Análisis de encuestas de hogares con R

Modulo 8: Modelos multinivel

CEPAL - Unidad de Estadísticas Sociales

Tabla de contenidos I

Introducción

Introducción a los Modelos Multinivel en Encuestas de Hogares

Los modelos multinivel, también conocidos como modelos de efectos mixtos o modelos jerárquicos, son una herramienta estadística fundamental para analizar datos de encuestas de hogares con estructuras jerárquicas o multinivel. Estas encuestas recopilan datos a niveles individuales (edad, género, educación) y a nivel del hogar (ingreso, propiedad de vivienda, ubicación geográfica).

Principales Características de los Modelos Multinivel:

1. **Análisis de Influencias:** Permiten entender cómo los factores a nivel del hogar e individual afectan las respuestas a las preguntas de la encuesta.
2. **Consideración de la Heterogeneidad:** Modelan efectos aleatorios y fijos, teniendo en cuenta la variación entre hogares y las relaciones promedio.
3. **Precisión Estadística:** Ofrecen estimaciones más precisas al considerar la estructura jerárquica de los datos y la heterogeneidad en la población.

Referencias Bibliográficas Relevantes

1. **Multilevel statistical models** Harvey Goldstein (2011): Clásico en el análisis multinivel, aborda modelos jerárquicos en encuestas de hogares, cubriendo regresión y varianza-covarianza.
2. **Data analysis using regression and multilevel/hierarchical models** Andrew Gelman y Jennifer Hill (2006): Introducción accesible a modelos jerárquicos con ejemplos de encuestas de hogares.
3. **Multilevel and longitudinal modeling using Stata** Sophia Rabe-Hesketh y Anders Skrondal (2012): Guía práctica para el análisis multinivel y longitudinal con ejemplos de encuestas de hogares.

Referencias Bibliográficas Relevantes

4. **A comparison of Bayesian and likelihood-based methods for fitting multilevel models"** - William J. Browne y David Draper (2006): Comparación de enfoques Bayesianos y basados en verosimilitud en modelos jerárquicos para encuestas de hogares.

5. **A brief conceptual tutorial of multilevel analysis in social epidemiology** - Juan Merlo et al. (2006): Introducción a modelos jerárquicos en epidemiología social con ejemplos de encuestas de hogares.

Estas referencias ofrecen una base sólida para comprender y aplicar modelos multinivel en el análisis de datos de encuestas de hogares.

Ejemplo de los modelos multinivel.

Para efectos de ejemplificar los conceptos que se presentarán en este capítulo, definamos una muestra con 6 estratos como se muestra a continuación:

```
encuesta <- readRDS("../Data/encuesta.rds")
encuesta_plot <- encuesta %>%
  dplyr::select(HHID, Stratum) %>% unique() %>%
  group_by(Stratum)  %>% tally() %>%
  arrange(desc(n)) %>% dplyr::select(-n) %>%
  slice(1:6L) %>%
  inner_join(encuesta) %>% filter(Expenditure < 700) %>%
  dplyr::select(Income, Expenditure, Stratum,
                Sex, Region, Zone)
encuesta_plot  %>% slice(1:10L)
```

Introducción a los modelos multinivel.

Income	Expenditure	Stratum	Sex	Region	Zone
697.3	296.1	idStrt017	Male	Norte	Rural
697.3	296.1	idStrt017	Female	Norte	Rural
697.3	296.1	idStrt017	Male	Norte	Rural
697.3	296.1	idStrt017	Female	Norte	Rural
526.8	294.8	idStrt017	Male	Norte	Rural
526.8	294.8	idStrt017	Female	Norte	Rural
526.8	294.8	idStrt017	Female	Norte	Rural
526.8	294.8	idStrt017	Male	Norte	Rural
526.8	294.8	idStrt017	Male	Norte	Rural
526.8	294.8	idStrt017	Female	Norte	Rural

Introducción a los modelos multinivel.

Se comenzará ajustando un modelo lineal cuya variable a modelar son los ingresos de los hogares y cuya variable explicativa son los gastos de los hogares sin considerar el efecto de los estratos del diseño muestral. A continuación, se muestra la gráfica:

$$\text{Ingreso}_i \sim \hat{\beta}_0 + \hat{\beta}_1 \text{Gasto}_i + \varepsilon_i$$

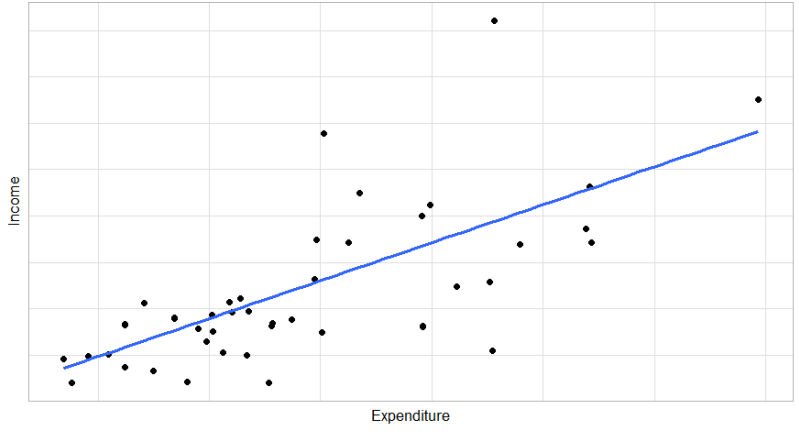

Figura 1: Modelo regresión simple

Introducción a los modelos multinivel.

Ahora se ajusta un modelo de regresión en donde el intercepto cambia de acuerdo con cada estrato.

```
B1 <- coef(lm(Income ~ Expenditure, data = encuesta_plot))[2]
(coef_Mod <- encuesta_plot %>% group_by(Stratum) %>%
  summarise(B0 = coef(lm(Income ~ 1))[1]) %>%
  mutate(B1 = B1))
```

Stratum	B0	B1
idStrt002	496.9	1.637
idStrt010	584.7	1.637
idStrt015	660.6	1.637
idStrt017	408.3	1.637
idStrt022	517.9	1.637
idStrt028	492.1	1.637

$$\text{Ingreso}_{ij} \sim \hat{\beta}_{0j} + \hat{\beta}_1 \text{Gasto}_{ij} + \varepsilon_{ij}$$

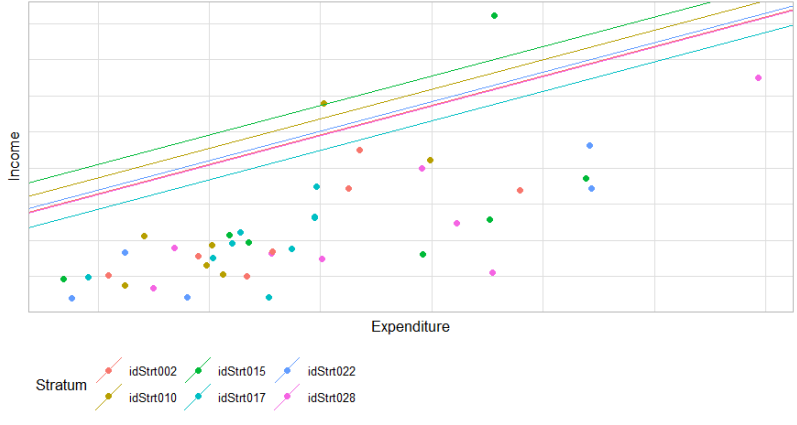

Figura 2: Modelo de regresión con intercepto variando por estrato

Introducción a los modelos multinivel.

Ahora se ajustará un modelo con pendiente aleatoria. Dicha pendiente se estimará para cada uno de los estratos definidos en el diseño muestral como se presenta a continuación:

```
B0 <- coef(lm(Income ~ Expenditure,
              data = encuesta_plot))[1]
(coef_Mod <- encuesta_plot %>% group_by(Stratum) %>%
  summarise(
    B1 = coef(lm(Income ~ -1 + Expenditure))[1]) %>%
  mutate(B0 = B0))
```

Stratum	B1	B0
idStrt002	1.727	29.56
idStrt010	2.303	29.56
idStrt015	1.837	29.56
idStrt017	1.672	29.56
idStrt022	1.478	29.56
idStrt028	1.495	29.56

$$\text{Ingreso}_{ij} \sim \hat{\beta}_0 + \hat{\beta}_{1j}\text{Gasto}_{ij} + \varepsilon_{ij}$$

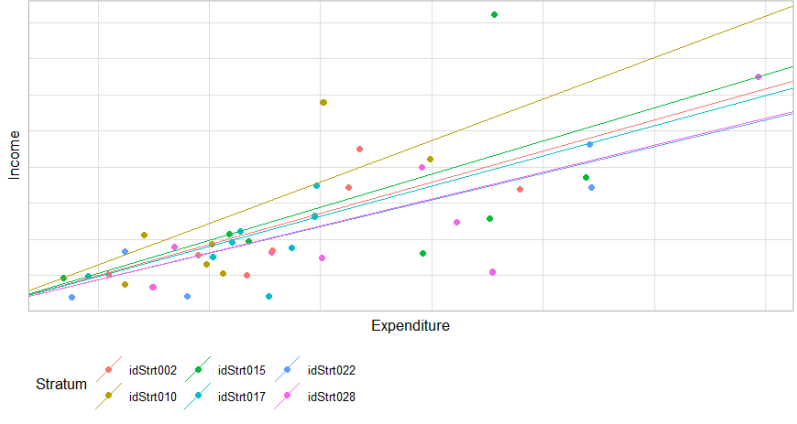

Figura 3: Modelo de regresión con pendiente variando por estrato

Introducción a los modelos multinivel.

Creando un gráfico con intercepto y pendientes aleatorias.

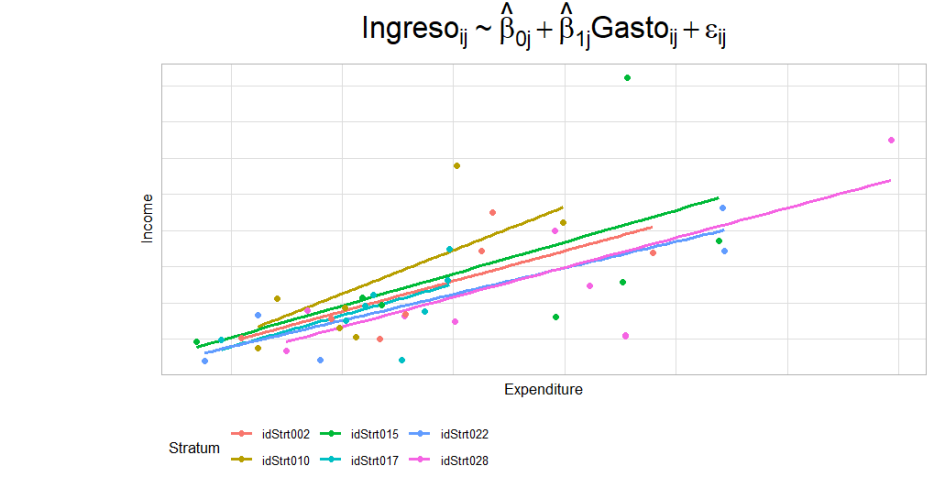

Figura 4: Modelo de regresión con intercepto y pendiente variando por estrato

Se puede observar que este modelo se ajusta mejor a los datos que el modelo anterior y que el modelo lineal clásico.

Modelos multinivel en muestras complejas

Modelos multinivel en muestras complejas

Dos tipos de índices son relevantes en los análisis multinivel:

- Los coeficientes de regresión, generalmente denominados como los parámetros fijos del modelo.
- Las estimaciones de la varianza, generalmente denominadas parámetros aleatorios del modelo.

Cualquier análisis de regresión multinivel siempre debe comenzar con el cálculo de las estimaciones de varianza de Nivel 1 y Nivel 2 para la variable dependiente.

Modelos multinivel en muestras complejas

- El primer paso recomendado en el análisis de regresión multinivel consiste en una descomposición de la varianza de la variable dependiente en los diferentes niveles.

Ejemplo La varianza del ingreso se descompondrá en dos componentes:

- La varianza dentro dentro del estrato
- la varianza entre los estratos.

Estos dos componentes de varianza se pueden obtener una regresión multinivel.

Modelos multinivel en muestras complejas

Un modelo básico es:

$y_{ij} = \beta_{0j} + \epsilon_{ij}$ con $\beta_{0j} = \gamma_{00} + \tau_{0j}$

donde

- y_{ij} = Los ingresos de la persona i en el estrato j.
- β_{0j} = El intercepto en el estrato j.
- ϵ_{ij} El residual de la persona i en el estrato j.
- γ_{00} = El intercepto en general.
- τ_{0j} = Efecto aleatorio para el intercepto.

donde, $\tau_{0j} \sim N\left(0, \sigma_{\tau}^2\right)$ y $\epsilon_{ij} \sim N\left(0, \sigma_{\epsilon}^2\right)$.

La correlación intra clásica esta dada por:

$$\rho = \frac{\sigma_{\tau}^2}{\sigma_{\tau}^2 + \sigma_{\epsilon}^2}$$

La ICC mide la similitud o correlación entre las observaciones dentro del mismo grupo o nivel en comparación con las observaciones de diferentes grupos.

Modelos multinivel en muestras complejas.

- Aunque existe evidencia suficiente de que las ponderaciones de muestreo deben usarse en el modelado multinivel (MLM) para obtener estimaciones no sesgadas[1], y también sobre cómo deben usarse estas ponderaciones en los análisis de un solo nivel, hay poca discusión en la literatura sobre qué y cómo usar pesos de muestreo en MLM.
- Actualmente, diferentes autores recomiendan cuatro enfoques diferentes sobre cómo usar los pesos de muestreo en modelos jerárquicos.

[1] Cai, T. (2013). Investigation of ways to handle sampling weights for multilevel model analyses. Sociological Methodology, 43(1), 178-219.

Modelos multinivel en muestras complejas.

Diferentes autores recomiendan diferentes enfoques sobre cómo usar los pesos de muestreo en modelos jerárquicos.

- Pfefermann et al. (1998) y Asparouhov (2006) aconsejan utilizar un enfoque de pseudomáxima verosimilitud para calcular estimaciones dentro y entre los diferentes niveles utilizando la técnica de maximización de mínimos cuadrados generalizados ponderados por probabilidad (PWGLS) para obtener estimaciones no sesgadas.[2][3]
- Rabe-Hesketh y Skrondal (2006) proporcionan técnicas de maximización de expectativas para maximizar la pseudoverosimilitud[4]

---

[2] Pfeffermann, D., Skinner, C. J., Holmes, D. J., Goldstein, H., & Rasbash, J. (1998). Weighting for unequal selection probabilities in multilevel models. Journal of the Royal Statistical Society: series B (statistical methodology), 60(1), 23-40.

[3] Asparouhov, T. (2006). General multi-level modeling with sampling weights. Communications in Statistics—Theory and Methods, 35(3), 439-460.

[4] Asparouhov, T., & Muthen, B. (2006, August). Multilevel modeling of complex survey data. In Proceedings of the joint statistical meeting in Seattle (pp. 2718-2726).

Estimación de pseudo máxima verosimilitud

La función de log-verosimilitud para la población esta dada por:

$$L_U(\theta) = \sum_{i \in U} \log\left[f(y_i; \theta)\right]$$

El estimador de máxima verosimilitud esta dada por:

$$\frac{\partial L_U(\theta)}{\partial \theta} = 0$$

La dificultad que encontramos aquí, es transferir los pesos muéstrales a los niveles inferiores, por ejemplo UPMs -> Stratum.

Estimación de pseudo máxima verosimilitud

Pfeffermann et al. (1998) argumentaron que debido a la estructura de datos agrupados, ya no se asume que las observaciones sean independientes y que la probabilidad logarítmica se convierta en una suma entre los elementos de nivel uno y dos en lugar de una simple suma de las contribuciones de los elementos.

Modelo Nulo

Asuma que la información dentro del estrato esta definida por el intercepto.

$$\begin{aligned} Ingreso_{ij} &= \beta_{0j} + \epsilon_{ij} \\ \beta_{1j} &= \gamma_{10} + \gamma_{11} Stratum_j + \tau_{1j} \end{aligned}$$

calculando de los Qweighted

Para tener estimaciones consistentes se calculan los pesos Qweighted siguiendo los pasos mostrados, tomando en este caso como covariables la edad del entrevistado, el sexo, la región y la zona donde reside.

```
mod_qw <- lm(wk ~ Age + Sex + Region + Zone,
             data = encuesta)
encuesta$wk2 <-   encuesta$wk/predict(mod_qw)
```

calculando los senate-weight

Adicionalmente, se calculan también los *senate-weight* para el ajuste de los modelos (Wk3, en el ajuste del modelo en R).

```
# Alternativa los Qweighted
n = nrow(encuesta)
encuesta <- encuesta %>% mutate(wk3 = n*wk/sum(wk))
encuesta %>% summarise(fep = sum(wk),
                       q_wei = sum(wk2),
                       fep2 = sum(wk3) )
```

fep	q_wei	fep2
150266	2602	2605

Comparando los pesos.

```
ggplot(encuesta, aes(x = wk2, y = wk3)) + geom_point() +
  theme_bw() + labs(x = "q-weighted", y = "senate-weight")
```

Comparando los pesos.

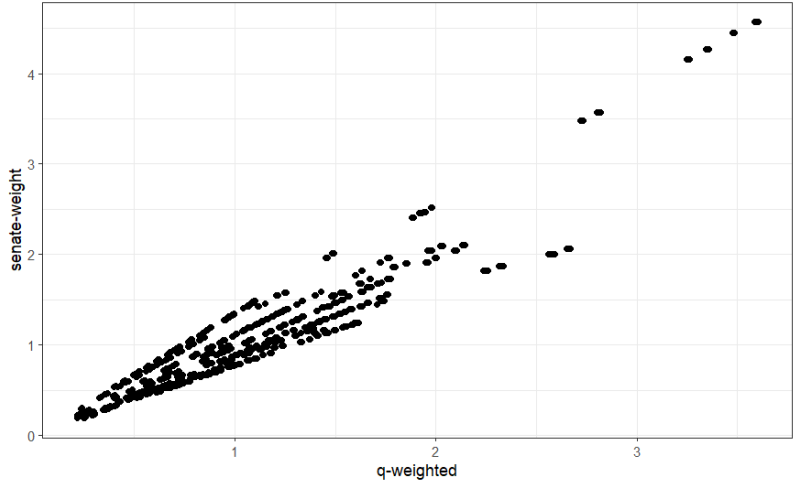

Figura 5: Comparando q-weighted Vs senate-weight

Modelo Nulo

Se realizarán los ajustes de los modelos utilizando los dos pesos mostrados anteriormente:

```
library(lme4)

mod_null  <- lmer(Income  ~ (1  |  Stratum),
                  data  =  encuesta,
                  weights  =  wk2)

mod_null2  <- lmer(Income  ~ (1  |  Stratum),
                   data  =  encuesta,
                   weights  =  wk3)
```

Modelo Nulo

Comparando los modelos obtenidos.

```
coef_mod_null <- bind_cols(coef(mod_null)$Stratum,
                           coef(mod_null2)$Stratum)
colnames(coef_mod_null) <- c("Intercept Mod 1",
                             "Intercept Mod 2")
coef_mod_null %>% slice(1:12)
```

Modelo Nulo

	Intercept Mod 1	Intercept Mod 2
idStrt001	630.7	630.1
idStrt002	505.4	506.2
idStrt003	481.3	484.7
idStrt004	959.6	954.5
idStrt005	514.6	515.9
idStrt006	433.8	438.2
idStrt007	467.5	470.5
idStrt008	371.6	376.4
idStrt009	207.6	218.1
idStrt010	591.6	592.1
idStrt011	588.8	588.3
idStrt012	352.0	361.2

Modelo Nulo

```
mod_null
```

```
Linear mixed model fit by REML ['lmerMod']
Formula: Income ~ (1 | Stratum)
   Data: encuesta
Weights: wk2
REML criterion at convergence: 39356
Random effects:
 Groups   Name        Std.Dev.
 Stratum  (Intercept) 281
 Residual             408
Number of obs: 2605, groups:  Stratum, 119
Fixed Effects:
(Intercept)
        584
```

Modelo Nulo

Correlación intraclases

```
performance::icc(mod_null)
```

ICC_adjusted	ICC_unadjusted	optional
0.3218	0.3218	FALSE

Modelo Nulo

Predicción dentro de los estrato es constante.

```
(tab_pred <- data.frame(Pred = predict(mod_null),
            Income = encuesta$Income,
            Stratum = encuesta$Stratum)) %>% distinct() %>%
  slice(1:6L) # Son las pendientes aleatorias
```

	Pred	Income	Stratum
1	630.7	409.87	idStrt001
6	630.7	823.75	idStrt001
10	630.7	90.92	idStrt001
13	630.7	135.33	idStrt001
18	630.7	336.19	idStrt001
22	630.7	1539.75	idStrt001

Scaterplot de y vs $\widehat{y}$

Si la predicción es correcta se espera estar sobre la linea de 45°

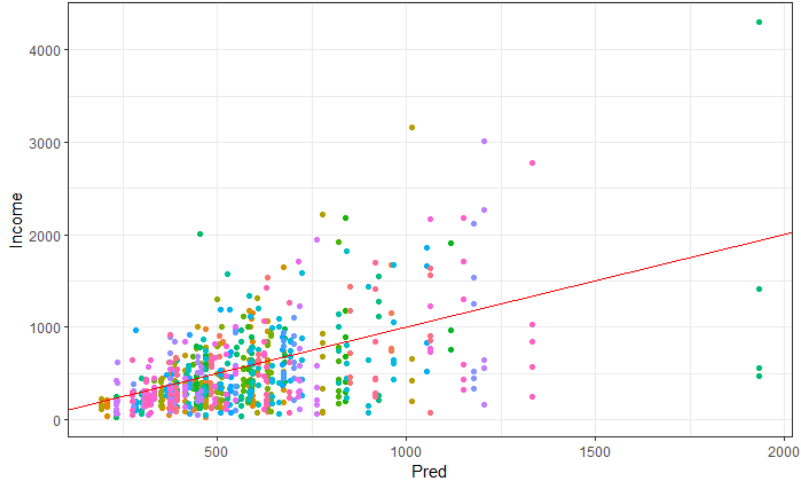

Figura 6: Predicción con el modelo nulo y los q-weighted

Modelo con intercepto aleatoria

- Un modelo con pendiente aleatoria permite que la relación entre una variable independiente y una variable dependiente varíe según otra variable explicativa.
- En un modelo con pendiente aleatoria, la pendiente puede cambiar según factores como el tiempo, la edad, el género o la ubicación geográfica.
- A diferencia de los modelos lineales simples, los modelos con pendiente aleatoria permiten ajustar la relación entre variables a curvas con pendientes variables.
- Explora cómo la relación entre variables se adapta a cambios en diferentes contextos, proporcionando una representación más realista de las complejidades en los datos.

Modelo con intercepto aleatoria

Consideremos el siguiente modelo

$$Ingreso_{ij} = \beta_0 + \beta_{1j} Gasto_{ij} + \epsilon_{ij}$$

donde β_{1j} esta dado como

$$\beta_{1j} = \gamma_{10} + \gamma_{11} Stratum_j + \tau_{1j}$$

```
mod_Int_Aleatorio <- lmer(
  Income ~ Expenditure  + (1 | Stratum),
  data = encuesta, weights  =  wk2)
performance::icc(mod_Int_Aleatorio)
```

ICC_adjusted	ICC_unadjusted	optional
0.1958	0.1022	FALSE

Modelo con intercepto aleatoria

Para cada estrato se tiene las siguientes estimaciones de β_{1j}

```
coef(mod_Int_Aleatorio)$Stratum %>% slice(1:8L)
```

	(Intercept)	Expenditure
idStrt001	248.257	1.202
idStrt002	152.988	1.202
idStrt003	139.765	1.202
idStrt004	292.650	1.202
idStrt005	-42.165	1.202
idStrt006	46.766	1.202
idStrt007	2.841	1.202
idStrt008	103.346	1.202

Modelo con intercepto aleatoria

Organizando los coeficientes para el gráfico.

```
Coef_Estimado <- inner_join(
  coef(mod_Int_Aleatorio)$Stratum %>%
       add_rownames(var = "Stratum"),
encuesta_plot %>% select(Stratum) %>% distinct())

ggplot(data = encuesta_plot,
       aes(y = Income, x = Expenditure,
           colour = Stratum)) +
  geom_jitter() + theme(legend.position="none",
    plot.title = element_text(hjust = 0.5)) +
  geom_abline(data = Coef_Estimado,
              mapping=aes(slope=Expenditure,
                          intercept=`(Intercept)`,
                          colour = Stratum))+
  theme_cepal()
```

Modelo con intercepto aleatoria

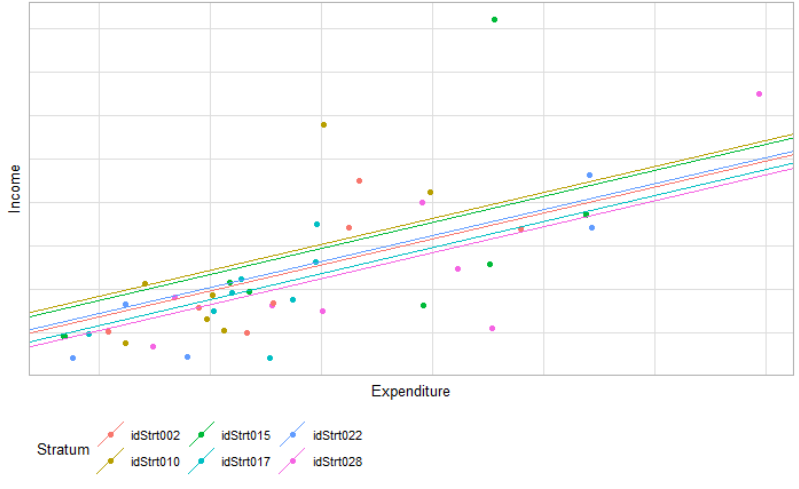

Figura 7: Modelo de intercepto aleatorio

Predicción del modelo

```
(tab_pred <- data.frame(
  Pred = predict(mod_Int_Aleatorio),
            Income = encuesta$Income,
            Stratum = encuesta$Stratum)) %>% distinct() %>%
  slice(1:6L) # Son las pendientes aleatorias
```

	Pred	Income	Stratum
1	664.4	409.87	idStrt001
6	719.6	823.75	idStrt001
10	337.3	90.92	idStrt001
13	348.9	135.33	idStrt001
18	560.9	336.19	idStrt001
22	890.5	1539.75	idStrt001

Scaterplot de y vs $\widehat{y}$

La predicción esta más cerca a la linea de 45 grados.

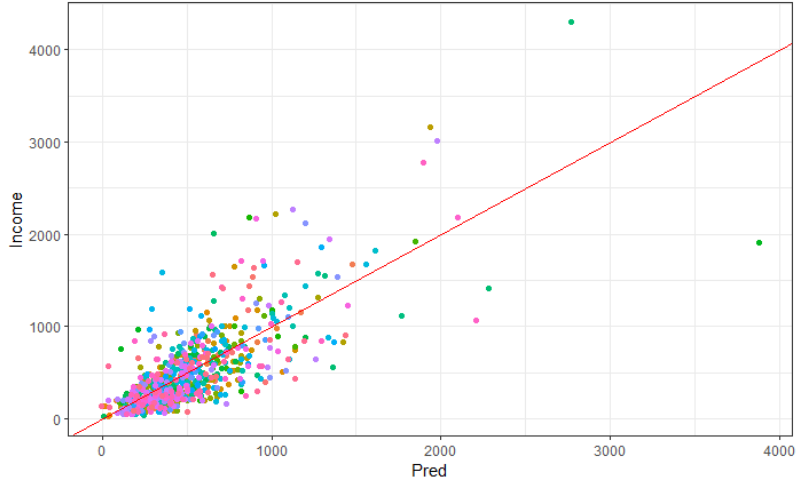

Figura 8: Scaterplot de y vs $\widehat{y}$

Modelo con intercepto y pendiente aleatoria

- Los modelos con intercepto y pendiente aleatoria incorporan tanto efectos fijos como efectos aleatorios para modelar la relación entre una variable de respuesta y variables predictoras.
- Los coeficientes de regresión (intercepto y pendiente) se consideran aleatorios en lugar de fijos.
- La variación en estos coeficientes entre unidades de análisis (individuos, grupos, regiones) se modela como efectos aleatorios.
- Útiles cuando los datos tienen una estructura jerárquica o de agrupamiento, con unidades de análisis agrupadas en diferentes niveles (estudiantes en escuelas, pacientes en hospitales).

Modelo con intercepto y pendiente aleatoria

- Captura la heterogeneidad en los coeficientes a través de diferentes niveles de agrupamiento.
- Ofrece una herramienta efectiva para abordar estructuras de datos complejas donde la variabilidad puede estar influenciada por múltiples niveles de agrupamiento.
- Proporciona una representación más realista al considerar la variabilidad inherente entre grupos en la relación entre variables predictoras y de respuesta.

Modelo con intercepto y pendiente aleatoria

La estructura del modelo es la siguiente:

$$Ingreso_{ij} = \beta_{0j} + \beta_{1j} Gasto_{ij} + \epsilon_{ij}$$

$$\beta_{0j} = \gamma_{00} + \gamma_{01} Stratum_j + \tau_{0j}$$

$$\beta_{1j} = \gamma_{10} + \gamma_{11} Stratum_j + \tau_{1j}$$

```
mod_Pen_Aleatorio <- lmer(
  Income ~ Expenditure  + (1 + Expenditure| Stratum),
  data = encuesta, weights  =  wk2)

performance::icc(mod_Pen_Aleatorio)
```

ICC_adjusted	ICC_unadjusted	optional
0.6971	0.4598	FALSE

Modelo con intercepto y pendiente aleatoria

```
coef(mod_Pen_Aleatorio)$Stratum %>% slice(1:14L)
```

	(Intercept)	Expenditure
idStrt001	-230.63	2.7761
idStrt002	31.07	1.6236
idStrt003	152.51	1.1621
idStrt004	230.80	1.3452
idStrt005	-97.05	1.2964
idStrt006	30.34	1.2039
idStrt007	37.63	1.0771
idStrt008	167.70	0.9018
idStrt009	30.46	0.7488
idStrt010	74.87	1.8971
idStrt011	275.83	0.6615
idStrt012	13.48	1.2099
idStrt013	178.63	1.1308
idStrt014	210.66	0.8120

Modelo con intercepto y pendiente aleatoria

```
Coef_Estimado <- inner_join(
  coef(mod_Pen_Aleatorio)$Stratum %>%
       add_rownames(var = "Stratum"),
encuesta_plot %>% select(Stratum) %>% distinct())

ggplot(data = encuesta_plot,
       aes(y = Income, x = Expenditure,
           colour = Stratum)) +
  geom_jitter() + theme(legend.position="none",
    plot.title = element_text(hjust = 0.5)) +
  geom_abline(data = Coef_Estimado,
              mapping=aes(slope=Expenditure,
                          intercept=`(Intercept)`,
                          colour = Stratum))+
  theme_cepal()
```

Modelo con intercepto y pendiente aleatoria

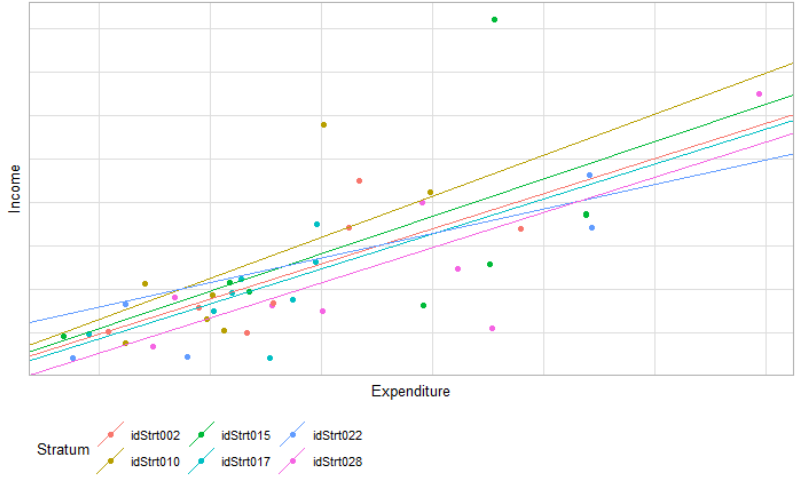

Figura 9: Modelo con intercepto y pendiente aleatoria

Predicción del modelo

```
(tab_pred <- data.frame(Pred = predict(mod_Pen_Aleatorio),
                        Income = encuesta$Income,
                        Stratum = encuesta$Stratum)) %>% distinct() %>%
  slice(1:8L) # Son las pendientes aleatorias
```

	Pred	Income	Stratum
1	730.855	409.87	idStrt001
6	858.279	823.75	idStrt001
10	-25.002	90.92	idStrt001
13	1.982	135.33	idStrt001
18	491.663	336.19	idStrt001
22	1253.017	1539.75	idStrt001
28	447.905	336.00	idStrt002
32	410.482	199.33	idStrt002

Scaterplot de y vs $\widehat{y}$

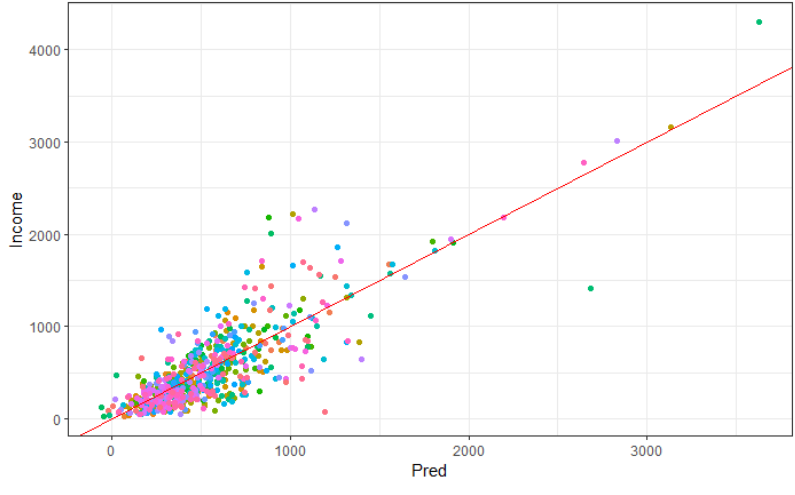

Figura 10: Scaterplot de y vs $\widehat{y}$

Modelo con intercepto y pendiente aleatoria

Para robustecer el modelo, se ajusta nuevamente, pero agregando la variable zona como se muestra a continuación:

$$Ingreso_{ij} = \beta_{0j} + \beta_{1j} Gasto_{ij} + \beta_{2j} Zona_{ij} + \epsilon_{ij}$$

$$\beta_{0j} = \gamma_{00} + \gamma_{01} Stratum_j + \gamma_{02} \mu_j + \tau_{0j}$$

$$\beta_{1j} = \gamma_{10} + \gamma_{11} Stratum_j + \gamma_{12} \mu_j + \tau_{1j}$$
$$\beta_{2j} = \gamma_{20} + \gamma_{21} Stratum_j + \gamma_{12} \mu_j + \tau_{2j}$$

donde μ_j es el gasto medio en el estrato j.

Modelo con intercepto y pendiente aleatoria

```
media_estrato <- encuesta %>% group_by(Stratum) %>%
  summarise(mu = mean(Expenditure))
encuesta <- inner_join(encuesta,
                       media_estrato, by = "Stratum")

mod_Pen_Aleatorio2 <- lmer(
  Income ~ 1 + Expenditure + Zone + mu +
    (1 + Expenditure + Zone + mu | Stratum ),
    data = encuesta, weights  =  wk2)
performance::icc(mod_Pen_Aleatorio2)
```

ICC_adjusted	ICC_unadjusted	optional
0.5039	0.2578	FALSE

Modelo con intercepto y pendiente aleatoria

```
(tab_pred <- data.frame(Pred = predict(mod_Pen_Aleatorio2),
                        Income = encuesta$Income,
                        Stratum = encuesta$Stratum)) %>% distinct() %>%
  slice(1:10L) # Son las pendientes aleatorias
```

	Pred	Income	Stratum
1	730.596	409.87	idStrt001
6	856.747	823.75	idStrt001
10	-17.710	90.92	idStrt001
13	9.004	135.33	idStrt001
18	493.794	336.19	idStrt001
22	1247.541	1539.75	idStrt001
28	453.736	336.00	idStrt002
32	417.001	199.33	idStrt002
36	562.474	685.48	idStrt002
41	578.061	900.33	idStrt002

Scaterplot de y vs $\widehat{y}$

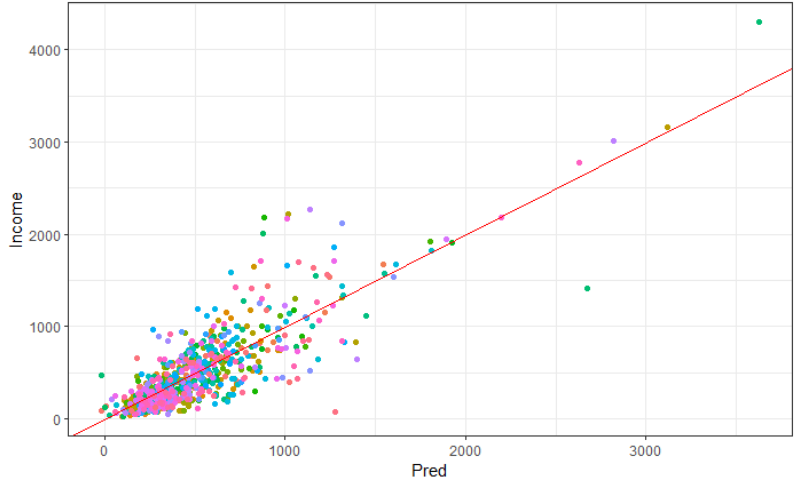

Figura 11: Scaterplot de y vs $\widehat{y}$

Modelo con intercepto y pendiente aleatoria

```
(Coef_Estimado <- inner_join(
  coef(mod_Pen_Aleatorio2)$Stratum %>%
    add_rownames(var = "Stratum"),
  encuesta_plot %>% select(Stratum, Zone) %>% distinct()
))
```

Stratum	(Intercept)	Expenditure	ZoneUrban	mu	Zone
idStrt002	51.36	1.594	26.43	-0.1161	Urban
idStrt010	94.60	1.984	146.70	-0.6633	Urban
idStrt015	22.64	1.746	-150.28	0.0202	Rural
idStrt017	52.29	1.578	44.36	-0.1239	Rural
idStrt022	41.45	1.132	24.32	0.2746	Urban
idStrt028	49.45	1.570	-85.97	0.0159	Urban

Modelo con intercepto y pendiente aleatoria

```
(Coef_Estimado<- Coef_Estimado %>%
   inner_join(media_estrato, by = "Stratum"))
```

Stratum	(Intercept)	Expenditure	ZoneUrban	mu.x	Zone	mu.y
idStrt002	51.36	1.594	26.43	-0.1161	Urban	286.2
idStrt010	94.60	1.984	146.70	-0.6633	Urban	255.8
idStrt015	22.64	1.746	-150.28	0.0202	Rural	357.0
idStrt017	52.29	1.578	44.36	-0.1239	Rural	244.8
idStrt022	41.45	1.132	24.32	0.2746	Urban	524.0
idStrt028	49.45	1.570	-85.97	0.0159	Urban	337.1

El modelo para el estrato *idStrt002* viene dado por:

$$\hat{y}_{ij} = 51.1 + 1.59 Expenditure_{ij} + 26.43 Zone_{ij} + (-0.1161)\,\mu_j$$

Modelo con intercepto y pendiente aleatoria

```
(Coef_Estimado %<>%  mutate(B0 = ifelse(
Zone == "Urban", `(Intercept)` + mu.y * mu.x + ZoneUrban,
               `(Intercept)` + mu.y * mu.x)) %>%
  select(Stratum, Zone, B0, Expenditure))
```

Stratum	Zone	B0	Expenditure
idStrt002	Urban	44.57	1.594
idStrt010	Urban	71.62	1.984
idStrt015	Rural	29.85	1.746
idStrt017	Rural	21.95	1.578
idStrt022	Urban	209.66	1.132
idStrt028	Urban	-31.15	1.570

Modelo con intercepto y pendiente aleatoria

```
ggplot(data = encuesta_plot,
       aes(y = Income, x = Expenditure,
           colour = Stratum)) +
  geom_jitter() +
  theme(legend.position = "none",
        plot.title = element_text(hjust = 0.5)) +
  facet_grid( ~ Zone) +
  geom_abline(
    data = Coef_Estimado,
    mapping = aes(
      slope = Expenditure,
      intercept = B0,
      colour = Stratum
    )
  ) +
  theme_cepal()
```

Modelo con intercepto y pendiente aleatoria

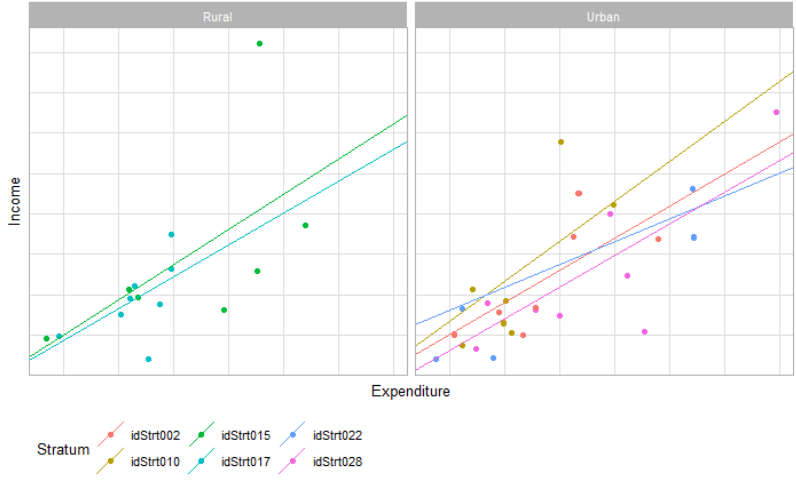

Figura 12: Modelo con intercepto y pendiente aleatoria

Modelos logísticos multinivel en muestras complejas

Introducción a los Modelos Logísticos Multinivel

- Los modelos logísticos multinivel extienden los modelos logísticos simples, adaptándose a la estructura jerárquica de los datos recopilados de individuos agrupados en diferentes niveles.
- En contextos jerárquicos (escuelas, ciudades, países), los modelos logísticos simples pueden no capturar adecuadamente la variación entre grupos y la estructura jerárquica de los datos.
- Permite estimar la varianza en las respuestas entre diferentes grupos, identificando fuentes de variabilidad y comparando la variabilidad entre grupos.
- Herramienta poderosa para analizar datos de respuestas binarias en entornos jerárquicos, proporcionando una comprensión más completa de la variación y estructura de los datos.

Introducción a los modelos logístico multinivel.

Sea la variable $y_{ij} = 1$ si el individuo i en el estrato j esta por encima de la linea de pobreza y $y_{ij} = 0$ en caso contrario, la variable y_{ij} se puede modelar mediante el modelo logístico:

$$Pr\left(y_{ij}\right) = Pr\left(y_{ij} = 1 \mid x_i : \beta\right) = \frac{1}{1 + \exp\left(-\beta_j x_{ij}\right)}$$

ó

$$\log\left(\frac{\pi_{ij}}{1 - \pi_{ij}}\right) = \beta_j x_{ij}$$

donde $\pi_{ij} = Pr\left(y_{ij} = 1 \mid x_i : \beta\right)$.

Ejemplos de modelo logit

```
encuesta_plot <- encuesta %>%
  dplyr::select(Stratum,Expenditure) %>% unique() %>%
  group_by(Stratum)  %>%
  summarise(sd = sd(Expenditure)) %>%
  arrange(desc(sd)) %>% dplyr::select(-sd) %>%
  slice(1:20L) %>%
  inner_join(encuesta) %>%
  dplyr::select(Poverty, Expenditure, Stratum,
         Sex, Region, Zone)
encuesta_plot %>% slice(1:15L)
```

Ejemplos de modelo logit

Poverty	Expenditure	Stratum	Sex	Region	Zone
NotPoor	3367.5	idStrt039	Male	Sur	Urban
NotPoor	3367.5	idStrt039	Female	Sur	Urban
NotPoor	3367.5	idStrt039	Male	Sur	Urban
NotPoor	312.1	idStrt039	Female	Sur	Urban
NotPoor	312.1	idStrt039	Female	Sur	Urban
NotPoor	312.1	idStrt039	Female	Sur	Urban
NotPoor	312.1	idStrt039	Male	Sur	Urban
NotPoor	226.5	idStrt039	Male	Sur	Urban
NotPoor	226.5	idStrt039	Female	Sur	Urban
NotPoor	616.3	idStrt047	Female	Sur	Urban
NotPoor	616.3	idStrt047	Female	Sur	Urban
NotPoor	616.3	idStrt047	Female	Sur	Urban
NotPoor	1385.7	idStrt047	Male	Sur	Urban
NotPoor	1385.7	idStrt047	Female	Sur	Urban
NotPoor	1385.7	idStrt047	Female	Sur	Urban

Ejemplos de modelo logit

```
encuesta <- encuesta %>% mutate(
  pobreza = ifelse(Poverty != "NotPoor", 1, 0))
encuesta_plot %<>% mutate(
  pobreza = ifelse(Poverty != "NotPoor", 1, 0))
```

```
ggplot(data = encuesta,
       aes(y = pobreza, x = Expenditure)) +
  geom_point() +
  geom_smooth(
    formula = y~x, method = "glm",
    se=FALSE,
    method.args = list(family=binomial(link = "logit"))) +
  theme_bw()
```

Ejemplos de modelo logit

Para poder observar la distribución la distribución de la variable pobreza, se presenta el siguiente gráfico:

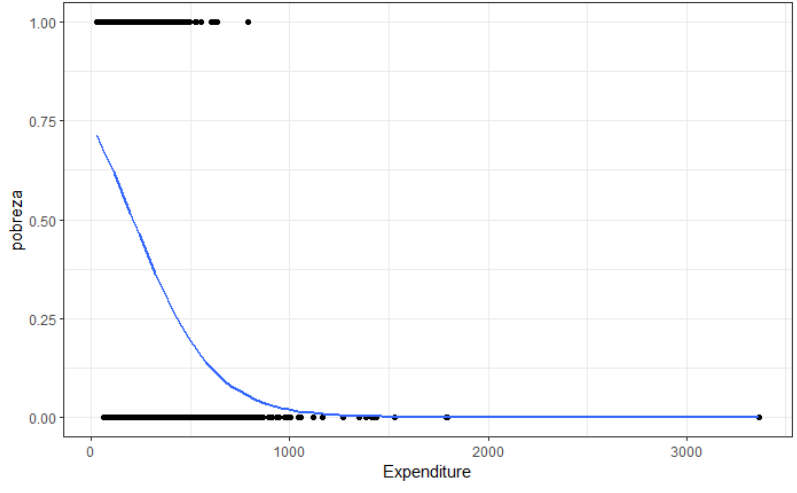

Figura 13: Modelo logit

Ejemplos de modelo logit

Crear una función auxiliar para calcular la probabilidad.

```
auxLogit <- function(x,b0,b1){
  1/(1+exp(-(b0+b1*x)))
}
```

El ajuste del modelo logístico se realiza con la función `glm` y la función link "logit". Ejecutando el siguiente código tenemos la pendiente intercepto fijo

```
B0 = coef(glm(pobreza~1,data = encuesta_plot,
     family=binomial(link = "logit")))
```

Ejemplos de modelo logit

A Continuación ajustamos el modelo sin intercepto por estrato.

```
(coef_Mod <- encuesta_plot %>% group_by(Stratum) %>%
  summarise(B1 = coef(glm(pobreza ~  -1 + Expenditure,
              family=binomial(link = "logit")))) %>%
mutate(B0 = B0)) %>% slice(1:6L)
```

Stratum	B1	B0
idStrt007	-0.0189	-0.8782
idStrt020	-0.0010	-0.8782
idStrt022	-0.0057	-0.8782
idStrt024	-0.0020	-0.8782
idStrt036	-0.0009	-0.8782
idStrt039	-0.0976	-0.8782

Ejemplos de modelo logit

A continuación, se grafican los diferentes modelos logísticos ajustados para cada uno de los estratos observándose que, hay una variación importante entre los estratos:

```
# Creando las variables respuesta
pred_logit <- coef_Mod %>%
  mutate(Expenditure = list(seq(0,2000, length =100))) %>%
    tidyr::unnest_legacy()
pred_logit %<>% mutate(Prob = auxLogit(Expenditure,B0,B1))

ggplot(data = pred_logit,
       aes(y = Prob, x = Expenditure, colour = Stratum)) +
  geom_line() +
   theme_bw() +
  theme(legend.position = "none")
```

Ejemplos de modelo logit

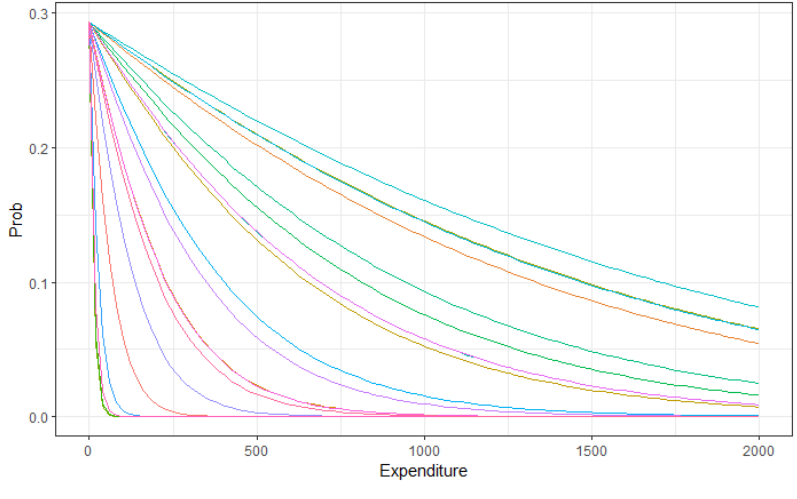

Figura 14: Modelo logit

Modelo logit nulo

Un modelo logístico básico o nulo se escribe de la siguiente manera:

$$\begin{aligned} logit(\pi_{ij}) &= \beta_{0j} + \epsilon_{ij} \\ \beta_{0j} &= \gamma_{00} + \tau_{0j} \end{aligned}$$

- $\pi_{ij} = Pr\left(y_{ij} = 1 \mid x_i : \beta\right)$.
- $\beta_{0j} =$ El intercepto en el estrato j.
- ϵ_{ij} El residual de la persona i en el estrato j.
- $\gamma_{00} =$ El intercepto en general.
- $\tau_{0j} =$ Efecto aleatorio para el intercepto.

donde, $\tau_{0j} \sim N\left(0, \sigma_\tau^2\right)$ y $\epsilon_{ij} \sim N\left(0, \sigma_\epsilon^2\right)$. La correlación intra clásica esta dada por:

$$\rho = \frac{\sigma_\tau^2}{\sigma_\tau^2 + \sigma_\epsilon^2}$$

Modelo Nulo

```
library(lme4)
mod_logist_null  <- glmer( pobreza  ~ ( 1  |  Stratum ),
                    data  =  encuesta,
              weights  =  wk2,
              family = binomial(link = "logit") )


coef( mod_logist_null )$Stratum %>% slice(1:12)
```

Modelo Nulo

	(Intercept)
idStrt001	-0.8334
idStrt002	-0.0133
idStrt003	-2.6023
idStrt004	-2.7770
idStrt005	-1.0268
idStrt006	1.0100
idStrt007	-1.0134
idStrt008	0.2035
idStrt009	2.1966
idStrt010	-0.5948
idStrt011	-1.2986
idStrt012	0.2825

Modelo Nulo

```
mod_logist_null
```

```
Generalized linear mixed model fit by maximum likelihood (Laplace
  Approximation) [glmerMod]
 Family: binomial  ( logit )
Formula: pobreza ~ (1 | Stratum)
   Data: encuesta
Weights: wk2
     AIC      BIC   logLik deviance df.resid
    2966     2978    -1481     2962     2603
Random effects:
 Groups  Name        Std.Dev.
 Stratum (Intercept) 1.29
Number of obs: 2605, groups:  Stratum, 119
Fixed Effects:
(Intercept)
     -0.802
```

Modelo nulo

```
performance::icc(mod_logist_null)
```

ICC_adjusted	ICC_unadjusted	optional
0.3342	0.3342	FALSE

```
(tab_pred <- data.frame(
  Pred = predict(mod_logist_null, type = "response"),
  pobreza = encuesta$pobreza,
  Stratum = encuesta$Stratum)) %>% distinct() %>%
  slice(1:6L) # Son las pendientes aleatorias
```

	Pred	pobreza	Stratum
1	0.3029	0	idStrt001
10	0.3029	1	idStrt001
28	0.4967	1	idStrt002
36	0.4967	0	idStrt002
61	0.0690	0	idStrt003
84	0.0586	0	idStrt004

Estimación de la propoción para y y $\widehat{y}$

```
weighted.mean(encuesta$pobreza, encuesta$wk2)
```

```
[1] 0.3859
```

```
weighted.mean(tab_pred$Pred, encuesta$wk2)
```

```
[1] 0.385
```

Modelo con intercepto aleatoria

EL modelo se define de la siguiente manera:

$$\begin{aligned} logit(\pi_{ij}) &= \beta_0 + \beta_{1j} Gasto_{ij} + \epsilon_{ij} \\ \beta_{1j} &= \gamma_{10} + \gamma_{11} Stratum_j + \tau_{1j} \end{aligned}$$

Siguiendo las ideas de la sección anterior, el ajuste del modelo en R se realiza de la siguiente manera:

```
mod_logit_Int_Aleatorio <- glmer(
  pobreza ~ Expenditure  + (1 | Stratum),
  data = encuesta, family = binomial(link = "logit"),
  weights  =  wk2)

performance::icc(mod_logit_Int_Aleatorio)
```

ICC_adjusted	ICC_unadjusted	optional
0.3151	0.1867	FALSE

Modelo con intercepto aleatoria

```
coef(mod_logit_Int_Aleatorio)$Stratum %>% slice(1:10L)
```

	(Intercept)	Expenditure
idStrt001	0.9889	-0.0066
idStrt002	1.8837	-0.0066
idStrt003	-0.7463	-0.0066
idStrt004	-0.1484	-0.0066
idStrt005	1.7155	-0.0066
idStrt006	3.2456	-0.0066
idStrt007	0.5601	-0.0066
idStrt008	1.6848	-0.0066
idStrt009	3.9332	-0.0066
idStrt010	1.1207	-0.0066

Modelo con intercepto aleatoria

Gráficamente, los modelos ajustados se muestran a continuación:

```
dat_pred <- encuesta %>% group_by(Stratum) %>%
  summarise(
    Expenditure = list(seq(min(Expenditure),
                           max(Expenditure), len = 100))) %>%
  tidyr::unnest_legacy()

dat_pred <- mutate(dat_pred,
       Proba = predict(mod_logit_Int_Aleatorio,
                       newdata = dat_pred , type = "response"))

ggplot(data = dat_pred,
       aes(y = Proba, x = Expenditure,
           colour = Stratum)) +
   geom_line()+   theme_bw() +
  geom_point(data = encuesta, aes(y = pobreza, x = Expenditure))+
  theme(legend.position = "none",
        plot.title = element_text(hjust = 0.5))
```

Modelo con intercepto aleatoria

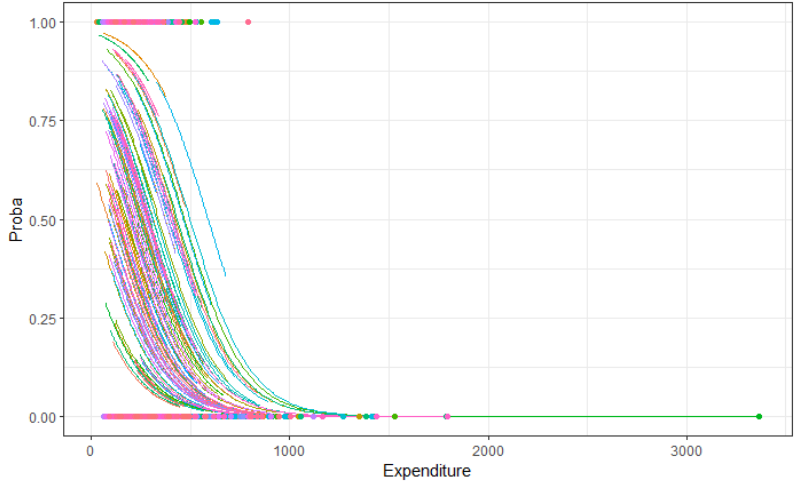

Figura 15: Modelo logit intercepto aleatoria

Predicción del modelo

Las predicciones del modelo se presentan a continuación:

```
(tab_pred <- data.frame(
  Pred = predict(mod_logit_Int_Aleatorio,
                 type = "response"),
           pobreza = encuesta$pobreza,
           Stratum = encuesta$Stratum,
           wk2 = encuesta$wk2)) %>% distinct() %>%
  slice(1:6L) # Son las pendientes aleatorias
```

Pred	pobreza	Stratum	wk2
0.2149	0	idStrt001	0.7770
0.2149	0	idStrt001	0.7501
0.2149	0	idStrt001	0.7463
0.2149	0	idStrt001	0.7717
0.2149	0	idStrt001	0.7438
0.1682	0	idStrt001	0.7507

Estimación de la propoción para y y $\widehat{y}$

Para verificar la calidad del modelo se realizan las estimaciones de las predicciones y de las variables observadas, teniendo estimaciones similares:

```
tab_pred %>%
  summarise(Pred = weighted.mean(Pred, wk2),
            pobreza = weighted.mean(pobreza,wk2))
```

Pred	pobreza
0.3855	0.3859

Logístico Multinivel con Coeficientes Aleatorios.

- Tanto el intercepto como la pendiente son variables aleatorias que varían entre los diferentes grupos de observación.
- La función logística se ajusta para cada grupo, permitiendo que los coeficientes del modelo varíen según el grupo de observación.
- Permite capturar la heterogeneidad en la relación entre variables predictoras y la respuesta en diferentes grupos, adaptándose a la variabilidad entre observaciones.

Logístico Multinivel con Coeficientes Aleatorios.

- La incorporación de coeficientes aleatorios mejora la precisión de las estimaciones y la capacidad del modelo para adaptarse a la variación entre grupos.
- Permite la inclusión de variables a nivel individual y de grupo, proporcionando una visión completa de la estructura jerárquica de los datos.
- La función logística se ajusta con coeficientes aleatorios para capturar las diferencias en la relación entre variables predictoras y respuesta en grupos específicos.

Logístico Multinivel con Coeficientes Aleatorios.

El modelo se define de la siguiente manera:

$$logit(\pi_{ij}) = \beta_{0j} + \beta_{1j} Gasto_{ij} + \epsilon_{ij}$$

Con

$$\beta_{0j} = \gamma_{00} + \gamma_{01} Stratum_j + \tau_{0j}$$

y

$$\beta_{1j} = \gamma_{10} + \gamma_{11} Stratum_j + \tau_{1j}$$

Logístico Multinivel con Coeficientes Aleatorios.

En R, el ajuste se hace de la siguiente manera:

```
mod_logit_Pen_Aleatorio <- glmer(
  pobreza ~ Expenditure  + (1 + Expenditure| Stratum),
  data = encuesta, weights  =  wk2,
  binomial(link = "logit"))
performance::icc(mod_logit_Pen_Aleatorio)
```

ICC_adjusted	ICC_unadjusted	optional
0.8859	0.6534	FALSE

Modelo con intercepto y pendiente aleatoria

```
dat_pred <- encuesta %>% group_by(Stratum) %>%
  summarise(
    Expenditure = list(seq(min(Expenditure),
                           max(Expenditure), len = 100))) %>%
  tidyr::unnest_legacy()

dat_pred <- mutate(dat_pred,
       Proba = predict(mod_logit_Pen_Aleatorio,
                       newdata = dat_pred , type = "response"))

ggplot(data = dat_pred,
       aes(y = Proba, x = Expenditure,
           colour = Stratum)) +
   geom_line()+   theme_bw() +
  geom_point(data = encuesta, aes(y = pobreza, x = Expenditure))+
  theme(legend.position = "none",
        plot.title = element_text(hjust = 0.5))
```

Modelo con intercepto y pendiente aleatoria

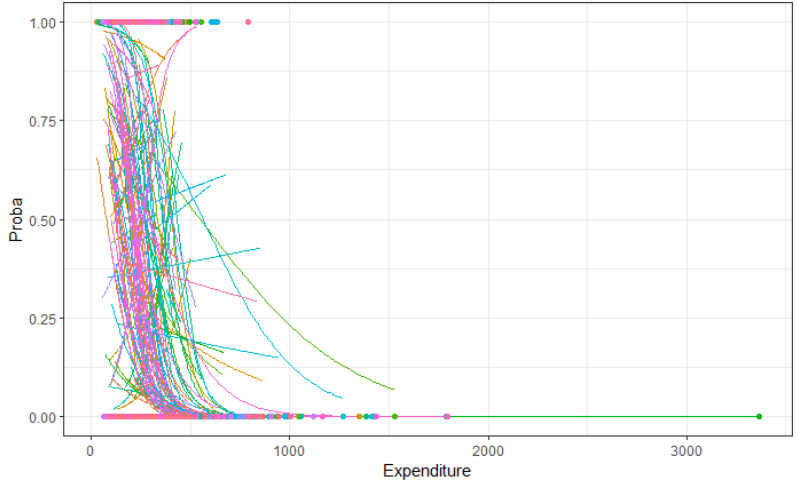

Figura 16: Modelo logit predicción

Predicción del modelo

Las predicciones se muestran a continuación:

```
(tab_pred <- data.frame(
  Pred = predict(mod_logit_Pen_Aleatorio,
                 type = "response"),
            pobreza = encuesta$pobreza,
            Stratum = encuesta$Stratum,
            wk2 = encuesta$wk2)) %>% distinct() %>%
  slice(1:6L)
```

Pred	pobreza	Stratum	wk2
0.0154	0	idStrt001	0.7770
0.0154	0	idStrt001	0.7501
0.0154	0	idStrt001	0.7463
0.0154	0	idStrt001	0.7717
0.0154	0	idStrt001	0.7438
0.0045	0	idStrt001	0.7507

Estimación de la propoción para y y $\widehat{y}$

```
tab_pred %>%
  summarise(Pred = weighted.mean(Pred, wk2),
            pobreza = weighted.mean(pobreza,wk2))
```

Pred	pobreza
0.3845	0.3859

Modelo con intercepto y pendiente aleatoria

Se ajusta un modelo agregando ahora la variable zona. La idea es entonces medir el porcentaje de pobreza discriminando por zona. El modelo es el siguiente:

$$logit(\pi_{ij}) = \beta_{0j} + \beta_{1j} Gasto_{ij} + \beta_{2j} Zona_{ij} + \epsilon_{ij}$$

donde

$$\begin{aligned}
\beta_{0j} &= \gamma_{00} + \gamma_{01} Stratum_j + \gamma_{02}\mu_j + \tau_{0j} \\
\beta_{1j} &= \gamma_{10} + \gamma_{11} Stratum_j + \gamma_{12}\mu_j + \tau_{1j} \\
\beta_{2j} &= \gamma_{20} + \gamma_{21} Stratum_j + \gamma_{12}\mu_j + \tau_{2j}
\end{aligned}$$

donde μ_j es el gasto medio en el estrato j.

Modelo con intercepto y pendiente aleatoria

El ajuste del modelo es el siguiente:

```
mod_logit_Pen_Aleatorio2 <- glmer(
  pobreza ~ 1 + Expenditure + Zone + mu +
    (1 + Expenditure + Zone + mu | Stratum ),
    data = encuesta, weights  =  wk2,
  binomial(link = "logit"))
performance::icc(mod_logit_Pen_Aleatorio2)
```

ICC_adjusted	ICC_unadjusted	optional
0.9598	0.8183	FALSE

Gráfica del modelo obtenido

Se grafican los modelos ajustados anteriormente:

```
dat_pred <- encuesta %>% group_by(Stratum, Zone, mu) %>%
  summarise(
    Expenditure = list(seq(min(Expenditure),
                           max(Expenditure), len = 100))) %>%
  tidyr::unnest_legacy()

dat_pred$Proba = predict(mod_logit_Pen_Aleatorio2,
                         newdata = dat_pred , type = "response")

ggplot(data = dat_pred,
       aes(y = Proba, x = Expenditure,
           colour = Stratum)) +
   geom_line()+   theme_bw() +facet_grid(.~Zone)+
  geom_point(data = encuesta, aes(y = pobreza, x = Expenditure))+
  theme(legend.position = "none",
        plot.title = element_text(hjust = 0.5))
```

Modelo con intercepto y pendiente aleatoria

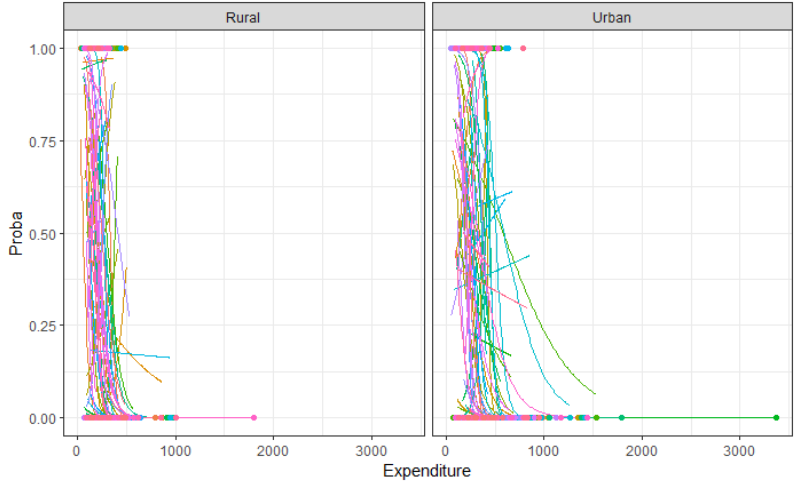

Figura 17: Modelo logit predicción

Predicción del modelo

Las predicciones del porcentaje de pobreza por zona se calculan a continuación:

```
(tab_pred <- data.frame(
  Pred = predict(mod_logit_Pen_Aleatorio2,
                 type = "response"),
            pobreza = encuesta$pobreza,
            Stratum = encuesta$Stratum,
            Zone = encuesta$Zone,
            wk2 = encuesta$wk2)) %>% distinct() %>%
  slice(1:5L)
```

Pred	pobreza	Stratum	Zone	wk2
0.0015	0	idStrt001	Rural	0.7770
0.0015	0	idStrt001	Rural	0.7501
0.0015	0	idStrt001	Rural	0.7463
0.0015	0	idStrt001	Rural	0.7717
0.0015	0	idStrt001	Rural	0.7438

Estimación de la propoción para y y $\widehat{y}$

Se verifica la calidad de las predicciones, obteniendo, como en los modelos anteriores, unas predicciones de buena calidad haciendo las comparaciones con las estimaciones de la variable observada para cada una de las zonas.

```
tab_pred %>% group_by(Zone) %>%
  summarise(Pred = weighted.mean(Pred, wk2),
            pobreza = weighted.mean(pobreza,wk2))
```

Zone	Pred	pobreza
Rural	0.4211	0.4298
Urban	0.3431	0.3437

¡Gracias!

Email: andres.gutierrez@cepal.org