `svy_vglm` realiza la estimación de los parámatros, sin embargo, presenta limitaciones para hacer las predicciones con el modelo, por lo tanto, podemos usar como alternativa.

```
library(CMAverse)
model_mul2 <- svymultinom(
  formula = Employment ~ Age + Sex + Region + Zone,
  weights = diseno_15$variables$wk2,
  data = diseno_15$variables
)
saveRDS(model_mul2, "Imagenes/08_MLG2/04_modelo_multi.rds")
```

Modelo multinomial función alternativa.

Parámetros estimados

	Estimate	Std. Error	t value	Pr(>|t|)
Inactive:(Intercept)	2.2901	0.5587	4.0992	0.0000
Inactive:Age	0.0248	0.0100	2.4721	0.0135
Inactive:SexMale	-2.2195	0.3162	-7.0182	0.0000
Inactive:RegionSur	-0.4361	0.4258	-1.0243	0.3058
Inactive:RegionCentro	0.3715	0.4910	0.7566	0.4494
Inactive:RegionOccidente	0.2537	0.4553	0.5573	0.5774
Inactive:RegionOriente	0.6175	0.5158	1.1973	0.2314
Inactive:ZoneUrban	-0.2346	0.2907	-0.8071	0.4197
Employed:(Intercept)	2.0929	0.5427	3.8563	0.0001
Employed:Age	0.0207	0.0096	2.1496	0.0317
Employed:SexMale	-0.5563	0.3053	-1.8219	0.0686
Employed:RegionSur	-0.2790	0.4106	-0.6794	0.4969
Employed:RegionCentro	0.2559	0.4679	0.5471	0.5844
Employed:RegionOccidente	0.0929	0.4439	0.2093	0.8342
Employed:RegionOriente	0.4705	0.5051	0.9314	0.3517
Employed:ZoneUrban	0.0567	0.2792	0.2033	0.8390

Predicción del modelo

El hacer uso de esta función podemos obtener de forma simple la predicción de las probabilidades

```
tab_pred <- predict(model_mul2, type = "probs") %>%
  data.frame()
tab_pred %>% slice(1:10)
```

Predicción del modelo

Unemployed	Inactive	Employed
0.0387	0.2237	0.7376
0.0151	0.5948	0.3901
0.0310	0.5551	0.4139
0.0908	0.1854	0.7238
0.0134	0.6005	0.3861
0.0317	0.5537	0.4146
0.0467	0.2157	0.7376
0.0173	0.5878	0.3949
0.0791	0.1921	0.7289
0.0295	0.2350	0.7355

Predicción del modelo

Las predicciones del modelo se realizan de la siguiente manera:

```
diseno_15$variables  %<>%
  mutate(predicion = predict(model_mul2))

diseno_15 %>% group_by(Employment) %>%
  summarise(Prop = survey_mean(vartype = c("se", "ci")))
```

Employment	Prop	Prop_se	Prop_low	Prop_upp
Unemployed	0.0448	0.0078	0.0294	0.0602
Inactive	0.3798	0.0150	0.3501	0.4096
Employed	0.5754	0.0131	0.5495	0.6013

¡Gracias!

Email: andres.gutierrez@cepal.org