Modelos multinivel

CEPAL

21/3/2022

Lectura de la base

```r
encuesta <- readRDS("../Data/encuesta.rds")
```

Creando theme_cepal

```r
theme_cepal <- function(...) theme_light(10) +
  theme(axis.text.x = element_blank(),
        axis.ticks.x = element_blank(),
        axis.text.y = element_blank(),
        axis.ticks.y = element_blank(),
        legend.position="bottom",
        legend.justification = "left",
        legend.direction="horizontal",
        plot.title = element_text(size = 20, hjust = 0.5),
        ...)
```

Introducción a los modelos multinivel.

```
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(1:6L)
```

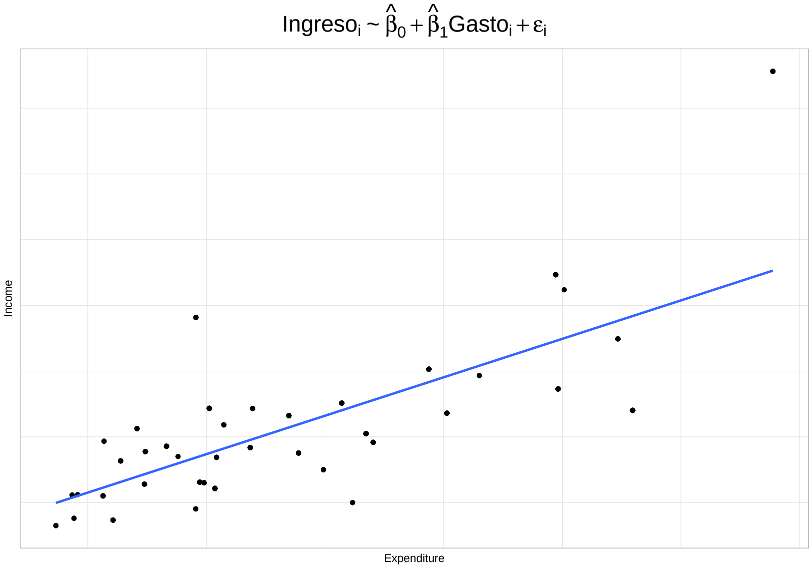
Introducción a los modelos multinivel.

Income	Expenditure	Stratum	Sex	Region	Zone
502.6	314.1	idStrt112	Male	Oriente	Rural
502.6	314.1	idStrt112	Female	Oriente	Rural
502.6	314.1	idStrt112	Male	Oriente	Rural
502.6	314.1	idStrt112	Male	Oriente	Rural
502.6	314.1	idStrt112	Female	Oriente	Rural
200.0	323.2	idStrt112	Female	Oriente	Rural

Introducción a los modelos multinivel.

```
ggplot(data = encuesta_plot, aes(y = Income, x = Expenditur
  geom_jitter() +
  theme( legend.position="none",
         plot.title = element_text(hjust = 0.5)) +
  geom_smooth( formula = y ~ x, method = "lm", se = F) +
  ggtitle(
    latex2exp::TeX("$Ingreso_{i}\\sim\\hat{\\beta}_{0}+\\ha
  theme_cepal()
```

Introducción a los modelos multinivel.



Introducción a los modelos multinivel.

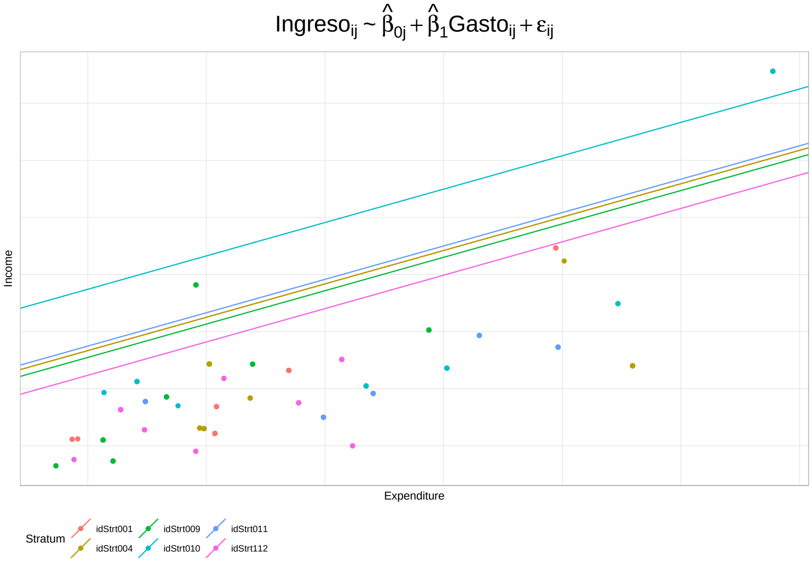
```r
B1 <- coef(lm(Income ~ Expenditure, data = encuesta_plot))
(coef_Mod <- encuesta_plot %>% group_by(Stratum) %>%
  summarise(B0 = coef(lm(Income ~ 1))[1]) %>%
  mutate(B1 = B1))
```

Stratum	B0	B1
idStrt001	416.7	1.169
idStrt004	416.5	1.169
idStrt009	392.8	1.169
idStrt010	631.5	1.169
idStrt011	432.5	1.169
idStrt112	330.0	1.169

Introducción a los modelos multinivel.

```
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_Mod,
              mapping=aes(slope=B1,
                          intercept=B0, colour = Stratum))
  ggtitle(
    latex2exp::TeX("$Ingreso_{ij}\\sim\\hat{\\beta}_{0j}+\\
  theme_cepal()
```

Introducción a los modelos multinivel.



Introducción a los modelos multinivel.

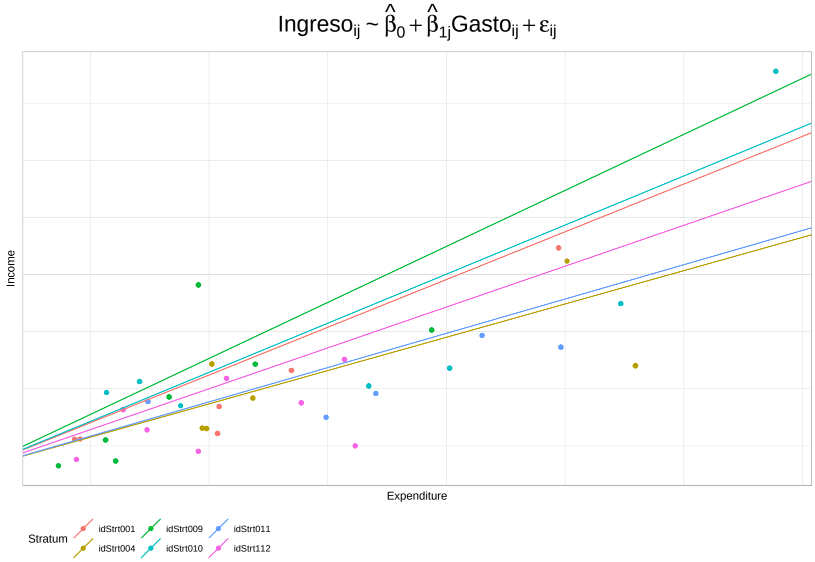
```
B0 <- coef(lm(Income ~ Expenditure, data = encuesta_plot))[1]
(coef_Mod <- encuesta_plot %>% group_by(Stratum) %>%
  summarise(B1 = coef(lm(Income ~ -1 + Expenditure))[1]) %>%
  mutate(B0 = B0))
```

Stratum	B1	B0
idStrt001	1.672	113.5
idStrt004	1.167	113.5
idStrt009	1.962	113.5
idStrt010	1.720	113.5
idStrt011	1.201	113.5
idStrt112	1.431	113.5

Introducción a los modelos multinivel.

```
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_Mod,
              mapping=aes(slope=B1,
                          intercept=B0, colour = Stratum))
  ggtitle(
    latex2exp::TeX("$Ingreso_{ij}\\sim\\hat{\\beta}_{0}+\\h
  theme_cepal()
```

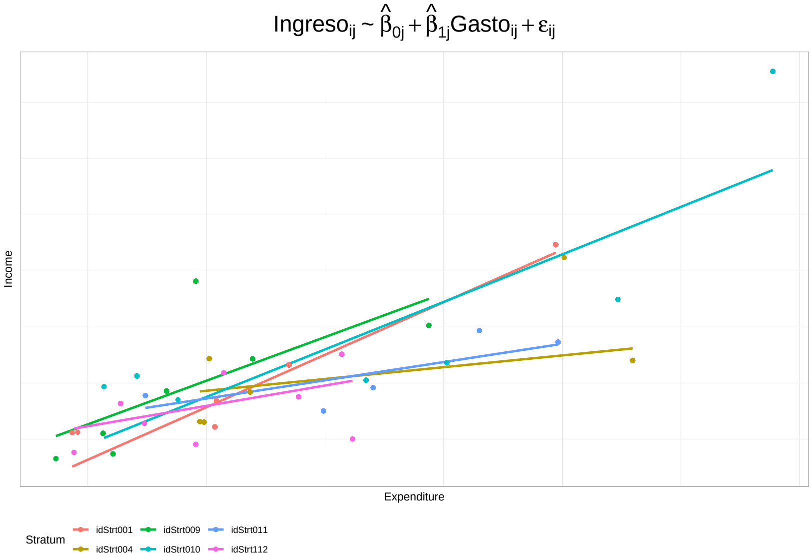
Introducción a los modelos multinivel.



Introducción a los modelos multinivel.

```
ggplot(data = encuesta_plot,
       aes(y = Income, x = Expenditure,
           colour = Stratum)) +
    geom_smooth( formula = y ~ x, method = "lm", se = F) +
  geom_jitter() + theme(legend.position="none",
    plot.title = element_text(hjust = 0.5)) +
  ggtitle(
    latex2exp::TeX("$Ingreso_{ij}\\sim\\hat{\\beta}_{0j}+\\
  theme_cepal()
```

Introducción a los modelos multinivel.



Introducción a los modelos multinivel.

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.

Introducción a los modelos multinivel.

- 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.

Introducción a los modelos multinivel.

Un modelo básico es:

$y_{ij} = \beta_{0j} + \epsilon_{ij}$

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

- $y_{ij} =$ Los ingresos de la persona i en el estrato j.
- $\beta_{0j} =$ 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:

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

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.

Introducción modelos multinivel.

- 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:

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

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.

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

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

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

Ajuste de pesos (alternativa a los Modelo q-weighted)

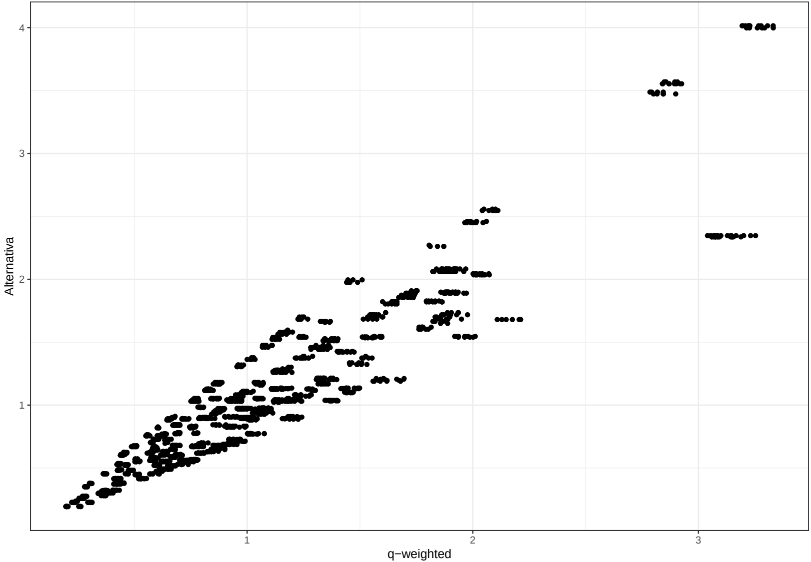
```
mod_qw <- lm(wk ~ Age + Sex + Region + Zone,
             data = encuesta)
encuesta$wk2 <-   encuesta$wk/predict(mod_qw)
# alternativa los q-weighted
n = nrow(encuesta)
encuesta <- encuesta %>% mutate(wk3 = n*wk/sum(wk))
encuesta %>% summarise(fep = sum(wk),
                       q_wei = sum(wk2),
                       fep2 = sum(wk3) )
```

fep	q_wei	fep2
150266	2420	2422

Comparando los pesos.

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

Comparando los pesos.



Modelo Nulo

```
library(lme4)

mod_null  <- lmer( Income  ~ ( 1  |  Stratum ),
                       data  =  encuesta,
                 weights  =  wk2 )
mod_null2  <- lmer( Income  ~ ( 1  |  Stratum ),
                       data  =  encuesta,
                 weights  =  wk3 )
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	424.0	427.1
idStrt002	955.4	949.2
idStrt003	349.4	356.8
idStrt004	423.2	423.9
idStrt005	170.8	179.2
idStrt006	402.3	406.1
idStrt007	307.0	310.6
idStrt008	634.1	632.7
idStrt009	401.7	405.0
idStrt010	628.3	628.4
idStrt011	423.2	425.5
idStrt012	586.8	586.9

Modelo Nulo

```
mod_null
```

```
## Linear mixed model fit by REML ['lmerMod']
## Formula: Income ~ (1 | Stratum)
##    Data: encuesta
## Weights: wk2
## REML criterion at convergence: 35124
## Random effects:
##  Groups   Name        Std.Dev.
##  Stratum  (Intercept) 242
##  Residual             298
## Number of obs: 2422, groups:  Stratum, 119
## Fixed Effects:
## (Intercept)
##         538
```

Modelo Nulo

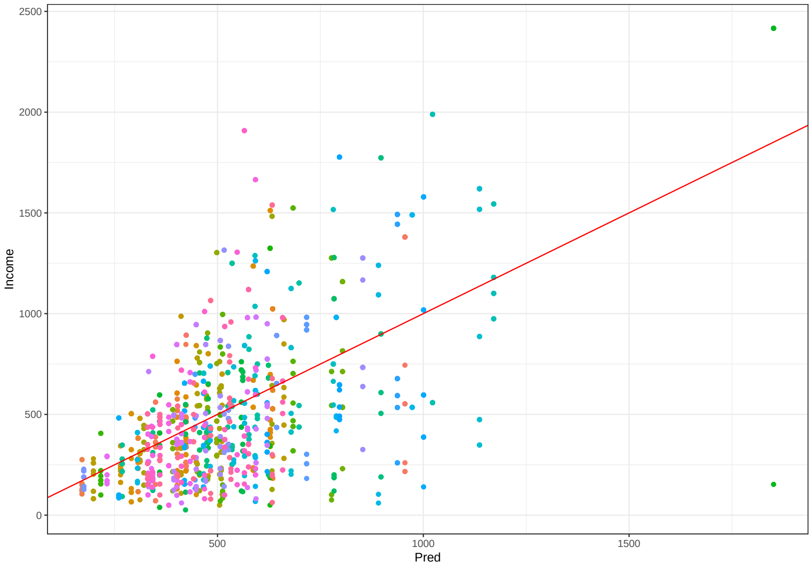
```r
#library(sjstats)
sjstats::icc(mod_null)
```

```
## # Intraclass Correlation Coefficient
##
##      Adjusted ICC: 0.397
##   Conditional ICC: 0.397
```

```r
(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	424	243.2	idStrt001
5	424	223.0	idStrt001
7	424	893.1	idStrt001
15	424	337.5	idStrt001
19	424	224.3	idStrt001

Scaterplot de y vs $\hat{y}$



Modelo con intercepto aleatoria

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

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

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

sjstats::icc(mod_Int_Aleatorio)
```

```
## # Intraclass Correlation Coefficient
##
##      Adjusted ICC: 0.280
##   Conditional ICC: 0.175
```

Modelo con intercepto aleatoria

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

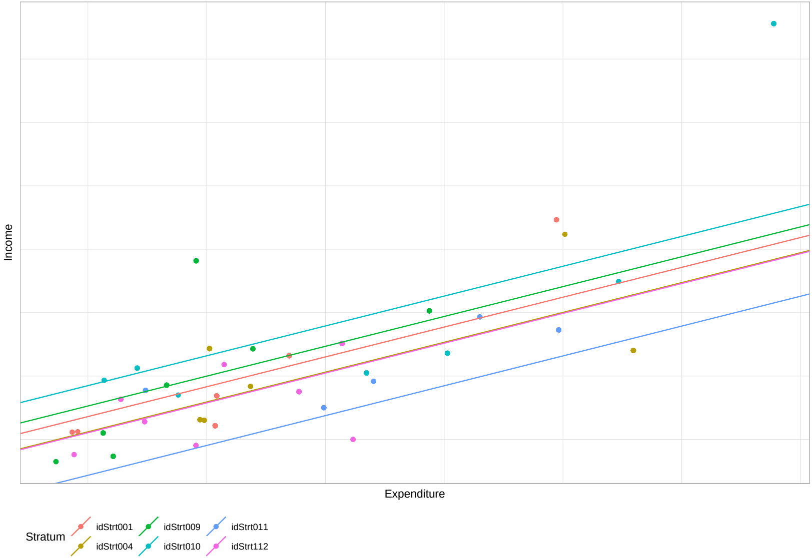
	(Intercept)	Expenditure
idStrt001	177.84	0.9416
idStrt002	537.77	0.9416
idStrt003	103.25	0.9416
idStrt004	129.82	0.9416
idStrt005	-23.29	0.9416
idStrt006	123.09	0.9416
idStrt007	122.92	0.9416
idStrt008	407.40	0.9416
idStrt009	211.30	0.9416
idStrt010	275.53	0.9416

Modelo con intercepto aleatoria

```
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

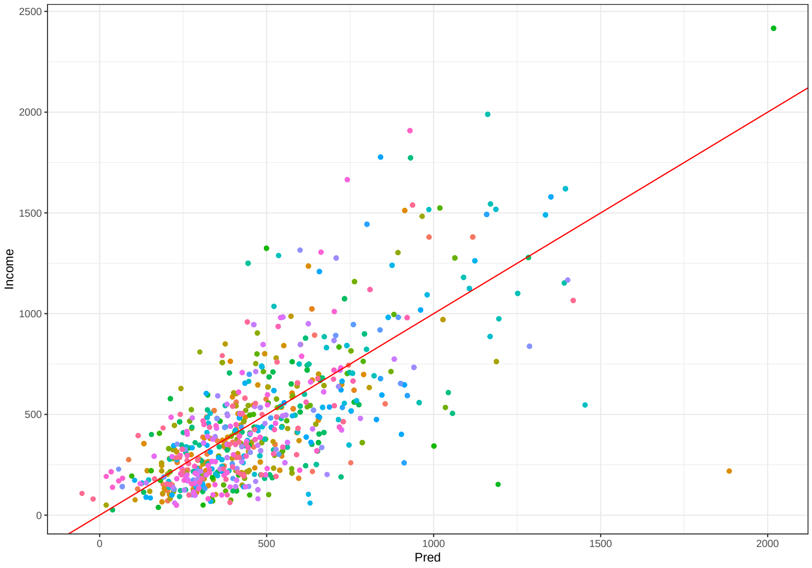


Predicción del modelo

```
(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	372.9	243.2	idStrt001
5	259.6	223.0	idStrt001
7	643.4	893.1	idStrt001
15	374.1	337.5	idStrt001
19	264.0	224.3	idStrt001
20	431.6	464.2	idStrt001

Scaterplot de y vs $\hat{y}$



Modelo con intercepto y pendiente aleatoria

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

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

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

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

sjstats::icc(mod_Pen_Aleatorio)
```

```
## # Intraclass Correlation Coefficient
##
##      Adjusted ICC: 0.726
##   Conditional ICC: 0.510
```

Modelo con intercepto y pendiente aleatoria

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

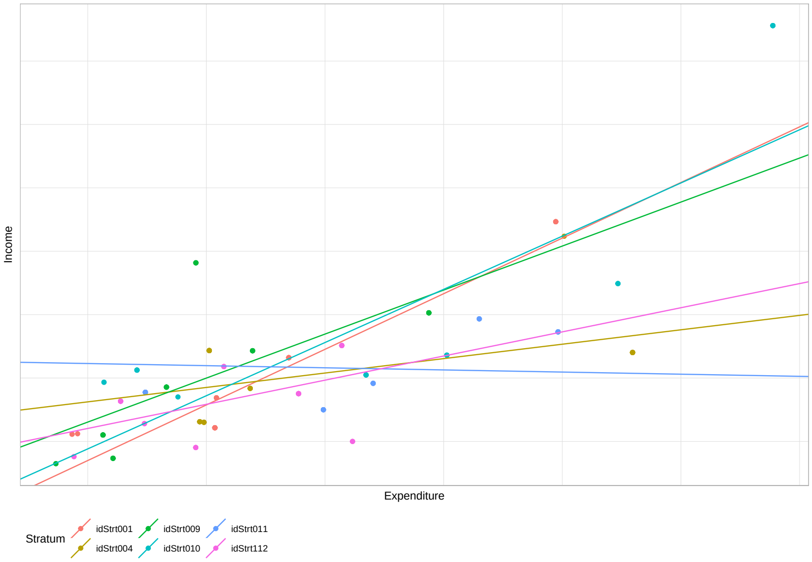
	(Intercept)	Expenditure
idStrt001	-35.7062	1.7543
idStrt002	-80.4434	2.4616
idStrt003	0.2362	1.2955
idStrt004	279.3118	0.4547
idStrt005	42.9532	0.5780
idStrt006	229.4414	0.5565
idStrt007	29.0805	1.4039
idStrt008	379.8502	1.1059
idStrt009	122.2335	1.3881
idStrt010	8.6536	1.6779

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

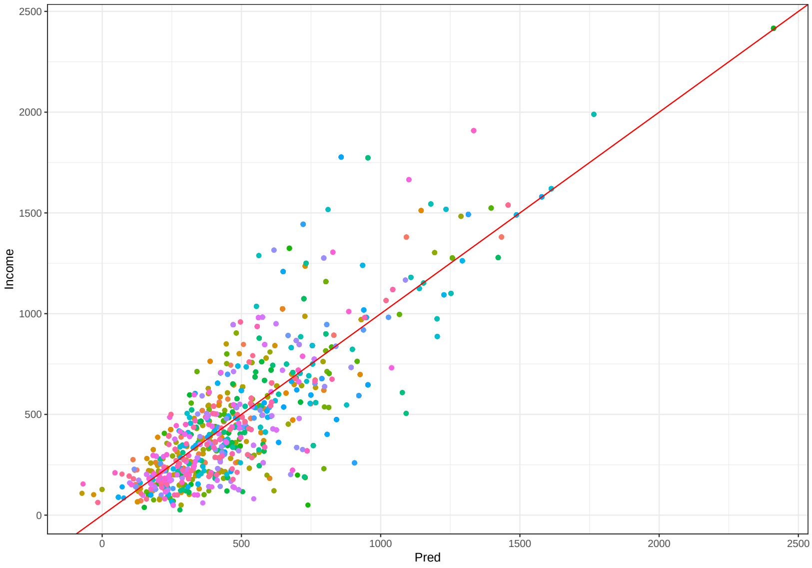


Predicción del modelo

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

	Pred	Income	Stratum
1	327.7	243.2	idStrt001
5	116.6	223.0	idStrt001
7	831.8	893.1	idStrt001
15	330.0	337.5	idStrt001
19	124.7	224.3	idStrt001
20	437.0	464.2	idStrt001

Scaterplot de y vs $\hat{y}$



Modelo con intercepto y pendiente aleatoria

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

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

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

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

donde μ_j es el gasto medio en el estrato j.

Modelo con intercepto y pendiente aleatoria

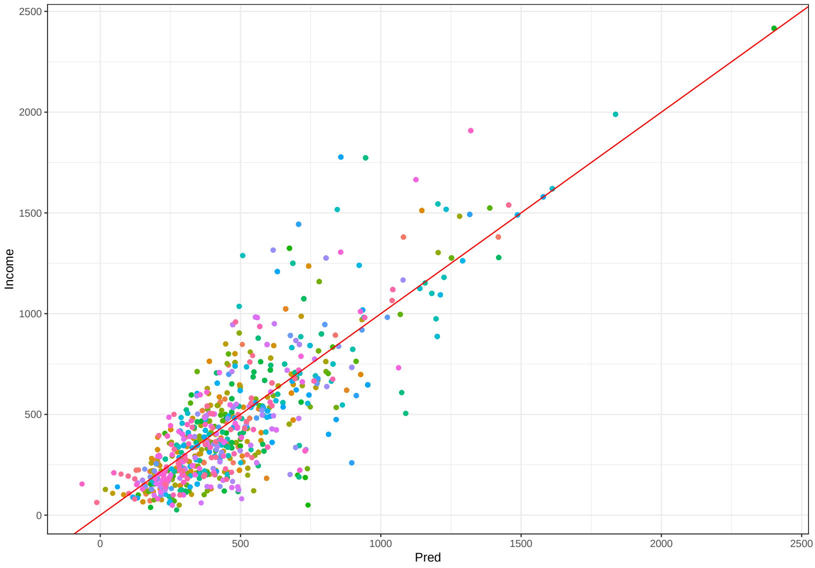
```r
media_estrato <- encuesta %>% group_by(Stratum) %>%
  summarise(mu = mean(Expenditure))
encuesta <- inner_join(encuesta,
                       media_estrato, by = "Stratum")

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

```
## # Intraclass Correlation Coefficient
##
##      Adjusted ICC: 0.596
##   Conditional ICC: 0.366
```

```r
(tab_pred <- data.frame(Pred = predict(mod_Pen_Aleatorio2),
           Income = encuesta$Income,
           Stratum = encuesta$Stratum)) %>% distinct() %>%
```

Scaterplot de y vs $\hat{y}$



Modelo con intercepto y pendiente aleatoria

```
as.data.frame( model.matrix(mod_Pen_Aleatorio2)) %>%
  distinct()
```

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
idStrt001	154.4	1.7418	77.353	-0.6954	Rural
idStrt004	112.6	0.4495	-15.163	0.6007	Urban
idStrt009	205.4	1.4888	8.512	-0.5310	Rural
idStrt010	155.2	1.6702	82.972	-0.6101	Urban
idStrt011	164.6	-0.0642	142.837	0.6058	Rural
idStrt112	106.5	0.7362	63.695	0.2932	Rural

Modelo con intercepto y pendiente aleatoria

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

Stratum	(Intercept)	Expenditure	ZoneUrban	mu.x	Zone	mu.y
idStrt001	154.4	1.7418	77.353	-0.6954	Rural	255.2
idStrt004	112.6	0.4495	-15.163	0.6007	Urban	305.5
idStrt009	205.4	1.4888	8.512	-0.5310	Rural	190.2
idStrt010	155.2	1.6702	82.972	-0.6101	Urban	365.9
idStrt011	164.6	-0.0642	142.837	0.6058	Rural	474.8
idStrt112	106.5	0.7362	63.695	0.2932	Rural	216.2

El modelo para el estrato *idStrt001* viene dado por:

$$\hat{y}_{ij} = 154.4 + 1.7418 Expenditure_{ij} + 77.353 Zone_{ij} + (-0.6954)\,\mu_j$$

$$\hat{y}_{ij} = 154.4 + 1.7418 Expenditure + 77.353\,(0) + (-0.6954)\,(255.2)$$

$$\hat{y}_{ij} = -23.07 + 1.7418 Expenditure$$

Modelo con intercepto y pendiente aleatoria

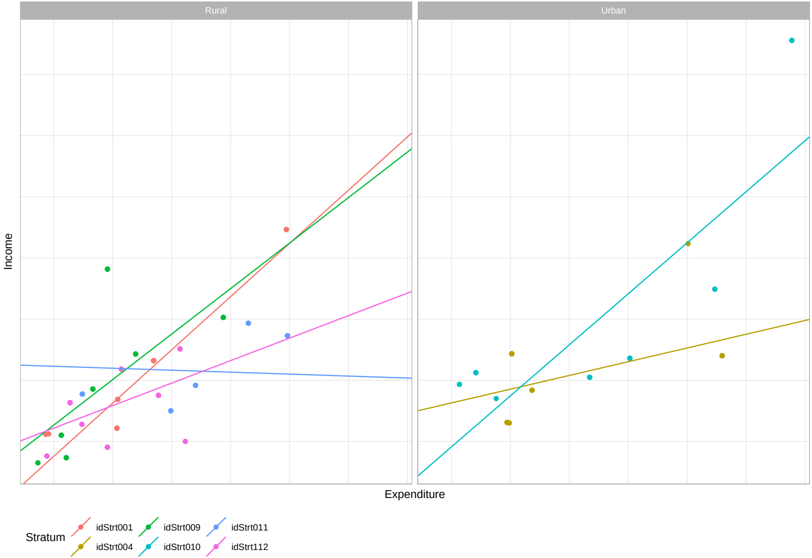
```
(Coef_Estimado %<>%  mutate(B0 = ifelse(
Zone == "Urban", `(Intercept)` + mu.y * mu.x + ZoneUrban,
              `(Intercept)` + mu.y * mu.x)) %>%
  select(Stratum, Zone, B0, Expenditure))
```

Stratum	Zone	B0	Expenditure
idStrt001	Rural	-23.04	1.7418
idStrt004	Urban	280.89	0.4495
idStrt009	Rural	104.46	1.4888
idStrt010	Urban	14.90	1.6702
idStrt011	Rural	452.24	-0.0642
idStrt112	Rural	169.86	0.7362

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



Introducción a los modelos logístico multinivel.

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

$$Pr\left(y_{ij}\right) = Pr\left(y_{ij} = 1 \mid x_i : \boldsymbol{\beta}\right) = \frac{1}{1 + \exp\left(-\boldsymbol{\beta}_j \boldsymbol{x}_{ij}\right)}$$

ó

$$\log\left(\frac{\pi_{ij}}{1 - \pi_{ij}}\right) = \boldsymbol{\beta}_j \boldsymbol{x}_{ij}$$

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

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

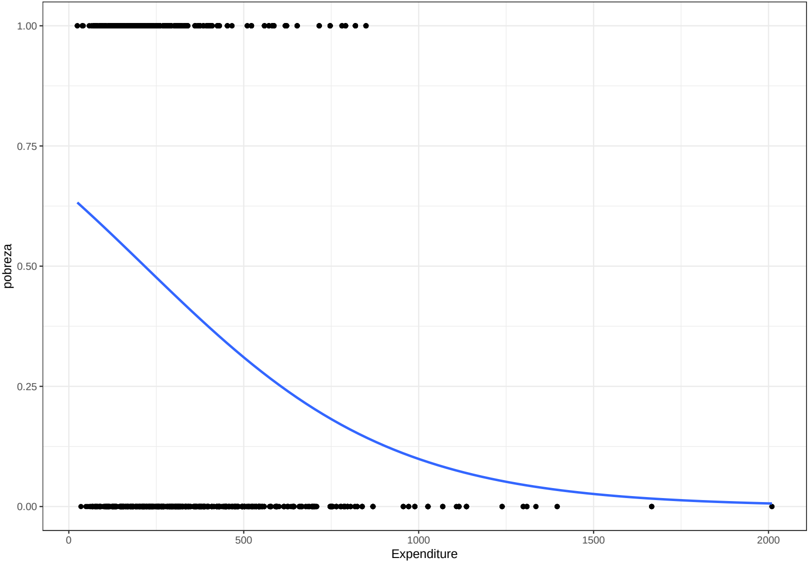
Poverty	Expenditure	Stratum	Sex	Region	Zone
NotPoor	148.5	idStrt011	Female	Norte	Rural
NotPoor	148.5	idStrt011	Female	Norte	Rural
NotPoor	148.5	idStrt011	Female	Norte	Rural
NotPoor	430.0	idStrt011	Male	Norte	Rural
NotPoor	430.0	idStrt011	Female	Norte	Rural
NotPoor	430.0	idStrt011	Female	Norte	Rural
NotPoor	148.5	idStrt011	Female	Norte	Rural
NotPoor	148.5	idStrt011	Female	Norte	Rural
NotPoor	148.5	idStrt011	Female	Norte	Rural
NotPoor	340.6	idStrt011	Female	Norte	Rural
NotPoor	340.6	idStrt011	Female	Norte	Rural
NotPoor	340.6	idStrt011	Male	Norte	Rural
NotPoor	496.4	idStrt011	Female	Norte	Rural
NotPoor	496.4	idStrt011	Male	Norte	Rural
NotPoor	496.4	idStrt011	Female	Norte	Rural

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



Ejemplos de modelo logit

```
auxLogit <- function(x,b0,b1){
  1/(1+exp(-(b0+b1*x)))
}
B0 = coef(glm(pobreza~1,data = encuesta_plot,
     family=binomial(link = "logit")))

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

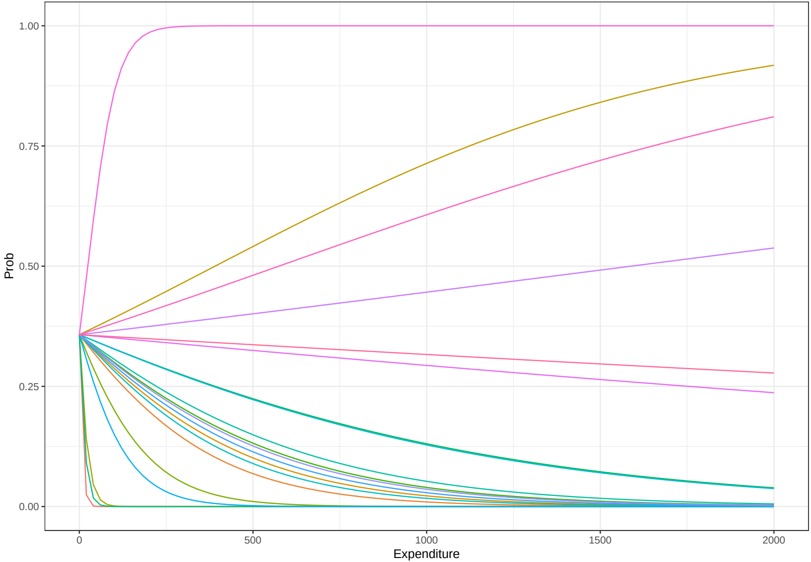
Stratum	B1	B0
idStrt011	-0.1537	-0.5862
idStrt020	-0.0041	-0.5862
idStrt024	-0.0032	-0.5862
idStrt038	0.0015	-0.5862
idStrt039	-0.0609	-0.5862
idStrt044	-0.0079	-0.5862

Ejemplos de modelo logit

```r
# 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")
```

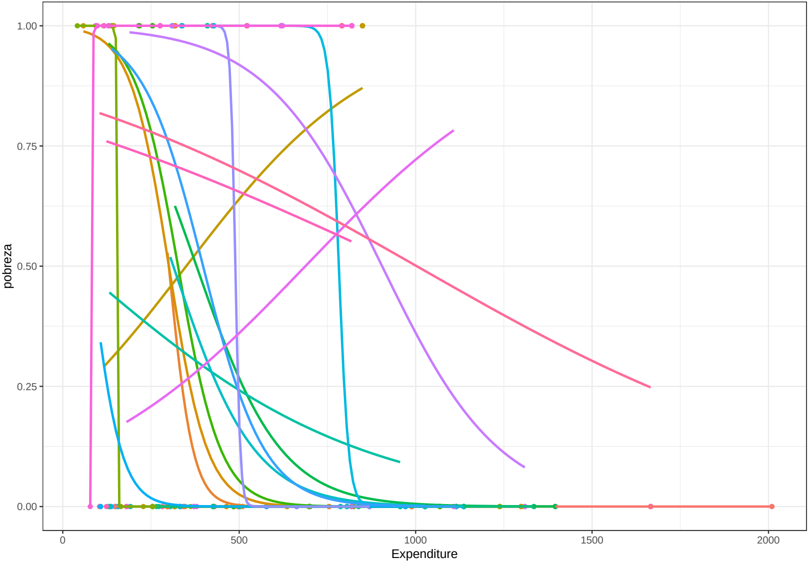
Ejemplos de modelo logit



Ejemplos de modelo logit

```
ggplot(data = encuesta_plot,
       aes(y = pobreza, x = Expenditure, colour = Stratum)) +
  geom_point() +
  geom_smooth(
    formula = y~x, method = "glm",
    se=FALSE,
    method.args = list(family=binomial(link = "logit")))+
  theme_bw() +
  theme(legend.position = "none")
```

Ejemplos de modelo logit



Un modelo básico es:

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

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

- $\pi_{ij} = Pr\left(y_{ij} = 1 \mid x_i : \boldsymbol{\beta}\right)$.
- $\beta_{0j} =$ 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	0.2849
idStrt002	-2.7120
idStrt003	-1.1122
idStrt004	0.7008
idStrt005	1.6889
idStrt006	1.4194
idStrt007	2.4396
idStrt008	1.3761
idStrt009	0.2984
idStrt010	0.2650
idStrt011	-2.9291
idStrt012	-0.8685

Modelo Nulo

```
library(sjstats)
mod_logist_null
```

```
## Generalized linear mixed model fit by maximum likelihoo
##   Approximation) [glmerMod]
##  Family: binomial  ( logit )
## Formula: pobreza ~ (1 | Stratum)
##    Data: encuesta
## Weights: wk2
##      AIC      BIC   logLik deviance df.resid
##     2618     2630    -1307     2614     2420
## Random effects:
##  Groups  Name        Std.Dev.
##  Stratum (Intercept) 1.44
## Number of obs: 2422, groups:  Stratum, 119
## Fixed Effects:
## (Intercept)
##      -0.605
```

Modelo Nulo

```
sjstats::icc(mod_logist_null)
```

```
## # Intraclass Correlation Coefficient
##
##      Adjusted ICC: 0.386
##   Conditional ICC: 0.386
```

```
(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	0.5707	1	idStrt001
5	0.5707	0	idStrt001
28	0.0623	0	idStrt002
46	0.2475	0	idStrt003
50	0.2475	1	idStrt003
64	0.6684	0	idStrt004

Estimación de la propoción total con y y $\hat{y}$

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

```
## [1] 0.4266
```

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

```
## [1] 0.4261
```

Modelo con intercepto aleatoria

$$logit(\pi_{ij}) = \beta_0 + \beta_{1j} Gasto_{ij} + \epsilon_{ij}$$

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

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

sjstats::icc(mod_logit_Int_Aleatorio)
```

```
## # Intraclass Correlation Coefficient
##
##      Adjusted ICC: 0.356
##   Conditional ICC: 0.304
```

Modelo con intercepto aleatoria

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

	(Intercept)	Expenditure
idStrt001	1.3104	-0.0039
idStrt002	-1.2031	-0.0039
idStrt003	-0.0682	-0.0039
idStrt004	1.9640	-0.0039
idStrt005	2.5457	-0.0039
idStrt006	2.5890	-0.0039
idStrt007	3.2405	-0.0039
idStrt008	2.2885	-0.0039
idStrt009	1.0931	-0.0039
idStrt010	1.7075	-0.0039

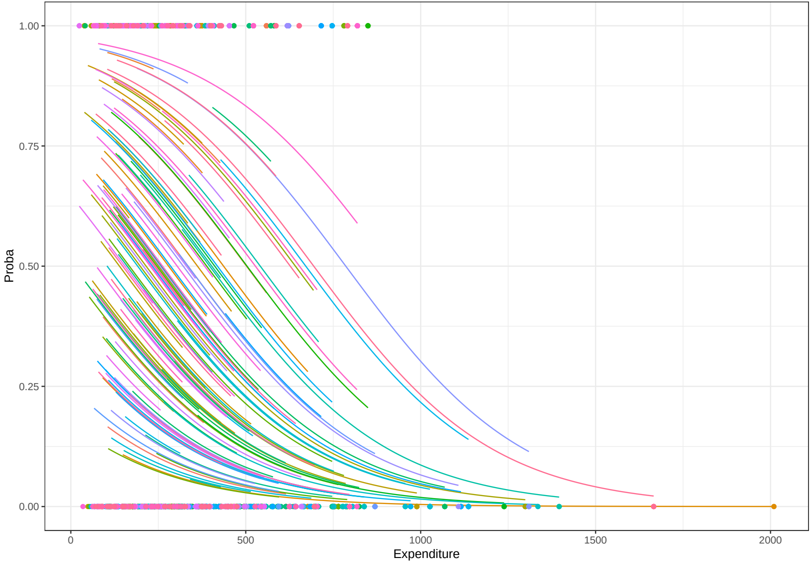
Modelo con intercepto aleatoria

```
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_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



Predicción del modelo

```
(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
0.6225	1	idStrt001	0.8791
0.6225	1	idStrt001	0.8636
0.6225	1	idStrt001	0.8574
0.6225	1	idStrt001	0.8557
0.7253	0	idStrt001	0.8977
0.7253	0	idStrt001	0.9012

Estimación de la propoción total con y y $\hat{y}$

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

Pred	pobreza
0.426	0.4266

Modelo con intercepto y pendiente aleatoria

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

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

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

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

sjstats::icc(mod_logit_Pen_Aleatorio)
```

```
## # Intraclass Correlation Coefficient
##
##      Adjusted ICC: 0.830
##   Conditional ICC: 0.677
```

Modelo con intercepto y pendiente aleatoria

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

	(Intercept)	Expenditure
idStrt001	1.2576	-0.0036
idStrt002	-0.9801	-0.0071
idStrt003	2.7782	-0.0182
idStrt004	-0.5403	0.0045
idStrt005	-0.3587	0.0112
idStrt006	3.4107	-0.0056
idStrt007	4.3334	-0.0078
idStrt008	5.4260	-0.0165
idStrt009	5.6493	-0.0287
idStrt010	2.3016	-0.0053

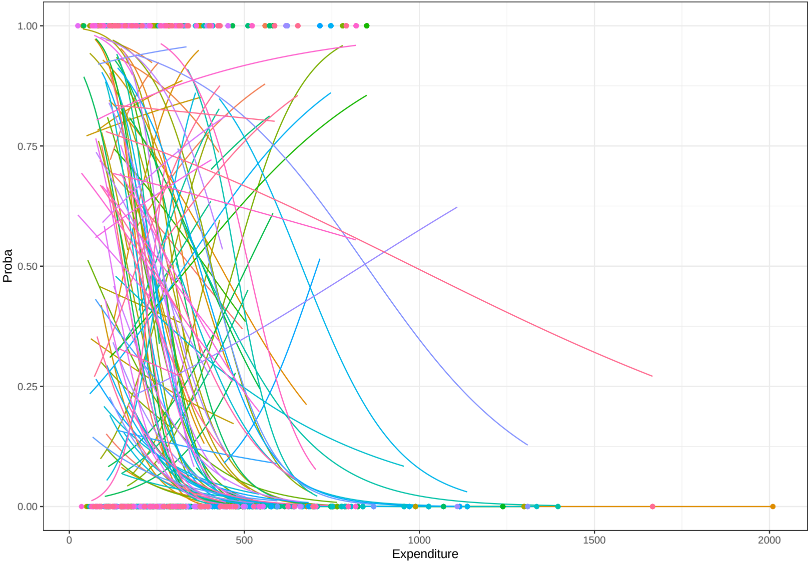
Modelo con intercepto y pendiente aleatoria

```
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



Predicción del modelo

```
(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) # Son las pendientes aleatorias
```

Pred	pobreza	Stratum	wk2
0.6241	1	idStrt001	0.8791
0.6241	1	idStrt001	0.8636
0.6241	1	idStrt001	0.8574
0.6241	1	idStrt001	0.8557
0.7197	0	idStrt001	0.8977
0.7197	0	idStrt001	0.9012

Estimación de la propoción total con y y $\hat{y}$

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

Pred	pobreza
0.4259	0.4266

Modelo con intercepto y pendiente aleatoria

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

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

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

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

donde μ_j es el gasto medio en el estrato j.

Modelo con intercepto y pendiente aleatoria

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

```
## # Intraclass Correlation Coefficient
##
##      Adjusted ICC: 0.786
##   Conditional ICC: 0.573
```

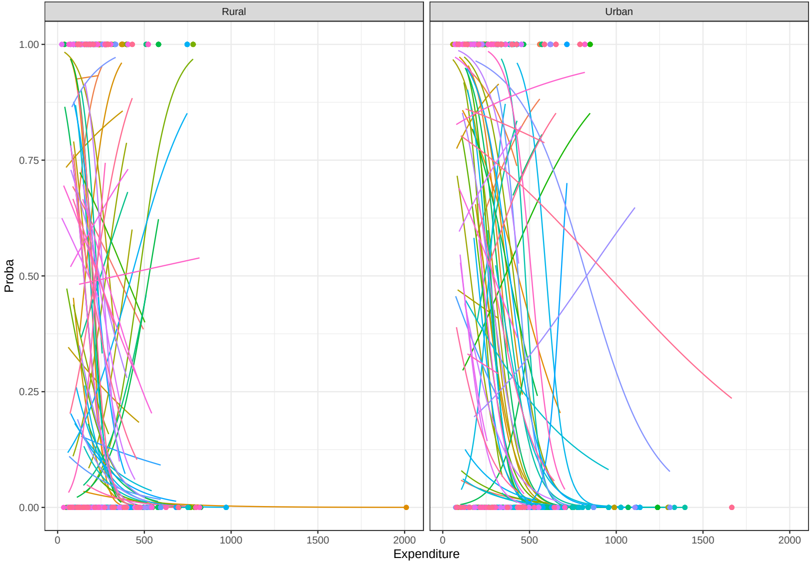
Gráfica del modelo obtenido

```
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



Predicción del modelo

```
(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(1:6L)
```

Pred	pobreza	Stratum	Zone	wk2
0.6074	1	idStrt001	Rural	0.8791
0.6074	1	idStrt001	Rural	0.8636
0.6074	1	idStrt001	Rural	0.8574
0.6074	1	idStrt001	Rural	0.8557
0.6933	0	idStrt001	Rural	0.8977
0.6933	0	idStrt001	Rural	0.9012

Estimación de la propoción total con y y $\hat{y}$

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

Zone	Pred	pobreza
Rural	0.4577	0.4584
Urban	0.3938	0.3939