Análisis de encuestas de hogares con R

Modulo 7: 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 apartado, usaremos la muestra de hogares

```
encuesta_hog <- readRDS("Imagenes/06_MLG1/ENIGH_HND_Hogar.rds")
encuesta_hog <- encuesta_hog %>% # Base de datos.
  filter(YDISPONIBLE_PER > 0 ) %>%
  transmute(
    LLAVE_HOGAR,
    estrato = haven::as_factor(F1_A0_ESTRATO),
    dam = haven::as_factor(F1_A0_DEPARTAMENTO),
    Area = haven::as_factor(F1_A0_AREA),
    ingreso_per  = ifelse(YDISPONIBLE_PER < 0 , 0 , YDISPONIBLE_PER) ,
    gasto_per = GASTO_CORRIENTE_HOGAR / CANTIDAD_PERSONAS,
    pobreza_LP = case_when(
      ingreso_per < 3046 & Area == "1. Urbana" ~ 1,
      ingreso_per < 1688  &
        Area == "2. Rural" ~ 1,
      TRUE ~ 0
    ),
    log_gasto = log(gasto_per + 500),
    log_ingreso = log(ingreso_per + 500),
     TIPOVIVIENDA = haven::as_factor(F1_A1_P1_TIPOVIVIENDA),
     TIENEVEHICULOS = haven::as_factor(F2_A2_P1_TIENEVEHICULOS),
    Factor
  )
dam_pba <- c("1. Atlántida", "2. Colon",
             "3. Comayagua", "4. Copan", "5. Cortes")
encuesta_hog  %>% slice(1:10L) %>% dplyr::select(dam:pobreza_LP)
```

Introducción a los modelos multinivel.

dam	Area	ingreso_per	gasto_per	pobreza_LP
1. Atlántida	1. Urbana	18.75	1346.0	1
1. Atlántida	1. Urbana	20.83	1831.3	1
4. Copan	1. Urbana	57.71	4483.3	1
1. Atlántida	1. Urbana	68.06	12584.5	1
10. Intibuca	1. Urbana	111.94	706.9	1
14. Ocotepeque	2. Rural	160.00	10721.9	1
15. Olancho	2. Rural	166.67	1975.9	1
9. Gracias A Dios	2. Rural	180.56	1541.7	1
2. Colon	1. Urbana	187.50	4106.4	1
2. Colon	2. Rural	196.06	1354.2	1

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 dams del diseño muestral. A continuación, se muestra la gráfica:

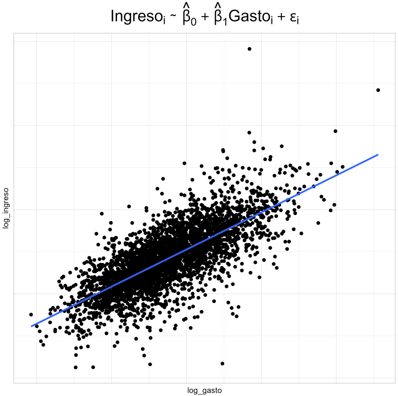


Figura 1: Modelo regresión simple

Introducción a los modelos multinivel.

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

```
modelo <- lm(log_ingreso ~ 0 + dam + log_gasto, data = encuesta_hog)
B1 <- as.numeric(coef(modelo)["log_gasto"])
B0 <- coef(modelo)[1:18]
names(B0) <- stri_replace_all_fixed( str = names(B0),
  pattern = "dam",  replacement = "" )
coef_Mod <- data.frame(dam = names(B0), B0 = as.numeric(B0),
           B1 = B1) %>% filter(dam %in% dam_pba)
coef_Mod
```

dam	B0	B1
1. Atlántida	0.8629	0.904
2. Colon	0.8513	0.904
3. Comayagua	0.9388	0.904
4. Copan	0.9545	0.904
5. Cortes	0.9543	0.904

Introducción a los modelos multinivel.

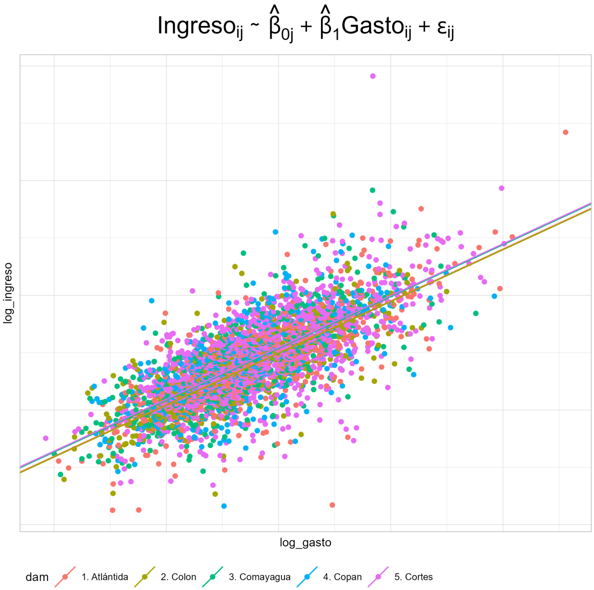


Figura 2: Modelo de regresión con intercepto variando por departamento

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

```r
modelo <- lm(log_ingreso ~ log_gasto:dam + 1, data = encuesta_hog)

B0 <- as.numeric(coef(modelo)["(Intercept)"])
B1 <- coef(modelo)[2:19]
names(B1) <- stri_replace_all_fixed(
  str = names(B1),  pattern = "log_gasto:dam", replacement = "")
coef_Mod <- data.frame(dam = names(B1), B0 = B0,
                       B1 = as.numeric(B1)) %>%
  filter(dam %in% dam_pba)

coef_Mod
```

dam	B0	B1
1. Atlántida	0.9383	0.8956
2. Colon	0.9383	0.8936
3. Comayagua	0.9383	0.9042

Introducción a los modelos multinivel.

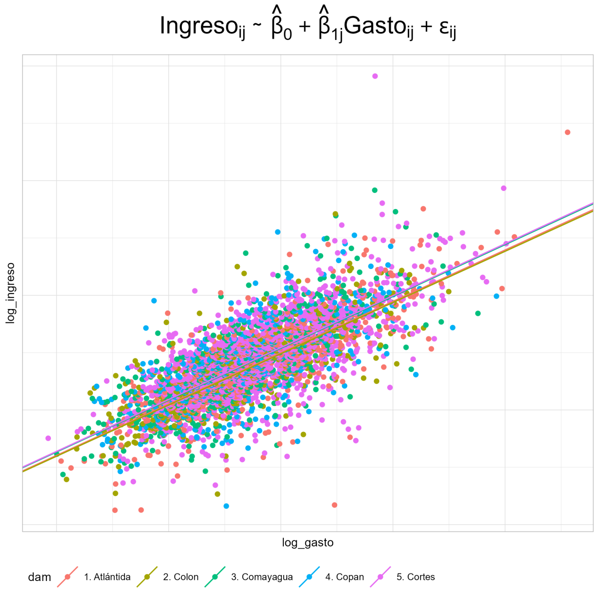


Figura 3: Modelo de regresión con pendiente variando por departamento

Introducción a los modelos multinivel.

Creando un gráfico con intercepto y pendientes aleatorias.

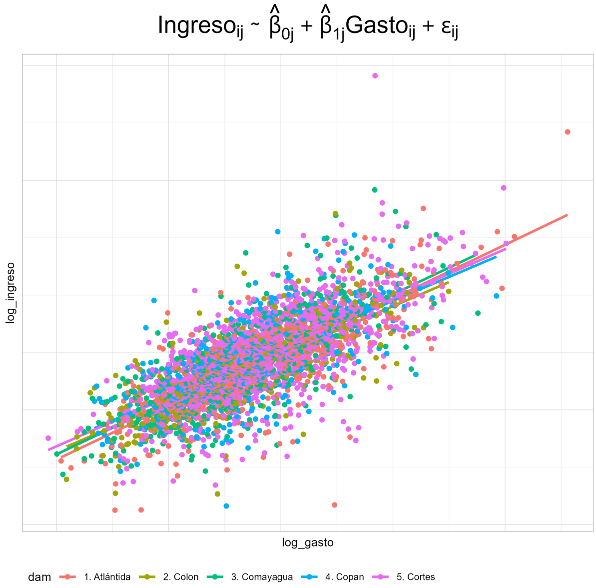


Figura 4: Modelo de regresión con intercepto y pendiente variando por dam

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 dam
- la varianza entre los dam.

Estos dos componentes de varianza se pueden obtener en 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 del hogar i en el dam j.
- $\beta_{0j} =$ El intercepto en el dam j.
- ϵ_{ij} El residual del hogar i en el dam j.
- $\gamma_{00} =$ El intercepto en general.
- $\tau_{0j} =$ Efecto aleatorio para el intercepto.

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

La correlación intra clásica esta dada por:

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

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:

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

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 -$>$ dam.

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 dam esta definida por el intercepto.

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

calculando de los Qweighted

Para tener estimaciones consistentes se calculan los pesos Qweighted siguiendo los pasos mostrados, tomando en este caso como covariables el área donde reside, tenencia de vehículo y el tipo de vivienda.

```
mod_qw <- lm( Factor ~  Area + TIPOVIVIENDA +  TIENEVEHICULOS ,
             data = encuesta_hog)
summary(predict(mod_qw))
```

Min.	1st Qu.	Median	Mean	3rd Qu.	Max.
131.9	201.1	202	299.2	202	878.6

```
encuesta_hog$Factor2 <-   encuesta_hog$Factor/predict(mod_qw)
```

calculando los senate-weight

Adicionalmente, se calculan también los *senate-weight* para el ajuste de los modelos (Factor3, en el ajuste del modelo en R).

```
# Alternativa los Qweighted
n = nrow(encuesta_hog)
encuesta_hog <- encuesta_hog %>%
  mutate(Factor3 = n * Factor / sum(Factor))
encuesta_hog %>% summarise(
  fep = sum(Factor),
  q_wei = sum(Factor2),
  fep2 = sum(Factor3)
)
```

fep	q_wei	fep2
2614746	8739	8739

Comparando los pesos

```r
qw0 <- ggplot(encuesta_hog, aes(x = Factor2, y = Factor3)) +
  geom_point() +   theme_bw() +
  labs(x = "q-weighted", y = "senate-weight")
```

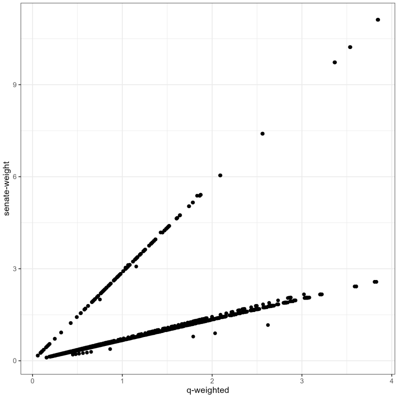


Figura 5: Comparando q-weighted Vs senate-weight

Modelo Nulo

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

```
library(lme4)

mod_null  <- lmer(log_ingreso   ~ (1  |  dam),
                  data  =  encuesta_hog,
                  weights  =  Factor2)

mod_null2  <- lmer(log_ingreso   ~ (1  |  dam),
                   data  =  encuesta_hog,
                   weights  =  Factor3)
```

Modelo Nulo

Comparando los modelos obtenidos.

```
coef_mod_null <- bind_cols(coef(mod_null)$dam,
                           coef(mod_null2)$dam)
colnames(coef_mod_null) <- c("Intercept Mod 1",
                             "Intercept Mod 2")
coef_mod_null
```

Modelo Nulo

	Intercept Mod 1	Intercept Mod 2
1. Atlántida	8.827	8.718
2. Colon	8.639	8.492
3. Comayagua	8.731	8.650
4. Copan	8.741	8.729
5. Cortes	8.878	8.872
6. Choluteca	8.904	8.747
7. El Paraíso	8.627	8.383
8. Francisco Morazán	8.929	8.860
9. Gracias A Dios	8.190	7.966
10. Intibuca	8.302	7.942
11. Islas De La Bahía	9.064	9.121
12. La Paz	8.543	8.313
13. Lempira	8.409	8.299
14. Ocotepeque	8.518	8.237
15. Olancho	8.627	8.458
16. Santa Bárbara	8.508	8.393
17. Valle	8.876	8.974
18. Yoro	8.664	8.599

Modelo Nulo

```r
mod_null
```

```
Linear mixed model fit by REML ['lmerMod']
Formula: log_ingreso ~ (1 | dam)
   Data: encuesta_hog
Weights: Factor2
REML criterion at convergence: 22067
Random effects:
 Groups   Name        Std.Dev.
 dam      (Intercept) 0.236
 Residual             0.776
Number of obs: 8739, groups:  dam, 18
Fixed Effects:
(Intercept)
       8.67
```

Modelo Nulo

Correlación intraclases

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

ICC_adjusted	ICC_unadjusted	optional
0.0849	0.0849	FALSE

Modelo Nulo

Predicción dentro de los dam es constante.

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

Pred	log_ingreso	dam
8.827	6.251	1. Atlántida
8.827	6.255	1. Atlántida
8.741	6.324	4. Copan
8.827	6.342	1. Atlántida
8.302	6.417	10. Intibuca
8.518	6.492	14. Ocotepeque

Scaterplot de y vs $\hat{y}$

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

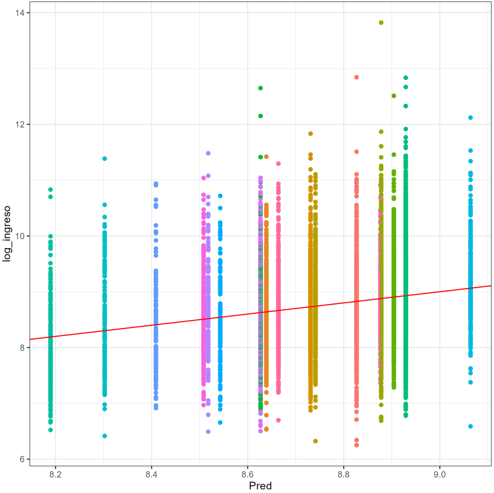


Figura 6: Predicción con el modelo nulo y los q-weighted

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

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

```
mod_Int_Aleatorio <- lmer(
  log_ingreso  ~ log_gasto   + (1 | dam),
  data = encuesta_hog, weights  =  Factor2)
performance::icc(mod_Int_Aleatorio)
```

ICC_adjusted	ICC_unadjusted	optional
0.0172	0.0072	FALSE

Modelo con intercepto aleatoria

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

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

	(Intercept)	log_gasto
1. Atlántida	0.9867	0.8916
2. Colon	0.9966	0.8916
3. Comayagua	1.0314	0.8916
4. Copan	1.0438	0.8916
5. Cortes	1.0485	0.8916
6. Choluteca	1.0670	0.8916
7. El Paraíso	1.0274	0.8916
8. Francisco Morazán	1.1409	0.8916

Modelo con intercepto aleatoria

Organizando los coeficientes para el gráfico.

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

plot_mod_Int_Aleatorio <-
  ggplot(data = encuesta_plot,
         aes(y = log_ingreso , x = log_gasto, colour = dam)) +
  geom_jitter() +
  theme(legend.position = "none",
        plot.title = element_text(hjust = 0.5)) +
  geom_abline(
    data = Coef_Estimado,
    mapping = aes(
      slope = log_gasto,
      intercept = `(Intercept)`,
      colour = dam
    ) ) +   theme_cepal()
```

Modelo con intercepto aleatoria

Figura 7: Modelo de intercepto aleatorio

Predicción del modelo

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

Pred	log_ingreso	dam
7.692	6.251	1. Atlántida
7.900	6.255	1. Atlántida
8.635	6.324	4. Copan
9.438	6.342	1. Atlántida
7.297	6.417	10. Intibuca
9.188	6.492	14. Ocotepeque

Scaterplot de y vs $\hat{y}$

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

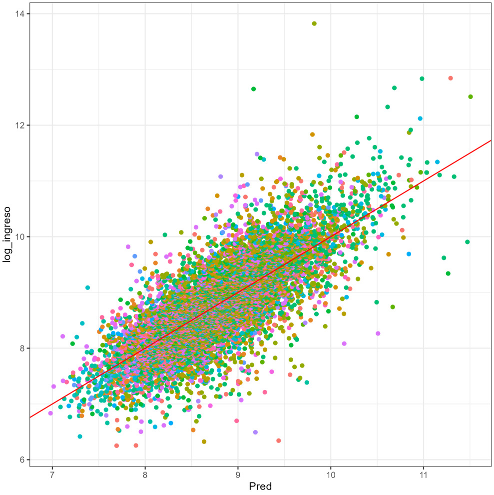


Figura 8: Scaterplot de y vs $\hat{y}$

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:

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

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

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

```
mod_Pen_Aleatorio <- lmer(
  log_ingreso  ~ log_gasto  + (1 + log_gasto| dam),
  data = encuesta_hog, weights  =  Factor2)

performance::icc(mod_Pen_Aleatorio)
```

ICC_adjusted	ICC_unadjusted	optional
0.0237	0.01	FALSE

Modelo con intercepto y pendiente aleatoria

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

	(Intercept)	log_gasto
1. Atlántida	0.7015	0.9245
2. Colon	1.4758	0.8352
3. Comayagua	0.7716	0.9220
4. Copan	1.3713	0.8531
5. Cortes	1.3878	0.8528
6. Choluteca	0.7372	0.9296
7. El Paraíso	0.8587	0.9117
8. Francisco Morazán	0.7349	0.9381
9. Gracias A Dios	0.9191	0.8943
10. Intibuca	0.8410	0.9082
11. Islas De La Bahía	1.0260	0.8895
12. La Paz	0.7560	0.9222
13. Lempira	1.0707	0.8915
14. Ocotepeque	1.9058	0.7705

Modelo con intercepto y pendiente aleatoria

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

plot_mod_Pen_Aleatorio <-ggplot(data = encuesta_plot,
       aes(y = log_ingreso , x = log_gasto,
           colour = dam)) +
  geom_jitter() + theme(legend.position="none",
    plot.title = element_text(hjust = 0.5)) +
  geom_abline(data = Coef_Estimado,
              mapping=aes(slope=log_gasto,
                          intercept=`(Intercept)`,
                          colour = dam))+
  theme_cepal()
```

Modelo con intercepto y pendiente aleatoria

Figura 9: Modelo con intercepto y pendiente aleatoria

Predicción del modelo

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

Pred	log_ingreso	dam
7.655	6.251	1. Atlántida
7.870	6.255	1. Atlántida
8.634	6.324	4. Copan
9.465	6.342	1. Atlántida
7.286	6.417	10. Intibuca
9.091	6.492	14. Ocotepeque
7.997	6.502	15. Olancho
7.735	6.523	9. Gracias A Dios

Scaterplot de y vs $\hat{y}$

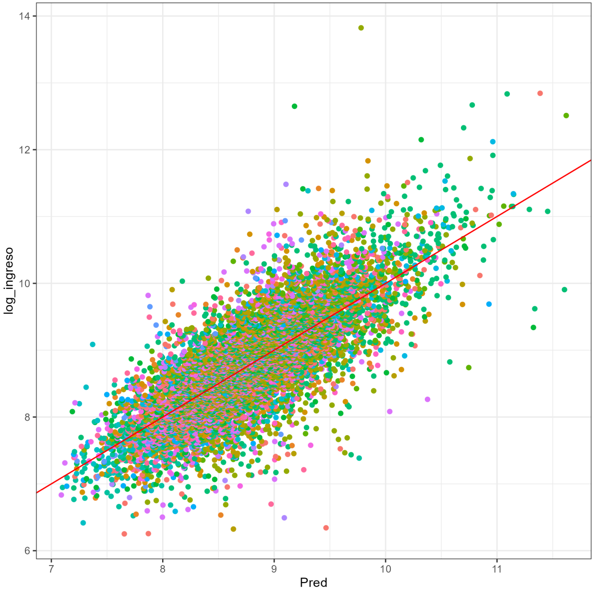


Figura 10: Scaterplot de y vs $\hat{y}$

Modelo con intercepto y pendiente aleatoria

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

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

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

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

donde μ_j es el gasto medio en el dam j.

Modelo con intercepto y pendiente aleatoria

```r
media_dam <- encuesta_hog %>% group_by(dam) %>%
  summarise(mu = mean(log_gasto))
encuesta_hog <- inner_join(encuesta_hog,
                           media_dam, by = "dam")

mod_Pen_Aleatorio2 <- lmer(
  log_ingreso  ~ 1 + log_gasto + Area + mu +
    (1 + log_gasto + Area + mu | dam ),
    data = encuesta_hog, weights  =  Factor2)
performance::icc(mod_Pen_Aleatorio2)
```

ICC_adjusted	ICC_unadjusted	optional
0.0291	0.0123	FALSE

Modelo con intercepto y pendiente aleatoria

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

Pred	log_ingreso	dam
7.689	6.251	1. Atlántida
7.901	6.255	1. Atlántida
8.634	6.324	4. Copan
9.470	6.342	1. Atlántida
7.394	6.417	10. Intibuca
8.699	6.492	14. Ocotepeque
8.004	6.502	15. Olancho
7.571	6.523	9. Gracias A Dios
8.554	6.533	2. Colon
7.645	6.545	2. Colon

Scaterplot de y vs $\hat{y}$

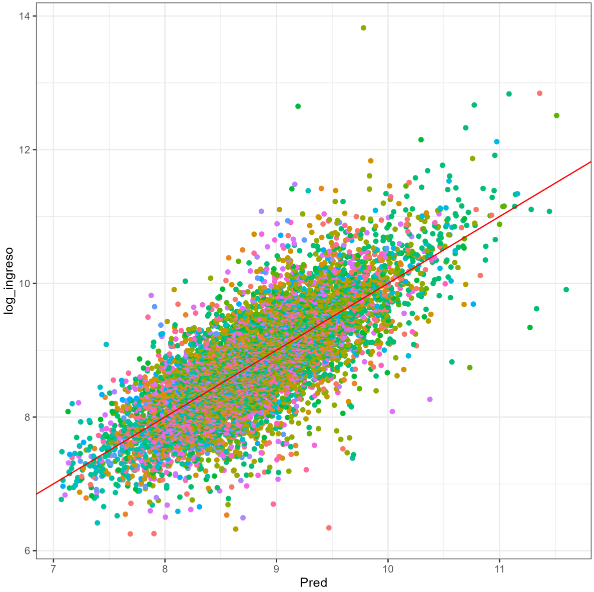


Figura 11: Scaterplot de y vs $\hat{y}$

Modelo con intercepto y pendiente aleatoria

```
(Coef_Estimado <- inner_join(
  coef(mod_Pen_Aleatorio2)$dam %>%
    add_rownames(var = "dam"),
  encuesta_plot %>% dplyr::select(dam, Area) %>% distinct()
))
```

dam	(Intercept)	log_gasto	Area2. Rural	mu	Area
1. Atlántida	1.7741	0.9093	-0.0854	-0.1050	1. Urbana
1. Atlántida	1.7741	0.9093	-0.0854	-0.1050	2. Rural
2. Colon	0.3209	0.8073	-0.1736	0.1645	1. Urbana
2. Colon	0.3209	0.8073	-0.1736	0.1645	2. Rural
3. Comayagua	0.7887	0.9091	-0.0823	0.0130	2. Rural
3. Comayagua	0.7887	0.9091	-0.0823	0.0130	1. Urbana
4. Copan	0.8749	0.8560	0.0010	0.0544	1. Urbana
4. Copan	0.8749	0.8560	0.0010	0.0544	2. Rural
5. Cortes	0.8007	0.8528	0.0167	0.0665	1. Urbana
5. Cortes	0.8007	0.8528	0.0167	0.0665	2. Rural

Modelo con intercepto y pendiente aleatoria

```r
(Coef_Estimado <- Coef_Estimado %>%
   inner_join(media_dam, by = "dam")) %>% slice(1:6L)
```

dam	(Intercept)	log_gasto	Area2. Rural	mu.x	Area	mu.y
1. Atlántida	1.7741	0.9093	-0.0854	-0.1050	1. Urbana	8.795
1. Atlántida	1.7741	0.9093	-0.0854	-0.1050	2. Rural	8.795
2. Colon	0.3209	0.8073	-0.1736	0.1645	1. Urbana	8.647
2. Colon	0.3209	0.8073	-0.1736	0.1645	2. Rural	8.647
3. Comayagua	0.7887	0.9091	-0.0823	0.0130	2. Rural	8.655
3. Comayagua	0.7887	0.9091	-0.0823	0.0130	1. Urbana	8.655

El modelo para el dam *1. Atlántida* viene dado por:

$$\hat{y}_{ij} = 1.7741 + 0.9093 log\_gasto_{ij} + (-0.9237)\, Zone_{ij} + (-0.10502)\, \mu_j$$

Modelo con intercepto y pendiente aleatoria

```
(Coef_Estimado %<>%  mutate(B0 = ifelse(
Area == "1. Urbana", `(Intercept)` + mu.y * mu.x + `Area2. Rural`,
               `(Intercept)` + mu.y * mu.x)) %>%
  dplyr::select(dam, Area, B0, log_gasto))
```

dam	Area	B0	log_gasto
1. Atlántida	1. Urbana	0.7650	0.9093
1. Atlántida	2. Rural	0.8504	0.9093
2. Colon	1. Urbana	1.5701	0.8073
2. Colon	2. Rural	1.7437	0.8073
3. Comayagua	2. Rural	0.9015	0.9091
3. Comayagua	1. Urbana	0.8192	0.9091
4. Copan	1. Urbana	1.3471	0.8560
4. Copan	2. Rural	1.3461	0.8560
5. Cortes	1. Urbana	1.4044	0.8528
5. Cortes	2. Rural	1.3877	0.8528

Modelo con intercepto y pendiente aleatoria

```
plot_mod_Pen_Aleatorio212 <-ggplot(data = encuesta_plot,
       aes(y = log_ingreso , x = log_gasto,
           colour = dam)) +
  geom_jitter() +
  theme(legend.position = "none",
        plot.title = element_text(hjust = 0.5)) +
  facet_grid( ~ Area) +
  geom_abline(
    data = Coef_Estimado,
    mapping = aes(
      slope = log_gasto,
      intercept = B0,
      colour = dam
    )
  ) +
  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 dam 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

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

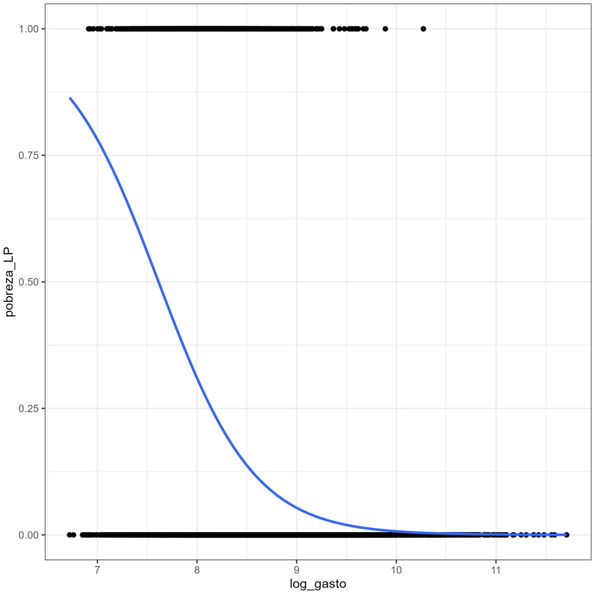
donde $\pi_{ij} = Pr\left(y_{ij} = 1 \mid x_i : \beta\right)$.

Ejemplos de modelo logit

```r
modl_logit1 <- ggplot(data = encuesta_hog,
       aes(y = pobreza_LP , x = log_gasto)) +
  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:



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_LP ~ 1,data = encuesta_plot,
     family=binomial(link = "logit")))
```

Ejemplos de modelo logit

A Continuación ajustamos el modelo sin intercepto por dam.

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

dam	B1	B0
1. Atlántida	-0.1836	-1.713
2. Colon	-0.1894	-1.713
3. Comayagua	-0.1904	-1.713
4. Copan	-0.1746	-1.713
5. Cortes	-0.2399	-1.713

Ejemplos de modelo logit

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

```
# Creando las variables respuesta
pred_logit <- coef_Mod %>%
  mutate(log_gasto = list(seq(0,20, length =100))) %>%
    tidyr::unnest_legacy()
pred_logit %<>% mutate(Prob = auxLogit(log_gasto,B0,B1))

modl_logit2 <- ggplot(data = pred_logit,
       aes(y = Prob, x = log_gasto, colour = dam)) +
  geom_line() +
   theme_bw() +
  theme(legend.position = "none")
```

Ejemplos de modelo logit

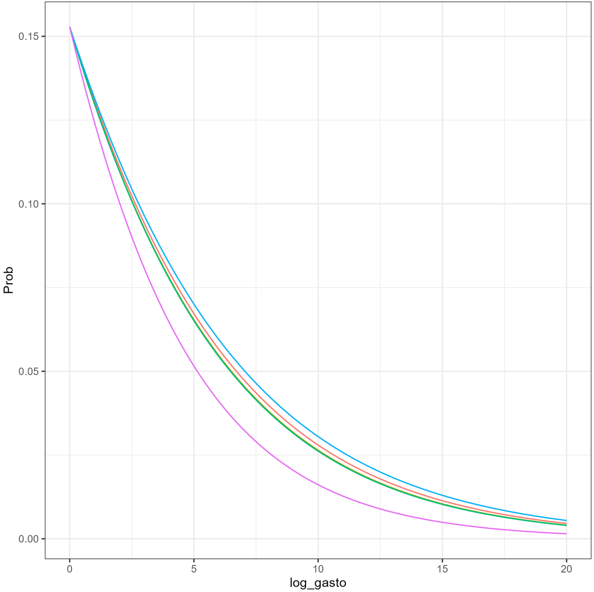


Figura 14: Modelo logit

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 dam j.
- $ _{ij} = $ El residual de la persona i en el dam j.
- $\gamma_{00} =$ El intercepto en general.
- $\tau_{0j} =$ Efecto aleatorio para el intercepto.

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

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

Modelo Nulo

```
library(lme4)
mod_logist_null  <- glmer( pobreza_LP  ~ ( 1  |  dam ),
                   data  =  encuesta_hog,
             weights  =  Factor2,
             family = binomial(link = "logit") )


coef( mod_logist_null )$dam
```

Modelo Nulo

	(Intercept)
1. Atlántida	-1.588
2. Colon	-1.548
3. Comayagua	-1.609
4. Copan	-1.631
5. Cortes	-1.907
6. Choluteca	-2.035
7. El Paraíso	-1.859
8. Francisco Morazán	-1.681
9. Gracias A Dios	-1.299
10. Intibuca	-1.686
11. Islas De La Bahía	-1.848
12. La Paz	-1.698
13. Lempira	-1.840
14. Ocotepeque	-1.715
15. Olancho	-1.693
16. Santa Bárbara	-1.390
17. Valle	-1.797
18. Yoro	-1.433

Modelo Nulo

```r
mod_logist_null
```

```
Generalized linear mixed model fit by maximum likelihood (Laplace
  Approximation) [glmerMod]
 Family: binomial  ( logit )
Formula: pobreza_LP ~ (1 | dam)
   Data: encuesta_hog
Weights: Factor2
      AIC       BIC    logLik deviance df.resid
     6983      6997     -3489     6979     8737
Random effects:
 Groups Name        Std.Dev.
 dam    (Intercept) 0.181
Number of obs: 8739, groups:  dam, 18
Fixed Effects:
(Intercept)
      -1.69
```

Modelo nulo

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

ICC_adjusted	ICC_unadjusted	optional
0.0099	0.0099	FALSE

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

	Pred	pobreza	dam
1	0.1697	1	1. Atlántida
3	0.1637	1	4. Copan
5	0.1562	1	10. Intibuca
6	0.1526	0	14. Ocotepeque
7	0.1554	0	15. Olancho
8	0.2144	0	9. Gracias A Dios

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

```
weighted.mean(encuesta_hog$pobreza_LP, encuesta_hog$Factor2)
```

```
[1] 0.1533
```

```
weighted.mean(tab_pred$Pred, encuesta_hog$Factor2)
```

```
[1] 0.153
```

Modelo con intercepto aleatoria

EL modelo se define de la siguiente manera:

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

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_LP ~ log_gasto  + (1 | dam),
  data = encuesta_hog, family = binomial(link = "logit"),
  weights  =  Factor2)

performance::icc(mod_logit_Int_Aleatorio)
```

ICC_adjusted	ICC_unadjusted	optional
0.0653	0.0395	FALSE

Modelo con intercepto aleatoria

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

	(Intercept)	log_gasto
1. Atlántida	17.20	-2.204
2. Colon	16.85	-2.204
3. Comayagua	16.87	-2.204
4. Copan	16.87	-2.204
5. Cortes	16.90	-2.204
6. Choluteca	16.33	-2.204
7. El Paraíso	16.16	-2.204
8. Francisco Morazán	16.92	-2.204
9. Gracias A Dios	17.14	-2.204
10. Intibuca	15.90	-2.204

Modelo con intercepto aleatoria

Gráficamente, los modelos ajustados se muestran a continuación:

```
dat_pred <- encuesta_hog %>% group_by(dam) %>%
  summarise(
    log_gasto = list(seq(min(log_gasto),
                         max(log_gasto), len = 100))) %>%
  tidyr::unnest_legacy()

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

plot_mod_Int_Aleatorio02 <- ggplot(data = dat_pred,
       aes(y = Proba, x = log_gasto,
           colour = dam)) +
   geom_line()+   theme_bw() +
  geom_point(data = encuesta_hog, aes(y = pobreza_LP, x = log_gasto))+
  theme(legend.position = "none",
        plot.title = element_text(hjust = 0.5))
```

Modelo con intercepto aleatoria

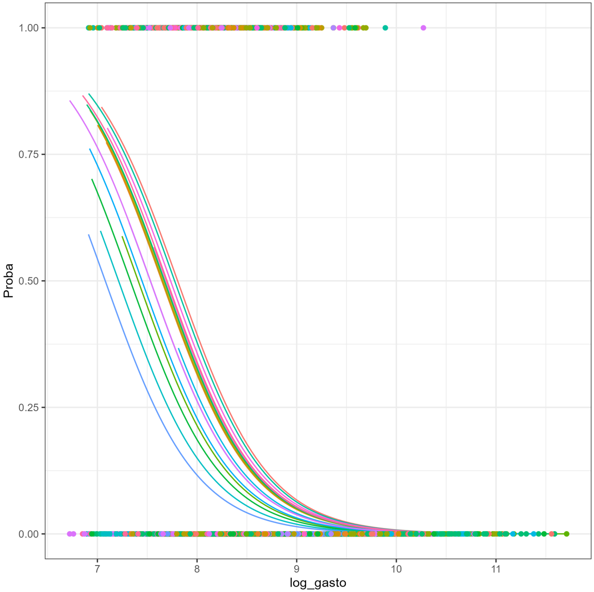


Figura 15: Modelo logit intercepto aleatoria

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_hog$pobreza_LP,
             dam = encuesta_hog$dam,
             Factor2 = encuesta_hog$Factor2)) %>% distinct() %>%
  slice(1:6L) # Son las pendientes aleatorias
```

Pred	pobreza	dam	Factor2
0.6517	1	1. Atlántida	0.7155
0.5280	1	1. Atlántida	0.7155
0.1305	1	4. Copan	0.3204
0.0244	1	1. Atlántida	0.6151
0.5639	1	10. Intibuca	0.9593
0.0188	0	14. Ocotepeque	3.3636

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

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, Factor2),
            pobreza = weighted.mean(pobreza,Factor2))
```

Pred	pobreza
0.1529	0.1533

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

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

y

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

Logístico Multinivel con Coeficientes Aleatorios.

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

```
mod_logit_Pen_Aleatorio <- glmer(
  pobreza_LP ~ log_gasto  + (1 + log_gasto| dam),
  data = encuesta_hog, weights  =  Factor2,
  binomial(link = "logit"))
# performance::icc(mod_logit_Pen_Aleatorio)
```

Modelo con intercepto y pendiente aleatoria

```
dat_pred <- encuesta_hog %>% group_by(dam) %>%
  summarise(
    log_gasto = list(seq(min(log_gasto),
                         max(log_gasto), len = 100))) %>%
  tidyr::unnest_legacy()

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

plot_mod_Pen_Aleatorio02 <-
ggplot(data = dat_pred,
       aes(y = Proba, x = log_gasto,
           colour = dam)) +
   geom_line()+   theme_bw() +
  geom_point(data = encuesta_hog, aes(y = pobreza_LP, x = log_gasto))+
  theme(legend.position = "none",
        plot.title = element_text(hjust = 0.5))
```

Modelo con intercepto y pendiente aleatoria

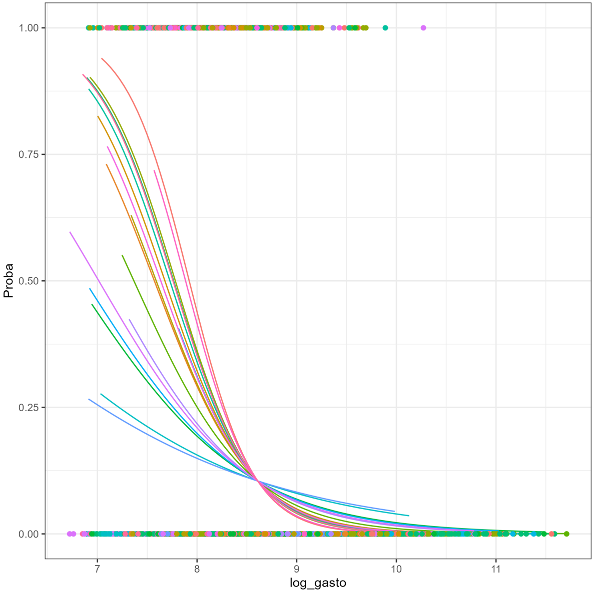


Figura 16: Modelo logit predicción

Predicción del modelo

Las predicciones se muestran a continuación:

```
(tab_pred <- data.frame(
  Pred = predict(mod_logit_Pen_Aleatorio,
                 type = "response"),
            pobreza = encuesta_hog$pobreza_LP,
            dam = encuesta_hog$dam,
            Factor2 = encuesta_hog$Factor2)) %>% distinct() %>%
  slice(1:6L)
```

Pred	pobreza	dam	Factor2
0.7775	1	1. Atlántida	0.7155
0.6276	1	1. Atlántida	0.7155
0.1248	1	4. Copan	0.3204
0.0076	1	1. Atlántida	0.6151
0.2674	1	10. Intibuca	0.9593
0.0403	0	14. Ocotepeque	3.3636

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

```
tab_pred %>%
  summarise(Pred = weighted.mean(Pred, Factor2),
            pobreza = weighted.mean(pobreza,Factor2))
```

Pred	pobreza
0.152	0.1533

Modelo con intercepto y pendiente aleatoria

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

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

donde

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

donde μ_j es el gasto medio en el dam j.

Modelo con intercepto y pendiente aleatoria

El ajuste del modelo es el siguiente:

```
mod_logit_Pen_Aleatorio2 <- glmer(
  pobreza_LP ~ 1 + log_gasto + TIENEVEHICULOS  + mu +
    (1 + log_gasto + TIENEVEHICULOS  + mu | dam ),
    data = encuesta_hog, weights  =  Factor2,
  binomial(link = "logit"))
# performance::icc(mod_logit_Pen_Aleatorio2)
```

Gráfica del modelo obtenido

Se grafican los modelos ajustados anteriormente:

```r
dat_pred <- encuesta_hog %>% group_by(dam, TIENEVEHICULOS , mu) %>%
  summarise(
    log_gasto = list(seq(min(log_gasto),
                         max(log_gasto), len = 100))) %>%
  tidyr::unnest_legacy()

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

plot_mod_Pen_Aleatorio002 <- ggplot(data = dat_pred,
       aes(y = Proba, x = log_gasto,
           colour = dam)) +
   geom_line()+   theme_bw() +facet_grid(.~TIENEVEHICULOS )+
  geom_point(data = encuesta_hog, aes(y = pobreza_LP, x = log_gasto))+
  theme(legend.position = "none",
        plot.title = element_text(hjust = 0.5))
```

Modelo con intercepto y pendiente aleatoria

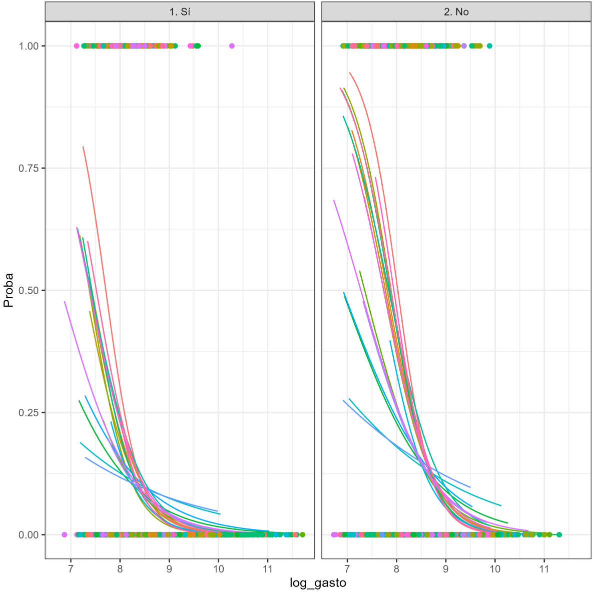


Figura 17: Modelo logit predicción

Predicción del modelo

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

```
(tab_pred <- data.frame(
  Pred = predict(mod_logit_Pen_Aleatorio2,
                 type = "response"),
           pobreza = encuesta_hog$pobreza_LP,
           dam = encuesta_hog$dam,
           TIENEVEHICULOS = encuesta_hog$TIENEVEHICULOS,
           Factor2 = encuesta_hog$Factor2)) %>% distinct() %>%
  slice(1:5L)
```

Pred	pobreza	dam	TIENEVEHICULOS	Factor2
0.8123	1	1. Atlántida	2. No	0.7155
0.6869	1	1. Atlántida	2. No	0.7155
0.1703	1	4. Copan	2. No	0.3204
0.0058	1	1. Atlántida	1. Sí	0.6151
0.2711	1	10. Intibuca	2. No	0.9593

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

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 tenencia de vehículo.

```
tab_pred %>% group_by(TIENEVEHICULOS) %>%
  summarise(Pred = weighted.mean(Pred, Factor2),
            pobreza = weighted.mean(pobreza,Factor2))
```

TIENEVEHICULOS	Pred	pobreza
1. Sí	0.0748	0.0757
2. No	0.2248	0.2262

¡Gracias!

Email: andres.gutierrez@cepal.org